Knowledge Discovery for Sustainable Urban Mobility

Marjan Momtazpour

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Science and Application

Naren Ramakrishnan, Chair
Madhav Marathe
Ratnesh Sharma
Chang-Tien Lu
B. Aditya Prakash

March 21, 2016
Blacksburg, Virginia

Keywords: Data mining, Urban computing, Smart grids, Electric vehicles.

Knowledge Discovery for Sustainable Urban Mobility

Marjan Momtazpour

(ABSTRACT)

Due to the rapid growth of urban areas, sustainable urbanization is an inevitable task for city planners to address major challenges in resource management across different sectors. Sustainable approaches of energy production, distribution, and consumption must take the place of traditional methods to reduce the negative impacts of urbanization such as global warming and fast consumption of fossil fuels. In order to enable the transition of cities to sustainable ones, we need to have a precise understanding of the city dynamics. The prevalence of big data has highlighted the importance of data-driven analysis on different parts of the city including human movement, physical infrastructure, and economic activities.

Sustainable urban mobility (SUM) is the problem domain that addresses the sustainability issues in urban areas with respect to city dynamics and people movements in the city. Hence, to realize an integrated solution for SUM, we need to study the problems that lie at the intersection of energy systems and mobility. For instance, electric vehicle invention is a promising shift toward smart cities, however, the impact of high adoption of electric vehicles on different units such as electricity grid should be precisely addressed. In this dissertation, we use data analytics methods in order to tackle major issues in SUM. We focus on mobility and energy issues of SUM by characterizing transportation networks and energy networks. Data-driven methods are proposed to characterize the energy systems as well as the city dynamics. Moreover, we propose anomaly detection algorithms for control and management purposes in smart grids and in cities. In terms of applications, we specifically investigate the use of electrical vehicles for personal use and also for public transportation (i.e. electric taxis). We provide a data-driven framework to propose optimal locations for charging and storage installation for electric vehicles. Furthermore, adoption of electric taxi fleet in dense urban areas is investigated using multiple data sources.

This work is supported in part by Virginia Tech and NEC Laboratories America, Inc.

Dedication

To my parents, Mehdi and Firouzeh, and to my beloved husband, Pejman.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Naren Ramakrishnan for his support, guidance, and patience through the course of this work. Indeed this thesis would not have been possible without his efforts.

I would especially like to thank Dr. Ratnesh Sharma for giving me the opportunity to collaborate with NEC Labs. I have spent great summers at NEC Labs which during that time, I have completed a significant portion of my research. Working with Ratnesh and other research scientists taught me to think differently about problems in various domains.

I would like to thank my thesis committee members, Prof. Madhav Marathe, Dr. Ratnesh Sharma, Dr. Chang-Tien Lu, and Dr. B. Aditya Prakash for their advice and comments.

I also acknowledge my professors, mentors, collaborators, and close friends for their support, comments, and suggestions. Special thanks goes to Dr. M. Shahriar Hossain whom I learned so much from. I like to thank Dr. Sanmay Das, Dr. Patrick Butler, Dr. Mohammad C. Bozchalui, Dr. Babak Asghari, and my fellow lab-mates for their guidance and assistance during past years.

Finally, I thank my amazing family, in particular my parents, for their endless love, encouragement, advice, and support that I have gotten over the years. Indeed without their support, I would not be able to make it this far. I would also like to thank my wonderful husband for all his support and endeavor during these past years. I learned so much from him in my academic life and he was always by my side through the good and bad times. Indeed without his love, patience, and encouragement, I would not be able to survive in graduate school.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Smart cities are going to become an inevitable part of our lives in the near future. According to the UN's World Urbanization Prospects, by 2050, more than 66% of the global population will live in the cities [123]. Sustainable urbanization is thus an important task for city planners with major challenges in resource management needing to be addressed satisfactorily across different sectors (e.g. water, energy, health care, and transportation). The modern prevalence of 'big data' techniques now supports entirely data-driven analysis on human movement, physical infrastructures, and economic activities at a city-scale. This aids in the creation of rapid prototyping tools for posing and answering questions of emerging urban settings.

One of the key aspects of modern cities will be sustainable energy systems. In fact, sustainable energy systems are considered as the backbone of all modern urban infrastructure systems. Due to the fast decline of fossil fuels, sustainable approaches of energy production, distribution, and consumption are now going to take the place of traditional methods [97]. The smart grid—a modernized electrical grid—is a safe, reliable, and cost effective grid that functions using advanced sensors and distributed computing technologies. Energy microgrids are a key building block of smart grids. By definition, a microgrid is a group of interconnected loads, energy storage, and power generation systems within clearly defined boundaries that act as a single controllable entity with respect to the grid.

Sustainable urban mobility (SUM) is a problem that lies at the intersection of sustainable energy systems and urban transportation. The advent of electric vehicles (EVs) and the new technologies deployed in smart grids are trends that are ushering in the study of SUM. Electric vehicle invention is a promising shift toward smart cities. EVs are going to take the place of gasoline-powered vehicles by benefiting the society from different perspectives. In this regard, new technologies deployed in smart grids have this potential to proactively respond to human mobility and dynamic behavior patterns in the city, helping to overcome various difficulties in energy networks.

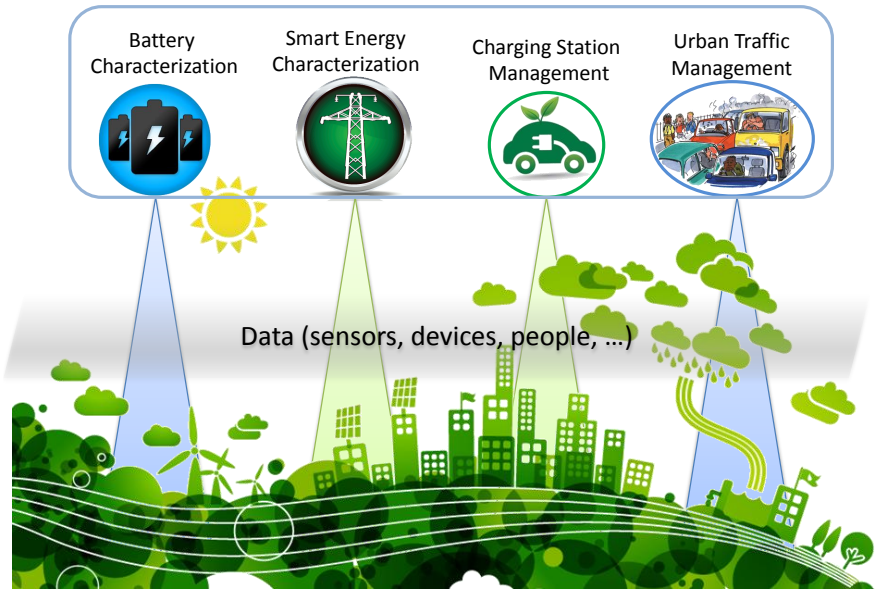


Figure 1.1: Schema of a smart city.

To realize an integrated solution for SUM, we need to study problems at the intersection of energy systems and mobility. Without considering the mobility aspect, any consideration about energy consumption and projections would be unrealistic. For instance, the high penetration of electric vehicles will place extra demand on the utility grid. To resolve such issues, new demand response strategies must be taken into account. Similarly, without considering the energy aspect, transportation planning will result in impractical outcomes. To clarify issues in SUM, consider the following scenarios:

1. EV Charging Station Placement: One of the key issues in deployment of EVs is the design and placement of charging infrastructure to support their operation. While it is better to install EV charging stations (CSs) in those places that are convenient for the drivers, placing them without considering the electricity consumption profile across the city is not realistic. Also, the impact of higher peak demand imposed on the grid by EVs should be evaluated using proper demand response strategies. For example, charging the EV at off-peak hours places less strain on the electric grid in addition to providing lower cost of electricity for consumers.

2. Electric Taxis in NYC: Manhattan is going to replace yellow taxicabs by EVs. By 2020, one-third of the city's fuel-based taxis are intended to be replaced by electric ones. While EV technology has improved in recent years, none of the existing EV designs can be considered as an economically perfect fit for taxi industry. It is estimated that one-third of the electric taxi fleet requires a network of 350 quick-charging stations (each one adds 50kW

of electric load to the grid) that cost about $20 million per year while the generated revenue from taxi drivers is about $13 million per year [78]. In addition to the aforementioned economic issues, there are at least two other operational issues. First, due to the day-long trips made by taxis, EV batteries need to be charged multiple times during a day whereas the charging process is usually time-consuming. For taxi drivers, time is money and hence it is preferred to deploy quick charging stations, in turn resulting in a huge amount of extra load on the grid. For instance, a level-3 charging process requires 50kW of electric load to charge 80% of an EV battery in 30 minutes. Second, most of the taxicabs (around 90% of them) are rented by two drivers to cover their 12-hour shift period which limits the time that could be used for low-cost level-2 charging. Also, taxi drivers prefer to use their rest time at shift change to charge their EVs. Shift change of NYC taxis happens at 5:00 PM at predetermined garages in clusters. Thus, maintaining an electric taxi fleet will create spikes of demand on the grid.

3. Failure in Grid and Electric Vehicles: EVs are ideal for fleet managers due to their high efficiency and easy maintenance. As EV fleets expand, concerns about quality and reliability of this technology will stand out. Utility companies have to provide reliable energy delivery through the smart grid during peak hours with the help of renewable energy sources at the time of high demand [26]. Understanding the interaction between city mobility (e.g., involving EVs) and power utility optimization will underlie careful policy design and implementation. For instance, offering rate incentives to consumers to prevent outages due to high demands of EVs at peak-hours is one solution. However, outage prevention and balancing the demand with the power grid capacity are not the only concerns in SUM. When the adaption rate of EVs in a populated urban area is increased, most of the city transportation will be performed using electric vehicles, either personal or public ones. In this situation, any unplanned outage and anomalous behavior in the power grid may result in catastrophic conditions. In a broader sense, while in smart cities, various technologies are used to improve the quality of life, the whole system is vulnerable to various kinds of failure due to natural disasters, physical faults, or security attacks [94]. Due to the interdependencies within and between different components in a smart city, one failure may cause multiple subsequent cascading failures which may in turn cause catastrophic conditions. Hence, continuous monitoring of the power grid and deployment of smart anomaly detection techniques is inevitable in SUM.

In this research, we use data analytics to address some of these challenging problems in a smart city context. The ultimate goal is to propose a data-driven solution for sustainable urban mobility in the context of smart cities. In the remaining parts of this chapter, we describe the thesis motivation and goals in detail and introduce the contributions of the thesis.

Figure 1.2: Issues in SUM.

1.1 Issues in SUM

Sustainable urban mobility has various benefits to the society such as improving the quality of life, contributing to better health and environment, improving mobility access, effective use of resources, and saving the costs [127]. In this section, we introduce different challenging issues in SUM along with several examples.

We have categorized issues in sustainable urban mobility into several groups: Urban Planning, Urban Environment, Energy Management, Infrastructure Investment, and Sharing Economy. Figure 1.2 represents these groups illustrating several identified problems at each group that scientists have been trying to solve them in the past years. Here, we will briefly introduce these categories:

- **Urban Planning:** Urban planning plays an important role for efficient use of various resources in a city such as land, energy, and water resources. Urban planning not only involves scientific and technical considerations, but also political processes. In transition to smart cities, various problems in sustainable urban mobility fall into the category of urban planning. Traffic monitoring to detect and prevent accidents and anomalous event is one major task that has been of huge interest. Finding locations of interests (e.g. choosing the best places for ambulance stations), detecting damages in road, assigning car-free zone, and other socio-economic issues are few examples in this category.

- **Urban Environment:** The most challenging environmental issues such as climate change, pollution, industrial and household waste, and overpopulation in cities result in many unwanted hazardous problems such as public health issues and global warming. Proper sustainable planning in urban areas is required to combat these growing threats. Use of sustainable energy resources (a.k.a green energy) and reduction of CO_2 emission (e.g. fuel-based cars) are two fundamental steps to overcome environmental issues.

- **Energy Management:** Energy management strategies try to optimize energy con-

sumption and minimize the associated costs by proper actions in monitoring, controlling, and conserving energy. Wide range of activities fall into this category including proper management of electricity consumption at homes (smart homes), production and distribution of renewable energy sources in microgrids, or outage prediction in utility grid. Energy management is tied with SUM from different aspects. For example, the extra load imposed on the grid by electric vehicles will arise concerns regarding the capacity of the grid. Hence, the need for mobile storages and proper installation of charging stations should be carefully addressed. Other strategies such as vehicle-to-grid (V2G) or advanced technologies such as smart lighting can be discussed here.

- **Infrastructure Investment:** Any investment for deploying smart infrastructure in cities should be addressed and assessed precisely. For example, electric utility infrastructure such as charging stations for electric vehicles is a challenging topic affecting various parameters in the city. Intelligent transport systems (such as Subway and BRT), installation of advanced parking meters are two other examples.
- **Sharing Economy:** Sharing resources in communities has two main advantages: affordability and optimization of resource consumption. In this regard, bike-sharing in various cities or car-sharing companies (e.g. Uber) have gained huge popularity in the past few years. Sharing economy has been beneficial in cities from different points of view such as CO_2 emission reduction and saving costs.

In this thesis, we have studied several issues in SUM such as charging infrastructure placement for electric vehicles (infrastructure investment), monitoring urban traffic (urban planning), monitoring and management of energy devices (energy management). Other issues such as urban noise and urban pollution, and problems under sharing economy have been left for future work.

1.2 Goals of the dissertation

In this thesis, the common thread in all of our studies is to promote the existing approaches to discover complex hidden relationships and behaviors in SUM which can be used for applications such as infrastructure design and investment in urban areas. In this section, we first describe the thesis motivations. Then, we describe our data-driven framework to study SUM. At the end, we mention our contributions and the datasets used in our studies.

1.2.1 Thesis Motivation

The rapid growth in urban populations has highlighted the importance of harnessing data-driven methods to aid in city planning, including in areas like stemming air pollution, controlling energy consumption, and relieving traffic congestion [139]. Urban computing [59] is

an emerging area which aims to foster human life in urban environments through the methods of computational science. It is focused on understanding the concepts behind events and phenomena spanning urban areas using available data sources, such as people movements and traffic flows. Organizing relevant data sources to solve compelling urban computing scenarios is itself an important research issue. With respect to SUM, we can categorize different urban computing issues into two groups: urban transportation and energy systems. In this section, after a short introduction about each of these categories of issues, we introduce the problems that we will study in this thesis.

Sustainability and Energy Systems:

Ushered by recent developments in various areas of science and technology, modern energy systems are going to be an inevitable part of our societies. Smart grids are one of these modern systems that have attracted many research activities in recent years. Before utilizing the next generation of smart grids, we should have a comprehensive understanding of the interdependent energy networks and processes. Next-generation energy system networks cannot be effectively designed, analyzed, and controlled in isolation from the social, economic, sensing, and control contexts in which they operate.

Smart cities are vulnerable to various kinds of failure [94] and due to the dependencies that exists both among the different components and within each individual sub-system, any fault in the system may cause multiple cascading failures which in turn may result in catastrophic conditions. As an example, failure in energy systems will bring several problems such as inability to charge EVs. Modern energy systems, such as smart grids, are complex and contain multiple subsystems and due to this complexity, characterization of these systems is not a straightforward task. Furthermore, in these systems, the complex relationships between various signals make the process of anomaly detection a complicated task. Hence, proper anomaly detection methods are needed.

In recent years, with the rapid growth in data logged from modern devices in a distributed system, the need for having stronger knowledge discovery methods has attracted significant attention [109]. Concomitantly, the size and complexity of these systems have become a burden for administrators in detecting failures and repairing them [35, 106]. These challenges inspired us to characterize and track anomalies in urban transportation and smart grid systems by correlating all monitored data across the system.

Sustainability and Urban Transportation:

One of the main concerns in large urban cities is to analyze and monitor traffic flows with a view toward characterizing both regularities and anomalies; detection of anomalies (e.g., caused by accidents, protests, sports, celebrations, disasters) for instance can be utilized to help mitigate congestion and diagnose bottlenecks. Proper method is required to handle a large amount of logged traffic data for better understanding of mobility in the city and for proper further decision making process.

Due to the fast decline of fossil fuels and global warming, sustainable approaches of energy

production, distribution, and consumption are now going to take the place of traditional methods [97]. Electric vehicles have gained a lot of interest for their potential in reducing CO_2 emissions [128]. However, to be prepared for a world laden with EVs, we must revisit various features in the design of energy distribution systems and we need to address multiple issues. One of the key issues in ushering in EVs is the design and placement of charging infrastructure to support their operation. Issues that must be addressed include [97]:

(i) prediction of EV charging needs based on their owners' activities;

(ii) prediction of EV charging demands at different locations in the city, and available charge of EV batteries;

(iii) design of distributed mechanisms that manage the movements of EVs to different charging stations; and

(iv) optimizing the charging cycles of EVs to satisfy users' requirements, while maximizing vehicle-to-grid profits.

In large metropolitan areas a significant load of the transportation is handled by taxis and hence, replacement of fuel-based taxicabs with *electric taxis* is considered as an important step toward a green city. Due to the characteristics of public transportation and the important differences between taxi fleet and personal vehicles, in addition to charging station placement for personal EVs, CS placement should be precisely addressed for taxi fleets in large metropolitan areas. There are several issues that must be addressed before ushering in electric taxi fleet adoption. The foremost is to study if the replacement of taxis with EVs is economically feasible for drivers and the CS owners. In a comprehensive planning effort, it is crucial to consider economic factors in the deployment of electric taxis and charging infrastructure design to ensure financial feasibility as well as long-term economic growth.

Another important issue for electric taxis is design and placement of charging infrastructure. Due to the long distances traveled by taxi drivers and their limited times to rest, fast chargers must be provided in a large scale, properly distributed in the city. Proper placement of charging stations results in optimal distribution of electricity load, maximization of revenue of service providers, and lead to increased availability of charging stations, and reduced range anxiety.

1.2.2 Use of Data Analytics in SUM

Different approaches and techniques have been used to face with a diverse range of smart city problems. In the literature, model-based methods, multi-agent systems, and game theoretic techniques have been proposed to simulate and model the cities to study the smart city issues. Smart cities are complex structures with multiple components and dynamic

interactions. Therefore, developing an effective and precise analytic model for a smart city is not a straightforward and easy task.

Sweda and Klabjan [116] developed a decision support system which is agent-based and is used to identify patterns in EV ownership and driving activities. This support system is used to enable strategic deployment of charging infrastructure. Gerding et al. [36] developed an online auction protocol. In this approach, EVs use agents to bid for available time slots and power. In another work, Escudero-Garzas and Seco-Granados [30] presented a framework for selection of the best charging station using two-way communication between CSs and EVs using a non-cooperative oligopoly game model that uses differentiated products theory and conjectural variations to provide a Nash equilibrium. Other techniques such as computer vision-based methods has been widely used in detecting objects in massive load of images. However, the application of computer vision in SUM has some limitations and is constrained to the availability of sufficiently good images and videos [31].

With the advances in sensor networks and telemetry, nowadays there are multiple devices and sensors that measure a diverse set of the city parameters and with the help of the data which is logged by these devices, we can understand the behavior of city. Using the provided city data, data analytic and machine learning methods may be deployed to deal with the activities in a city and to propose effective solutions.

In this dissertation, we aim to use data analytic methods in order to tackle major issues in SUM. We deploy various data mining techniques such as clustering and regression methods in combination with data visualization techniques for the purpose of knowledge discovery and representation.

1.2.3 A Framework to Study SUM

Urban mobility is a complex system and this complexity is increased when we are facing with highly populated cities. We need to have a precise understanding of the city behavior and a well-designed policy to enable a transition of cities to sustainable ones [87],[29]. In order to have this, we need to develop an in-depth analysis framework to analyze current mobility systems and the assessment of the potential for transition to smart cities. Due to the increase in volume of mobility in large metropolitan areas, negative impacts on health, quality of life, climate, space, and landscape are also increasing. A fundamental approach is needed to handle the interactions that exist among transportation, energy networks, economic infrastructures, and social activities in a sustainable mobility system.

In this dissertation, we propose a top-down data-driven framework as a unified approach to face with SUM issues. The overall framework is illustrated in Figure 1.3. In this framework, we identify and investigate the problems and issues in SUM in multiple steps as follows:

1. **Problem Identification:** In this step, we need to identify all the components and

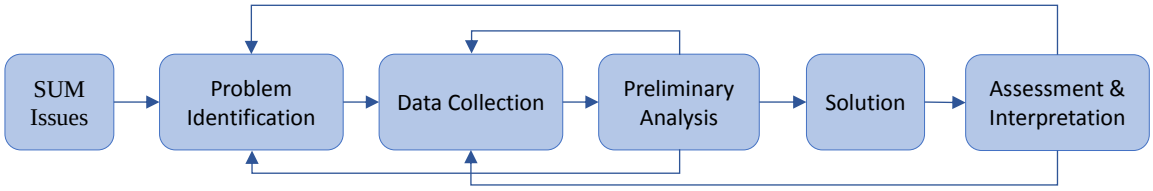


Figure 1.3: Overall framework for SUM issues.

parameters that are involved in the problem. As an example, for the problem of CS placement for personal vehicles, distribution of electricity consumption in the city and the points of interest of the EV owners are important factors.

2. **Data Collection:** In this step, we need to gather all the datasets that are required for the problem. If real datasets are not available for some particular parameters, realistic synthetic datasets may be deployed here. As an example, for the problem of CS placement for personal vehicles, we need to have datasets about the mobility patterns of potential EV owners in the city as well as electricity consumption data.

3. **Preliminary Analysis:** In this step, we perform initial analysis to understand the underlying structures and relationships. This step is necessary due to the complexity and dynamic behaviors that can be observed in a highly populated city or an advanced sustainable energy system. During this step, we may observe that some new parameters and/or datasets should be added to the model. As an example, for the problem of CS placement for personal vehicles, we need to understand how electricity consumption is changed during the day. As another example, in monitoring and outlier discovery of transportation network we need to look for hotspot locations and peak-time periods.

4. **Solution:** In this step, we need to come up with a data analytic solution for the problem.

5. **Assessment and Interpretation:** After proposing a solution for an issue in SUM, assessment, interpretation, and validation of results are necessary. Comparison with the state-of-the-art methods using different metrics such as accuracy can be performed in this step. For applications such as CS placement or recommendation algorithms, real implementation is ultimately required to observe the actual performance of the proposed solution.

1.2.4 Thesis Contributions

In this thesis, we provide data-driven solutions for various problems in SUM. These problems are related to each other from different perspectives, including structural relationship,

common data sources, and their importance regarding SUM. The studied problems are as follows:

- **Urban Traffic Characterization:** Accurate models of large-scale traffic flows need to be developed with applications to many areas, including outlier detection and characterization. For this problem, we analyze taxi trips in New York City logged in 2013. Our goals are to infer knowledge about the pattern of locations with respect to their transportation profiles, to find hotspot locations, and to detect and track anomalies over time.

- **Energy System Characterization and Anomaly Detection:** In order to analyze large-scale data logged from complex energy systems (such as microgrid), we need to infer invariants to capture the functional and operational relationships. Here, we aim to use a more realistic approach to discover hidden patterns and indirect relationships among devices by employing latent variables in regression models. Specifically, we harness hidden factors derived by factor analysis and use them in regression models. We perform various experiments on synthetic and real datasets. Furthermore, we use graph representations for better visualization of relationships which aids in discovering system-wide anomalies.

- **Charging Station Placement for Personal EVs:** We propose a new framework to address the problem of charging and storage infrastructure design for EVs by adopting an urban computing approach. Furthermore, due to the additional load imposed to the network by EVs, appropriate storage units must be deployed beside the charging stations. There are several works that consider the problem of load management for EV charging and its impact on the grid [88, 77]. However, there is no previous work that addresses the coordinated impact of placement over an urban infrastructure and its solution thereof.

- **Charging Station Placement for Electric Taxi Fleet:** We propose a data-driven framework for mass scale electric taxi adoption to replace fuel-based taxicabs of NYC. This framework determines various parameters including optimal adoption rate of electric taxis, desired number of fast charging stations, and optimal locations to install charging infrastructures with respect to various parameters such as revenue, electricity load, land price, and points of interest. Furthermore, an online recommendation system is proposed to suggest the best available charging station aiming to minimize detour and waiting times. Experimental results indicate that the proposed method can be effectively deployed for high adoption of electric taxis in metropolitan areas.

Structural relationships w.r.t. SUM: Figure 1.4 shows the structural relationships in SUM that were taken into considerations in this thesis. This figure illustrates how SUM issue is divided into various interacting problems. The inter-relations and dependencies among these problem domains are also depicted in this figure. In this dissertation, we study

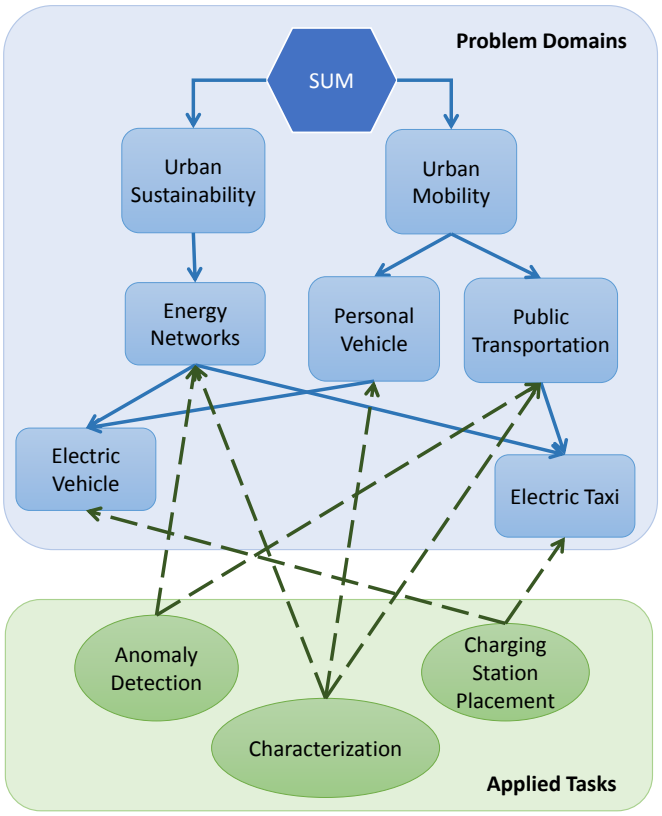


Figure 1.4: Structural relationship of problems and tasks in SUM.

various challenging problems such as anomaly detection in energy systems, and placement of charging stations for personal vehicles and taxis. As one may notice, there are various correlating parameters and factors that can be added to this figure. However, in this research we focus on the most important problems in mobility and sustainability.

In a typical urban area, both public and private transportation systems are used widely. Therefore, in this dissertation we study two different transportation systems: personal vehicles and taxis. These two devices are the most important types of transportation that have high effect on SUM and can be replaced by electric vehicles to reduce the CO_2 emission. Personal vehicles and taxis have different patterns in urban mobility. For example, taxis mainly transport to/from locations of interest such as downtown areas. Also, the charging requirements of taxis and personal vehicles are different due to their different usage patterns.

Datasets: Figure 1.5 shows how each problem uses different datasets for the purpose of knowledge discovery in sustainable urban mobility. As this figure illustrates, people's activity in terms of transportation, consumption, and living conditions have been studied. Profile of people dataset contains demographic information of people such as their age, income, and personal vehicles. Transportation activity dataset includes activity of people whether

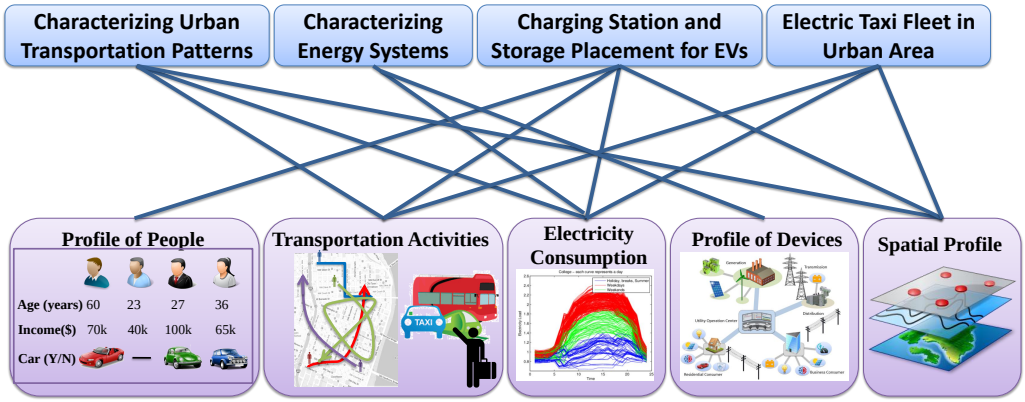


Figure 1.5: Datasets used in studied research problems.

they use their personal vehicles or they use public transportation. For example, pick-up and drop-off locations of passengers, as well as the time of trip is logged. For personal vehicles, type of activity (e.g. school, shopping) and origin and destination of trips are recorded in the dataset. Electricity consumption dataset contains electricity load of each building at each time based on the type of activities. Profile of devices dataset contains logged measurements from devices in energy systems such as temperature and voltage. Spatial profile dataset contains geographical information, and other attributes such as land price index. As we can observe, each research problem uses a set of datasets depending on its specific requirements and one dataset may be deployed in several problems.

Summary of Contributions:

In this part, we provide a summary of thesis contributions. We can categorized contributions into two groups: applied contributions and theoretical ones. Each chapter of this thesis focuses on a specific problem in SUM which uses one or more of the following contributions:

Theoretical Contributions:

- Summarizing the behavior and signature of patterns in datasets using a graph and detecting outliers based on the graph changes. In Chapter 2 and Chapter 3, graph is used to show average traffic flow and invariant relationships and outliers are discovered based on the change in these graphs.
- Proposing latent factor analysis regression to reveal hidden correlations among time series in a cyber-physical system in Chapter 3.
- Proposing the use of average probabilistic flow graph for the purpose of spatio-temporal characterization and anomaly detection in Chapter 2.

Applied Contributions:

- Developing a recommendation and assignment algorithm to assign electric vehicles to charging stations in Chapter 5 and Chapter 4.
- Proposing a data-driven approach to find ideal locations for charging stations based on various data sources such as electricity usage and points of interest in Chapter 5 and Chapter 4.
- Proposing a data-driven method to determine which taxis are suitable to be replaced by EVs and number of required charging stations in Chapter 5.

1.3 Organization of Dissertation

The design of a smart and sustainable city faces with various challenging issues such as energy consumption minimization, anomaly detection, and placement of new infrastructure. In this dissertation we aim to study these issues using data analytics approaches. The rest of this report, dedicated to the above issues, is organized as follows:

Chapter 2) Characterizing Urban Transportation Patterns: First, we aim to understand and characterize city mobility using data mining techniques. In this chapter, transportation in cities in terms of traffic flows of taxi drivers is analyzed using graph mining approaches. Here, we look at the yellow taxicab system of Manhattan. This study is useful in development of accurate models of large-scale traffic flows with applications to many areas, including outlier detection and characterization. This chapter is published in the 4th ACM SIGKDD International Workshop on Urban Computing [82].

Chapter 3) Characterizing Energy Systems: In order to analyze large-scale data logged from complex energy systems (such as microgrid), we need to infer invariants to capture the functional and operational relationships. In this chapter, we focus on invariant discovery algorithm using latent factor approach to monitor a cyber-physical system. We performed our experiments on a microgrid system (provided by NEC Labs) and other testbeds. This chapter is published in the 21th ACM SIGKDD Conference on Knowledge Discovery and Data Mining [84].

Chapter 4) Charging and Storage Infrastructure Design for Electric Vehicles: Electric vehicles are touted as the sustainable alternative to reduce our over-reliance on fossil fuels and stem our excessive carbon emissions. As the use of EVs becomes more widespread, planners in large metropolitan areas have begun thinking about the design and installation of charging stations city-wide. Unlike gas-based vehicles, EV charging requires a significant amount of time and must be done more periodically, after relatively shorter distances. In this chapter, we present a novel framework to support charging and storage infrastructure design for EVs. We develop coordinated clustering techniques to work with network models of urban environments to aid in placement of charging stations for EVs. Furthermore, we evaluate the network before and after the deployment of charging stations, to recommend

the installation of appropriate storage units to overcome the extra load imposed on the network by the charging stations. This chapter is published in ACM SIGKDD International Workshop on Urban Computing [80] and in the ACM Transactions on Intelligent Systems and Technology [81].

Chapter 5) Electric Taxi Fleet in Urban Area: In this chapter, we propose a data-driven framework for mass scale electric taxi adoption to replace fuel-based taxicabs of NYC. This framework determines various parameters including optimum adoption rate of electric taxis, desired number of fast charging stations, and optimal locations to install charging infrastructures w.r.t various parameters such as revenue, electricity load, land price, and points of interest. Furthermore, an online recommendation system is proposed to suggest the best available charging station aiming to minimize detour and waiting times. Experimental results indicate that the proposed method can be effectively deployed for high adoption of electric taxis in metropolitan areas.

Chapter 6) Conclusion: In this chapter, after a brief conclusion of thesis, future paths are proposed.

Chapter 2

Characterizing Urban Transportation Patterns

2.1 Introduction

One of the main concerns in large urban cities is to analyze traffic flows with a view toward characterizing both regularities and anomalies; detection of anomalies (e.g., caused by accidents, protests, sports, celebrations, disasters) for instance can be utilized to help mitigate congestion and diagnose bottlenecks.

In this chapter, we aim to provide a data analytic framework to infer knowledge about the mobility patterns, to find hotspot locations, and to detect and track anomalies over time in a public transportation system. One of the main objectives of city planning is to resolve such issues and move toward a smarter city [10]. The datasets gathered from wireless technologies (e.g. sensors and GPS) represent traffic flows which are beneficial to understand human mobility in such dynamic and complex cities and ease the future decision making processes. For this purpose, we use the dataset of taxi trips in New York City logged in 2013. Using clustering approaches, we categorize the city into smaller units and characterize locations based on their daily taxi pick-up and drop-off demands. We also develop a novel probabilistic graph representation of traffic flow to infer common profiles of each location. We investigate locations by extracting their local and egonet features in the traffic flow graph. Then, we extract the role of each location in graph using role extraction methods and finally detect spatio-temporal outliers using the extracted roles.

Our contributions are thus:

- Developing a novel average probabilistic flow graph to capture the behavior of traffic flow in each location.
- Characterizing interesting locations using anomaly detection methods applied over the

average probabilistic flow graph.

- Using role extraction and role-change detection to understand normal roles of each location and find spatio-temporal anomalies in dynamic graphs.

2.2 Related Work

Taxi Datasets: Taxis do not have pre-specified routes and schedules and can provide a unique insight into the mobility of people through a city. There have been several works on mining taxi GPS traces in three main categories: social dynamics, traffic dynamics, and operational dynamics [15]. In social dynamics, the behavior of a group of people is studied for several purposes such as to identify hotspots, to characterize locations based on functionality, and to find frequent trajectories and connectivity (linkage) between regions. The results are useful as a guide to future decision making [15]. In traffic dynamics, the dynamics of congestion levels of vehicles have been studied. For example, travel time and speed or adverse traffic events or even air pollution resulting from traffic can be analyzed [15]. As an example, Wang et al. [126] estimate travel time of a path in a sparse trajectory dataset using tensors. In operational dynamics, the goal is to provide useful information to drivers (and passengers). Ranking drivers, taxi-finding strategies [136], taxi ride-sharing, route planning, anomaly detection (accident, road repair, fraudulent drivers), and route prediction (travel time estimation) are a few tasks that have been studied under this category [15].

Graph Mining: One of the popular graph mining techniques applied here (e.g., to bike usage data) is community clustering. In [33], the authors clustered bike-sharing stations according to their usage profiles using Poisson mixtures. Bike demand for NYC bikes has been analyzed in [112]. The authors in [7] used community detection with a gravity model applied to bicycle flow dataset. Cheng et al. [25] used an ARIMA model to capture autocorrelation in road traffic data locally and dynamically. Min et al. [79] used a hybrid spatio-temporal method to forecast traffic also using an ARIMA model. The authors of [129] used spatio-temporal random effect models to predict traffic flows.

Liu et al. [70] investigate inter-urban movements from a check-in dataset to analyze the underlying patterns of trips and spatial interactions by fitting gravity models. For visualization, the authors in [142] proposed a flow clustering method to cluster flows in order to avoid cluttering while revealing abstracted flow patterns. In order to estimate missing data or to find structure of different units in different problem domains, matrix factorization and tensor decomposition have been studied previously [126]. As an example, Lian et al. [65] exploit weighted matrix factorization to view mobility records in location-based social networks for point-of-interest recommendation. As another example, in [140], noise situations in NYC are modeled using tensor decompositions. In [39], a tensor is created for traffic flow state and clustering methods are used to derive traffic states.

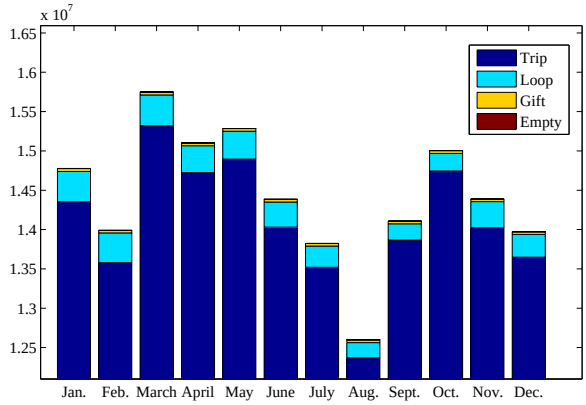
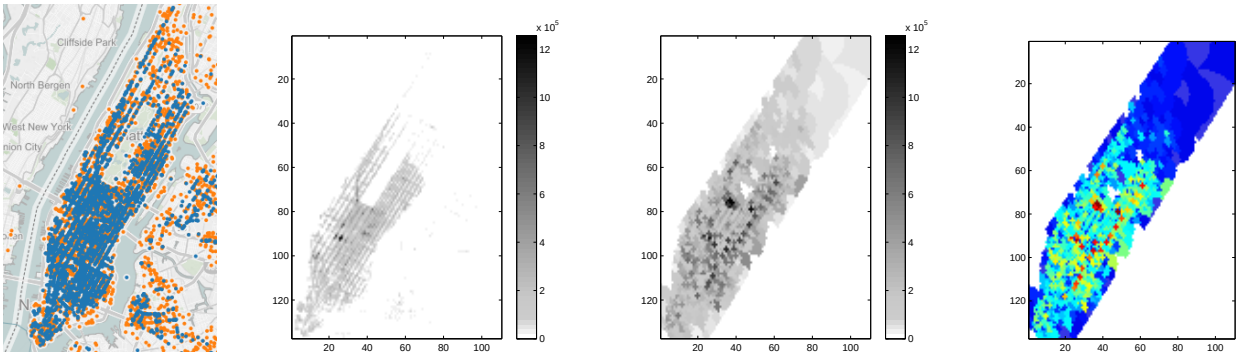


Figure 2.1: Noisy data vs. valid data.

Anomaly Detection: Liu et al. [67] proposed an algorithm to construct outlier causality trees based on spatio-temporal properties of detected outliers. Chawla et al. [16] proposed a two-step mining framework to infer the root causes of anomalies in road traffic data. Shafer et al. [106] proposed a novel approach using Kalman filtering as a state estimation model for mining large bursty time series and to find trends and anomalies. Xu et al. [131] identify and rank crossroads in a road network using a tripartite graph. Finally, the authors in [45] find all road segment outliers which have different traffic load than their expected values.

There is a rich literature of methods for anomaly detection in graphs [49]. For instance, Akoglu et al. [2] find anomalies using egonet features of the graph. According to [2], anomalies can be of different types including near-cliques, stars, heavy vicinities, and dominant heavy links. In another work, a non-negative residual matrix factorization method has been proposed to detect anomalies in graphs [122]. An ensemble of different methods to detect anomalies in dynamic graphs is proposed in [99]. A detailed survey on anomaly detection in graphs can be found in [3]. Furthermore, various methods used for anomaly detection in dynamic graphs have been reviewed in [98].

Gupta et al. [38] provided a comprehensive overview of outlier detection methods for various forms of temporal data including spatio-temporal data and temporal networks. Various studies in spatial and spatio-temporal outlier detection have been reported in the literature. For example, Liu et al. [68] proposed the first outlier detection framework for spatial categorical data with presenting a new metric, Pair Correlation Ratio (PCR) to measure the relevance among spatial objects. In another work, Chen et al. [17] proposed two algorithms to detect spatial outliers. Shekhar et al. [110] proposed a general framework to detect outliers in spatial datasets with underlying graph structure. Kou et al. [60] proposes two spatial outlier detection algorithms based on KNN graph. Chen et al. [18] presented issues in Generalized Local Statistical framework for spatial outlier detection methods. They also designed robust estimation and outlier detection methods based on this framework. Also, Wu et al. [130] proposed a novel spatio-temporal random effects model that outperforms traditional traffic volume prediction models such as spatio-temporal ARMA.



(a) (b) (c) (d)

Figure 2.2: (a) Map of Manhattan with pick-ups (blue) and drop-offs (orange) at 5am on March 3rd, 2013. (b) Total number of pick-ups and drop-offs for each block in grid. (c) Population of each cluster. (d) Clusters of blocks.

2.3 Preliminary Analysis

2.3.1 Dataset Description and Pre-processing

The dataset contains Yellow cab trips of NYC in 2013 (raw size ~ 45GB) which is publicly available[1]. Information such as pick-up and drop-off geographical coordinates as well as time, distance, and price of trips have been logged in this dataset. The total number of trips is 173,179,759. A pre-processing step has been performed to remove missing values and noisy data such as invalid geographical coordinates, loops (trips with the same pick-up and drop-offs), gifts (trips with zero traveled distance but with registered payment), and trips with no passenger(s). The portion of invalid data compared to the valid part is shown in Fig. 2.1.

Since Manhattan is one the most complex and highly populated urban areas in the world, we focused on Manhattan. For simplicity, we considered a rectangular area, where latitudes are bounded between 40.7 and 40.85, and longitudes are between -74.02 and -73.90. Total trips made within Manhattan after noise removal is 143,329,066 in 365 days.

2.3.2 City Decomposition

One simple and popular approach to split up the city into blocks is grid decomposition. We divide Manhattan into predetermined equal-sized blocks to logical blocks of Manhattan. However, due to the high number of vacant blocks and also behavioral similarity of adjacent ones, clustering adjacent blocks is recommended [15]. Various techniques have been used for

[1] http://www.andresmh.com/nyctaxitrips/

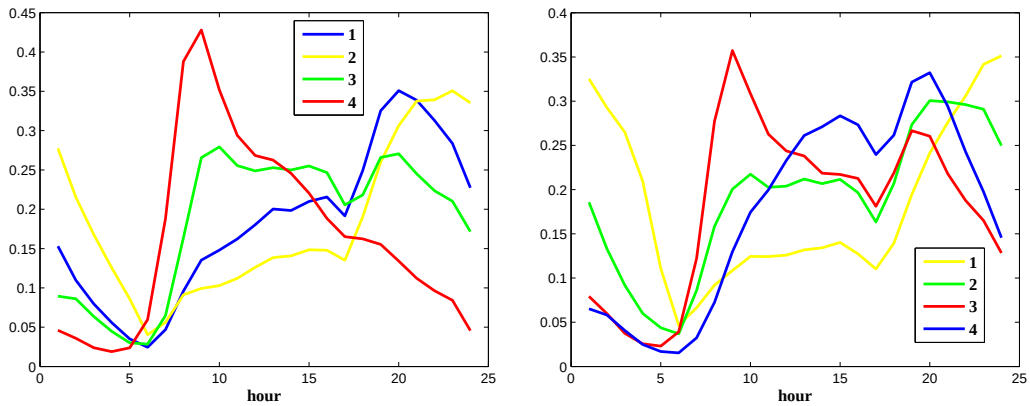
this purpose. As an example, Cao et al. [14] used a spatial clustering algorithm to analyze composition of cities in terms of their functional behavior. They used k-means via PCA to cluster individuals into groups. Here, we applied hierarchical spatial clustering on less populated blocks (blocks with less than 200 pick-ups/drop-offs per day). The number of blocks after clustering reduced from 15,070 to 1,204 clusters.

The population of trips in terms of number of pick-ups and drop-offs is illustrated in Fig. 2.2. Fig. 2.2(a) shows an example of the exact latitude and longitude of pickups and drop-offs in Manhattan at 5am on March 3rd, 2013. The total number of pickups and drop-offs for each block before clustering is shown in Fig. 2.2(b). Population of each group after clustering is shown in Fig. 2.2(c). For a better visualization, the differentiation between clusters is illustrated in Fig. 2.2(d). It should be noted that the total number of pick-ups and drop-offs are calculated over the year.

2.3.3 Location Characterization

Discovering functional regions (residential, business, etc.) or categorizing people (student, workers, etc.) is valuable for city planners to comprehend activities of individuals and decide on the placement of new infrastructures [139, 51, 81]. We calculate the average daily demand to find communities with similar daily activity. K-spectral centroid (KSC) clustering is a time series clustering method which deploys a similarity metric that is scale and shift invariant. We use KSC-clustering with initial clusters driven using k-means (k=4) on averaged pick-ups and drop-offs. An adaptive wavelet-based incremental approach of this algorithm is used for this purpose [132]. Clustering results for pick-up profiles are shown in Fig. 2.3(a) and the geographical areas related to each curve are shown in Fig. 2.4(a). The corresponding results for drop-offs are shown in Fig. 2.3(b) and Fig. 2.4(b). As these figures depict, yellow-colored areas are the ones with higher activities during the night (7pm-6am). The north side of Manhattan has higher activities in terms of pick-up demands while the southeastern part of Manhattan has higher activities in both pick-ups and drop-offs. Red colored areas have high demand in the morning (8am-10am) and red colored drop-off locations are indicators of the business areas of the city. Other than characterization of functionality of locations, these results are helpful for recommendation on where to pickup a passenger or to find the taxis.

The total number of trips per each month at each hour is shown in Fig. 2.5(a). Months are clustered according to their temperature values in 2013. Red, green, and blue indicate warm, mild, and cold temperatures, respectively. As this figure depicts, the demand during colder months is higher. According to the above observations, for simplicity, in the rest of the chapter we divide the 24 hours in a day into four 6-hour time slots.



(a) (b)

Figure 2.3: Clustering locations based on their daily profile at (a) pick-up, and (b) drop-off.

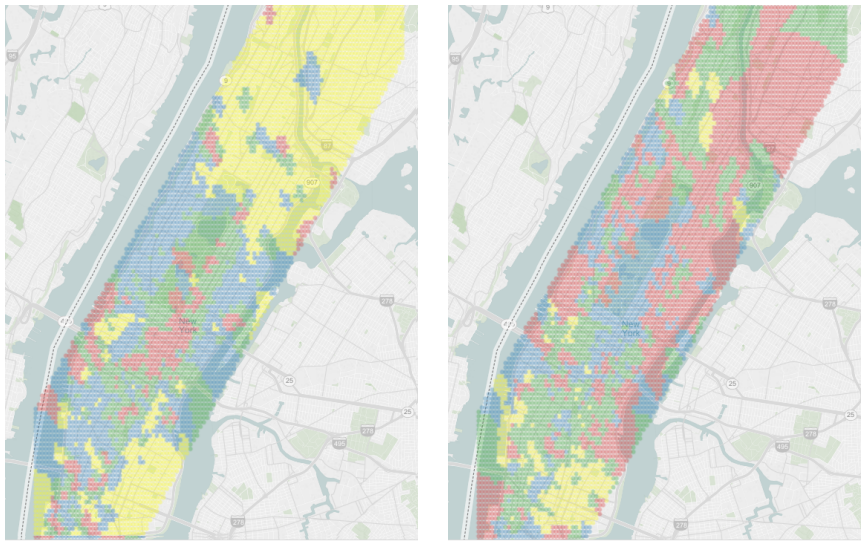
2.4 Traffic Flow Graph

In order to derive an abstract explanation of behavior of people, we create an average network graph of transportation flows. This average graph helps us to understand the normal behavior of taxi transportation in the city. Following the construction of such a graph, at each timestamp, we compare the traffic graph for that instant with the averaged graph to detect anomalies. Anomalies will thus denote areas that have characteristics distinct from the average graph. For these purposes, after extracting local and egonet features of the graph, we employed two graph mining methods to explain the dynamics of traffic flow. The first method is to extract network signatures of locations using power law relations (as proposed in [2]). The other method is to use role extraction methods to understand the structural behavior of nodes and to detect outliers by finding significant changes in role memberships.

2.4.1 Graph Model and Feature Extraction

In our graph model, we assume each node is a clustered area (from Section 2.3.2) and each edge represents the existence of at least one trip between two areas. Hence, the graph will be a directed one and the weight of an edge shows the number of trips. It should be noted that at different times, the dynamics of the graph changes in terms of the existence and weight of an edge, not in terms of the number of nodes.

Definition 1 ***Traffic flow graph:*** *Traffic flow graph of taxis at time t of day d is denoted by $\boldsymbol{G}_t^d(\boldsymbol{V}, \boldsymbol{E}_t^d)$, where $\boldsymbol{V} = \{v_1, v_2, ..v_n\}$ is a set of nodes. Each node, v_i, in this graph is a geographic area (clustered area) and each edge, e_{ij}, represents the trips from the ith area to the jth area.*



(a) (b)

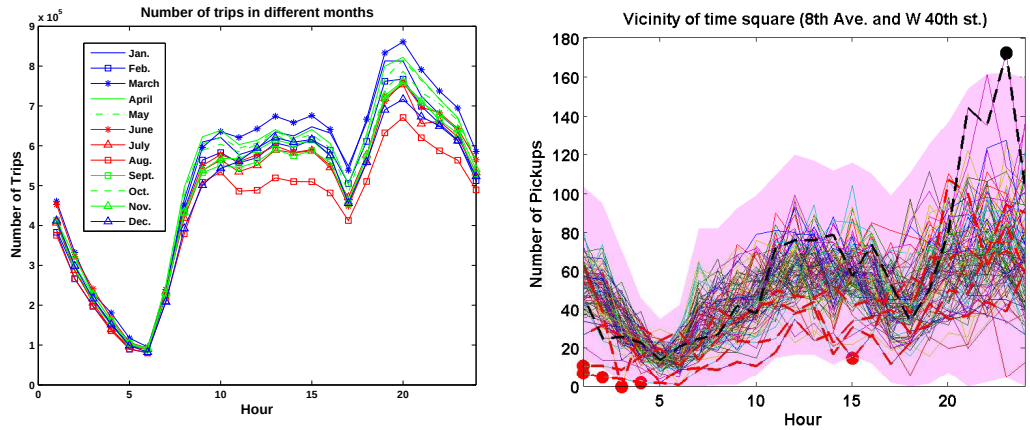
Figure 2.4: Clustered areas based on their daily profile on (a) pick-up, and (b) drop-off.

Each element of the underlying adjacency matrix (A), a_{ij}, is one if and only if there is an edge between the ith and j^{th} areas in $\boldsymbol{E}_t^d$. Each element of the weight matrix, w_{ij}, denotes the number of trips from v_i to v_j. As stated before, we look at 6-hour time ranges: (1) 1am-6am, (2) 7am-12pm, (3) 1pm-6pm, and (4) 7pm-12am. Hence, per day, d, depending on the nature of the day (weekend/weekday), we have four different graphs: $G_1^d, G_2^d, G_3^d, G_4^d$.

Local and Egonet Features

In order to extract signatures of the traffic network of taxis, for each node we extract the following local and egonet features:

- Degree In: The number of edges going into the node,
- Degree Out: The number of edges going out from the node,
- Weight In: The total weights of edges going into the node,
- Weight Out: The total weights of edges going out from the node,
- $DistIn_i$: The average geographical distance traveled to reach the node,
- $DistOut_i$: The average geographical distance traveled from the node,
- N_i: Number of nodes in egonet i,
- E_i: Number of edges in egonet i,



(a) (b)

Figure 2.5: (a) Number of trips at each month at each hour. Blue, green, and red colors represent cold, medium, and hot temperatures. (b) Example of traffic burst (black) and absenteeism (red) in the number of pick-ups in one location. Each curve represents one day of the year. The shaded pink area shows the normal range of variation.

- $\mathcal{W}_i$: Total weight of egonet i,
- λ_i: Principal eigenvalue of the weighted adjacency matrix of egonet i, and
- Clustering coefficient: the ratio of links that exists in egonet i divided by maximum possible number of links that could exist.

Egonet features are helpful in characterizing the topology and relationships between nodes in the graph. Akoglu et al. [2] find anomalies in a static graph using power law representations. According to their definition, anomalous nodes can have different types w.r.t their egonet features (near-clique, stars, heavy vicinities, and dominant heavy links). We deploy this approach to identify interesting and unusual locations that have such behaviors in their egonets. More details are provided in Section 2.5.

The run-time complexity depends on the graph representation. It can be shown that using an adjacency and weight matrix, some of graph features such as Degree In can be estimated in $O(n^2)$. However, other features such as clustering coefficient have higher complexity (i.e. $O(n^3)$).

2.4.2 Average Probabilistic Flow Graph

One might use a simple approach to look at locations individually to find the profile of each location and also to find occurrence of outliers. Fig. 2.5(b) shows such an example where traffic bursts (black dots) and absenteeism (red dots) occurred in terms of deviation in the

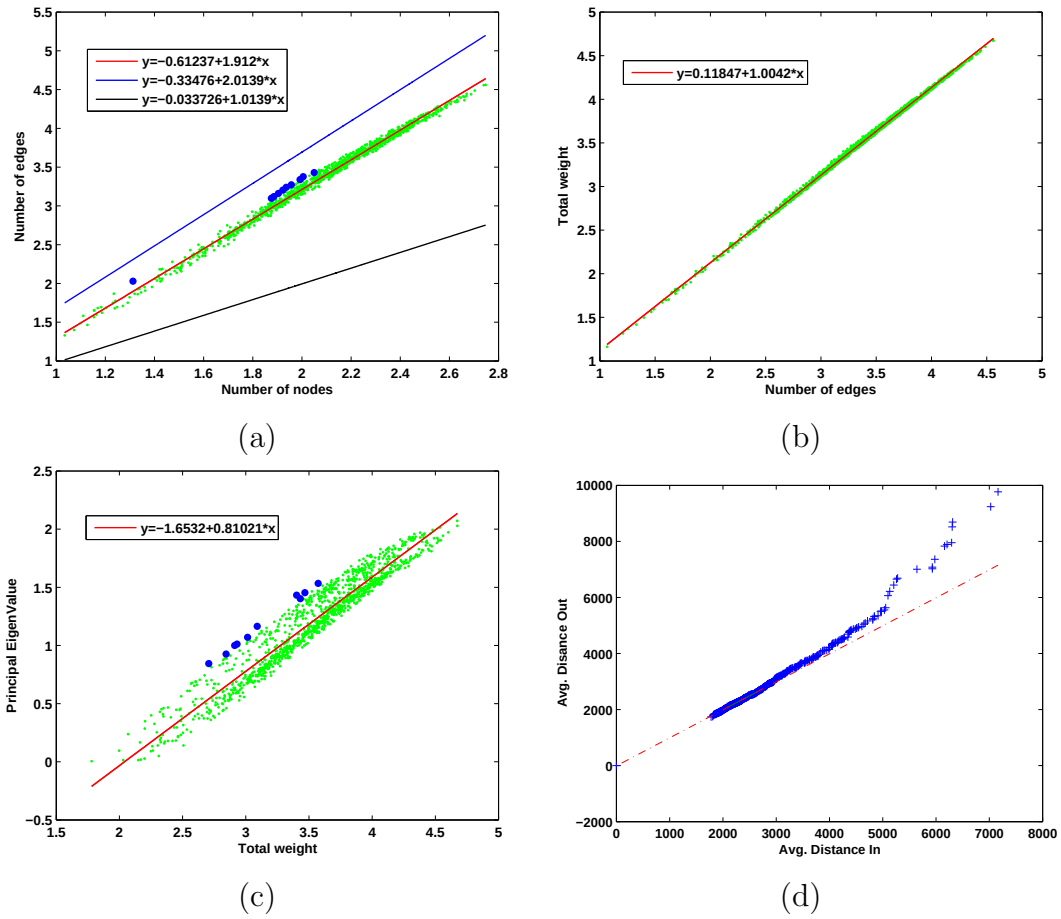


Figure 2.6: Illustration of relationships of features of the APF graph for an evening period (1pm-6pm) during weekdays. (a) E_i vs N_i, (b) E_i vs. $\mathcal{W}$, (c) $\mathcal{W}$ vs. λ_i, (d) Q-Q plot of $DistIn$ and $DistOut$. Green dots in figures (a) to (c) are nodes and blue dots represent outliers.

number of pick-ups in one location (vicinity of Times Square – 8th Ave and 40th St). The shaded pink area shows the normal range of variations in the number of pick-ups ($\mu \pm 3\sigma$). Any values outside this range can be labeled as an outlier. We refer to the values higher (or lower) than the normal range as bursts (respectively, absenteeism). However with this technique the relationships between different locations cannot be recovered.

The average graph of traffic flow will be an indicator of normal pattern of locations throughout the year. We calculate this graph using a probabilistic approach. Also this graph can be considered as a baseline model to identify anomalies at different times/days of the year.

Definition 2 ***Average Probabilistic Flow (APF) Graph:*** *A graph of taxi flows at time t averaged over a set of days, S, $|S| \leq 365$ is $\boldsymbol{G}_t^S(\boldsymbol{V}, \boldsymbol{E}_t^S)$, where $\boldsymbol{V} = \{v_1, v_2, ..v_n\}$ is a set of nodes. The edges $\boldsymbol{E}_t^S$ represent the probability of trips between nodes.*

In this definition, S is the set of days that is used for the construction of the APF graph. It should be mentioned that choosing the right set of days depends on our target and our data analytics model. As an example, while we can assume that S includes all the days of a whole year, one may differentiate between the weekends and weekdays. In the latter case, for each type of day, one separate set of days should be constructed. As another example, each month of a year may be analyzed separately and hence, we may construct 12 different sets. In our studies, S is chosen based on type of days (weekend/weekdays) in four different time slots (morning, afternoon, evening, night) throughout the year.

Since we aim to suppress the effect of specific events—where traffic flow happens during a few times with high ratio—the elements of the probabilistic adjacency matrix (A^S) are calculated as follows:

$$a_{ij}^S = \frac{1}{|S|} \sum_{d=1\,,d\in S}^{365} a_{ij}^d. \tag{2.1}$$

This equals the average number of days that have at least one trip from v_i to v_j and can be interpreted as the empirical probability of having an edge from i^{th} to j^{th} areas. Similarly, the elements of the weight matrix are calculated as follows:

$$w_{ij}^S = \frac{1}{|S|} \sum_{d=1\,,d\in S}^{365} w_{ij}^d. \tag{2.2}$$

This equals the total number of trips from v_i to v_j in S divided by the number of days ($|S|$) which can be interpreted as the expected number of daily trips from v_i to v_j. Since city mobility patterns differ between weekdays and weekends, we perform separate experiments on each category of days. It should be noted that for weekends $|S| = 104$ and for weekdays $|S| = 261$.

Calculating Local and Egonet Features

In a regular graph, $\boldsymbol{G}_t^d(\boldsymbol{V}, \boldsymbol{E}_t^d)$, we say v_j is in the egonet of v_i, if $e_{ij} \in \boldsymbol{E}_t^d$ or $e_{ji} \in \boldsymbol{E}_t^d$. However in the APF graph, we have to describe features in probabilistic terms. Hence, the probability of the existence of node v_j in egonet i depends on the probability of existence of e_{ij} and e_{ji}. Assuming that the presence of these two edges are independent, the following equation is used to calculate the probability of having node v_j in egonet i:

$$\begin{aligned} P_j^i &= Prob(v_j \in Egonet_i) \\ &= Prob(e_{ij} \in \mathbf{E}_t^d \cup e_{ji} \in \mathbf{E}_t^d) = a_{ij}^S + a_{ji}^S - a_{ij}^S a_{ji}^S. \end{aligned} \tag{2.3}$$

Recall that a^S_{ij} is the probability that an edge exists from v_i to v_j, i.e. $a^S_{ij} = Prob(e_{ij} \in \mathbf{E}^d_t)$. Then, the expected number of nodes in the egonet i can be determined as follows:

$$N_i = 1 + \sum_{k=1,k\neq i}^{n} P^i_k. \tag{2.4}$$

Note that the ith node always exists in egonet i (with probability of 1).

In order to calculate the expected number of edges in egonet i, we need to consider e_{kj}, if both j^{th} and k^{th} nodes are in egonet i. The probability of existence of both nodes in egonet i is $P^i_k P^i_j$. Hence, the expected number of edges in egonet i is

$$E_i = \sum_{k=1,k\neq i}^{n} (a^S_{ik} + a^S_{ki}) + \sum_{k=1,k\neq i}^{n} \left(\sum_{j=1,j\neq i,j\neq k}^{n} P^i_k P^i_j (a^S_{kj} + a^S_{jk}) \right)$$

where the first summation is the expected number of edges that are connected to v_i while the second summation represents the expected number of edges that are not connected to v_i. In a similar way, the expected total weight in egonet i is:

$$\mathcal{W}_i = \sum_{k=1,k\neq i}^{n} (w^S_{ik} + w^S_{ki}) + \sum_{k=1,k\neq i}^{n} \left(\sum_{j=1,j\neq i,j\neq k}^{n} P^i_k P^i_j (w^S_{kj} + w^S_{jk}) \right).$$

The expected principal eigenvalue of egonet i is derived from the weighted adjacency matrix of egonet. Let us assume that $\mathbf{\Omega}^S_i$ is the weight matrix of egonet i, and $\{v_j, v_k\} \in Egonet_i$. Each element of the weight matrix of egonet i is calculated as follows:

$$\Omega^S_{i_{jk}} = \begin{cases} w^S_{ji} & k = i, j \neq i \\ w^S_{ik} & j = i, k \neq i \\ P^i_k P^i_j w^S_{jk} & j \neq i, k \neq i, k \neq j \\ 0 & O.W. \end{cases} \tag{2.5}$$

where $P^i_k P^i_j$ is the probability that both of the nodes v_k and v_j are in egonet i.

The average geographical distance originating from v_i, $DistOut_i$, and the average geographical distance going into v_i, $DistIn_i$, are calculated as follows:

$$\begin{aligned} DistOut_i &= \frac{1}{\sum_{j=1}^{n} a_{ij}} \sum_{j=1}^{n} a_{ij} dist(i,j), \\ DistIn_i &= \frac{1}{\sum_{j=1}^{n} a_{ji}} \sum_{j=1}^{n} a_{ji} dist(j,i), \end{aligned} \tag{2.6}$$

where $dist(j,i)$ is the distance between v_i and v_j, and $dist(j,i) = dist(i,j)$. Note that the averages of Eq. 2.6 are weighted averages where higher weights are given to those edges that

have higher probability of existence in the average graph. It should be noted that in this study, Euclidean distance is used to measure the distances although other distance metric such as Manhattan distance can be used.

The run-time complexity of feature extraction for average flow graph is $O(n^3)$.

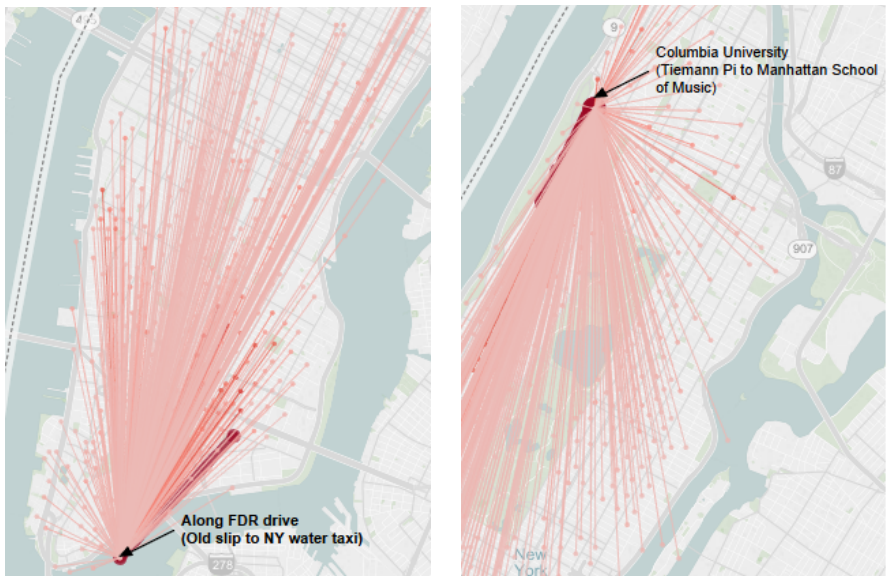
2.5 Behavior Analysis using the Average Probabilistic Flow Graph

The egonet features driven from a graph can identify different behavioral patterns of each node [2]. In what follows, we deployed an anomaly detection method based on egonet features [2] on eight APF graphs (four time periods during weekends and four time periods during weekdays) to find locations of interest and investigate their behavior. Since we perform our experiment on an average graph of transportation over a year, outliers detected using this method do not necessarily imply anomalous locations. In fact, this method reveals a set of locations with uncommon features that made them different from the rest of locations. Similar to [2], we analyze the following three pairs of features:

(1) E_i vs N_i: Comparing the number of edges with the number of nodes in each egonet is helpful in detecting near-cliques and stars. According to [2], the number of nodes and number of edges of egonets follow a power law ($E_i \propto N_i^{\alpha}, 1 \leq \alpha \leq 2$) where in our experiments for APF graphs, α ranges between 1.716 and 1.94. This range of variation for α indicates that most of the nodes have a near-clique pattern. The logarithmic scale for one of the APF graphs (1pm-6pm in weekdays) is shown in Fig. 2.6(a). The red line shows the least square error fit on data. Also blue and black lines have the slope of 2 (cliques) and 1 (stars), respectively.

(2) $\mathcal{W}_i$ vs E_i: Comparing the total weight with the number of edges in each egonet is helpful in detecting heavy vicinities. The total weight and number of edges follow a power law ($\mathcal{W}_i \propto E_i^{\beta}, \beta \geq 1$) [2]. In our experiments for APF graphs, β ranged upto 1.023 which reveals that no heavy vicinity node is observed in the traffic flow graph. As Fig. 2.6(b) depicts, all nodes have similar behavior and no particular node deviates from the fitting line.

(3) λ_i vs $\mathcal{W}_i$: Comparing the principal eigenvalue of weighted adjacency matrix with total weight is helpful in detecting dominant pairs (strongly connected pair of nodes). According to [2], the relationship between these two features follows a power law ($\lambda_i \propto \mathcal{W}_i^{\nu}, 0.5 \leq \nu \leq 1$) where smaller values of ν indicate uniform distribution of weights while larger values indicate the existence of dominant edges in egonet. In our experiment with APF graphs, ν ranged from 0.766 to 0.906 which means most of the nodes have dominant pairs in their egonets. Fig. 2.6(c) shows an example where most of the nodes have near-heavy edges (ν is close to one). Blue nodes in this figure shows egonets that deviate from the fitting line.



(a) (b)

Figure 2.7: Illustration of a dominant pair (dark red link) in two locations: (a) Along FDR drive during 1am-6am on weekends, (b) Columbia University during 1pm-12am on weekends.

2.5.1 Locations of Interest

Typically, nodes that deviate from the fitting line are considered as outliers. As we stated before, because we are only studying APF graphs the outliers found by this method should be considered as locations of interest since their patterns are unique (compared to the rest of Manhattan). In Fig. 2.6(a) and Fig. 2.6(c), blue circles show the top ten nodes with highest outlier score. Similar to [2], the outlier score for anomaly detection is calculated as a summation of normalized local outlier factor (LOF) and d_f, where d_f is a distance to the fitting line ($y = Cx^{\theta}$) which is calculated as follows:

$$d_f(i) = \frac{max(y_i, Cx_i^{\theta})}{min(y_i, Cx_i^{\theta})} * log(|\frac{(y_i - Cx_i^{\theta})}{y_i}| + 1).$$

Note that d_f represents the distance between normal behavior (Cx^{θ}) and the observed value(y) in a normalized logarithmic scale. Higher deviations from the normal behavior (Cx^{θ}) result in larger values of d_f.

2.5.2 Discussion on Interesting Points

In what follows, we mention a few samples of discovered interesting locations that either have high feature values or deviate from the fitting line. It is interesting to know that most of the top attractions of NYC such as the Empire State Building, Rockefeller Center, and the Metropolitan museum of art are *not* included in our list. This suggests that people perhaps

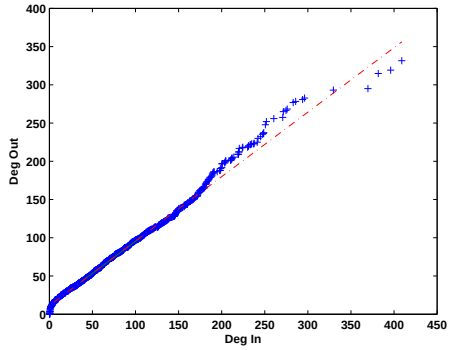


Figure 2.8: Q-Q plot of Degree-In and Degree-Out of APF graph for evening(1pm-6pm) of weekdays.

use other types of transportation (e.g. subway) to travel to these attractions, or they chose nearby locations as their pick-up and drop-off points.

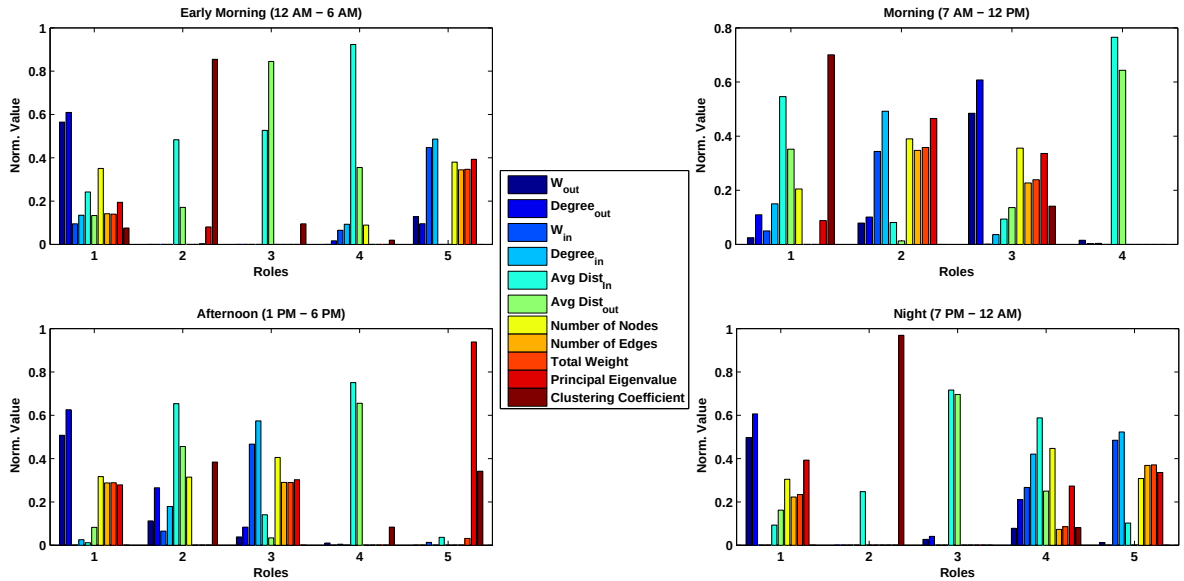
Fig. 2.6(d) illustrates Q-Q plots of geographical distances (In vs. Out) in a sample APF graph. As this figure depicts, the distribution of In and Out distances are the same. This is also true for the In and Out degrees of nodes (Fig. 2.8).

Some locations such as the New York Presbyterian Hospital (covering Fort Washington Ave, from W 161st St to W 173rd St) have high geographical distance to their neighborhoods, indicating that trips to/from this hospital are longer compared to others. High Bridge Park and Claremont Park have similar behavior.

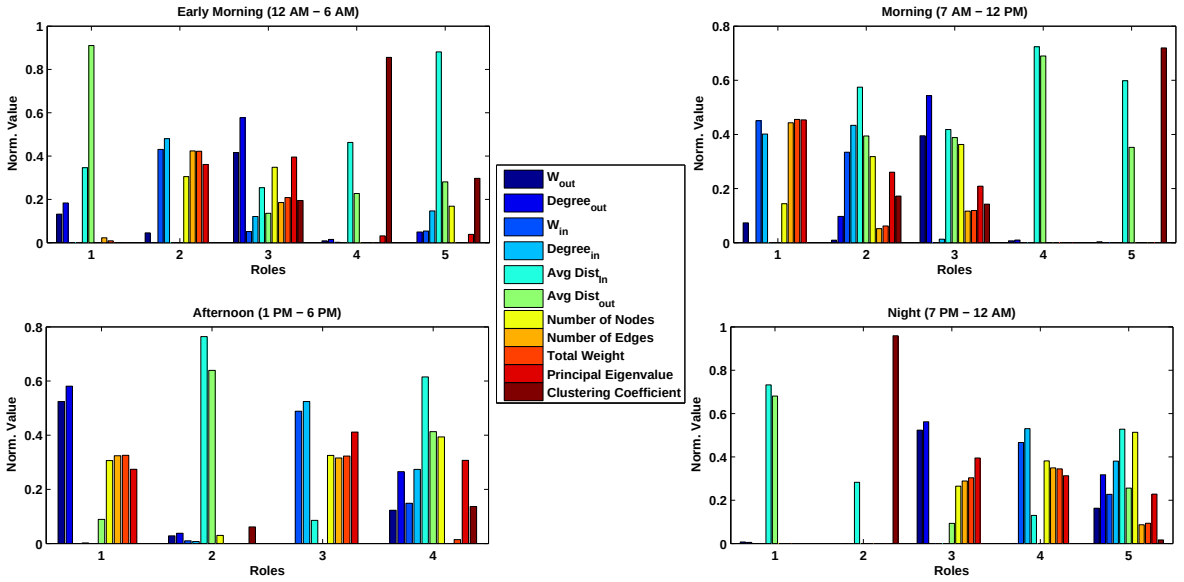
Two examples of discovered dominant pairs are shown in Fig. 2.7. The dark red colored links indicate dominant edges.

The area between Avenue C, E 5th St, and E 3rd St has a near-clique pattern during 7pm-12am. On the other hand, MalcolmX Blvd from W 137th St to W 147th St, during 7am-12pm on weekends has a star pattern.

The egonet features of the following locations have high value which made them hotspots: Penn Station (7th Ave and W 31st St) during 7am-6pm on weekends has a high number of nodes, edges, weights, and large eigenvalues. The area covering Penn station (SE), US post office, and intersection of 8th Ave and W 31st St has a high number of incoming and low outgoing trips from 7am-12am. The area covering Chelsea market and Google HQ, in 1am-6am, has a high number of nodes, edges, total weights, and large eigenvalue. The area in Central park (SW), Columbus Circle, 7th Ave, 55th St, and 8th Ave from 7am-12am in weekdays has a high number of nodes, edges, weight, and large eigenvalue.



(a)



(b)

Figure 2.9: Extracted roles at different time ranges for (a) weekdays, and (b) weekends.

2.6 Role Extraction In Graph

Role extraction (RolX) is a non-parametric, scalable and efficient approach that finds similar structural behaviors and patterns of the graph [41, 101, 102]. The three main steps of this method are feature extraction, feature grouping, and model selection. First, features (global,

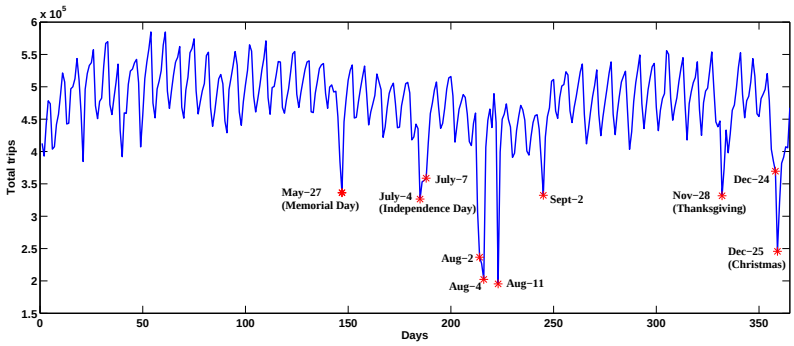


Figure 2.10: Total trips traveled at each day. Days with high variations represent anomalies.

local, egonet) for each node must be extracted to create a $n \times f$ node-feature matrix X. The next step is to generate a rank r approximation of X using non-negative matrix factorization (NMF). NMF is used to simplify interpretation of roles and memberships by creating non-negative low rank matrices (H_t^S and F_t^S) as follows:

$$H_t^S, F_t^S = \arg\min_{H,F} \frac{1}{2}||X_t^S - HF||_F^2, \; s.t. \, H \geq 0, \; F \geq 0,$$

where X_t^S is the node-feature matrix for the APF graph, G_t^S, and $||.||_F$ is Frobenius norm. Membership of a node to each role can be estimated using the rows of $H_{n\times r}$ while columns of $F_{r\times f}$ are used to determine the relationship between the role membership and feature values. Since NMF generally results in sparse representations of the original matrix, it is a better candidate for role extraction compared to other factorization methods. The third step is to select the number of roles, r, using the minimum description length (MDL) criterion to compress X. In other words, the objective is to minimize the description length L which is equal to the summation of the coding cost and the cost of model description. MDL selects the number of behavioral roles, r, such that the model complexity (number of bits) and model errors are balanced:

$$L = r(n+f) + (-\frac{1}{2\sigma^2}||X_t^S - H_t^S F_t^S||_F^2),$$

where σ^2 is variance of X_t^S. Details can be found in [41]. The running time complexity of RolX is linear on the number of edges ($\boldsymbol{E}_t$), and specifically is $O(f\boldsymbol{E}_t + nfr)$ [41].

The number of detected roles as well as the normalized feature values at each time of weekdays and weekends are shown in Fig. 2.9. As these figures illustrate, some features were significant in defining the extracted roles such as degrees, weights, and geographical distances.

2.6.1 Spatio-temporal Anomaly Detection

The total number of trips per day is shown in Fig. 2.10. As this figure depicts, the number of trips decreased significantly in specific days with most of them being holidays. This

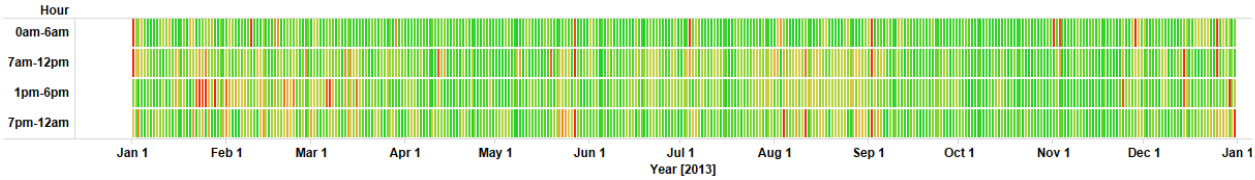


Figure 2.11: Outliers for different times of a day during a year. Red colors represent higher score of outliers.

might be due to the decrease in the number of available taxi drivers rather than decrease in demand (the present analysis and data availability cannot make this distinction). We use the extracted roles from the APF graph to detect anomalies during the whole year. Several techniques have been proposed to detect changes in dynamic networks. As an example, [104] defined an event as a subset of nodes in the network that are close to each other and have high activity levels. However, Rossi et al. [101] track node memberships over time (temporal dependencies of roles and nodes) to discover anomalies. Network dynamics (structural patterns in network over time) can be analyzed using this method. In another work, Rossi et al. [102] proposed dynamic behavioral mixed-membership model to capture roles of the nodes. They identify dynamic patterns in node behaviors and then using prediction on future structural changes in the node, they identify unusual changes in behavior transitions.

In this chapter, we use roles extracted from APF graphs and compare the roles with the profile of each day to discover graphs with high variations compared to the average behavior. Therefore, for each graph, G_t^d, we calculate the node-feature matrix X_t^d. Let us assume that based on the APF graph, the role R_i has been assigned to the ith node (i.e. $R_i = \arg\min_k(dist(F^S_{t_{k,:}}, X^d_{t_{i,:}}))$ [2]. Then we calculate the total distance of graph to the assigned roles in feature space as follows:

$$\Delta_t^d = \sum_{i=1}^{n} ||X^d_{t_{i,:}} - F^S_{t_{R_i,:}}||_2,$$

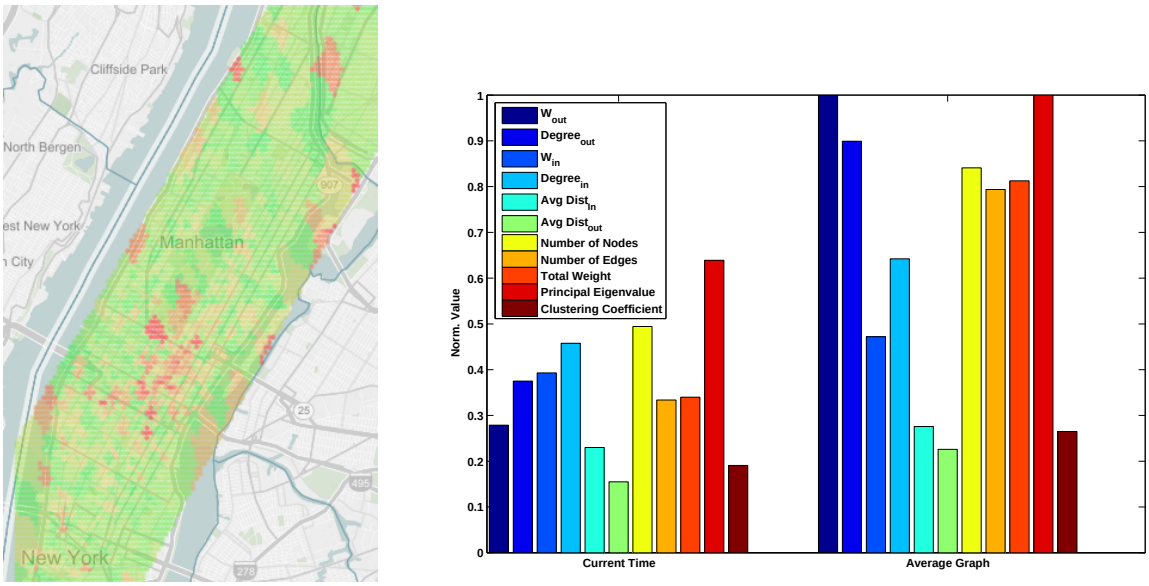
where n is total number of nodes.

Since we are seeking graphs with high variations, we extract graphs with variations outside the normal range. For this purpose, we calculate the following average and standard deviations over all days at time t:

$$\mu_t = \frac{\sum_{d=1,d\in|S|}^{365} \Delta_t^d}{|S|}, \; \sigma_t = \sqrt{\frac{1}{|S|-1} \sum_{d=1,d\in|S|}^{365} (\Delta_t^d - \mu_t)^2},$$

where $|S|$ is number of days. Based on our assumption, an anomaly will occur if the changes in the graph deviates more than a predefined threshold, $\mu \pm \tau\sigma$. Fig. 2.11 shows the variation

[2] $X^d_{t_{i,:}}$ is the ith row of matrix X_t^d



(a) (b)

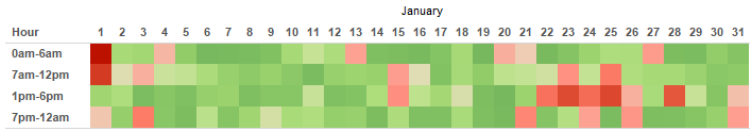
Figure 2.12: (a) Degree of deviation from assigned roles for Labor day in late morning (7am-12pm). (b) Comparison of assigned role of (Park Ave and Lexington Ave and 62nd St and 60th St) in APF graph and features of Labor Day in late morning.

degree of each part of the day compared to the original assigned roles. Here, we assumed that τ is equal to 3. Red color shows high variations while green color shows low variations. The result is compatible with Fig. 2.10. Table 2.6.1 illustrates the amount of variations for major holidays and cultural events. As an example, the last day of the year (1pm-12am) and first day of the year (1am-12pm) are the ones that have medium to high variations. This result is helpful to understand when the normal behavior of movements in terms of taxi trips changes significantly. As an example, we looked at what happens at each location during Labor day (7am-12pm). The variation at each location is show in Fig. 2.12(a). Fig. 2.12(b) shows the difference of features of APF graph (assigned role) vs. that on Labor day for one specific location. The results are helpful for decision-makers and traffic management. Regarding the run-time complexity, if we have f features, Δ_t^d can be calculated in $O(nf)$ and μ and σ can be calculated in $O(|S|)$.

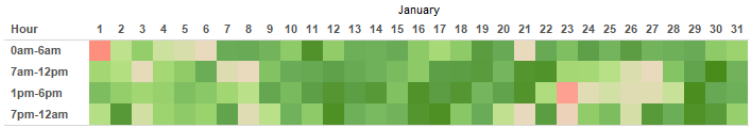
As we mentioned before, the average flow graph is constructed over a set of days, S, and in our studies, this set is chosen based on type of days (weekend/weekdays) in four different time slots (morning, afternoon, evening, night) throughout the year. However, outlier detection can be performed on a different timespan. In that case, results will be more specified based on the types of activities in that particular set. For example, Figure 2.13 (a) and (b) represent the difference in terms of detected outliers when S is set to one year vs. one month, respectively. In both cases, 8 different sets were chosen to cover weekends/weekdays and four time slots in each day. As it is illustrated in Fig. 2.13 (b), in January, the holidays will have higher effect on APF graph and hence, fewer outliers have been detected. As figure

Table 2.1: Federal, religious, and cultural holidays with their deviation degrees

Name	1-6	7-12	1-6	7-12
New Year's Day (1/1)	7.1σ	3.5σ	-.7σ	1.6σ
Inauguration Day. (1/20)	1.7σ	-1.1σ	.σ	.1σ
Martin Luther King, Jr. Day (1/21)	1.6σ	1.2σ	.7σ	2.1σ
groundhog day (2/2)	1.3σ	.7σ	1.8σ	1.4σ
Chinese New Year (2/10)	.3σ	.3σ	.7σ	-.5σ
Lincoln BD (2/12)	-.2σ	1.5σ	1.8σ	1.5σ
Valentine's Day (2/14)	.1σ	.6σ	.8σ	1.2σ
George Washington's Birthday (2/18)	2.2σ	.9σ	.9σ	1.2σ
Mothers day (5/12)	-.9σ	-.2σ	-1.σ	-.6σ
Memorial Day (5/27)	2.8σ	4.1σ	-1.σ	3.1σ
Independence Day (7/4)	5.1σ	2.1σ	-.5σ	-.7σ
Eid al-Fitr (8/8)	.2σ	-1.5σ	-1.7σ	-1.9σ
Labor Day (9/2)	5.7σ	3.8σ	-.4σ	4.3σ
Columbus Day (10/14)	-.2σ	-.3σ	-1.4σ	-.2σ
Eid al-Adha (10/15)	-.2σ	-1.1σ	-1.2σ	-.9σ
Halloween 10/31	.5σ	-.6σ	.6σ	-.3σ
Diwali (11/3)	2.7σ	1.9σ	.3σ	-.6σ
Veterans Day (11/11)	-.3σ	-.4σ	.1σ	-.5σ
Thanksgiving Day, (11/28)	3.6σ	1.6σ	-.8σ	-.6σ
Christmas day (12/25)	6.σ	3.4σ	-.5σ	2.σ
New Year's Eve (12/31)	.7σ	.3σ	1.8σ	5.3σ



(a)



(b)

Figure 2.13: (a) Outliers discovered in January when APF graph is created using one year. (b) Outliers discovered in January when APF graph is created using January data.

2.13 (a) indicates, creating the average flow graph on a larger set (the whole year) will lead to more accurate results since the detected outliers would be the ones that deviate from the average behavior of a typical day. In fact, by looking at a small set of days with large number of anomalous outliers, the average flow graph will be created using those anomalies and hence, true outliers cannot be discovered using the proposed method.

2.6.2 Performance Measurement and Comparison

We performed various experiments to measure the performance of the proposed anomaly detection method. As the performance metrics, we measured precision, recall, and accuracy. Furthermore, for comparison purposes we implemented a graph similarity based outlier detection algorithm [38, 89] using snapshot of weight matrix at each time. In this graph similarity based method, for each category of days (i.e. weekends and weekdays) we first calculate the average graph of the year. Then, for each day, we measure the similarity distance between the day's snapshot graph and the corresponding average graph. As we defined in Eq. 2.2, for all the days that belong to set S, $w^S_{t,ij}$ is the expected number of trips that occur during the time frame t, from v_i to v_j. Let us assume that for a specific day, d, the actual number of trips in time frame t from v_i to v_j is $w^d_{t,ij}$. Then, the weight-distance [89] between day d and the APF graph is calculated as follows:

$$\Delta_t^{sim,d} = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\left(w^d_{t,ij} - w^S_{t,ij}\right)^2}, \tag{2.7}$$

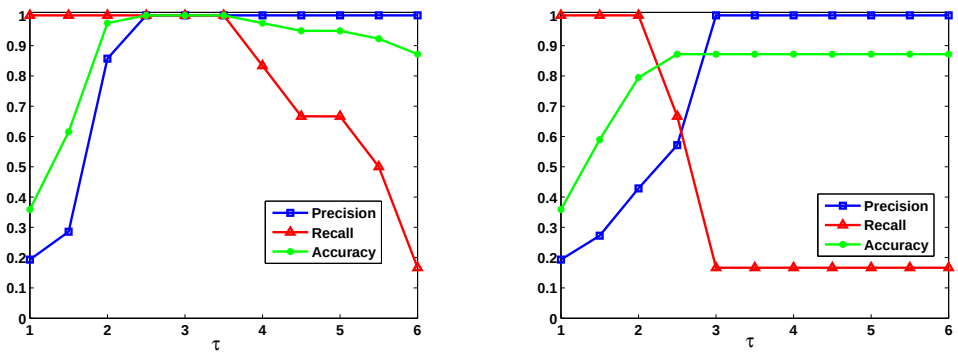
where n is the number of nodes. We calculate the following average and standard deviations over all days at time t:

$$\mu_t^{sim} = \frac{\sum_{d=1,d\in|S|}^{365}\Delta_t^{sim,d}}{|S|}\ ,\ \sigma_t^{sim} = \sqrt{\frac{1}{|S|-1}\sum_{d=1,d\in|S|}^{365}(\Delta_t^{sim,d} - \mu_t^{sim})^2},$$

where $|S|$ is number of days. Then, as we have assumed before, an anomaly will occur if the changes in the graph deviates more than a predefined threshold, $\mu^{sim} \pm \tau\sigma^{sim}$. Regarding the run-time complexity of this method, $\Delta_t^{sim,d}$ can be calculated in $O(n^2)$ and μ and σ can be calculated in $O(|S|)$.

In order to measure the performance of the anomaly detection algorithms, we need a ground-truth, i.e. set of days with known anomalous behavior as well as normal days. For this purpose, we picked 6 days with known anomalous behavior including New Year's Day, Memorial Day, Independence Day, Labor Day, Thanksgiving Day, and Christmas Day. We also included 3 days before and after of these days to our ground truth as the days with normal behavior.

In this section, we change the value of τ from 1 to 6 with step 0.5 and measured the value of recall, precision, and accuracy on the ground-truth set. Since for each particular day (and



(a) (b)

Figure 2.14: Comparison of precision, recall, and accuracy with respect to τ in APF graph (a) versus Graph similarity distance approach (b).

for the APF Graph) we partitioned the day to four 6-hour sections, for each algorithm, we have four metrics. in other words, Δ_t^d and $\Delta_t^{sim,d}$ for day d are measured for four different time periods, t, including 12am-6am, 7am-12pm, 1pm-6pm, and 7pm-12am. To determine that if a particular day is an anomaly, we used the maximum value of measured metrics over four time slots as follows:

$$\Delta_{max}^d = \max\left\{\Delta_{12^{AM}-6^{AM}}^d, \Delta_{6^{AM}-12^{PM}}^d, \Delta_{12^{PM}-6^{PM}}^d, \Delta_{6^{PM}-12^{AM}}^d\right\}$$
$$\Delta_{max}^{sim,d} = \max\left\{\Delta_{12^{AM}-6^{AM}}^{sim,d}, \Delta_{6^{AM}-12^{PM}}^{sim,d}, \Delta_{12^{PM}-6^{PM}}^{sim,d}, \Delta_{6^{PM}-12^{AM}}^{sim,d}\right\}$$

Results are illustrated in Fig. 2.14. As this figure represents, in traditional method, as τ increases, recall decays faster comparing to the proposed method. Also, for the proposed method there exists a range of values of τ that results in perfect accuracy, precision, and recall (i.e. $2 \leq \tau \leq 4$).

2.7 Discussion

In this chapter, we applied graph mining approaches to understand the dynamic behavior of taxi trips in a highly populated city. For this purpose, using power-law relationships of egonet features and role extraction using non-negative matrix factorization, we discovered locations of interest as well as outlier days (and locations) at different times. Event prediction methods using the APF graph and utilizing this approach to recommend the placement of new infrastructure are possible directions of future work.

Chapter 3

Characterizing Energy Systems

3.1 Introduction

The analysis of large scale data logged from complex systems, such as microgrids, often entails the discovery of invariants capturing functional as well as operational relationships underlying such large systems. The size and complexity of such modern devices have become a burden for administrators in fault detection and localization [35, 106]. These challenges inspired us to characterize and track anomalies in energy systems by correlating all monitored data across the system. According to [46], "*detecting anomalies that occur only within individual variables is often trivial, while detecting correlation anomalies is much harder and is practically important in fault analysis of complicated dynamic systems*". In a complex system, such as a smart grid (Fig.3.1), while some of the relationships between time series can be directly observed, other mutual dependencies are significantly complex to extract computationally.

A typical system may include tens of time series with hundreds of mutual dependencies, where a large number of them are not directly observable. In the past, researchers have tried to infer existing linear relations using regression models [48] or by harnessing the structure of causal networks [6]. However, due to the complexity of modern systems, we must go beyond direct linear correlations in understanding them. In this chapter, we aim to use a more realistic approach to discover hidden patterns and indirect relationships among devices by employing latent variables in regression models. Specifically, we harness hidden factors derived by factor analysis and use them in regression models. We perform various experiments on synthetic and real datasets. Furthermore, we use graph representations for better visualization of relationships which aids in discovering system-wide anomalies. Results show that the use of invariants derived with latent factors helps us to monitor large scale complex systems and discover outliers more precisely. We also propose a ranking method to score system-wide anomalies.

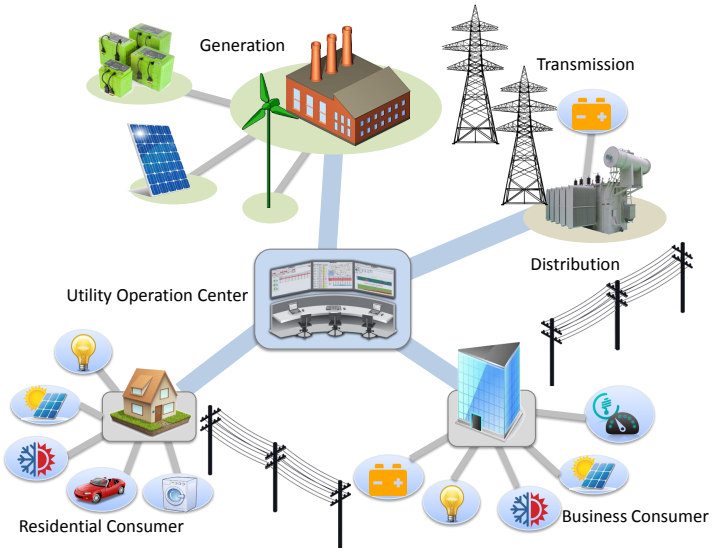


Figure 3.1: A typical example of a smart grid.

Our key contributions are thus:

- Proposing latent factor analysis regression to reveal hidden correlations among time series in a system.
- Summarizing the discovered invariants into an invariant graph of the system.
- Detecting system outliers based on the change in the graph of invariants and ranking time series for fault localization.

Our work, invariant detection, is different from traditional methods such as outlier mining or frequent episode mining. In invariant detection, we observe the hidden structure of relationships between observed variables and illustrate them using invariant graph. As a matter of fact, due to the complex nature of these systems and existence of a large number of interacting signals and components, the dynamic behavior of the system cannot be easily revealed by the traditional methods. On the other hand, for efficient discovery of faults and finding the culprit, one needs to understand the correlations that exist between various parts of the system in steady state. Using invariant analysis we can model the underlying relationships to characterize such systems precisely.

3.2 Background

The high complexity of modern distributed cyber-physical systems urges us to enhance the self-management capabilities of these systems. Cyber-physical systems such as microgrid systems have a high degree of heterogeneity (in terms of shape, trend, and periodicity) that

requires us to have a general tool to profile a variety of behaviors. Moreover, due to the nature of these systems, we may observe abrupt regime changes, seasonal patterns, and pairwise relationships among time series [28]. Ding et al. proposed an ensemble of different approaches to tackle these problems in [28]. As stated in [109], traditional computational techniques cannot be used to model complex cyber-physical systems for data analytic purposes in a straightforward manner. There have been multiple research efforts to model complex dynamic systems such as inferring/visualizing the input-output relationships or predicting state switches/changes [109].

Guofei et al. [47] proposed a concept named *flow intensity* and used the ARX (autoregressive exogenous) model to quantify the relationship between each pair of flow intensities. If such a relationship holds all the time, they are considered as invariants of the underlying system. This model has been successful in characterizing complex systems and in supporting different system management tasks such as fault detection and localization. However, one of the main disadvantages of this method, as cited in [35], is that the complexity of algorithm in order to find all invariants is high. In this model, they look at two flow intensities (timeseries) where one of them is considered as input and the other one as output signal. Note that the differentiation of input and output time series are unknown and such labeling can occur only after examining both directions and evaluating which assignments lead to higher scores. The ARX model posits the following relationship between two flow intensities of y (output) and x (input):

$$y(t) + a_1 y(t-1) + ... + a_u y(t-u) = b_0 x(t-l) + ... + b_v x(t-l-v) \quad (3.1)$$

where u, v, and l are the order of the model and determine the number of previous steps that are affecting the current output. a_i's and b_j's are coefficient parameters that reflect how strongly a previous step is affecting the current output. Equation 3.1 can be solved using a least squares method (LSM) and the fitness score will indicate whether the model fits the observed data appropriately [47].

3.3 Problem Formulation

Let us assume that we have observed a set of n time series, $\mathcal{D} = \{x_1(t), \ldots, x_n(t)\}$, measured at various points in one or more cyber-physical systems. For a time series $x_i(t)$, we represent the vector of samples at time steps t_k, ..., t_{k+w} as follows:

$$\mathcal{X}_i^{k:k+w} = [x_i(t_k), x_i(t_{k+1}) \ldots, x_i(t_{k+w})]^T . \quad (3.2)$$

Furthermore, we use X_i to represent the time series $x_i(t)$ as a random variable. In other words, $x_i(t)$ is a time series whose samples are drawn from a random distribution represented by random variable X_i.

In any type of cyber-physical system, there are various correlations and inter-dependencies among time series. In large cyber-physical systems, having sufficient level of knowledge about these inter-dependencies is crucial to preform accurate system management tasks. In the following definition, we formally define what we mean by dependency between two time series.

Definition 3 ***Approximate Dependency:*** *At time step t_m, time series $x_j(t) \in \mathcal{D}$ approximately depends on $x_i(t) \in \mathcal{D}$, if and only if, there exists a function $f : \mathbb{R} \to \mathbb{R}$ that for appropriately small $\epsilon > 0$:*

$$\hat{x}_j(t_m) = f(\mathcal{X}_j^{1:m-1}, \mathcal{X}_i^{1:m}) \tag{3.3}$$

and

$$|x_j(t_m) - \hat{x}_j(t_m)| < \epsilon. \tag{3.4}$$

We depict this dependency by $x_j(t) \underset{\epsilon}{\rightarrow} x_i(t)\Big|_{t_m}$.

When the dependency between two time series does not change over time, we say that these two time series are system-invariants.

Definition 4 ***System Invariants:*** *Two time series, $x_j(t) \in \mathcal{D}$ and $x_i(t) \in \mathcal{D}$, are system-invariant up to time T within range of ϵ if and only if at least one of the following rules satisfied:*

$$\exists f : \mathbb{R} \to \mathbb{R} \quad \textit{and} \quad \forall 0 \leq t \leq T \quad : \quad x_j(t) \underset{\epsilon}{\rightarrow} x_i(t)\Big|_{0 \leq t \leq T}$$

or

$$\exists f : \mathbb{R} \to \mathbb{R} \quad \textit{and} \quad \forall 0 \leq t \leq T \quad : \quad x_i(t) \underset{\epsilon}{\rightarrow} x_j(t)\Big|_{0 \leq t \leq T}.$$

We show invariant time series by $x_i(t) \overset{\epsilon}{\leftrightharpoons} x_j(t)$.

Based on the nature of the system, dependencies between time series can be linear or non-linear and this is modeled by the function f. In complex cyber-physical systems, when we have a large number of time series, it is appropriate to represent the invariants in the form of a graph.

Definition 5 ***Invariant Graph:*** *Graph $G = (V, E)$, with the set of vertices $V = \{v_1, \ldots, v_n\}$ and the set of edges $E = \{e_1, \ldots, e_m\}$, is called an invariant graph of a system with observed time series $\mathcal{D} = \{x_1(t), \ldots, x_n(t)\}$, where $e = (v_i, v_j) \in E$ if and only if $x_i(t) \overset{\epsilon}{\leftrightharpoons} x_j(t)$.*

From Definition 5, it is obvious that the vertex v_i is equivalent to the time series $x_i(t)$. It should be noted that system invariants and invariant graph represent features of a system under its normal condition. However, in the presence of anomalies, when the behavior of system deviates from its normal condition, these dependencies may disappear. In other words, while two times series, $x_i(t)$ and $x_j(t)$, may be invariant under normal conditions, the invariant feature may not hold when an anomaly condition appears in the system.

Definition 6 ***Broken Invariants:*** *We say that system invariant $x_i(t) \overset{\epsilon}{\leftrightharpoons} x_j(t)$ is broken at time $T = t_m$, if and only if, time series $x_i(t)$ and $x_j(t)$ satisfy the following conditions:*

$$\exists f : \mathbb{R} \to \mathbb{R} \quad \textit{and} \quad \forall 0 \leq t < T = t_m :$$

$$\left(x_j(t) \underset{\epsilon}{\to} x_i(t) \Big|_{t<T} \wedge \quad \left| x_j(t_m) - f(\mathcal{X}_j^{1:m-1}, \mathcal{X}_i^{1:m}) \right| \geq \epsilon \right)$$

or

$$\left(x_i(t) \underset{\epsilon}{\to} x_j(t) \Big|_{t<T} \wedge \quad \left| x_i(t_m) - f(\mathcal{X}_i^{1:m-1}, \mathcal{X}_j^{1:m}) \right| \geq \epsilon \right).$$

In some cases, the existence of unseen factors has an impact on the observed values of the system which cause them to have a specific behavior. However, uncovering those hidden factors behind all the underlying electro-mechanical devices is a challenging task. Characterizing these factors can help us to reveal the hidden relationships between potential time series whether they have indirect or complex relationships. Figure 3.2 (a) shows an example of relationships among a set of time series, $(x_1, x_2, \cdots, x_n)$. In reality, the relationships can be direct (solid lines) or indirect (dashed lines). Previous works tried to reveal the direct relationships among time series (which is shown in Figure 3.2 (b)). However, despite the simplicity of these linear methods, sometimes they result in a sparse graph of invariants where tracking all time series is impossible. Moreover, these methods are not able to capture the underlying hidden relationships and results in poor detection of system outliers. In this chapter, we aim to uncover those hidden relationships with the help of hidden factors as latent variables. Hidden factors, $(f_1, f_2, \cdots, f_n)$, are considered as a higher level in hierarchy of the system and have an impact on the whole observed variables. An example of relationships in a system with hidden factors is illustrated in Figure 3.2 (c).

Definition 7 ***Latent Variable:*** *In a cyber-physical system with the set of observed time series $\mathcal{D} = \{x_1(t), \ldots, x_n(t)\}$, an unobserved time series $h(t)$ is a latent variable when two or more observed time series are functions of $h(t)$. In other words,*

$$\begin{aligned} &\exists \mathcal{D}' \subset \mathcal{D} \quad \textit{where} \quad \forall x(t) \in \mathcal{D}', \quad \exists g_x : \mathbb{R} \to \mathbb{R} \quad : \\ &\quad \forall t_m \geq 0, \quad x(t_m) = g_x(\mathcal{H}^{1:m}) \end{aligned} \tag{3.5}$$

where similar to Eq. 3.2, $\mathcal{H}^{1:m}$ is defined as follows:

$$\mathcal{H}^{1:m} = \left[h(t_1), h(t_2) \ldots, h(t_m) \right]^T.$$

It should be noted that each cyber-physical system may have more than one latent variable. Also, existence of a latent variable does not mean that all the observed time series should be directly related to that variable.

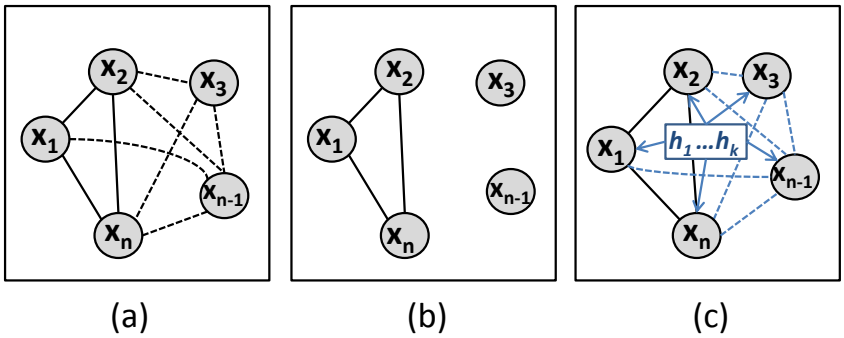


Figure 3.2: Different models of relationships: (a) Relationships in reality, (b) Relationships in the ARX model and (c) Relationships in ARX with a latent model.

3.4 Invariant Discovery

In this section, we describe a framework for invariant graph discovery and anomaly detection. For this purpose, we first extract latent variables using factor analysis and incorporate hidden factors into the regression model. Then we construct the invariant graph using a search algorithm. Finally, we use the constructed graph as a normal invariant graph and deploy it for the purpose of anomaly detection in the system. By discovering the broken invariants and ranking them, one may be able to find fault(s) and localize them.

3.4.1 Factor Analysis

Let us assume that we have a set of n random variables (input variables), denoted by X_1, ..., X_n. Also, assume that there are k hidden (latent) factors in the system, denotes by H_1, ..., H_k. Furthermore, assume that the observed variables are modeled as linear combinations of latent variables. Then we derive latent variables using the *factor analysis* method. Factor analysis is a well-studied field and is used to determine the main latent sources behind the observed data variation [27]. Although factor analysis is similar to principal component analysis (PCA), it is used more in predictive models due to its generalizability (e.g., factor loadings can remain consistent for different subsets of variables) [115]. Some might think of the factor model as generative models where the data is produced based on factors.

In factor analysis, for one sample of data extracted from random variable distributions, we have:

$$X_i - \mu_i = \lambda_{i1} H_1 + ... + \lambda_{ik} H_k + \zeta_i, \tag{3.6}$$

where μ_i is the expected value of X_i, H_j's are unobserved random variables and λ_{ij}'s are unknown constants ($i \in 1, \cdots, n$ and $j \in 1, \cdots, k$ where $k < n$). Also, ζ_i's are independently distributed error terms with zero mean and finite variance ($Var(\zeta_i) = \psi_i$). In other words, by Eq. 3.6, each of the X_i's random variables is related to k hidden random variables, known as latent factors.

In matrix notation, we have:

$$\mathbf{X} - \mu = \mathbf{\Lambda H} + \mathbf{Z}, \tag{3.7}$$

where $\mathbf{X} = (X_1, \cdots, X_n)^T$ is a data sample vector, μ is the expected values of data samples, $\mathbf{\Lambda}$ is an $n \times k$ matrix named as loading matrix, $\mathbf{H} = (H_1, \cdots, H_k)^T$ is a vector of latent factors, and $\mathbf{Z} = (\zeta_1, \cdots, \zeta_n)'$ is the vector of error.

It is assumed that $\mathbf{H}$ and $\mathbf{Z}$ are independent, and $E(\mathbf{Z}) = \mathbf{0}$, $E(\mathbf{H}) = \mathbf{0}$, $Cov(\mathbf{Z}) = Diag(\psi_1, ..., \psi_n) = \psi$, and $E(\mathbf{HH^T}) = \mathbf{\Phi}$. Furthermore, it is assumed that the data has a multivariate normal distribution, $\mathbf{X} = \mathcal{N}(\mu, \mathbf{\Sigma})$. Based on these assumptions, we will have:

$$\mathbf{\Sigma} = \mathbf{\Lambda \Phi \Lambda^T} + \psi. \tag{3.8}$$

Since $\mathbf{X}$ has a multivariate normal distribution, the actual distribution function of elements of sample covariance matrix, $\mathbf{S}$, can be expressed as a Wishart distribution with $m-1$ degrees of freedom, $m\mathbf{S} \sim \mathcal{W}_n(\mathbf{\Sigma}, m-1)$, where m is the number of samples.

The log-likelihood of the Wishart distribution can be expressed as follows:

$$log\, L = -\frac{m-1}{2}\big(log|\mathbf{\Sigma}| + tr(\mathbf{S\Sigma^{-1}})\big), \tag{3.9}$$

where the terms independent of $\mathbf{\Sigma}$ are dropped.

It is obvious that maximization of L is equivalent to minimizing the following function:

$$Q = log\,|\mathbf{\Sigma}| + tr(\mathbf{S\Sigma}^{-1}). \tag{3.10}$$

One can find the latent variables by taking the partial derivatives of Eq. 3.10 with respect to the elements of loading matrix and errors constrained by Eq. 3.8. For simplicity, it is convenient to assume that $\mathbf{\Phi} = \mathbf{I}$ and $\mathbf{\Lambda^T \Psi^{-1} \Lambda}$ is diagonal.

There are different types of criteria to determine the number of factors such as criteria based on eigenvalues, discrepancy of approximation, or overall discrepancy [92]. Here, we use the Kaiser criterion which drops those with eigenvalues of less than 1. Indeed, the number of factors, must be lower than the number of observed variables, $k < n$. More details can be found in [40, 53].

3.4.2 Latent Factor Auto Regression with Exogenous input (LFRX)

Having n time series, $\mathcal{D} = \{x_1(t), \cdots, x_n(t)\}$, related to a cyber-physical system, similar to the ARX model [47], we can rewrite Eq. 3.1 as:

$$\hat{x}_j(t) = \sum_{p=1}^{u} a_p x_j(t-p) + \sum_{p=0}^{v} b_p x_i(t-l-p), \tag{3.11}$$

where $x_i(t), x_j(t) \in \mathcal{D}$.

As acknowledged widely [47, 35, 20, 19, 108], a drawback of ARX is that relationship discovery is done based on the existence of direct linear relationships between two observed time series. In other words, at each time ARX considers a pair of time series without considering the underlying relationships and hidden patterns. To address this issue, we deploy latent factors in the ARX model to recover the complex relationships. If we use latent factors in the above regression model, we will have:

$$\hat{x}_j(t) = \sum_{p=1}^{u} a_p x_j(t-p) + \sum_{p=0}^{v} b_p x_i(t-l-p) + \sum_{p=0}^{w} \sum_{q=1}^{k} c_{pq} h_q(t-p), \tag{3.12}$$

where $h_p(t)$'s are the latent factor time series that have been built based on the latent factor random variables, as discussed in the previous subsection (Eq. 3.6). Also, a_p's, b_p's, and c_{pq}'s are the regression weights that are determined in the learning phase. Note that in Eq. 3.12, in addition to the regression weights, latent factors are also unknown and should be estimated in the learning phase.

It should be noted that here we incorporate the previous values of $x_j(t)$ as well as values of exogenous variable, $x_i(t)$, and hidden variables, $h_q(t)$'s, to estimate new value of $x_j(t)$. In matrix notation, Eq. 3.12 will change to:

$$\hat{x}_j(t) = \mathbf{A}^T \mathcal{X}_j^{t-u:t-1} + \mathbf{B}^T \mathcal{X}_i^{t-l-v:t-l} + \mathrm{Tr}(\mathbf{C}^T \boldsymbol{\mathcal{H}}), \tag{3.13}$$

where $\mathbf{A}_{u\times 1}$, $\mathbf{B}_{(v+1)\times 1}$, $\mathbf{C}_{(w+1)\times k}$ are matrices of coefficients. Also, $\boldsymbol{\mathcal{H}}_{(w+1)\times k}$ is a matrix that represents all the latent factors, i.e. $\boldsymbol{\mathcal{H}} = \left[\mathcal{H}_1^{t-w:t} \cdots \mathcal{H}_k^{t-w:t}\right]$. In our experiments, we assume $u = v = w$ and their values are estimated using cross-validation. Also, due to the lack of delay in our datasets, we assume l is zero. In order to solve Eq. 3.13, first we derive latent factors, $\boldsymbol{\mathcal{H}}$, using factor analysis of Subsection 3.4.1 and then we incorporate them into the regression model to estimate the weights.

3.4.3 Invariant Graph Construction

Based on Definition 4, in order to discover system invariants we need to identify time series that have persistent approximate dependencies. While time series may have nonlinear dependencies, in this chapter we consider linear relationships and use ARX and LFRX for this purpose.

The search algorithm that extracts system invariants is shown in Algorithm 1. In this algorithm, for each pair of time series, we first assume that they have a direct linear relationship and we fit them using an ARX model (Eq. 3.11). The ARX model for time series $x_i(t)$ and $x_j(t)$ is illustrated by θ_{ij}^{ARX}. Then, we assume that there might be an indirect relationship through latent variables and hence, we use LFRX model to learn θ_{ij}^{LFRX}. As defined in

Definition 3, to determine if $x_j(t)$ depends on $x_i(t)$, we need to compare the estimation error with an acceptable threshold, ϵ. However, since in a specific cyber-physical system different time series have different range of values, it is more appropriate to use normalized error measurements. For this purpose, when we estimate $x_j(t)$ based on $x_i(t)$, we can evaluate the relative absolute error (RAE) defined by the following equation:

$$e_{j,i}^{RAE}(t) = \frac{|\hat{x}_j(t) - x_j(t)|}{\sum_{t=t_s}^{t_e} |\overline{x}_j(t) - x_j(t)|}, \tag{3.14}$$

where $x_j(t)$ is the observed value, $\hat{x}_j(t)$ is the estimated value based on $x_i(t)$, and $\bar{x}_j$ is the sample mean of observed values.

According to [48] for each pair of time series, $x_i(t)$ and $x_j(t)$, we calculate a score to measure their dependencies. The following normalized score may be used for the evaluations:

$$\mathcal{F}_{j,i}(t) = 100(1 - e_{j,i}^{RAE}(t)). \tag{3.15}$$

A higher score indicates stronger dependency between the time series. It should be noted that RAE is a specific example of normalized error measurement and one can easily extend the algorithm to use other error measurement approaches including RMSE, specifically when time series have the same range of variations.

In Algorithm 1, lines 8 to 15 are dedicated to estimate individual values of $x_j(t)$ based on $x_i(t)$ and calculate the scores for ARX and LFRX models. In order to discover invariants and choose between direct or indirect relationships, we consider the following criteria:

- For all the time steps, score should be greater than or equal to a specific threshold. We name this threshold as *minimum acceptable score* and denote it by τ. Then, we should have:
$$\min_t(\mathcal{F}_{ji}(t)) \geq \tau. \tag{3.16}$$
- Since higher score depicts stronger relationships, in choosing between ARX and LFRX, the one with the better overall score is chosen.
- Since linear invariants represent simpler relationships, higher priority is given to ARX-based invariants using a guard bound, Δ, which we name as the *ARX superiority threshold*. In other words, ARX-based invariants are selected when:
$$\sum_{t=t_s}^{t_e} \mathcal{F}_{ji}^{ARX}(t) \geq \sum_{t=t_s}^{t_e} \mathcal{F}_{ji}^{LFRX}(t) - \Delta. \tag{3.17}$$

When calculated scores satisfy Eq. 3.16, we will say that $x_i(t)$ and $x_j(t)$ are invariant and based on Eq. 3.17, the type of invariant is chosen to be direct (ARX) or indirect (LFRX).

The resulted invariants are added to the sets of ARX and LFRX invariants denoted by S_{ARX} and S_{LFRX}, respectively. In Algorithm 1, lines 16 to 21 are dedicated to this process.

After finding system invariants, the final step (line 24 in Algorithm 1) is to construct the invariant graph, $G = (V, E)$. This is a straightforward task which is performed based on Definition 5. The total number of iterations of this algorithm is $O(tn^2)$ where t is the length of time series. At each iteration (lines 9 to 14), models are learned with a time complexity which is a function of t^2 and various constants $(w, v, u, \cdots)$. This results in an overall complexity of $O(Cn^2t^3)$.

3.4.4 Outlier Detection using Broken Invariants

After constructing the invariant graph (in Subsection 3.4.3), we can use this graph for detecting abnormalities in the system. For this purpose, using Definition 6, at each time step we check whether each of the graph edges is broken or not. We then rank the time series in order to localize the source of abnormality. In what follows we first describe the alerting algorithm, followed by a metric for alerting threshold estimation and finally the ranking method for fault localization.

Anomaly alerting algorithm: The alerting algorithm is illustrated in Algorithm 2. In this algorithm, in order to prevent generations of multiple alerts consecutively, we use an alert filtering mechanism by imposing a counting strategy with alert threshold of α. When the number of consecutive violations of a specific invariant goes beyond α, the algorithm invokes an alert to the system administrator, who may use this for further investigations. Time complexity of this algorithm is $O(|E|)$ at each time-step.

Anomaly detection threshold: According to model-based FDI methods used in control theory and similar to [108], in order to reduce false alarms the following approach is used for detection of broken invariants. The difference between the predicted value, $\hat{x}_j(t)$, and the actual value, $x_j(t)$, is recorded and whenever this difference deviates more than a predetermined threshold, ϵ_0, an invariant will be broken:

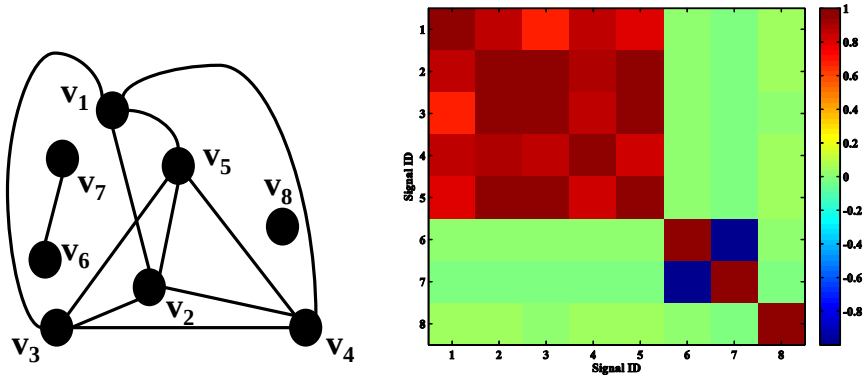
$$|\hat{x}_j(t) - x_j(t)| > \epsilon_0. \tag{3.18}$$

The threshold ϵ_0 can be estimated based on the observed values in the training period. According to [108], ϵ_0 is assumed to be 10% larger than the tolerance of deviations from the actual values:

$$\epsilon_0 = 1.1 * arg_r \left\{ Prob(|\hat{x}_j - x_j| < r) = 0.995 \right\}, \tag{3.19}$$

where r is greater than 99.5% of the residuals observed in the training data.

Ranking time series for fault localization: In complex cyber-physical systems with a large number of invariants, one single fault in the system may lead to a large number of



(a) (b)

Figure 3.3: (a) Invariant graph of synthetic data. (b) Correlation matrix of synthetic time series.

broken invariants. Hence, for fault localization we need to rank the invariant graph vertices according to the number of their broken edges. Similar techniques have also been used in [34]. For this purpose, we use the following score to rank the vertices after the occurrence of an alarm. Assuming that an alarm is generated at time t, for each vertex, v_j, we calculate the following score:

$$\rho_j = \frac{d_j^{normal} - d_j(t)}{d_j^{normal}}, \tag{3.20}$$

where d_j^{normal} is the degree of v_j in normal condition and $d_j(t)$ is the degree of v_j after alarm generation at time t. It is obvious that higher value of ρ_j indicates v_j has lost more edges which may potentially be due to the occurrence of a fault at $x_j(t)$.

3.5 Experimental Results

We perform our experiments on several datasets. We aim to show how our method (ARX + LFRX) can discover the invariants, how it can improve the accuracy of system, and how it can find the anomalies happening throughout the network. First, we perform our analysis on a synthetic dataset to recover indirect invariants. Next, we use a datasets from real cyber-physical systems: a microgrid system. In this dataset, there are multiple factors and measurements with various temporal and spatial dependencies.

3.5.1 Synthetic Data

Dataset Description: At the first step, we perform our experiment on a synthetic dataset to verify our method for the discovery of indirect hidden relationships. For this purpose, we generate eight signals and compare the results of ARX with our method (integration

of ARX and LFRX). In this experiment, we add a Gaussian noise with zero mean and standard deviation of 0.1 to one of the time series in order to test the invariant graph under abnormalities. The ground truth graph and its corresponding correlation matrix are shown in Fig. 3.3 (a) and (b), respectively. As Fig. 3.3 (a) shows, V_6 and V_7 are correlated to each other. V_8 is isolated and all the remaining nodes are correlated to each other. However, the hidden relationship between signals is not observable in Fig. 3.3 (a). In fact, V_3, V_4, V_5 are generated using V_1 and V_2. The relationship between signals is given in the following equations:

$$V_1(t) = 0.9V_1(t-1) - 0.02V_1(t-2) - 0.01V_1(t-3) + 0.09 + \eta$$

$$V_2(t) = 2(V_1(t-1) - V1(t-2)) + 0.5(V_2(t-1) + V2(t-2))$$

$$V_3(t) = V_1(t-1) + V_2(t) - V_1(t),\ V_4(t) = 3V_1(t-1) + V_2(t-1)$$

$$V_5(t) = 3V_1(t-1) - V_2(t-1)$$

$$V_6(t) = 1 + 0.01R(t, 100),\ V_7(t) = 1 - 0.01R(t, 100)$$

$$V_8(t) = 2e^{10^{-4}t} + R(t, 600)e^{-10^{-4}t}$$

where $R(t, T)$ is a rectangular function of t, oscillating between -1 and 1 with period of T and η is a Gaussian noise with zero mean and standard deviation of 0.01.

In order to consider various situations, with and without presence of hidden relationships, we perform multiple experiments with different subset of the above signals.

Results and Discussion: Recovered graph for both methods in normal and abnormal condition are shown in Figure 3.4. In this figure, each row denotes an experiment involving a subset of synthetic time series, where white nodes represent the one with injected noise. As it is shown, in all cases the ARX + LFRX method has recovered the planted invariants and the recovered graph matches the ground truth. In both methods, in the presence of an anomaly, the invariants attached to the corrupted signal (white node) are broken. However, in some cases such as (a) and (b) where the ARX method cannot recover the existing relationships, at the time of anomaly, it was not able to detect it correctly. In figures (a) to (d), time series V_1 and V_2 are not measured and hence, time series V_3, V_4, and V_5 have indirect relationships. It is obvious from Fig. 3.4 that the proposed method (ARX + LFRX) is able to discover the corresponding invariants while ARX, with the same parameter settings, has failed to discover them.

3.5.2 Microgrid

Dataset Description: We performed our experiments on a microgrid system where several devices are operating in a distributed setting (Fig. 3.5). In this setting, the control unit tries to minimize the amount of energy based on various criteria and hence the microgrid shows

a complex behavior in the logged measurements. This dataset which is provided by NEC Labs contains logged data from multiple sources such as loads (primary, secondary), battery, PMU (measurement unit outside of the microgrid), solar system (PV), weather (inside and outside parameters), and air cooling unit. There are total of 84 features measured from July 7th to August 7th, 2014. Due to the different sampling rates of each device, time series are re-sampled with a unique rate to be aligned to a specific window-time. Figure 3.6 shows a sample plot of three time series from different units during one week.

Results and Discussion: The invariant graph derived by our proposed method is represented in Fig. 3.7 (a). In this figure, each node represents one of the features and the set of features that belong to a specific device are illustrated using the same color. Also, the size of each node is proportional to its degree. Furthermore, invariants derived by ARX and LFRX methods are shown by red and blue edges, respectively. The total number of invariants that ARX + LFRX discovered is 2285 where 797 of them are indirect and 1488 of them are direct invariants.

The average estimation errors of traditional ARX and the proposed method are compared in Table 3.1. As this table shows, the average error of the proposed method for each device is dramatically lower than the error resulted by ARX approach. This means that invariants are selected with higher accuracy using ARX + LFRX. Also, Table 3.1 compares the proposed method with ARX in terms of the number of invariants between devices (Inter-Device) and within each device (Intra-Device). Inter-Device edges are visualized in Fig. 3.8(a) where edge thickness represents the total number of invariants between devices. From this figure, we can observe the high complexity of inter-dependencies between measurements of devices. For example, energy produced by photovoltaics (PV) has effect on battery, PMU, loads, and the temperature of environment.

After occurrence of an abnormal behavior, the topology of the invariant graph changes (i.e., depending on the nature of anomaly, some edges are removed from the graph). By comparing consecutive graphs, one is able to detect outliers in the system. We detect changes in the invariant graph in two different situations: when an additional device is switched off/on and when secondary load is turned off. Results are shown in Fig. 3.7 (b) and (c), respectively. As we may observe from Fig. 3.7 (b), two edges connecting the secondary load to PMU and PV are broken. This is due to the change in the energy consumption behavior of the system. On the other hand, as Fig. 3.7 (c) depicts, a large number of invariants between secondary load and other devices are broken. This is due to the disconnection of the secondary load. It should be noted that measurements of the secondary load were among the top five anomalies returned by our outlier ranking method. Also, the number of remaining invariants between devices are shown in Fig. 3.8 (b) and (c).

Figure 3.9 shows some examples of anomalies in the microgrid system. Each figure shows a pair of time series that under normal condition are invariant. Abnormal conditions are shown in darker colors. The gap between occurrence of anomalies and normal time series

depicts the time difference between them. As an example, Fig. 3.9 (a) shows the detection of sudden change in the red curve which is the State of Health (SOH) of the battery. This change is detected using the broken link between SOH and Reactive power in Channel C of the PMU. Detecting such anomalies is crucial for microgrid operators.

It should be noted that since we do not know the labeling of real dataset, we are unable to evaluate our method using precision and recall metrics. Nevertheless, we calculated precision and recall under different scenarios. As an example, when 10 nodes have random injected noise, by looking at the top 10 ranked results, precision and recall were equal. This value is 0.51 for the ARX method, whereas for ARX + LFRX, it is 0.68.

3.6 Related Work

Smart Grid and Power System Analytics: Power grids comprise a large number of elements and processes that are highly dynamic and complex. Traditionally, power system operational studies are primarily based on a quasi-steady-state assumption with static and explicit models that largely ignores dynamic characteristics of loads and control devices. The classic weighted least square (WLS) estimator, combined with methods such as largest normalized residual test and hypothesis testing identification, is extensively used for system diagnosis and outlier identification [1]. Recent developments in smart grids have revealed to us insight into stochastic operating behaviors and dynamics that we were never able to observe before. In particular, the widespread deployment of smart meters, renewable generation, smart load controls, energy storage, and plug-in hybrid vehicles will require fundamental changes in the operational concepts and principal components of the grid, in order to achieve real-time operation and control.

Fraud detection and particularly detection of energy theft is one of most important concerns in the smart grid [50, 76]. Data analytic methods can play an important role in identifying abnormal consumption trends and possible malicious activities in such systems. Daisuke et al. [76] used ARMA and LOF methods in an adversarial environment to detect attacks in data collected using advanced meter infrastructure (AMI). Rong et al. [50] compared classification-based, state-based, and game theory-based methods in energy-theft detection schemas.

One area that has witnessed significant developments is in the use of phasor measurement units (PMUs). Chen et al. [22] use PCA for online monitoring of PMU data for the purpose of early event detection. Khan et al. [57] proposed a parallel fluctuation approach using MapReduce techniques. At the lower level, We [83] proposed an integrated data-driven framework to study the behavior of battery systems in microgrids using clustering, regression, and spectral clustering of time series for the purposes of high level characterization of usage behavior and online parameter estimation.

Invariant Discovery and Structure Learning: Sharma et al. [108] used ARX for invariant discovery in distributed systems and discussed the challenges in fault localization for data centers. Shan et al. [107] have extracted overlay invariants based on pairwise invariant networks for fault detection and capacity planning in distributed systems. Due to the time complexity of invariant discovery of large scale systems, Ge et al. [35] developed an effective pruning techniques based on the identified upper bounds. In some applications, the existence of anomalies in invariant graphs yields many broken links which makes it difficult for a system expert to manually inspect each broken link. Hence, Ge et al. in [34] proposed two different methods of ranking metrics according to the anomaly levels occurring in invariant networks.

In a closely related area, viz. causal modeling of time-series data, Arnold et al. [6] used the concept of Granger causality to infer the structure of the causal network given set of time series. These authors compared performance of the exhaustive Granger method and a Lasso-Granger method with benchmark methods including the VAR and SIN methods. However, in [6], the main goal was to construct causal graphs instead of addressing data with correlated variables. Subsequently, Liu et al. [69] used a hidden Markov random field regression framework to infer temporal causal structures. Cheng et al. [24] use time order relationships to capture temporal dependence structures underlying multivariate time series.

Anomaly Detection in Graphs: Akoglu et al. [3] provide an extensive survey of anomaly detection methods in graphs spanning different settings: unsupervised, (semi-) supervised approaches, static, dynamic, attributed, and plain graphs. In dependency graphs, for the purpose of anomaly detection, Ide et al. [46] used sparse structure learning to compute correlation anomaly scores of each variable using neighborhood selection approaches.

3.7 Conclusion

Invariant discovery is an exciting research field which aims to discover underlying relationships in cyber-physical systems. We used latent factor regression analysis and combined it with the ARX model (ARX + LFRX) to recover underlying direct and indirect relationships. These invariants are helpful in decision making and monitoring processes such as outlier detection. We tested our models on several datasets and results showed that with the help of latent factors, the accuracy of discovered invariants was higher than traditional methods. Investigating other topologies involving latent variables (such as a mesh network) and heuristic search algorithms to reduce the computational complexity are some of the directions for future research.

Algorithm 1: Invariant Search Algorithm

Input: x_i, $i \in \{1, .., n\}$: set of time series, Δ: ARX superiority threshold, τ: minimum acceptable score, t_s and t_e: start and end time of training dataset.

Output: G: Invariant Graph.

```
S_ARX = {};
S_LFRX = {};
for i = 1 to n do
    for j = 1 to n do
        if i == j then
            Continue;
        end
        foreach t_s ≤ t ≤ t_e do
            Learn an ARX model, θ_ji^ARX, using Eq. 3.11;
            Calculate x̂_j^ARX(t) using θ_ji^ARX;
            Compute F_ji^ARX(t) with Eq. 3.15;
            Learn an LFRX model, θ_ji^LFRX, using Eq. 3.12;
            Calculate x̂_j^LFRX(t) using θ_ji^LFRX;
            Compute F_ji^LFRX(t) with Eq. 3.15;
        end
        if (Σ_{t=t_s}^{t_e} F_ji^ARX(t) ≥ Σ_{t=t_s}^{t_e} F_ji^LFRX(t) − Δ) and (min_t(F_ji^ARX(t)) ≥ τ) then
            S_ARX = S_ARX ∪ {x_i ⇐ x_j};
        end
        if (Σ_{t=t_s}^{t_e} F_ji^LFRX(t) > Σ_{t=t_s}^{t_e} F_ji^ARX(t) + Δ) and (min_t(F_ji^LFRX(t)) ≥ τ) then
            S_LFRX = S_LFRX ∪ {x_i ⇐ x_j};
        end
    end
end
Construct Graph, G = (V, E), using S_ARX and S_LFRX;
return G;
```

Algorithm 2: Alerting Algorithm

Input: $x_i(t)$, $i \in \{1, .., n\}$: set of time series, $G = (V, E)$:Invariant Graph, t_e: start time of monitoring, α: alerting threshold.

```
foreach t > t_e do
    foreach e_{i,j} ∈ E do
        Use Definition 6 to check if e_{i,j} is broken;
        if e_{i,j} is broken then
            cnt_{ij} ← cnt_{ij} + 1;
        else
            cnt_{ij} ← 0;
        end
        if cnt_{ij} > α then
            Invoke an alert;
            cnt_{ij} ← 0;
        end
    end
end
```

Table 3.1: Performance evaluation result of ARX and (ARX + LFRX)

Device (No. of Signals)	Avg. Error (ARX + LFRX)	Avg. Error (ARX)	Intra-Device Edges (ARX + LFRX)		Intra-Device Edges(ARX)	Inter-Device Edges (ARX + LFRX)		Inter-Device Edges(ARX)
			Total	Latent		Total	Latent	
PMU (28)	0.0028	0.1346	102	73	102	810	265	755
PV (9)	0.0119	0.1146	26	5	26	187	63	175
Battery (9)	3.7×10^{-14}	0.2764	21	12	21	239	81	226
Primary LD (9)	2.2×10^{-4}	0.1460	35	22	29	235	66	236
Secondary LD (9)	2.3×10^{-6}	0.0434	36	0	36	53	19	52
Weather (13)	0	0.1342	57	17	57	260	94	244
Air Cooling (7)	0.2852	0.2903	11	5	11	213	75	203

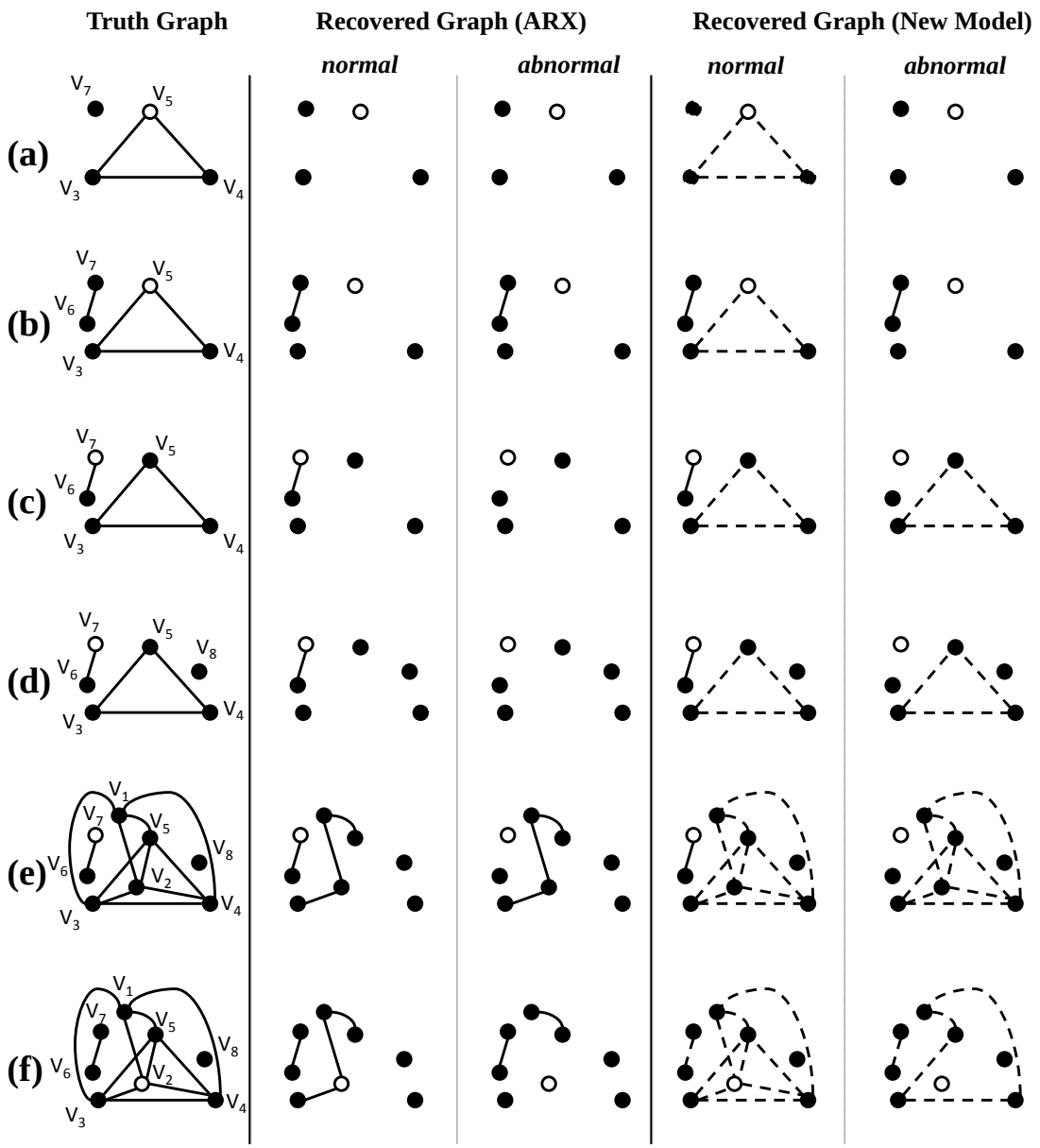


Figure 3.4: Invariant graphs discovered using ARX and the proposed method (ARX + LFRX) under normal and abnormal conditions. First column shows the ground truth. In the abnormal condition, an anomaly is injected into each graph at one variable (white node). Rows (a) to (f) show different combinations of time series in Fig. 3.3. Direct invariants are shown in solid lines and indirect invariants are shown in dashed lines.

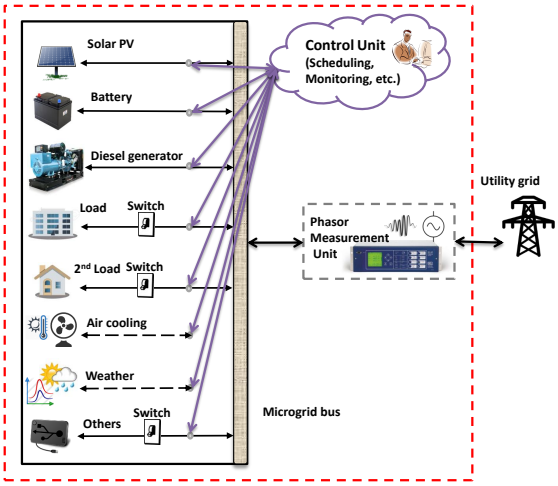


Figure 3.5: Schematic view of the NEC microgrid setup.

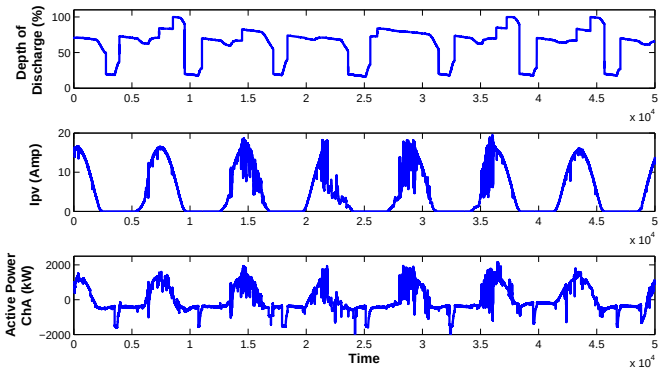
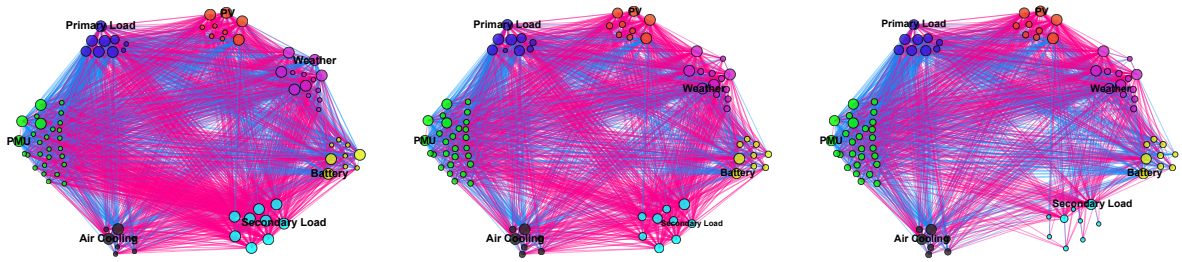
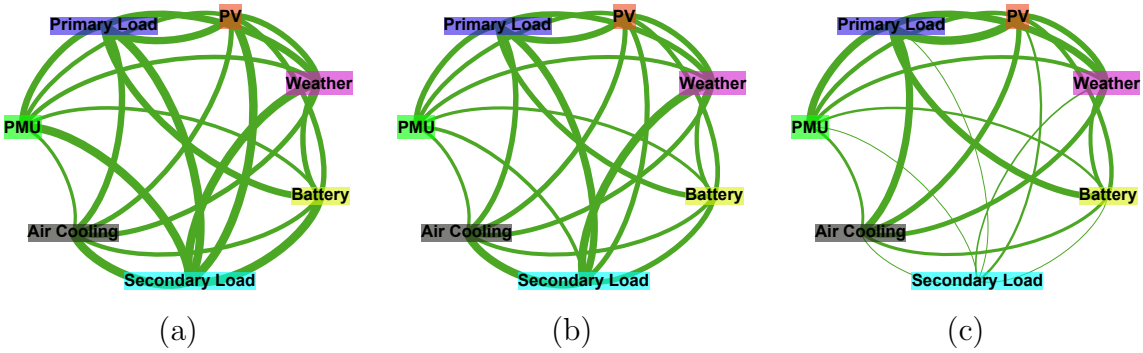


Figure 3.6: Three different time series of NEC microgrid dataset during one week.



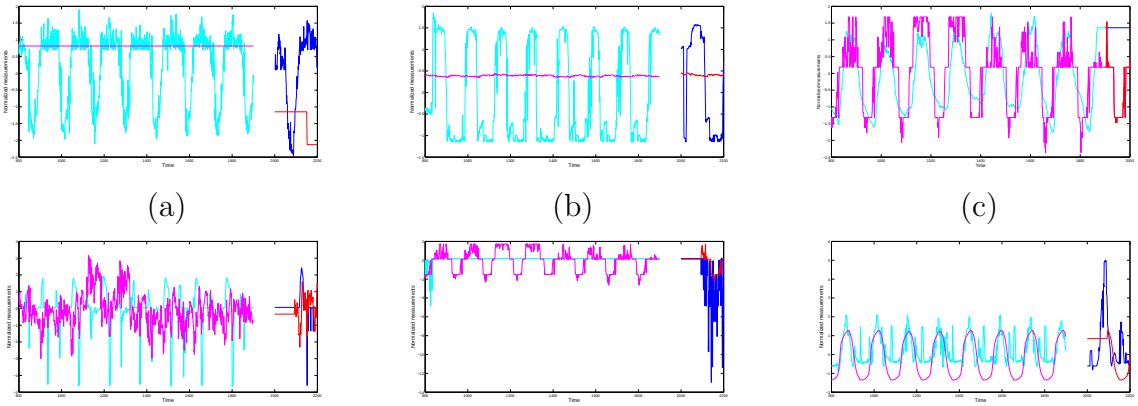
(a) (b) (c)

Figure 3.7: (a) Invariant graph of microgrid under normal condition (ARX + LFRX). (b) Outlier happens when an additional device is switched off/on in the system. (c) Outlier happens when the secondary load is disconnected.



(a) (b) (c)

Figure 3.8: (a) Invariant graph of inter-devices of microgrid (ARX + LFRX). (b) Outlier happens when additional device is switched off/on in the system. (c) Outlier happens when secondary load is disconnected.



(a) (b) (c)

(d) (e) (f)

Figure 3.9: Invariant time series at normal and abnormal conditions: (a) Reactive power Channel C vs. State of Health of Battery (b) Power factor vs. Voltage of channel A in PMU (c) Outside temperature vs. Power Factor of primary load (d) Battery Current vs. Inverter output voltage (e) Peak voltage of secondary load vs. Power factor (f) Current magnitude of Channel A vs. Peak power of primary load.

Chapter 4

Charging and Storage Infrastructure Design for Electric Vehicles

4.1 Introduction

In this chapter, we propose a novel framework to address the problem of charging and storage infrastructure design for EVs by adopting an urban computing approach. We develop coordinated clustering techniques to work with network models of urban environments to aid in placement of charging stations for an electrical vehicle deployment scenario. Furthermore, we evaluate the network before and after the deployment of charging stations, to recommend the installation of appropriate storage units to overcome the extra load imposed on the network by charging stations. We demonstrate multiple factors that can be simultaneously leveraged in our framework in order to achieve practical urban deployment. Our ultimate goal is to help realize sustainable energy system management in urban electrical infrastructure by modeling and analyzing networks of interactions between electric systems and urban populations.

Due to the additional load imposed to the network by EVs, appropriate storage units must be deployed beside the charging stations. There are several works that consider the problem of load management for EV charging and its impact on the grid [88, 77]. However, there is no previous work that addresses the coordinated impact of placement over an urban infrastructure and its solution thereof. Here, we consider charging costs and storage placement problems in addition to the problem of charging station placement. We develop an algorithm to assign EVs to the nearest charging stations by minimizing charging cost and travel distance. After assigning charging stations to EVs, additional load of each charging station is calculated and used to determine appropriate storage deployment for each location.

Although we assume that each charging station uses storage to offset the impact of charging on the grid, other alternative solutions can also be used, such as upgrading transmission lines

or using a Vehicle-to-Grid (V2G) strategy. While we can upgrade the transmission lines, the distribution infrastructure still remains a bottleneck. In fact, upgrading transmission lines is not a complete solution, albeit expensive and time consuming, discounting the regulatory hurdles. The other solution is based on using V2G. However, there are no EVs today which provide V2G capability to owners and business models utilizing such capability are still uncertain from a utility perspective. Hence, this will not affect the proposed methodology.

Here, we use network datasets organized from synthetic population studies, originally designed for epidemiological scenarios, to explore the EV charging station placement problem. The dataset was organized for the SIAM Data Mining 2006 Workshop on Pandemic Preparedness [8] and models activities of an urban population in the city of Portland, Oregon. The supplied dataset [12] tracks a set of synthetic individuals in Portland and, for each of them, provides a small number of demographic attributes (age, income, work status, household structure) and daily activities representing a normative day (including places visited and times). The city itself is modeled as a set of aggregated activity locations, two per roadway link. A collection of interoperable simulations—modeling urban infrastructure, people activities, route plans, traffic, and population dynamics—mimic the time-dependent interactions of every individual in a regional area. This form of 'individual modeling' provides a bottom-up approach, mirroring the contact structure of individuals and is naturally suited for formulating and studying the effect of intervention policies and considering 'what-if' scenarios.

4.2 Related Work

We survey the related work in two categories: mining GPS datasets and smart grid analytics. GPS datasets have emerged as a popular source for modeling and mining in urban computing contexts. They have been used to extract information about roads, traffic, buildings, and people behaviors [134], [137], [67]. The range of applications is quite varied as well, from anomaly detection [67] to taxi recommender systems [137] that aims to maximize taxi-driver profits and minimize passengers' waiting times. The notion of location-aware recommender systems is a key topic enabled by the increasing availability of GPS data, e.g., recommending points of interest to tourists [141]. We survey these works in greater detail next.

In [134], Yuan et al. proposed a framework to discover regions of different functionalities based on people movements. They adapt algorithms from the topic modeling literature, by mapping a region as a document and a function as a topic so that human movements become 'words' in this model. The focus of [137] and [135] is different: here, GPS data is used to mine the fastest driving routes for taxi drivers. In [137], Yuan et al. mined smart driving direction from GPS trajectory of taxis, and in [135] they consider driver behavior using other metrics such as driving strategies and weather conditions.

Clusters of moving objects in a noisy stadium environment are detected using DBSCAN

algorithm [32] in [103]. This task supports monitoring a stadium for groups of individuals that exhibit concerted behavior. In [117], the authors estimate distributions of travel-time from GPS data for use in routing and route-recommendation.

Our work here is different from the above works in that we use a synthetic population dataset and routes are based on people's travel habits that are mapped using geographical coordinates and road infrastructures. We are also not *per se* interested in mining the routes but to use the route information to better support charging infrastructure placement.

Smart grid analytics has emerged as a promising approach to usher in the promise of smart grid benefits. Researchers have begun to explore the problems concomitant with EV penetration in urban areas, especially unacceptable increases in electricity consumption [97]. A promising way to approach this problem is to understand the interactions between grid infrastructure and urban populations. While smart grids and EVs have been studied previously from technical and AI point of views, there is a limited number of research on smart grids from an urban computing perspective.

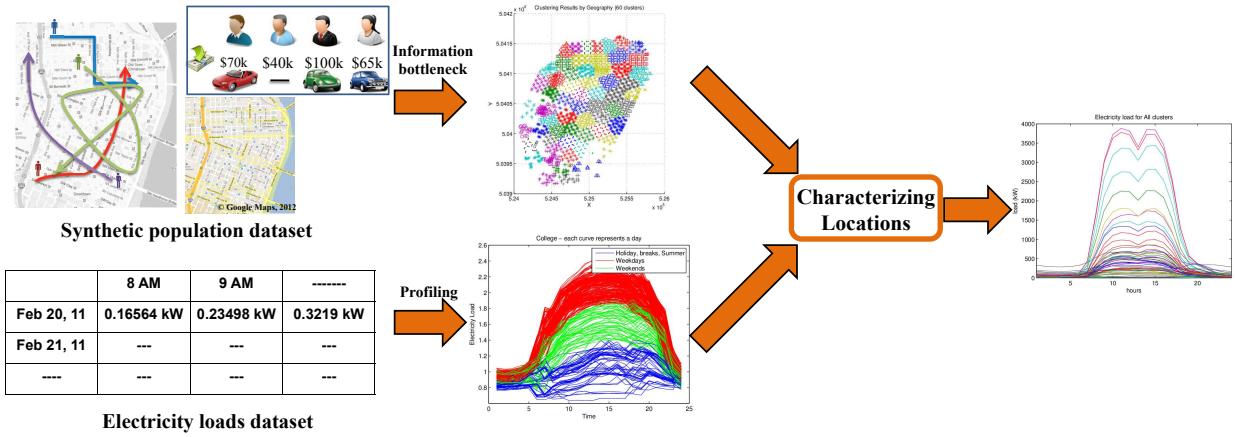
In this space, agent-based systems have been proposed to simulate city behavior in terms of agents with a view toward designing decentralized systems and maximizing grid profits as well as individuals' profit [97]. In [5], information from smart meters is used for forecasting energy consumption patterns in a university campus micro-grid, whose results can be used for future energy planning.

Significant research has been done to improve cost and reliability of energy storage systems [43]. Energy storage is used to perform an operation when there is not enough electricity. In [74], a solution is proposed to balance energy production against its consumption. In addition, authors in [11] try to design a general architecture in smart grid to have a significant gains in net cost/profit considering electric vehicles.

4.3 Methodology

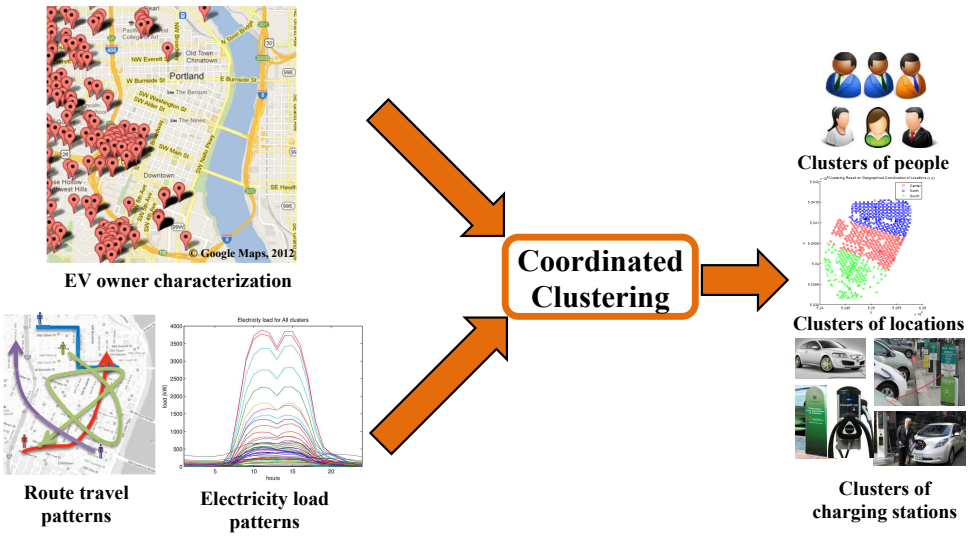
Our overall methodology is given in Figure 4.1. We describe each of the steps in our approach next. At a basic level, we integrate two basic types of data to formulate our data mining scenario. The first data, as described earlier, is a synthetic population of people and activities representing the city of Portland and the second dataset is electricity consumption profile of each location. Notice that the proposed methodology is a generic approach and can be applied to real-world data and the fact that we use synthetic data here is only due to our lack of access to real-world data to test our proposed methodology.

The synthetic dataset contains 243,423 locations of which 1,779 belong to the downtown area and of further interest for our purposes. Each location is represented by geographical [x,y] coordinate adopting the Universal Transverse Mercator coordinate system (UTM) [12]. There are a total of 1,615,860 people in the entire city. Information about them is orga-

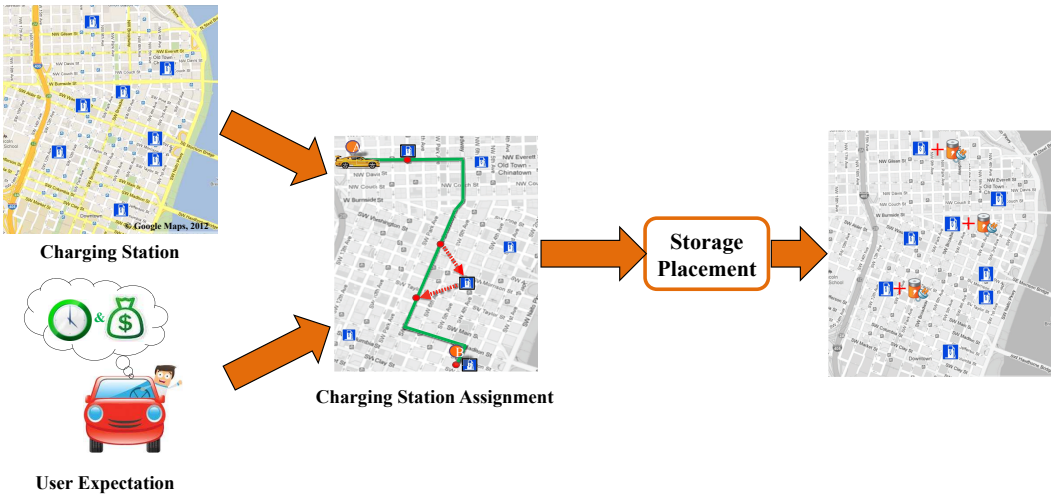


	8 AM	9 AM	-------
Feb 20, 11	0.16564 kW	0.23498 kW	0.3219 kW
Feb 21, 11	---	---	---
----	---	---	---

(a) Discovering location functionalities and characterizing electricity loads.



(b) Coordinated clustering of people, locations, and charging stations.



(c) Charging station assignment and storage placement.

Figure 4.1: Overview of our methodology.

nized into households, and for each household we have the details of number of people in the household including age, gender, income, and his/her house ID. The typical movement patterns of people in a 27 hour period (which includes a typical day) are also available. A total of 8,922,359 movements are provided. In addition to starting and ending locations for people's movements, this dataset also provides the *purpose* of the movement, categorized into nine types: {Home, Work, Shop, Visit, Social/Recreational, Serve Passenger, School, College, and Other}. A given person moves from one location to another location at a specific time for a specific purpose (from the nine mentioned above) and stays in that location for a specified period of time. These movement types can thus be utilized for further detailed studies. We also have the ability to map the locations using Google Maps and calculate distances of travel between locations.

To this dataset, we augment information about electricity consumption of each location and simulate the effects of EVs on its electricity demand profile. Since actual electricity consumption data for each location is not available until all the consumers have smart meters installed and in operation for some time, we approximate electricity load profile using the existing data (organized by NEC Labs, America).

It is clear that the electricity load of each location greatly depends on the functionality of that location and hence our first approach is to utilize an information bottleneck type approach [121] to characterize locations. Our aim is to cluster locations based on geographical proximity but such that the resulting clusters are highly informative of location function. This is thus our first application of a coordinated clustering formulation, and falls in the scope of clustering with side information. Next, we integrate the electricity load information to characterize usage patterns across clusters with a view toward helping identifying locations to place charging infrastructure.

Our next step is to more accurately characterize usage patterns of likely EV owners. A specific set of clusters from the previous pipeline is used and characterized using high-income attributes as the likely owners of EVs. We then bring in additional factors of locations that influence EV charger placement, e.g., residentiality ratio, load on the location, charging needs, and typical duration of stay in the location. Some of these factors (such as distance traveled) are in turn determined by mapping the home-to-work and work-to-home trajectories of EV owners and their stop locations. In the proposed method, three datasets are used. Two datasets describe locations and one of them describes people. Since each location has a set of features which do not depend on its coordination, we use one dataset to describe specific features of each location and another dataset that only consists of geographical coordinates. In addition to datasets that describe locations, we use a separate dataset for people with different income.

Choosing a right set of locations to install charging stations depends on many features. These features can be categorized into two groups: 1) Features of people who visit those locations. It is better to assume that these people have EVs, and because we assume that people with higher income have EVs, it is preferable to choose locations which people with

higher incomes visit frequently. 2) Features of locations such as electricity load. In fact, we are looking for a set of candidate locations that have similar features and also, are visited by same type of people. Among different data mining approaches [96], clustering techniques can identify similarities and can categorize locations into different sets.

We use a coordinated clustering formulation to simultaneously cluster three datasets in a relational setting. Coordinated clustering tries to cluster different datasets such that relationships between items in each dataset are preserved. Here, we try to identify best locations to install charging stations where certain groups of people visit those locations. Candidate locations for charging stations are the ones that have specific characteristics such as low electricity load. However, we try to find those locations that have direct relationship with a specific group of people (people with high income). Obviously, type of features in people dataset is different from locations dataset. Due to this difference, and due to many-to-many relationship between locations and people we cannot use regular clustering approaches such as k-means. Our coordinated clustering framework builds upon the previous work [44] which generalizes relational clustering between two non-homogeneous datasets. This problem is a bit non-trivial since one of the relations is a many-to-many relation and another is a one-to-one relation. The final set of coordinated clusters are then used as interpretation and as a guide to charger placement.

After locating the homes of EV owners, we can determine their trajectories and their stop locations. Then, based on this data, we can estimate their travel distances. This helps us estimate charging requirements of EVs, during a day. With the help of the distribution of electricity load in the city and charging needs of EVs, we determine proper locations for installing charging stations in city with respect to specific parameters.

In addition, we come up with the actual scenario for each EV owner, who needs charging to see which locations are the best ones for him with respect to charging cost and waiting time of EV owner. After measuring additional load of each charging station, we calculate the size of storage they need in order to reduce net load. Finally, we consider the economical aspects of storage deployment.

4.4 Algorithms

As described in Section 4.3, our methodology comprises the following six major steps to determine candidate locations for charging stations:

i discovering locations' functionalities using an information bottleneck method;

ii electricity load estimation and integration with results of previous step;

iii studying the behavior of EV owners and calculating specific parameters relevant to their usage patterns;

iv candidate selection for charging stations using coordinated clustering techniques;

v finding appropriate charging stations for each user while maximizing user benefits; and

vi calculating the actual load of charging stations and storage placement.

Each of these steps are detailed next.

4.4.1 Discovering Location Functionalities

We use information bottleneck methods to characterize locations with a view toward defining the specific purpose of the location. The idea of information bottleneck methods is to cluster data points in a space (here, geography) such that the resulting clusters are highly informative of another random variable (here, function). We focus on 1779 locations in the downtown Portland area whose geographies are defined by (x,y) coordinates and whose functions are given by a 9-length profile vector $P = [p_1, p_2, ..., p_9]$, where p_i is the number of travels incident on that location for the ith purpose (recall the different purposes introduced in the previous section).

Figure 4.2 (a) describes the results of a clustering based on Euclidean metrics between locations whose results are aggregated in Figure 4.2 (b) into a revised clustering that also preserves information about activities of people at these locations. It is worth mentioning that in this part of our method, we desire to consider nearby locations and their electricity loads. Hence, the most appropriate approach for distance measurement is using Euclidean distance. The population distribution of these clusters over time is shown in Figure 4.2 (c) which reveals characteristic changes of crowds around peak hours and lunch times. One final analysis that will be useful is to evaluate each of the discovered clusters with respect to what we term as the *residentiality ratio*. The residentiality ratio for a location is the percentage of people who use that location as a home w.r.t. all people who visit that location (in downtown Portland, many locations have combined home-work profiles, and hence the calculation of residentiality ratio becomes relevant). Figure 4.2 (d) reveals one cluster with relatively high residentiality ratio among three others.

4.4.2 Electricity Load Estimation

In order to uncover patterns in electricity load distributions, we now characterize each of the discovered clusters using typical profiles gathered from public data sources such as the California End User Survey (CEUS) and other sources of usage information. Figure 4.3 presents daily electricity consumption profile across large offices, small offices, residential buildings, and colleges for one year. By clustering this data across the year, we can discern important patterns associated with different types of consumption during the year. For

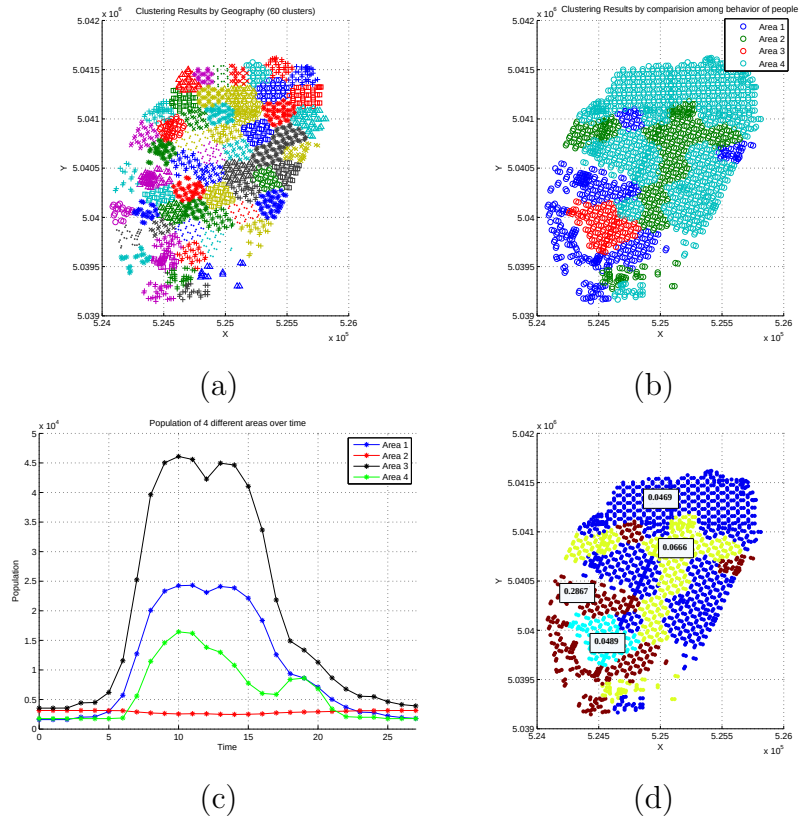


Figure 4.2: (a) Clustering downtown locations based on geographic coordinates. (b) Clustering over the previous clustering with people's activities as side-information. (c) Dynamic population of the four discovered clusters over a typical day. (d) Computed residentiality ratio revealing one primary residential cluster.

instance, in the college setting, we can discern three types of consumption patterns: holiday breaks (including summer), weekdays, and weekends.

Our next step is to compute the electricity load leveraging the above patterns but w.r.t. our network model of the urban environment. Recall that our network model is based on population dynamics but typical electricity load sources are based on square footage calculations. We map these factors using well-accepted measures, i.e., by considering the average square footage occupied by one person in a residential area as 600 sq ft [13], small office as 200 sq ft [124], large office as 200 sq ft [124], college as 50 sq ft [118], retail area as 50 sq ft [118], and other classes as 200. Further, the minimum population for an office to be considered as a large office is set to 300.

Based on some exploratory data analysis, we selected a weekday in the past (specifically, 18th March, 2011) and used the electricity load data of this day to map to the network model. Consider that in a specific hour, N people go to location l in which n_i of them come for the

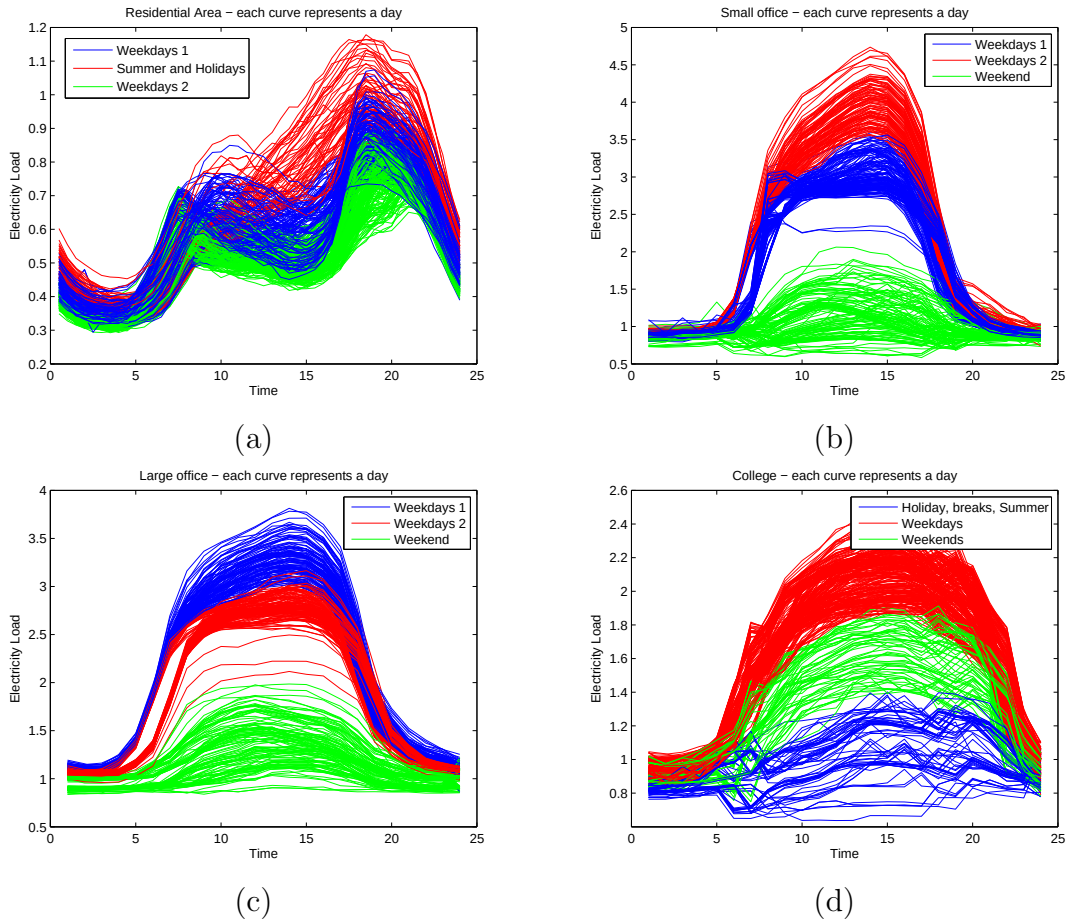


Figure 4.3: (a) Electricity usage in residential areas. (b) Electricity usage in small office areas. (c) Electricity usage in large office areas. (d) Electricity usage in college areas.

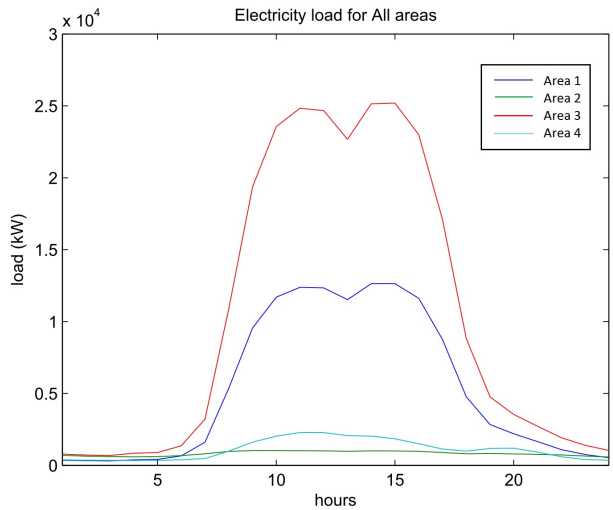


Figure 4.4: Electricity loads for four characterized location clusters.

purpose of p_i while $\sum_{i=1}^{9} n_i = N$. Then, the electricity load for that location is computed as

$$E_l = \sum_{i=1}^{9} \frac{n_i A_{p_i} E_{p_i}}{1000}, \tag{4.1}$$

where A_{p_i} is the average square footage per person for the purpose P_i and E_p is electricity consumption of building type p. It worth mentioning that E_p is from public data sources (California End User Survey (CEUS)) organized by NEC Labs. Observe that a single location can serve multiple purposes and the above equation marginalizes across all uses. For example, if there are 360 people in one location, and 10 of them are in the building for the purpose of home and 350 are for the purpose of office, the total electricity consumption of building would be calculated as $(10 \times 600 \times E_{p_{\text{home}}}/1000) + (350 \times 200 \times E_{p_{\text{office}}}/1000)$ where 600 and 200 are average square footage per person for the different categories, as mentioned earlier. The above methodology enables us to characterize electricity loads in terms of the four location clusters characterized in the previous step (see Figure 4.4).

4.4.3 Characterizing EV users

Currently only a small percentage of people use EVs, and this figure is correlated with high income. Based on [85] and [111], only 6 percent of people in the US have income more than 170,000 USD. In our synthetic dataset, 329,218 people make an income greater than 60,000 USD. To explore a hypothetical scenario, we posed the question:

> What if 6.31% of 329,218 people from Portland bought EVs? What charging infrastructure is necessary to support this scenario?

Based on [56] this is a realistic assumption if we consider different penetration scenarios in U.S in forecasted EV adoption in 2012-2022. Based on our modeling of these people's movements and patterns, we aim to identify the best locations for charging stations.

Figure 4.5 (a) gives the distribution of EV users in our potential scenario. We can notice several clusters around high-income neighborhoods. With the aid of Google Maps, we can estimate the amount of time an EV owner drives and how far he/she travels on a regular week day. Figure 4.5 (b) gives the distribution of distances traveled by these users.

Assuming EV owners charge their cars at their respective homes for beginning/end of day situations, our goal is now to identify candidate charging locations during other times. Candidate charging stations will be a critical issue in near future as the number of EVs increases [75]. Let us assume that the EV of a person P consumes E_P^C KWh energy per 100 Km. Also, assume that the battery of this vehicle can save E_P^S KWh. Then the estimated total distance that P can travel with his vehicle before he needs to charge its battery is

$$\Delta_P = \frac{100E_P^S}{E_P^C}, \tag{4.2}$$

As an example, for the Chevrolet Volt [37], with $E_P^S = 16$ KWh and $E_P^C = 22.4$ KWh per 100 Km, the EV can travel 71.43 Km before it needs to be recharged.

If the total traveling distance of P in a day is D_P then the number of times that P needs to charge his vehicle is N_P and is determined as follows:

$$N_P = \left\lfloor \frac{D_P}{\Delta_P} \right\rfloor, \tag{4.3}$$

As an example, if we assume that an EV's battery can save 16 KWh energy [37], an electric car can go for 71.43 Km before it needs to be charged [119].

Due to the long duration of charging process, we have a constraint to install charging stations only in destinations that people visit. Assume that V_L is the set of EV owners who visited location L during the day. Then $|V_L|$ is the total number of EV owners who have visited location L. However, there is a greater chance for a location to be a charging station if people with higher charge needs visit that location. Hence, the charge needs of location L is determined based on equation 4.4.

$$W_L = \sum_{P \in V_L} N_P, \tag{4.4}$$

Charging needs is an estimation to see in which locations, EV owners will probably charge their EVs. It does not mean that vehicles will certainly charge at every location. Here, we say that "there is a greater chance for a location to be a charging station if people with higher charge needs visit that location". Equation (4) does not mean that vehicles will charge at

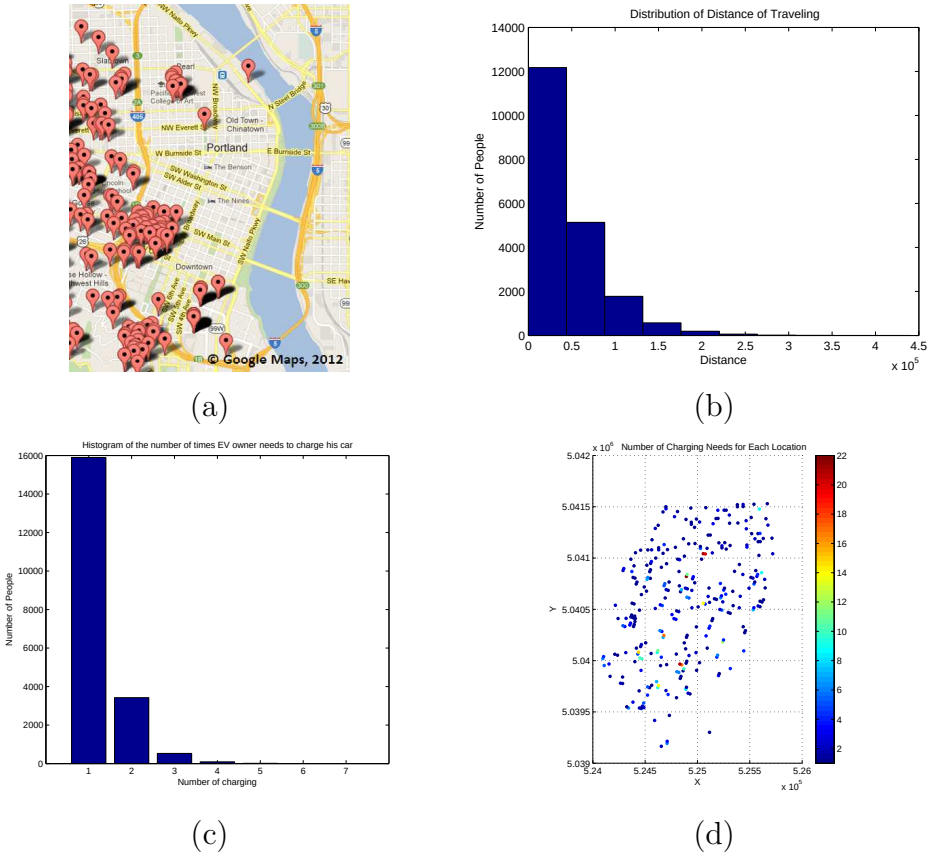


Figure 4.5: (a) EV household locations. (b) Distribution of distances people travel in their EVs. (c) Charging needs for EVs. (d) Number of charging needs (more than zero) per location.

Table 4.1: Table of notations

Variable	Description	Variable	Description
B	relationship matrix	$C_{(x)}$, $C_{(y)}$	cluster indices
$\mathbf{m}_{i,\mathcal{X}}$, $\mathbf{m}_{j,\mathcal{Y}}$	prototype vectors for clusters	$v_i^{(\mathbf{x}_s)}$, $v_j^{(\mathbf{y}_t)}$	cluster membership indicators
k_x,k_y	number of clusters	w_{ij}	contingency table entry
$w_{i.}$	row-wise counts of contingency table entries	$w_{.j}$	column-wise counts of contingency table entries
α_i	row-wise random variable	β_j	column-wise random variable
U	uniform distribution	$\mathcal{F}$	objective function

every visit to such locations. It is just a measure that indicates which locations have higher chance of experiencing charging needs. For example, if location A is visited by 10 EVs where all of them need to be charged only once during a day, then charging need of location A would be 10. On the other hand, if location B is visited by 8 EVs in which all of them need to be charged twice, then the charging need of location B would be 16, which is higher than A's. These numbers show that with higher probability people in location B require charging needs, compared with A.

Figure 4.5 (c) depicts the histogram of how many times an EV needs to be charged. Also, Figure 4.5 (d) depicts the charge needs of downtown locations.

4.4.4 Charging Station Placement using Coordinated Clustering

Since charging EVs is not an instantaneous process, it is helpful to place charging stations at those locations where people visit for an extended period of time. The average duration of stay of people in each location is an important feature in this regard. The right choice of EV charging stations thus depends on the regular electricity load of the area, the amount of time that people spend in the location, and the number of times that EV owners need to charge their vehicles [56]. Hence, based on EV owners' traveling routes during peak and off-peak hours, we can arrive at a set of candidate regions for charging stations. In this section, we describe how coordinated clustering can be used for charging station placement. Notations are summarized in Table 4.1.

Let $\mathcal{X}$ be the income dataset and $\mathcal{Y}$ be the locations datasets. $\mathcal{X} = \{\mathbf{x}_s\}, s = 1, \ldots, n_x$ is the set of vectors in dataset $\mathcal{X}$, where each vector is of dimension l_x, i.e., $\mathbf{x}_s \in \mathbb{R}^{l_x}$. Currently, our income dataset contains only one dimension. Similarly, locations dataset $\mathcal{Y} = \{\mathbf{y}_t\}, t = 1, \ldots, n_y, \mathbf{y}_t \in \mathbb{R}^{l_y}$. Locations are denoted by two dimensions (latitude and longitude) in our current database. The many-to-many relationships between $\mathcal{X}$ and $\mathcal{Y}$ are represented by an $n_x \times n_y$ binary matrix B, where $B(s,t) = 1$ if $\mathbf{x}_s$ is related to $\mathbf{y}_t$, else $B(s,t) = 0$. Let $C_{(x)}$ and $C_{(y)}$ be the cluster indices, i.e., indicator random variables, corresponding to the income dataset $\mathcal{X}$ and location dataset $\mathcal{Y}$ and let k_x and k_y be the corresponding number of clusters. Thus, $C_{(x)}$ takes values in $\{1, \ldots, k_x\}$ and $C_{(y)}$ takes

values in $\{1, \ldots, k_y\}$.

Let $\mathbf{m}_{i,\mathcal{X}}$ be the prototype vector for cluster i in income dataset $\mathcal{X}$ (similarly $\mathbf{m}_{j,\mathcal{Y}}$). These are the variables we wish to estimate/optimize for. Let $v_i^{(\mathbf{x}_s)}$ (likewise $v_j^{(\mathbf{y}_t)}$) be the cluster membership indicator variables, i.e., the probability that income data sample $\mathbf{x}_s$ is assigned to cluster i in the income dataset $\mathcal{X}$ (resp). Thus, $\sum_{i=1}^{k_x} v_i^{(\mathbf{x}_s)} = \sum_{j=1}^{k_y} v_j^{(\mathbf{y}_t)} = 1$. The traditional k-means *hard* assignment is given by:

$$v_i^{(\mathbf{x}_s)} = \begin{cases} 1 & \text{if } ||\mathbf{x}_s - \mathbf{m}_{i,\mathcal{X}}|| \leq ||\mathbf{x}_s - \mathbf{m}_{i',\mathcal{X}}||, \ i' = 1 \ldots k_x, \\ 0 & \text{otherwise.} \end{cases}$$

(likewise for $v_j^{(\mathbf{y}_t)}$). Ideally, we would like a continuous function that tracks these hard assignments to a high degree of accuracy. Such a continuous function for the cluster membership can be defined as follows.

$$v_i^{(\mathbf{x}_s)} = \frac{\exp(-\frac{\rho}{D}||\mathbf{x}_s - \mathbf{m}_{i,\mathcal{X}}||^2)}{\sum_{i'=1}^{k_x} \exp(-\frac{\rho}{D}||\mathbf{x}_s - \mathbf{m}_{i',\mathcal{X}}||^2)} \tag{4.5}$$

where ρ is a user-settable parameter and D is the point set diameter which depends on the data. An analogous equation holds for $v_j^{(\mathbf{y}_t)}$. Since our method operates over the prototypes, and uses membership probabilities to compute the probability distribution of the contingency table, it is mandatory that the functions are smooth and continuous everywhere in the system. These are the essential properties of our objective function. Any smooth and continuous membership function should work similarly. However, equation 4.5 has the advantage of involving Kreisselmeier-Steinhauser (KS) envelope function [61] that is smooth and infinitely differentiable. As a result, our objective function can be optimized using any standard local and global optimizer.

We prepare a $k_x \times k_y$ contingency table to capture the relationships between entries in clusters across income dataset $\mathcal{X}$ and locations dataset $\mathcal{Y}$. To construct this contingency table, we simply iterate over every combination of data entities from $\mathcal{X}$ and $\mathcal{Y}$, determine whether they have a relationship, and suitably increment the appropriate entry in the contingency table:

$$w_{ij} = \sum_{s=1}^{n_x} \sum_{t=1}^{n_y} B(s,t) v_i^{(\mathbf{x}_s)} v_j^{(\mathbf{y}_t)}. \tag{4.6}$$

We also define

$$w_{i.} = \sum_{j=1}^{k_y} w_{ij}, \quad w_{.j} = \sum_{i=1}^{k_x} w_{ij},$$

where $w_{i.}$ and $w_{.j}$ are the row-wise (income cluster-wise) and column-wise (locations cluster-wise) counts of the cells of the contingency table, respectively.

We also define the row-wise random variables $\alpha_i, i = 1, \ldots, k_x$ and column-wise random variables $\beta_j, j = 1, \ldots, k_y$ with probability distributions as follows

$$p(\alpha_i = j) = p(C_{(y)} = j | C_{(x)} = i) = \frac{w_{ij}}{w_{i.}}. \tag{4.7}$$

$$p(\beta_j = i) = p(C_{(x)} = i | C_{(y)} = j) = \frac{w_{ij}}{w_{.j}}. \tag{4.8}$$

The row-wise distributions represent the conditional distributions of the clusters in dataset in $\mathcal{X}$ given the clusters in $\mathcal{Y}$; the column-wise distributions are also interpreted analogously.

After we construct the contingency table, we must evaluate it to see if it reflects a coordinated clustering. In coordinated clustering, we expect that the contingency table will be nonuniform. We can expect that the contingency table will be an identity matrix when $k_x = k_y$. To keep the formulation and the implementation generic for different number of clusters in two dataset, we need to optimize the variables (cluster prototypes) in such a way that the contingency table is far from its uniform case. For this purpose, we compare the income cluster (row-wise) and locations cluster (column-wise) distributions from the contingency table entries to the uniform distribution.

We use KL-divergences to define our unified objective function:

$$\mathcal{F} = \frac{1}{k_x} \sum_{i=1}^{k_x} D_{KL} \left(\alpha_i || U \left(\frac{1}{k_y} \right) \right) + \frac{1}{k_y} \sum_{j=1}^{k_y} D_{KL} \left(\beta_j || U \left(\frac{1}{k_x} \right) \right), \tag{4.9}$$

where D_{KL} is the KL-divergence between two distributions and U indicates the uniform distribution over a row or a column. The idea of KL divergence is to estimate discrimination of information (Minimum Discrimination Information (MDI)) that leads us to use it as our divergence measure. Similar techniques that follow the MDI principle have the potential to be a part of our objective function.

Note that the row-wise distributions take values over the columns $1, \ldots, k_y$ and the column-wise distributions take values over the rows $1, \ldots, k_x$. Hence, the reference distribution for row-wise variables is over the columns, and vice versa. Also, observe that the row-wise and column-wise KL-divergences are averaged to form $\mathcal{F}$. This is to mitigate the effect of lopsided contingency tables ($k_x \gg k_y$ or $k_y \gg k_x$) wherein it is possible to optimize $\mathcal{F}$ by focusing on the "longer" dimension without really ensuring that the other dimension's projections are close to uniform.

Maximizing $\mathcal{F}$ leads to rows (income clusters) and columns (locations clusters) in the contingency table that are far from the uniform distribution as required by the coordinated clusters. It is equivalent to minimizing $-\mathcal{F}$.

The coordinated clustering formulation presented thus far can have some degenerate solutions where large number of data points in both datasets are assigned to the same cluster leading to

a huge overlap of relationships. To mitigate this, we add two more terms with the objective function.

$$\mathcal{F}_{\mathcal{R}} = -\mathcal{F} + D_{KL}\left(p\left(\alpha\right)||U\left(\frac{1}{k_x}\right)\right) + D_{KL}\left(p\left(\beta\right)||U\left(\frac{1}{k_y}\right)\right), \tag{4.10}$$

where $p(\alpha)$ and $p(\beta)$ are defined as follows:

$$p\left(\alpha\right) = \frac{1}{n_x}\sum_{s=1}^{n_x} V^{(\mathbf{x}_s)} \tag{4.11}$$

$$p\left(\beta\right) = \frac{1}{n_y}\sum_{t=1}^{n_y} V^{(\mathbf{y}_t)}. \tag{4.12}$$

It should be noted that function $\mathcal{F}_{\mathcal{R}}$ is expected to be minimized. This is the reason why $-\mathcal{F}$ is used in the formula for $\mathcal{F}_{\mathcal{R}}$.

Finally, we describe how to integrate three datasets: income, location, and station properties. Let $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$ be these three datasets, respectively. There are two sets of relationships, existing between $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Y}$, $\mathcal{Z}$. The objective function for these three datasets and two sets of relationships is defined as follows:

$$\mathcal{F}_{\mathcal{XYZ}} = \mathcal{F}_{\mathcal{R}}\left(\mathcal{X},\mathcal{Y}\right) + \mathcal{F}_{\mathcal{R}}\left(\mathcal{Y},\mathcal{Z}\right). \tag{4.13}$$

Here $\mathcal{F}_{\mathcal{R}}\left(\mathcal{X},\mathcal{Y}\right)$ refers to the objective function described in Eq. 4.10 with the income dataset $\mathcal{X}$, and locations dataset $\mathcal{Y}$. $\mathcal{F}_{\mathcal{R}}\left(\mathcal{Y},\mathcal{Z}\right)$ refers to the same objective function but input datasets are locations $\mathcal{Y}$, and station property $\mathcal{Z}$. In all our experiments, we minimize $\mathcal{F}_{\mathcal{XYZ}}$ to apply coordinated clustering between income, locations, and station property datasets.

4.4.5 Charging Station Assignment based on User Expectations

After determining candidate charging stations, we need to assess the effect of installing charging stations at those locations, and evaluate the changes in electricity load. In addition, from a business point of view, it is important to study the size of storage needed at those locations.

First, we need to evaluate candidate charging stations resulting from our co-clustering algorithm. One solution is to see whether these set of candidates are even used by EV owners. In order to understand which locations tend to be charging stations from EV owners's point

of view, we need to identify the desired locations of each person. These locations are the ones that minimize cost of charging for EV owners. On the other hand, since the process of charging an EV typically takes several hours, user tends to charge his car in those locations where he stays for at least a few hours. Obviously, from a business point of view, we not only consider those locations that meet users' criteria (charging cost), but also aim to optimize charging station in terms of electricity load and size of storage.

In this subsection, we show how to determine where users desire to charge their vehicles with respect to cost of charging and change of route for each user. In what follows, we develop an algorithm to assign charging stations to users. Of course, users have the freedom to select their charging stations. We assume that they are intuitively looking for the cheapest options. Also, we assume that users desire to minimize their detour and their waiting time (for charging). These assumptions were considered in the assignment algorithm. This assignment requires an estimate of storage sizes of charging stations. For this reason, we need to know the exact schedule of users to calculate the overall electricity load of each location. We assume that detour, cost, and waiting time are important issues in selecting charging stations for all users. (It should be noted that the goal here is to estimate the storage size, not to suggest charging stations to users.)

To the best of our knowledge, there is limited work on the "where to charge" problem in the literature. In [58], authors try to find the cheapest tour between customer destination locations to fill gas. Our work is different from [58] for a variety of reasons. For example, in our problem:

1. Sequence of stop points for each user is determined.
2. We do not have a boundary on the number of times that an EV owner can charge his car.
3. Price of charging in each location varies based on duration of stay of user in that location.
4. In some locations, car battery will be charged partially.

Before explaining our algorithm, it is worth mentioning that there are different standards for charging stations. Charging time of each EV depends on its capacity and the charging level of the charger. Levels of charging for EVs can be categorized into three levels: level 1, level 2, and level 3 (DC power). Different levels have different power consumption and hence, prices are different. Furthermore, rate of charging (the time that it takes to charge a battery for 1 KWh) is different for each level.

The algorithm for estimating desired charging locations based on user point of view is as follows:

Algorithm 3: User-based Candidate Charging Stations (UCCS)

Input: *Route*: consists of sequence of locations.

Output: ChargingStations: consists of best locations to charge as well as level of charging and time of charging at those locations.

CS= $RUCCS(Route(1), R, Route)$ `/* At first each car is fully charged (R) */`
$MinFailure = \min(\text{CS}(Failure))$;
$MinFailureSet$ =subset of CS with $Failure$ equal to $MinFailure$;
$ChargingStations = arg\min(MinFailureSet(Cost))$;
return $ChargingStations$

For each user, we invoke Algorithm 3 (UCCS). This algorithm takes the route of one user as input and calculates the best locations for charging as well as level of charging and the time of charging. Algorithm 3 calls Algorithm 4 to compute all feasible sets of charging stations in the route that user travels. Then, Algorithm 3 only retains those sets that have minimum number of failures, i.e. minimum number of times that car has to switch to gas because of empty battery. After that, it selects a set of charging stations which has a minimum cost of charging.

Algorithm 4 (RUCCS) is a recursive function for finding all feasible sets of charging stations. It takes the current location, remaining charge in the EV, and the route of user as inputs and calculates sets of candidate charging stations. This algorithm works as follows:

Let us assume that currently the EV is at location L_j, and that the available charge of battery is equal to C_j. Also, assume that d, the distance that the EV can travel from L_j without charging its battery, can be computed. This distance is determined in Line 2. Here, R is the capacity of the battery and D is the distance that EV can travel with a fully charged battery. In Line 3 of the algorithm, we determine A as the set of locations that are located on the route of EV, and are at most d meters away from L_j. It is obvious that if the last point of the route is in A, we do not need to recharge the battery (Lines 4-6). On the other hand, the EV must recharge its battery in at least one of the locations in A; otherwise after d meters, it should switch to gas.

However, when A is empty, there is no way to recharge the battery of EV. In that case, EV must switch to gas and we say that a $failure$ has happened. After a failure, in the next subsequent stop point, L_{j+1}, EV's battery must be recharged. In this case, we recursively call RUCCS for L_{j+1} (Lines 8-23). Here, $MaxC_{j+1,k}$ is the maximum possible charge of battery which is determined based on duration of stay of the car, and level of charge, k. However, because the capacity of battery is R, the actual value of the charge is calculated in line 11 and is shown by $C_{j+1,k}$. Cost of this charge is determined in line 12 by $CostC_{j+1,k}$.

If A is not empty (line 24), we must choose the most feasible location in A for re-charging the battery. Therefore, in Lines 25-32, for each location in A, and for each charging level, k, we calculate the amount of possible charge in that location ($C_{i,k}$), the cost of charging

($CostC_{i,k}$), and the maximum distance that the car can travel ($MaxD_{i,k}$), if we charge it in that location with that charging level. For each charging level, the best stop point for re-charging the car is the one that if we recharge our vehicle there, we can travel further with respect to the current location, L_j.

Choosing the best members of A for re-charging is performed in Line 33. Then, if the best stop point for charging level k is L_{idx}, we recursively call RUCCS with inputs L_{idx}, $C_{idx,k}$, and *Route* (Line 34). After returning from a recursive call of RUCCS for a location such as L_i, (Lines 13 and 34), we have several sets of stop points that are considered as feasible sets located after L_i. These sets are determined with this assumption that L_i is a charging station too. Hence, we have to add L_i to all of these sets before returning from the current iteration of the algorithm (Lines 14-19 and 35-40). Also, because we want to consider all feasible solutions to choose the best one, we have to keep all the results that are determined for different charging levels. This step is performed in Lines 20 and 41. It should be noted that time complexity of RUCCS for the ith vehicle is $O(3^{|Route_i|})$ where $|Route_i|$ is the number of stop points in the route of this vehicle. While this complexity is exponential, due to the small number of stop points that each vehicle has, the overall time complexity is acceptable.

After determining the most feasible locations from the users perspective, i.e. locations that minimize charging cost and number of failure's, we must match existing charging stations with the new locations. Since, we cannot establish charging station for each location that users want, we choose those charging stations that were extracted from Section 4.4.4 and assign each user to them based on distance to charging stations. Hence, for each charging station, we know when and how many times it will serve EVs. In order to select the best charging stations for a user, we use a *nearest charging station* assignment policy. Therefore, if the desired location for charging is L_i and S_c is the set of available charging stations, we use C_i instead of L_i where

$$C_i = arg\min distance(L_i, C_j) \quad , \quad \text{for all } C_j \text{ in } S_c, \tag{4.14}$$

where, $distance(A, B)$ measures the distance between locations A and B. It should be mentioned that any method of distance measurement (Euclidean, Manhattan, ...) can be used in this function.

With this policy, detours are minimized. After assigning charging stations, the amount of electricity load added to charging stations based on their serving time will be calculated.

Algorithm 4: Recursive Function (RUCCS)

Input: L_j is the current location, C_j is available charge of car at location L_j, and *Route* consists of sequence of locations.

Output: CS which consists of sets of candidate charging stations. Each candidate charging set (CS_i) has the following fields:
- $CS_i(points)$ is the ordered set of locations where user must charge his car.
- $CS_i(level)$ is level of charging at each location in $CS_i(points)$.
- $CS_i(costs)$ is cost of charging at each location.
- $CS_i(Failure)$ is the number of failure during trip.

```
CS = {};
d = C_j * D/R
A = set of stop points in distance d of L_j;
if Route(end) is in A then
    return CS;
end
if |A| = 0 then                    /* failure will happen and it must switch to gas */
    L_{j+1} = next subsequent stop point in Route;
    for k = 1 to 3 do
        MaxC_{j+1,k} =maximum possible charge at L_{j+1} with level k;
        C_{j+1,k} = min(MaxC_{j+1,k}, R);
        CostC_{j+1,k} =cost of charging at L_{j+1} with level k;
        CS^k = RUCCS(L_{j+1}, C_{j+1,k}, Route);
        for each candidate set, CS^k_m, in CS^k do
            CS^k_m(points) = [L_{j+1}    CS^k_m(points)];
            CS^k_m(levels) = [k    CS^k_m(levels)];
            CS^k_m(costs) = [CostC_{j+1,k}    CS^k_m(costs)];
            CS^k_m(failure) = CS^k_m(failure) + 1;
        end
        CS=CS ∪ CS^k;
    end
    return CS;
else
    for each point, L_i, in A do
        for k = 1 to 3 do
            MaxC_{i,k} =maximum possible charge at L_i with level k;
            C_{i,k} = min(C_j − (dist(L_j,L_i)*R)/D + MaxC_{i,k}, R);
            CostC_{i,k} =cost of charging at L_i with level k;
            MaxD_{i,k} = D/R * C_{i,k} + dist(L_j, L_i);
        end
    end
    for k = 1 to 3 do
        idx = arg max_{L_i∈A}(MaxD_{i,k});
        CS^k = RUCCS(L_{idx,k}, C_{idx,k}, Route);
        for each candidate set, CS^k_m, in CS^k do
            CS^k_m(points) = [L_{idx,k}    CS^k_m(points)];
            CS^k_m(levels) = [k    CS^k_m(levels)];
            CS^k_m(costs) = [CostC_{idx,k}    CS^k_m(costs)];
            CS^k_m(failure) = CS^k_m(failure) + 1;
        end
        CS=CS ∪ CS^k;
    end
    return CS;
end
```

4.4.6 Storage Placement

In previous section, we determined profile of electricity load at each location before and after charging station deployment. Profile of electricity load after installing charging stations is determined based on number of cars that are charged at each location and their corresponding level and duration of charging. On the other hand, each location has a predetermined capacity which is the maximum electricity load that it can tolerate. When electricity load of a location increases and goes above its capacity, we need to place storage to meet the electricity demand of that location. In this regard, the efficiency of storage is also important. Here, we assume that the desired utilization of storage in all locations is 80% i.e. at most 80% of the capacity of a storage is used in a day. That ensures us that storage will not discharged to no more than 80% of total capacity. Due to the small size of storage at some locations we aggregate storages of nearby locations. For this purpose, we use DBSCAN [32] to locate dense areas and calculate the needed storage size of each cluster as a summation of storages over all locations in that cluster.

From a business point of view, placing storage at a charging station must have a adequate revenue for storage owners. In addition, putting storage at locations is advantageous to city in terms of reducing the peak of electricity load in urban area.

To investigate the revenue of storage units, we consider each charging station in turn and compute the revenue of storage. Here, revenue refer to the amount of funds that storage owners will save from selling energy to consumers. Revenue can be achieved by selling energy during the day and recharging the storage during the night (with off-peak rate). In addition, to observe profile of charging stations based on their load curves, we use the clustering algorithm introduced in [132], i.e., the K-Spectral Centroid (K-SC) algorithm for time series data using a similarity metric invariant to scaling and shifting. In [132], authors apply adaptive wavelet-based incremental approach to K-SC to use it for large datasets. K-SC clustering proved to be an effective clustering method when scaling is not important. By applying this method, we can understand different types of charging stations based on their load curves and finally, locate the best locations to put storage in order to get high revenue.

4.5 Results

Figure 4.6 describes the coordinated clustering scenario. As illustrated in this figure, we use three datasets: people (income), coordinates (x,y) of location, and features (load, charge need, stay) of location. First dataset contains information about income of people and second dataset has information about the geographic coordinates of each location. However, the third dataset contains the characteristics of locations for charging station placement. Electricity load of buildings, charging need of people in that location and duration of stay in each location are three features in this dataset.

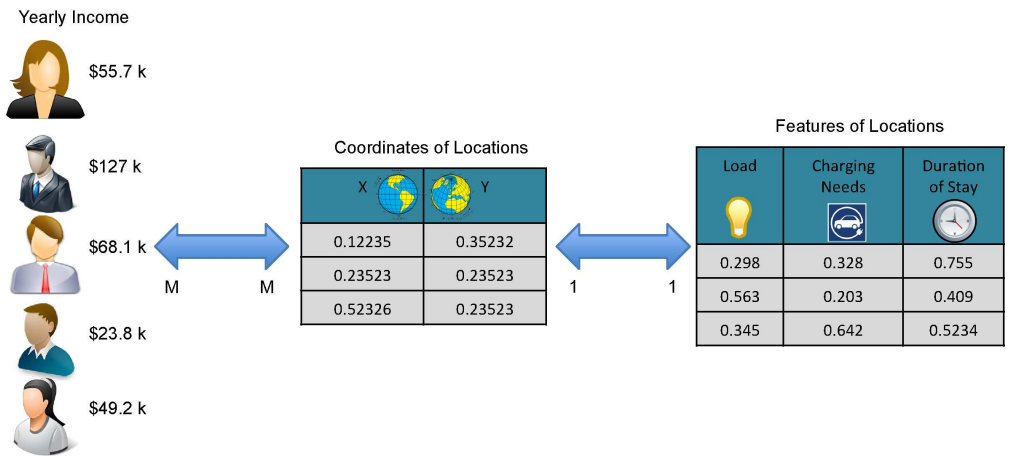


Figure 4.6: Coordinated clustering schema.

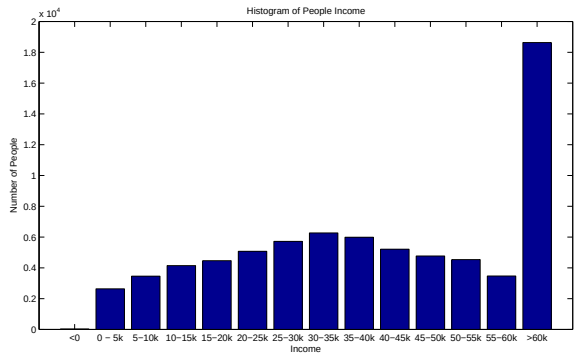
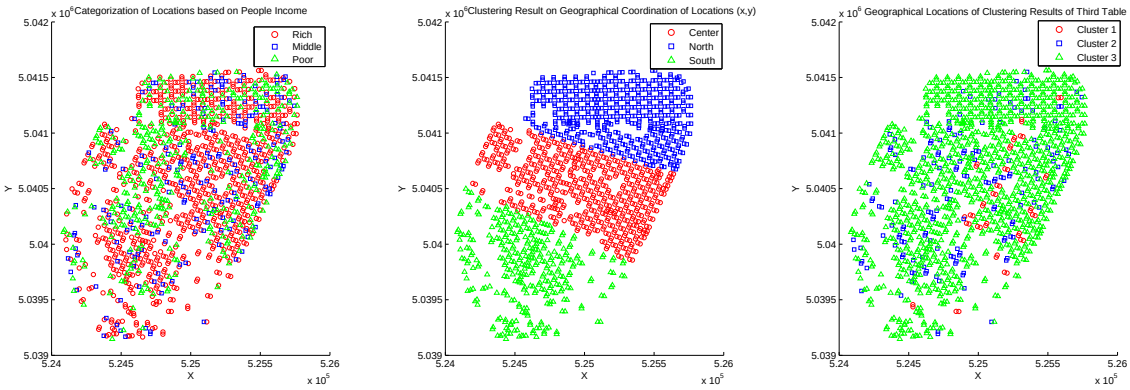


Figure 4.7: Distribution of income.

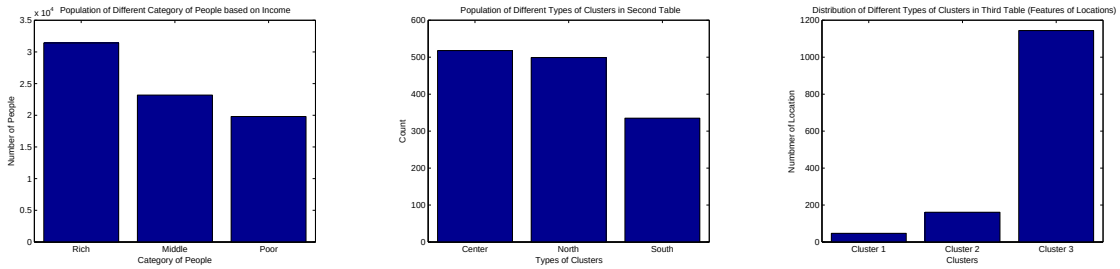
We begin with some preliminary observations about our data. Figure 4.7 depicts the distribution of people based on their income, indicating that a significant number of people have high income, leading to a large number of EV users. We experimented with coordinated clustering involving many settings. Figure 4.8 depicts three clusters of locations based on each of the attribute sets in our schema. Note that because the clusters are mapped onto (x,y) geographical locations, locality is apparent only in Figure 4.8 (b).

Profiles of these clusters are described in detail in Figure 4.9. Of particular interest to us is the view from the perspective of EV attributes, i.e., Figure 4.9 (c). Details of these clusters are explored in greater detail in Table 4.3. Ideal locations for charging stations for EVs must have a relatively low current electricity load (to accommodate the installation of charging infrastructure), high charging needs (population profiles), and high staying duration [56]. As can be seen from Table 4.3, cluster 2 from the third dataset fits these requirements. Greater insights into the three clusters from the viewpoint of these three attributes is shown in Figure 4.10, supporting the choice of locations in cluster 2 as the right candidates for locating charging stations. As we mentioned before, we try to identify locations with specific features while certain group of people (people with high income) visit those locations. Although based on the clusters of first dataset (people), we must choose locations where mostly people with



(a) (b) (c)

Figure 4.8: Results of coordinated clustering (3 clusters) when viewed through the attributes of each domain: (a) Clusters based on income. (b) Clusters based on geographical location. (c) Clusters based on EV charging station attributes.



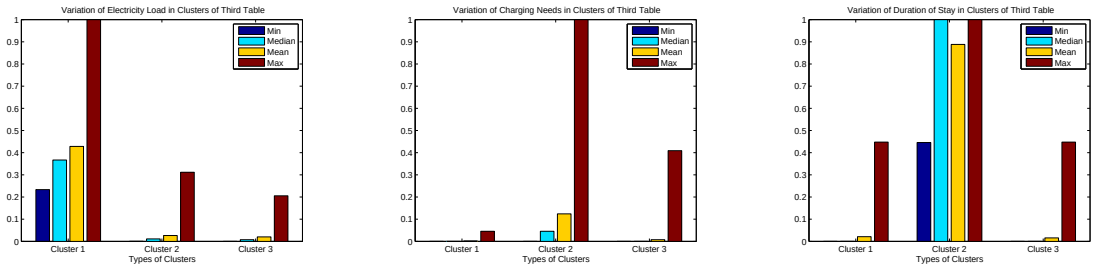
(a) (b) (c)

Figure 4.9: Profiles of clusters obtained from coordinated clustering w.r.t. each of the three domains: (a) Income attributes. (b) Location attributes. (c) EV charging station attributions.

greater salary affordances visit, the distribution of high income versus low income people in clusters 1, 2 and 3 in third dataset (locations) are almost similar. This is illustrated in Table 4.2. With respect to distribution of high income people, cluster 1 is better to be selected. However, cluster one is not a good choice for installing charging stations because 45% of its locations are those with high electricity load. Between other two clusters (cluster 2 and cluster 3), cluster 2 is better because it has low electricity load, high charging need, and high duration of stay.

With the aid of clustering, we can predict which locations are the best candidates to install charging stations. However, the effect of installing charging stations in these locations on other metrics such as the price of charging and electricity load of buildings must be evaluated.

Since we are looking only at downtown area of Portland, we do not have any information about exact location of other charging stations outside of downtown. Here, we padded our



(a) (b) (c)

Figure 4.10: Detailed inspection of clusters for their suitability for locating EV charging stations: (a) Distribution of electricity loads. (b) Distribution of charging needs. (c) Distribution of duration of stay. An ideal cluster should have (low, high, high) values respectively, suggesting that cluster 2 is best suited.

Table 4.2: Profiles of clusters in third dataset (location's features)

Cluster	% of People with High income	% of Locations with High Elec. load	% of Locations with High Charging need	% of Locations with High Stay
1	0.43	0.45	0	0.02
2	0.41	0.06	0.15	0.88
3	0.41	0.05	0.01	0.01

downtown area by 500 meters from each side (if we suppose downtown has a rectangular shape). Then, assuming that those cars in the padded area can be served by our current charging stations, we run the algorithm 3 for each car. The distance between charging station and current location of car must be minimized because charging at charging station with Level 1 or 2 will take several hours and people prefer to charge their cars at those locations that they stay longer. In reality, users can charge their cars anywhere in vicinity ($\sim$1 mile) of their desired buildings (e.g. the driver can park his car at nearest charging station and walk to his office). Furthermore, we need to have information for two types of movements (riding to charging station, and walking to office). Since, the distances are not too long, using Euclidean distance to measure distances is not troublesome and makes computations easier. Furthermore, the actual information about roads of the area was not available to use and the dataset consists only origin and destination of each movement.

Specifications of three levels of charging for Portland are summarized in Table 4.4 based on

Table 4.3: Characteristics of clusters in third dataset (location's features)

Cluster	Elec. Load	Charging Need	Stay Duration
1	High	Low	Low
2	Low	High	High
3	Low	Low	Low

Table 4.4: Characteristics of charging stations

Level	Description	Elec. Load(kW)	Cost(¢/kWh)			Time(h)
			on-peak	mid-peak	off-peak	
1	110V outlet,16 Amp	2.2	16.62	10.85	7.77	8
2	220V charger, 16 Amp	3.3	16.62	10.85	7.77	4
3	400V DC, 125 Amp	50	10.89	6.36	3.63	0.5

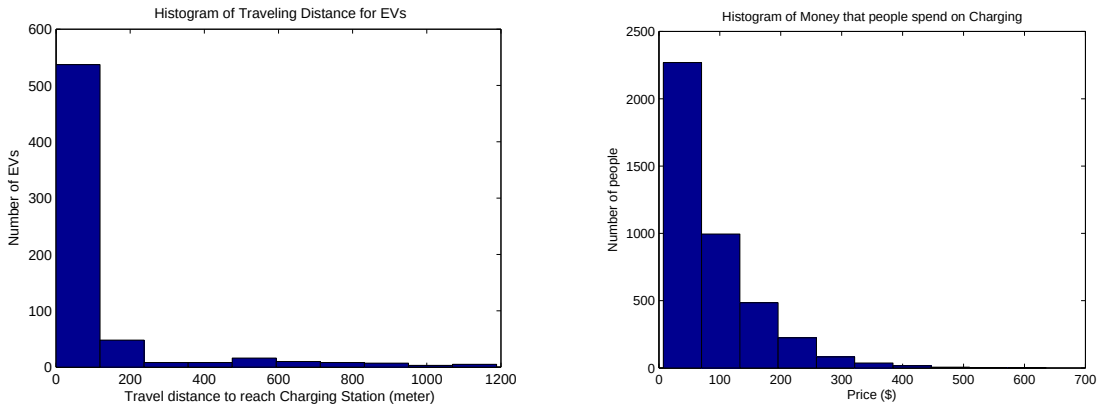
PGE [90, 91]. From [91], Schedule 7 is chosen for level 1 and 2 and Schedule 32 is selected for level 3. It is worth mentioning that prices (tariff rate) are based on time of use policy (TOU). The definition of On-peak, Mid-peak, and Off-peak is inspired from the electricity loads in our dataset:

- On-peak: 6 AM to 10 AM and 5 PM to 8 PM
- Mid-peak: 10 AM to 5 PM and 8 PM to 10 PM
- Off-peak: 10 PM to 6 AM

Prices at Table 4.4 are for both buying electricity from the grid and from charging station (by EVs). The type of charging depends on time of stay. If an EV stays for 8 hours, it can charge by charging level 1 which is cheapest option. If an EV stays for 4 hours, it can use level 2 charger whereas if the EV needs to be charged in 30 minutes, it can use the level 3 (DC) option. Price of charging in level 3 is very high compared to level 1 and level 2. For example, cost difference (a complete charging) between charging by DC and level 1 would be $50 * 10.89 - 2.2 * 16.62 = 507.936$ cents or \$5. Hence, the overall impact of level of charging (1, 2, and DC) is very high on charging stations and on users in cost and electricity load points of view.

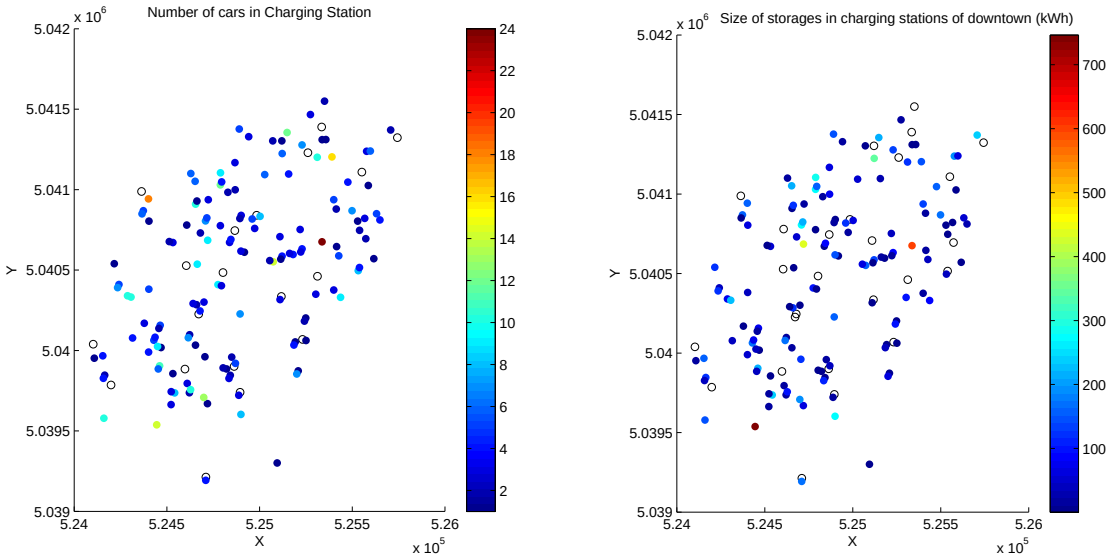
Experiments show that average distance traveled by each car is 8.4881 meters and that the maximum distance traveled in this experiment is 1188.2 meters. Figure 4.11 (a) depicts the histogram of distance between location of current stop point and available charging station. This result is promising since we considered part of the boundaries of downtown while there might be a charging station in that area. The number of charging stations based on our clustering algorithm is 161 while number of locations that people liked to charge their cars is 367. The histogram of expenses that all EV owners in Portland will pay daily for charging is shown in Figure 4.11 (b).

Also, number of cars that are served at each charging station is important from a business point of view, to study revenue of charging station owners. As Figure 4.12 (a) shows this is zero for some charging stations (black circles) and they can be removed from consideration as charging station candidates. Based on this figure, we can place appropriate charging infrastructure at those locations that serve certain number of cars.



(a) (b)

Figure 4.11: (a) Histogram of distance between current stop point of location and available charging station (meter). (b) Histogram of expenses people pay for charging during a day.



(a) (b)

Figure 4.12: (a) Number of cars that served by each charging station. (b) Size of storage in charging stations (kWh). Note that some charging stations (black circle) are useless and can be removed.

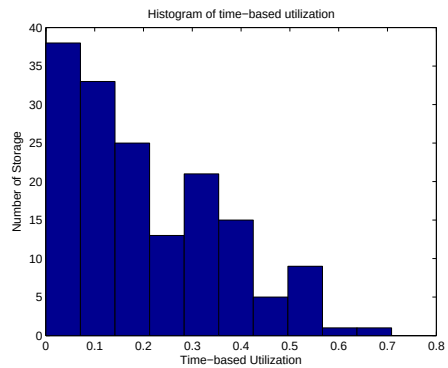


Figure 4.13: Histogram of time-based utilization.

It should be noted that the number of failures in our algorithm is 48. This highlights the number of cases where an EV must switch to gas in order to continue its route. Experiments show that all of these 48 cases was due to the nature of our dataset, i.e. distance between locations was more than maximum possible distance of travel with a full battery.

Those charging stations that provide service to cars will add extra load to the location. This load might be more than the capacity of the location. Here, we assume that maximum load of one location during a day is equal to its capacity. We must place storage to those locations that require extra electricity. For determining the size of storage at each charging station, we must look at values of location's capacity and electricity load after adding EV. To compute size of storages, we would like to assume that storage will not discharge to more than 80%. For example, 50kWh should be selected for a net daily load of 40kWh. The size of required storage should be calculated from the area below the curve of new electricity load (kW $\times$ hr) and above the capacity (net peak load)(kW). Typically, storage will be charged at night and used during the day and it should be sized to cover a day's net load.

Figure 4.12 (b) shows how many locations need to have storage. Again, black circle means there is no need for storage at this location. Based on our assumption, utilization is 80%. However, the time-based utilization (i.e. the percentage of time that storage has been used in a day) is shown in Figure 4.13. Obviously, the value of time-based utilization cannot be 1 in this case, because storage needs to be recharged over night to be used for the next day.

After determining storage sizes, we can aggregate them to minimize number of storage units. This aggregation is based on vicinity of locations. Hence, we used DBSCAN to find dense areas and take summation of the storage size over all locations in each cluster. This is shown in Figure 4.14. In this figure, small black circles represent those locations where they could not be grouped by other locations and considered as noise in DBSCAN algorithm. small red dots represent center of each group. As an example, in upper-right side, overal storage size of two locations (purple circles) is 252 kWh.

To study the amount of saving for each storage, we assume that each storage will charge at night with off-peak rate. Hence, during the day in on-peak and mid-peak hours, storage

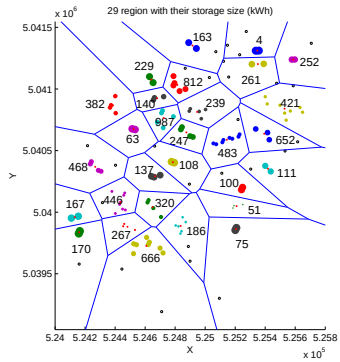


Figure 4.14: Aggregated regions: Value of storage size (per kWh) is shown for each region (sum of all storages).

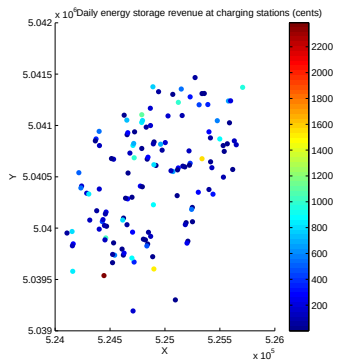


Figure 4.15: Revenue of energy storage at charging stations.

owners will sell energy to consumers (EV owners). Hence, the difference between price of selling and price of recharging will be considered as revenue of storage. Figure 4.15 shows energy storage revenue for each charging station that has storage unit. This value of revenue is calculated for one typical day.

To observe the profiles of charging stations based on their load curves after adding EVs and after adding storage, we used the K-SC clustering approach which has been described earlier. Here, the value of electricity load before adding EV, after adding EV, and after storage deployment during 24 hours were considered as a vector of 24*3 elements. The profiles of charging stations are categorized into 4 clusters which the prototype of each cluster is shown in Figure 4.16. It should be noted that this clustering is invariant to shift and scale and that is why the value of load after storage deployment is higher than maximum value of load before considering EV. Figure 4.17 depicts an example of actual curves for one charging station in cluster 1. It is obvious that storage deployment will ensure that the value of electricity load will not go higher than the capacity at each location. Figure 4.16 is important in understanding the behavior of charging stations. Also this figure is helpful in deciding between using a mobile storage unit and a stationary one.

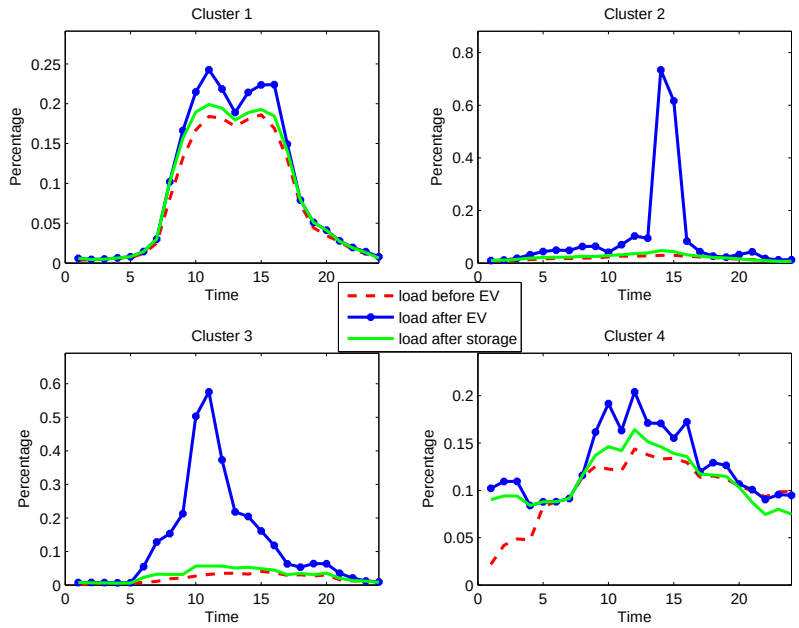


Figure 4.16: Profiles of four types of charging stations based on load curves.

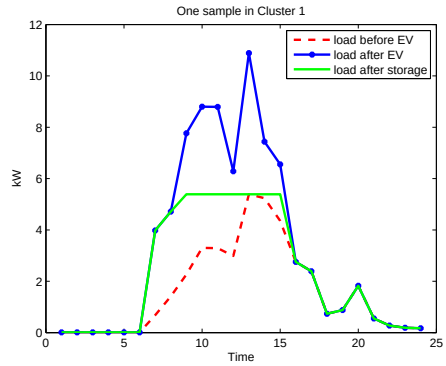


Figure 4.17: Profile of one charging station in cluster 3.

In Figure 4.16, charging stations in clusters 1 and 4 have little impact on the peak load, whereas those in cluster 2 and 3 significantly increase peak demand of the system. Therefore, using energy storage for charging stations in cluster 2 and 3 would make more sense than in clusters 1 and 4. Based on number of charging stations in each cluster, 43% of charging stations (in cluster 2 and 3) are candidates for storage deployment. On the other hand, if there is no possibility of adding energy storage, charging stations in clusters 1 and 4 would have much less impact on the grid and will be accepted by utilities with less opposition. Also, one can deploy mobile storage units for charging stations in clusters 1 and 4.

Figure 4.18 shows the amount of daily revenue achieved by storage deployment for each cluster. In this figure, locations in cluster 2 and 3 have highest revenue compared with cluster 1 and 4. Total revenue in cluster 1 and 4 is 6547.2 cents while total revenue in cluster

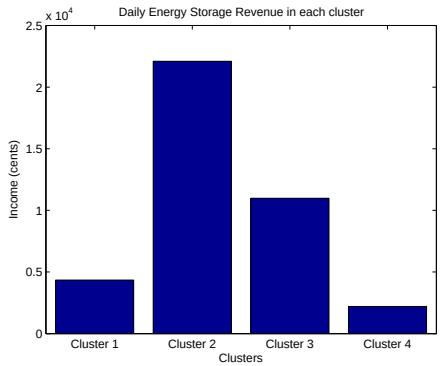


Figure 4.18: Revenue of energy storage at each cluster.

2 and 3 is 33081.0 cents. Based on this, one can justify using stationary battery storage in candidate charging stations (cluster 2 and 3).

4.6 Discussion

Electrical vehicles are going to become more popular in near future. We have demonstrated a systematic data mining methodology that can be used to identify locations for placing charging infrastructure as well as storage infrastructure as EV needs grow. In addition, we identified candidate locations for deployment of stationary energy storages to utilize existing electricity infrastructure. The results presented here can be generalized to a temporal scenario where we accommodate a growing EV population and to design charging infrastructure to accommodate additional scenarios of smart grid usage and design.

The methodology presented in this chapter mostly incorporates demand data from the electricity infrastructure and future work would incorporate information from the electricity supply side too. Information such as loading level of electricity feeders and remaining excess capacity of feeders for EV charging stations can be integrated in the methodology to improve the placement of EV charging stations. Also, there are several measures that were not considered here, such as life of battery, peak shaving reduction, adding PVs to current system, and details of economic analysis in charging stations and energy storage deployment. Incorporating these aspects is a direction of future work. Finally, the analysis presented here integrates a small range of datasets, each of which has adequate coverage over regions of interest. To overcome regions of data sparsity, we could employ the use of surrogate models like Gaussian processes [95], which can enable the integration of a greater variety of datasets.

Chapter 5

Electric Taxi Fleet in Urban Area

5.1 Introduction

In large metropolitan areas a significant load of the transportation is handled by taxis and hence, replacement of fuel-based taxicabs with *electric taxis* is considered as an important step toward a green city. There are several issues that must be addressed before ushering in electric taxi fleet adoption such as investigating the feasibility of replacement of taxis with EVs. In a comprehensive planning effort, it is crucial to consider economic factors in the deployment of electric taxis and charging infrastructure design to ensure financial feasibility as well as long-term economic growth. Another important issue for electric taxis is the design and placement of charging infrastructure. Due to the long distances traveled by taxi drivers and their limited times to rest, fast chargers must be provided in a large scale, properly distributed in the city. Proper placement of charging stations results in optimal distribution of electricity load, maximization of revenue of service providers, and lead to increased availability of charging stations, and reduced range anxiety.

In this chapter, we investigate the efficiency of electric taxi deployment on a mass scale in Manhattan. To summarize, our contributions of the chapter are threefold:

- Proposing a data-driven method to determine which taxis are suitable to be replaced by EVs and number of required charging stations.
- Proposing a data-driven approach to find ideal locations for charging stations based on various data sources such as electricity usage and points of interest.
- Developing an online recommendation algorithm for suggesting charging station to taxi drivers.

5.2 Related Works

In this section, we provide an overview of the relevant researches reported in the literature.

Electric Vehicles: A *life cycle analysis* of possible alternative vehicle technologies for London taxis has been performed in [9], indicating that EVs are more efficient cars than plug-in hybrid electric vehicles and fuel-based taxis. Yang et al. [133] proposed an approach to maximize the profit of electric taxis with respect to electricity prices and time-varying income. In another work, EVs are integrated with virtual power plant to balance the electrical grid by acting as an energy storage [54]. Their proposed algorithm determines which EVs should absorb excess electricity and which ones should provide energy during shortages in a specific time.

Charging Station Placement: Placement of charging station infrastructures for EVs is a non-deterministic polynomial-time hard problem [62]. In this regard, in [81] we proposed a coordinated clustering approach to find optimal placement of charging stations with respect to city movement and electricity consumption. Furthermore, after proposing an algorithm to assign each EV to the best charging stations, we determine the required size of energy storage units for charging stations for peak-shaving. Other methods such as mixed integer programming [21], genetic programming [42], and Bayesian game [72] have been proposed to find optimal placement of charging stations.

Charging Guidance for EVs: Proposing the optimum charging schedule and charging station for EVs that need to be charged has been studied using different techniques such as adaptive particle swarm optimization [86], genetic algorithm [4], and optimization of hybrid EVs in case of having time-varying electricity price [66]. Zhang et al. [138] studied delay-optimal charging scheduling of EVs at a charging station with multiple charge points with the ultimate goal of minimizing the mean waiting time for EVs under the long-term constraint on the cost using Markov Decision Process (MDP) framework. On the other hand, Lu et al. [71] proposed a dispatching strategy for fleet of electric taxis considering taxi demand, the state of charge of battery, and the availability of battery charging stations in order to lower the waiting time of recharging. Maalej et al. [73] proposed an optimal battery charging schedule and planning system for long trips. Their method predicts vehicle energy consumption based on historical driving power and an online mass estimation. Their cost function is minimized with respect to battery degradation, trip duration, and charging energy cost.

Energy management and planning: In SUM, management and proper planning of energy devices such as storage units and EV batteries is an important task. There have been several works in this regard. Styler and Nourbakhsh [114] proposed a data-driven adaptive optimization approach for real-time energy management task on hybrid electric vehicles. In [83], we proposed a data-driven approach to characterize battery storage systems and predict their efficiency over time. Li et al. [64] proposed a fuzzy logic energy-management system using battery state of EVs to ensure that the engine operates in the vicinity of its maximum

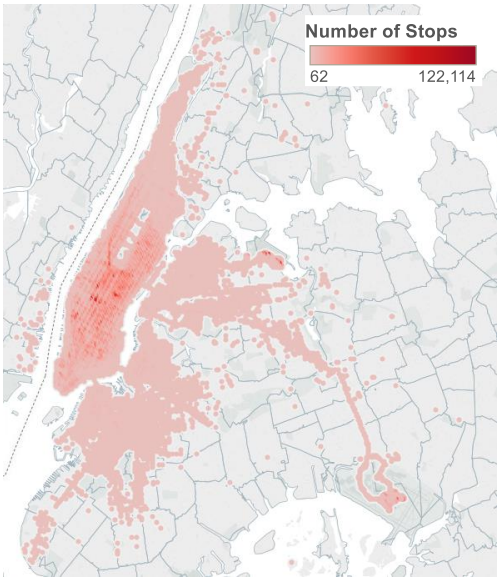


Figure 5.1: Distribution of taxi stop points in NYC.

fuel efficiency region while preventing the battery from over-discharging. Tianheng et al. [120] presented a new control strategy to improve fuel economy of EVs by predicting energy demand using Neural Networks and tracking an SOC reference to distribute the energy between the engine and the motors.

Jin et al. [52] studied EV charging scheduling problems from a customers perspective by considering the demand and aggregators revenue. They proposed linear programming (LP) based optimal schemes for the static charging scheduling scenarios and effective heuristic algorithms for the dynamic ones. Chen et al. [23] proposed an online intelligent energy management controller using neural networks and dynamic programming methods to improve fuel economy of EVs. They have analyzed different driving conditions (e.g. in highway or urban). In another work, Larsson et al. [63] focused on ways to reduce the computational demand and memory requirement of dynamic programming (DP) in energy management problem of EVs. Valogianni et al. [125] proposed an adaptive management of EV storage algorithm to adjust EV charging using reinforcement learning to learn individual behavior and to schedule charging on a weekly horizon.

Kalesar and Seifi [55] developed a distribution system planning algorithm to obtain optimal number and location of substation using genetic algorithm. On the other hand, Bose in [100] presented an algorithm to determine the optimal locations of base stations to provide maximum possible coverage without performing an exhaustive search.

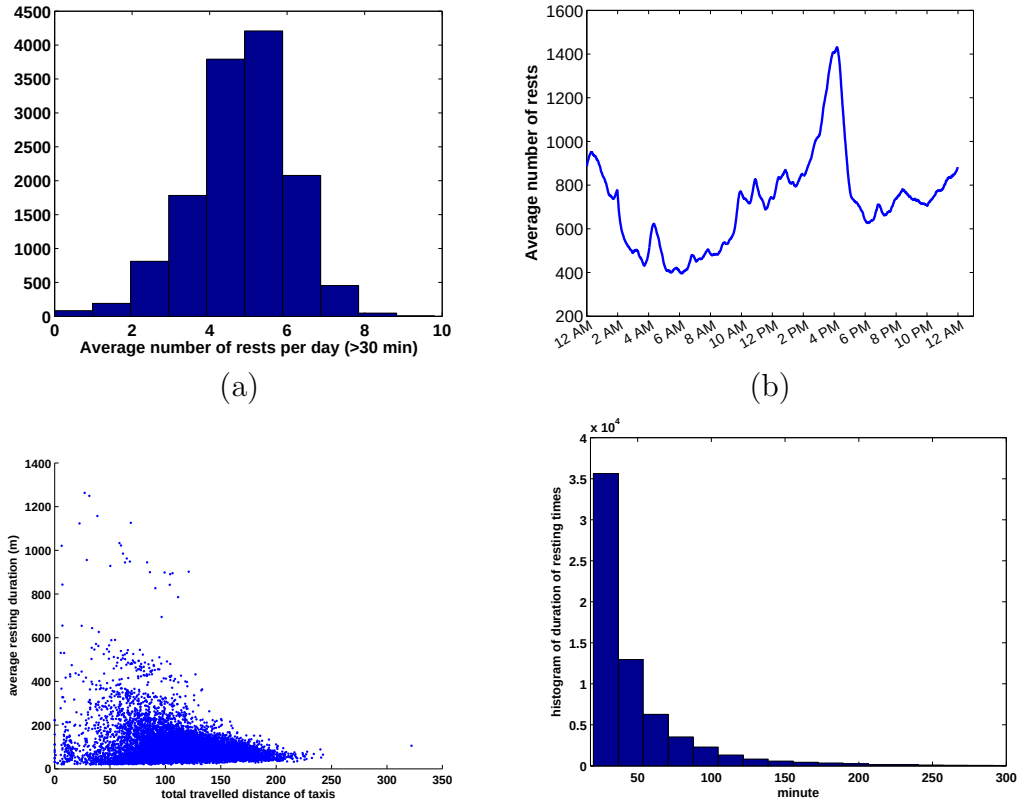


Figure 5.2: Various statistics of rests for taxis: (a) Histogram of average number of rests. (b) Distribution of start time of rest during a day. (c) Resting durations versus daily traveled distance of taxis. (d) Histogram of rest duration.

5.3 Overview of Framework and Datasets

High adoption of EVs need specific considerations regarding electricity consumption of the city as well as extra maintenance and installation costs. Therefore, with respect to the trade-offs between the efficiency of infrastructures and devices and the budget of city planners, various issues must be addressed before high adoption of EVs in a taxi fleet of large metropolitan areas. In this work, we propose a two-step data-driven framework for the deployment of electric taxis. In the first step, using integer linear programming, we search among fuel-based taxis to find the best candidates to be replaced by EVs. Furthermore, we determine the ideal number of required CSs. In the second step, we aim to find the optimal places to install CSs with respect to minimizing detours and electricity consumption. We also develop a recommendation system to suggest the best available CS for each car in an online manner. There are various parameters, such as vehicle expenses and electricity costs, that should be involved in electric taxi adoption problem. Table 5.1 shows the symbols and specifications used in this chapter. The specific value of these parameters are introduced in

Table 5.1: Table of notations

Symbol	Description
C_Δ	Extra payment per trip for electric taxis
n_i	Number of trips made by taxi i
$\mathcal{T}_E$	Set of electric taxis
B_F ,B_E	Price of fuel-based taxi and electric taxi
Y_l	Expected life of car (years)
ϕ	Size of full charge (kWh)
γ_F , γ_E	Miles per gallon or per full charge of battery
C_{Sell} , C_{Buy}	Cost of electricity at charging station (per kWh)
C_F , C_E	Price of fuel per gallon or a full battery charge
ν_F , ν_E	Maintenance cost of fuel-based car and EV (per mile)
d_i	Avg. distance traveled by taxi i
ν_C	Avg. daily maintenance/installation fee of a CS
m,k,β	Number of taxis, charging stations, and areas in city
e_i	Indicates candidacy of ith taxi to be electric one
K,M	Maximum charging stations and electric taxis
E_{curr}	Total daily electricity consumption of city
θ	Tolerance of extra electricity usage
r_i	Average number of rests in a day
ρ	Tolerance of number of rest
D_i	Avg. distance between area i and points of interest
E_i	Total daily electricity consumption of area i
Λ_i	Land price index in area i
k_i	Charging capacity of area i
(ζ_x^i, ζ_y^i)	Geographical coordinate of center of area i
(L_x^j, L_y^j)	Geographical coordinate of the jth point of interest
l	Number of points of interest
K_{area}	Maximum allowable charging station in one area
δ , $v_{i,j}$	Reachability threshold and reachability indicator
$\mathcal{S}_C^i$, $\mathcal{S}_T^i$	Status of the ith charging station and the ith taxi

Table 5.2.

5.3.1 Dataset Description

According to a survey conducted in 2013, on average, passengers are willing to pay 55 cents per trip to ride electric taxis [78]. Furthermore, based on the same survey, in our experiments, we assume that EV's model is Nissan Leaf and the efficient life time of a taxi is 5 years. In this chapter, we use the following datasets:

Electricity Usage: Charging station placement and EV adoption in a large metropolitan area cannot be performed without appropriate knowledge about the electricity consumption

Table 5.2: Parameter settings

Symbol	Value	Symbol	Value	Symbol	Value
ν_F	5.4 cents	ν_E	4 cents	ϕ	24 kWh
γ_F	29	γ_E	75 miles	Y_l	5 years
B_F	\$20000	B_E	\$30000	C_Δ	55 cents
C_{Buy}	6 cents	C_{Sell}	50 cents	nuC	12990/365
C_F	3.6 \$	C_E	$\phi\ Csell$	β	106
ρ	1	θ	0.1		

in the city. In this chapter, we use electricity consumption data of New York city per ZIP code, which is available online[1].

Land Index: The average value of residential houses (either to rent or to buy) per Zip code is available online[2] and can be used as an indicator of the land prices in Manhattan.

NYC Taxicab Trips: We used dataset of taxicabs of NYC in 2013 which is available online[3]. Information such as pick-up and drop-off locations as well as time, trip distance, price of the trip, and passenger counts have been logged in this dataset. For EV adoption and CS placement phases, we use one month of data (March 2013) which includes more than 15 million trips. The framework can be easily extended to the whole year. Also, the first 10 days of April has been used for evaluation phase of this work. The pick-up and drop-off locations of taxis for one month of data are illustrated in Fig. 5.1. As this figure illustrates, some locations such as Manhattan, Brooklyn, Queens, and JFK Airport have high density of taxis. Since battery charging is a time consuming task, the rest time statistics of the drivers should be included in the model. Fast battery charging of an EV takes about 30 minutes and hence, we assume that a rest time should be at least 30 minutes long. A few statistics regarding the taxi rest times are illustrated in Fig. 5.2. Fig. 5.2(a) shows the histogram of number of rests that a taxicab takes during a day while Fig. 5.2(b) shows the distribution of rest times in a 24-hour day. It can be observed that most of the taxis take rests five times a day and a large number of rest times are concentrated around 4:00 PM which is known as the current shift-change in NYC. Fig. 5.2(c) shows the monthly average duration of resting to the total traveled distance per each taxi. As this figure depicts, a car with longer trips has shorter resting times which increases the need for fast charging. The histogram of rest time duration is shown in Fig. 5.2(d). As this figure indicates, most of the times, this amount is less than 50 minutes.

[1] http://energyzip.org/
[2] http://www.city-data.com/zipmaps/New-York-New-York.html
[3] http://www.andresmh.com/nyctaxitrips/

5.4 Electric Taxi Adoption

In order to replace a set of fuel-based taxis with electric ones, we need to maximize the overall replacement revenue, which is determined by taxis and the charging infrastructure. While EVs are more expensive than their fuel-based equivalents, their energy and maintenance costs are lower. Furthermore, studies showed that passengers are willing to pay more to ride electric taxis [78]. Hence, the overall revenue of each taxi is calculated as the sum of the revenue that is made by owning an EV (i.e. lower charging cost, lower maintenance cost, and fare increase) compared to a fuel-based vehicle. The revenue of charging station owner is determined based on the price of buying electricity from the grid and selling it to taxi drivers as well as installation and maintenance costs. Thus, the overall revenue of electric taxi deployment per day is:

$$\sum_{i \in \mathcal{T}_E} \Bigg\{ C_\Delta n_i + \frac{B_F - B_E}{365 Y_l} + \Bigg(\frac{\phi}{\gamma_E}(C_{Sell} - C_{Buy}) + \frac{C_F}{\gamma_F} - \frac{C_E}{\gamma_E} + \nu_F - \nu_E \Bigg) d_i \Bigg\} - \nu_C k \tag{5.1}$$

where $\mathcal{T}_E$ is the set of the deployed electric taxis, n_i is the average number of daily trips, and d_i is the average daily traveled distance made by the ith taxi. Also, $\frac{B_F - B_E}{365 Y_l}$ is the difference between the average daily price of an electric taxi and a fuel-based one while $\frac{C_F}{\gamma_F} - \frac{C_E}{\gamma_E}$ and $\nu_F - \nu_E$ are the difference between the energy price and maintenance cost of EVs and fuel-based cars per mile, respectively. Furthermore, $\frac{\phi}{\gamma_E}(C_{Sell} - C_{Buy})$ is the revenue of charging infrastructure per traveled mile and $\nu_C k$ is the total maintenance and installation cost of k charging stations per day. While revenue and loss of gas station providers may be included in the formulation, we do not consider them in this study. In fact, both drivers and CS owners need incentives to participate in electric taxi deployment and hence, gas station owners are out of the scope in this scenario. Furthermore, in this study the price of selling and buying electricity is static due to the insufficient data. However, the model can be easily extended to include dynamic prices.

We assume that due to the budget constraints, there are upper bounds on the total number of EVs and CSs that can be adopted. On the other hand, due to the grid limited capacity, energy consumption should be limited. Moreover, since taxi drivers prefer to charge their batteries in their rest times, each driver should have enough rest times during his shift to be able to handle the charging requirements. These constraints, together with the revenue

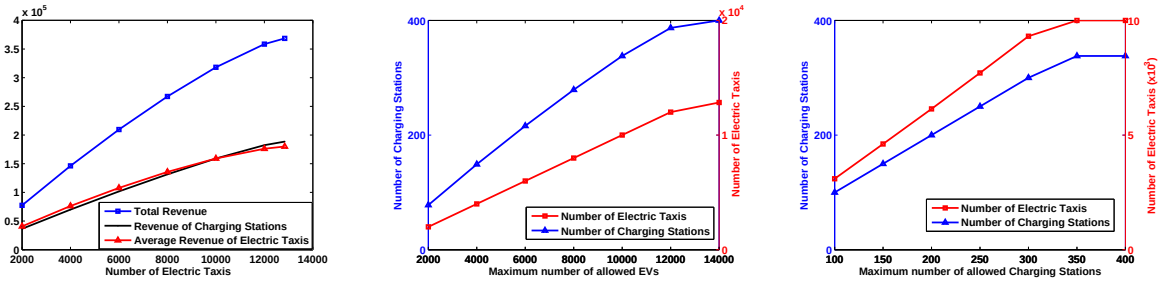
maximization function, can be summarized in the following optimization problem:

$$\begin{aligned}
\max \sum_{i=1}^{m} e_i \Bigg\{ & C_\Delta n_i + \frac{B_F - B_E}{365 Y_l} + \Bigg(\frac{\phi}{\gamma_E}(C_{Sell} - C_{Buy}) \\
& + \frac{C_F}{\gamma_F} - \frac{C_E}{\gamma_E} + \nu_F - \nu_E \Bigg) d_i \Bigg\} - \nu_C k \\
s.t. \quad & k \leq K \\
& \sum_{i=1}^{m} e_i \leq M \\
& \sum_{i=1}^{m} \frac{e_i d_i}{\gamma_E} \leq \min \left\{ 48k, \frac{\theta E_{curr}}{\phi} \right\} \\
& e_i \frac{d_i}{\gamma_E} \leq r_i + \rho \quad , \quad i = 1, \cdots, m
\end{aligned} \tag{5.2}$$

where $\mathbf{e} = [e_1 \cdots e_m]^T$ indicates which taxis are chosen as EVs ($e_i = 1$) and which ones are remained as fuel-based ($e_i = 0$). Also, k is the number of charging stations and $\sum_{i=1}^{m} e_i$ is the total number of deployed EVs which are upper bounded by K and M, respectively. Note that γ_E is the expected distance that an EV can travel with a fully charged battery and hence, $\frac{d_i}{\gamma_E}$ is the expected number of times that the ith EV should be charged. Since we have to use fast chargers for the taxi fleet, each charging takes 30 minutes and hence, k chargers can handle upto $48k$ of full charges during a day. In Eq. 5.2, ϕ is the needed energy for a full charge and θE_{curr} is the extra daily energy that the grid can tolerate. Also, r_i is the average number of rest times that the ith taxi takes in a day. The average number of required charges must be less than or equal to the average number of rests for each taxi driver. This can be penalized by a threshold, ρ. The unknown variables are $\mathbf{e}$ and k and the goal is to determine which taxis should be replaced by EVs and how many charging stations are required in the whole region.

5.4.1 Experiments

Various experiments have been performed in order to study the performance of the proposed method. Results are illustrated in Fig. 5.3. Fig. 5.3(a) compares the revenue of deployment of electric taxis and CSs versus regular fuel-based taxis. The total revenue is the summation of the average daily revenue of electric taxis and the revenue of charging station owners. As this figure depicts, by increasing the number of electric taxis, the revenue increases which is expected since the maintenance and charging costs of EVs are lower than the fuel-based vehicles. Effect of different values of M and K on the resulted number of electric taxis and charging stations (i.e. $\sum e_i$ and k) is shown in Fig. 5.3(b) and (c), respectively. Results are more promising compared with the proposed 350 chargers for less than 5000 electric taxis in previous studies [78]. It should be noted that choosing the right values for K and M



(a) (b) (c)

Figure 5.3: (a) Average daily revenue for different number of deployed electric taxis. (b) Number of charging stations and electric taxis for different values of M. (c) Number of charging stations and electric taxis for different values of K.

depends on the allocated budget for taxi fleet deployment and during the design process, target values of K and M should be determined by the stakeholders and policy makers.

5.5 Charging station placement

After determining the proper EV adoption rate and the number of required CSs, a proper charging infrastructure must be designed. There are several concerns that need to be taken into account. First of all, the overload electricity imposed by CSs on the grid must be minimized. One way to mitigate this issue is to avoid placing CSs in areas with high electricity consumption. Furthermore, CSs must be placed in a reachable distance from taxis stop points. We assume that drivers prefer to charge their batteries after dropping off a passenger and before picking up the next one. Hence, it is preferred that the distance between CSs and those locations be minimized. Moreover, to reduce the total installation cost, we aim to install CSs in places with cheaper land prices.

We decompose the city into different areas based on their ZIP-codes. In order to integrate all the above metrics (electricity consumption, land price, and reachability distance), for area i, we combine them in a linear function as follows:

$$k_i \left(\psi \frac{D_i - \mu_D}{\sigma_D} + \alpha \frac{E_i - \mu_E}{\sigma_E} + \lambda \frac{\Lambda_i - \mu_\Lambda}{\sigma_\Lambda} \right) \tag{5.3}$$

where E_i is the average daily electricity consumption, Λ_i is the land price index, and D_i is the average distance from points of interest (POI) to the center of the ith area, i.e.

$$D_i = \frac{1}{l} \sum_{j=1}^{l} \sqrt{(\zeta_x^i - L_x^j)^2 + (\zeta_y^i - L_y^j)^2}$$

Note that (ζ_x^i, ζ_y^i) is geographic coordinates of area i, (L_x^j, L_y^j) is the geographic coordinates of the jth location, and l is the number of POIs. The magnitude of Λ_i represents the value

of land in area i. This value is calculated based on the average value of residential houses (either to rent or to buy). Due to the different nature of parameters, a normalization step is done in Eq. 5.3. The expectation and standard deviation of each metric are represented by μ and σ, respectively. The importance of electricity, land, and distance to POI can be tuned with α, ψ, and λ which in our experiments are set to 1, 2, and 0.5, respectively. The overall approach to find the proper locations for CSs is formulated as the following integer linear programming problem:

$$\begin{aligned} &\min \sum_{i=1}^{\beta} k_i \left(\psi \frac{D_i - \mu_D}{\sigma_D} + \alpha \frac{E_i - \mu_E}{\sigma_E} + \lambda \frac{\Lambda_i - \mu_\Lambda}{\sigma_\Lambda} \right) \\ s.t. \quad &\sum_{i=1}^{\beta} k_i = k \\ &k_i \leq \min \left\{ k_{area}, \frac{\theta E_i}{48\phi} \right\} \quad , \quad i = 1, \cdots, \beta \\ &\sum_{j=1}^{\beta} v_{i,j} k_j \geq 1 \quad , \quad i = 1, \cdots, \beta \end{aligned} \tag{5.4}$$

where k_i is the unknown variable and represents the number of CSs that should be installed in area i. Note that the total number of CSs must be equal to the ideal number of charging stations found in Section 5.4 (i.e. k). An upper bound has been used on the number of CSs in each area, k_{area}, to prevent concentration of large number of charging stations in one place. Furthermore, electricity load at the time of simultaneous charging in one area must not go beyond the tolerance threshold of the grid ($\frac{\theta E_i}{48}$). Note that this constraint considers the low capacity of areas with low energy consumption, where k_i is bounded by electricity usage of area(E_i). In order to evenly distribute the charging stations across all areas, a constraint is applied to let the CSs be at a reachable distance, δ miles, from all areas. The value of $v_{i,j}$ indicates if area i is in a reachable distance from area j, i.e.

$$v_{i,j} = 1 \iff \sqrt{(\zeta_x^i - \zeta_x^j)^2 + (\zeta_y^i - \zeta_y^j)^2} \leq \delta.$$

Algorithm 5: Online Charging Station Recommendation

```
Procedure Recommender (S_C, S_T^i)
    if S_T^i(Charge) ≤ Critical Charge then
        CS_j = FindCharger(S_C, S_T^i);
        return Charge at CS_j;
    else
        if S_T^i(Charge) ≤ Early Charge then
            CS_j = FindCharger(S_C, S_T^i);
            if CS_j is reachable by S_T^i(Charge) then
                if S_C^j(In Queue) < S_C^j(Capacity) then
                    return Charge at CS_j;
                end
            end
        end
    end
    return Postpone Charging;

Procedure FindCharger (S_C, S_T^i)
    for kj = 1 to k do
        Dist_j = Distance between taxi i and CS_j;
        Wait_j = S_C^j(In Queue) − S_C^j(Capacity);
    end
    ϱ = Distance reachable by S_T^i(Charge) ;
    Reachables = {∀j | Dist_j ≤ ϱ};
    if Reachables ≠ ∅ then
        CS = arg min_{j∈Reachables} Wait_j;
    else
        Exception: Failure in finding a proper CS;
        Vicinities = {∀j | Dist_j ≤ αϱ};
        if Vicinities ≠ ∅ then
            CS = arg min_{j∈Vicinities} Wait_j;
        else
            CS = arg min_{j=1,···,k} Dist_j;
        end
    end
    return CS;
```

5.5.1 Experiments

We performed experiments for placement of charging stations with $M = 8000$ and $K = 400$ which has been resulted in 279 CSs. The average distance of each area to the POIs are shown in Fig. 5.4(a). As this figure illustrates, a large number of POIs are concentrated in Manhattan area which is consistent with Fig. 5.1. Fig. 5.4(b) and (c) show the average daily electricity usage and the land price index of areas, respectively. Higher values in Fig. 5.4(c) indicate more expensive areas. Fig. 5.4(d) illustrates the distribution of charging stations across NYC. As this figure depicts, charging stations have been placed in areas with less electricity, less land price, and in shorter distance to the stop points.

It should be noted that the proposed solution can be used as an interactive recommendation tool for decision makers. In this study, we prioritize the importance of three parameters (electricity, land, and distance to POI) by setting α, ψ, and λ to 1, 2, and 0.5, respectively. However, one can investigate other scenarios by changing these values. In fact, the desired outcome depends on the city planners' and decision makers' interests. Although in our study we consider electricity, land price, and POI, one can easily extend the solution by including other metrics such as neighborhood safety and crime index.

5.6 Charging Station Recommendation System

In this section, an online recommendation system is developed to suggest an appropriate CS to each driver. In this model, we aim to reduce the detours and waiting times in the queue. The algorithm is illustrated in Algorithm 5. Here, the status of the ith taxi is illustrated by $\mathcal{S}_T^i$ which includes the current charge of the ith taxi and its current location. The status of charging station j is denoted by $\mathcal{S}_C^j$ which includes its location, its capacity (i.e. maximum number of plug-in chargers in area j), and its queue length (i.e. number of taxis that have been assigned to CS_j and are waiting in the queue). $\mathcal{S}_C$ shows the status of all CSs together.

The charging status of a taxi is compared with two thresholds: *critical charging* and *early charging* thresholds. If the charge level of a taxi goes below the critical charging threshold, the driver should charge the battery at the first available charger. However, when the charge level is less than the early charging threshold but is greater than the critical one, the driver can postpone the charging process. In this situation, the system recommends the driver a CS if and only if, the charger is in a reachable distance and has free charging slot (i.e. not fully occupied). The main function in the recommendation algorithm is *Recommender*. If any of the aforementioned thresholds passed, the Recommender calls the *FindCharger* function. At first, FindCharger function looks for CSs in a reachable distance from the taxi, ϱ, which is the distance that the taxi can travel with its current battery charge. If there was no CSs in the ϱ miles radius of the taxi, we have faced with a catastrophic condition that the taxi needs to be charged while there is no CSs in the appropriate distance. Under this condition, the algorithm generates an exception. However, since we need to charge the taxi anyway, the algorithm searches for chargers in a larger area, $\alpha\varrho$, $\alpha > 1$. In this work, we set α to 3. Under severe conditions, when there is no CSs in $\alpha\varrho$ miles of the taxi, the closest charging station is selected. Since the taxi needs to be charged, a backup strategy such as towing the car to the charging station may be used here. Note that these cases will increase the total number of failed trips. In any case, among the selected charging stations, the one with the shortest queue length is returned as the optimal charging station. The complexity of algorithm is $O(k)$ and can be run in real-time.

5.6.1 Experiments

We used one month of data (March) for the electric taxi deployment and CS placement and evaluated the performance of the recommendation system on the first 10 days of April. We studied the impact of charging thresholds on the performance of the system and results are reported in Fig. 5.5 and 5.6. Let us assume that the ith taxi has a trip request that starts at time t. If at time t, the taxi is at charging station, the trip request cannot be answered and we name it as a *missed trip*. Otherwise, the trip can be started. If the taxi has enough battery charge to finish the trip successfully, we say that the trip was *successful*. Otherwise, the trip is named as a *failed trip*. Fig. 5.5(a) and (b) show the ratio of successful, failed, and missed trips w.r.t different values of early charging threshold and critical charging threshold, respectively. As these figures depict, higher values of early charging threshold increase the performance of algorithm while higher values of critical charging threshold decrease the performance.

Fig. 5.5 (c) and (d) illustrate how utilization and CS income vary w.r.t. the charging thresholds. Similar to the previous experiment, higher values of early charging threshold and lower values of critical threshold increase the performance. Fig. 5.6 shows the utilization of CSs at each area when early and critical charging thresholds are 0.25 and 0.05, respectively. Blue color and numbers in the pie charts represent utilization values.

Since the electric taxi adoption and CS placement phases are performed on one month of data, we may face with over-fitting issue. In other words, in Sections 5.4 and 5.5, all the parameters are optimized w.r.t. March dataset while in the rest of the year, trips may be different and this results in sub-optimal behaviors. To avoid the over-fitting issue, we study the effect of having more CSs than the proposed values resulted in Section 5.4. Hence, if the electric taxi adoption problem suggests to have k CSs, we use $(1 + \eta)k$ CSs for the placement and recommendation parts, where $0 \leq \eta \leq 1$ is an *expansion factor*. As Fig. 5.7 depicts, when expansion factor increases, utilization declines. This indicates that from the utilization point of view, results of Section 5.4 are optimal. However, the ratio of successful trips increases as more electric taxis can be served due to the increased number of CSs. Based on this result, using an expansion factor of $\eta = 0.1$ will result in a more reliable performance.

5.7 Conclusion

We proposed a data-driven framework for electric taxi adoption and charging infrastructure design in large metropolitan areas and an online recommendation system to suggest the best available charger to taxis. Comparing to the outcomes of previous studies, the proposed framework results in more promising outcomes as fewer charging stations are required for more EVs. It should be noted that the proposed framework includes multiple constraints

and parameters that mainly depend on the budget and policies dictated by the stakeholders. Therefore, an interactive tool may be developed based on the proposed framework such that various constraints and priorities can be changed by an expert to achieve an appropriate solution.

In the proposed framework, two problems (i.e. electric taxi adoption and charging station placement) are modeled by integer linear programming (ILP). ILP problems are generally NP-hard and hence, heuristic solutions are deployed to solve the electric taxi adoption and charging station placement problems. Further evaluations reveal that in our implementations and experiments, determined solutions were a close approximation of the optimal results.

To mitigate CO_2 emission and build a sustainable transportation system, integration of ride-sharing and electric taxis can be investigated as future work. Furthermore, integration of electric taxis with smart grid entities (e.g. energy storage) to alleviate the extra electricity load during peak hours is suggested.

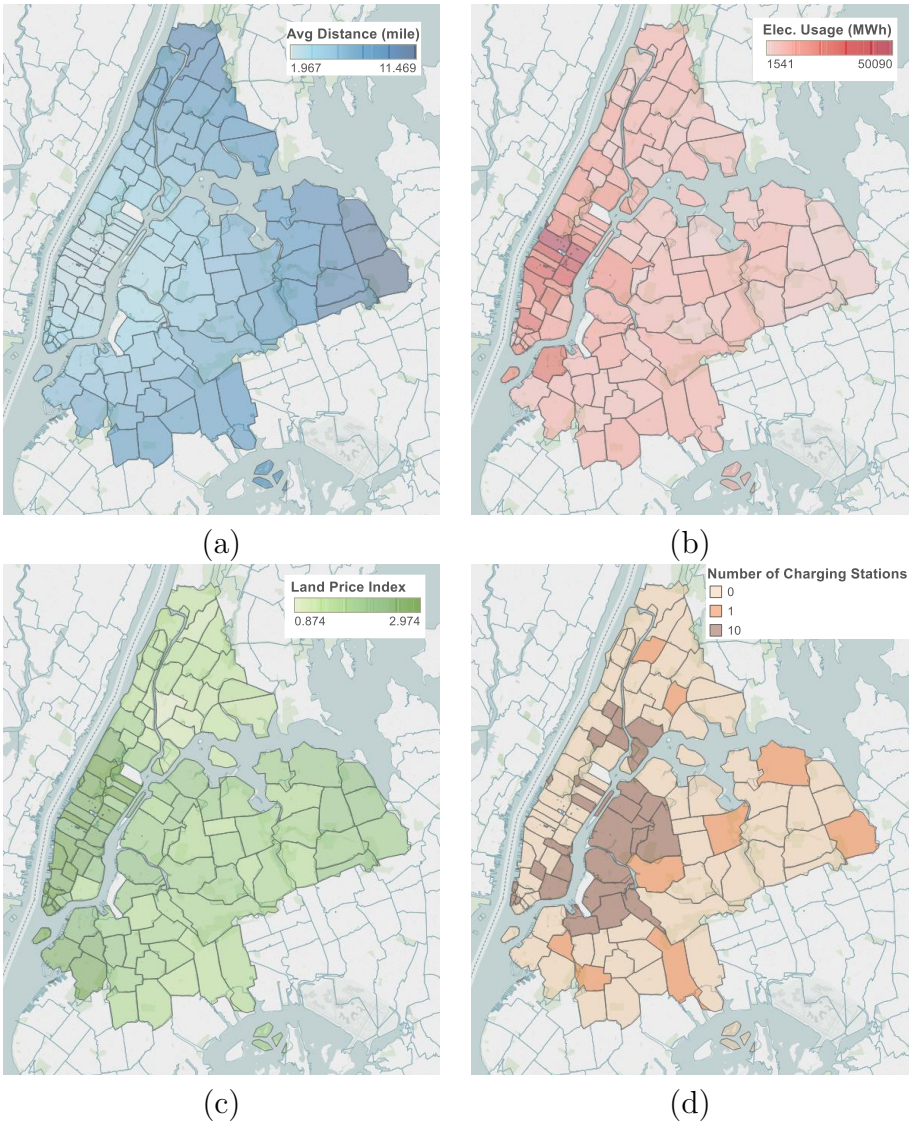


Figure 5.4: (a) Average distance of points of interest from center of areas. (b) Average daily electricity usage. (c) Land Price Index. (d) Proposed placement of charging stations across areas. Different colors represent different number of charging stations.

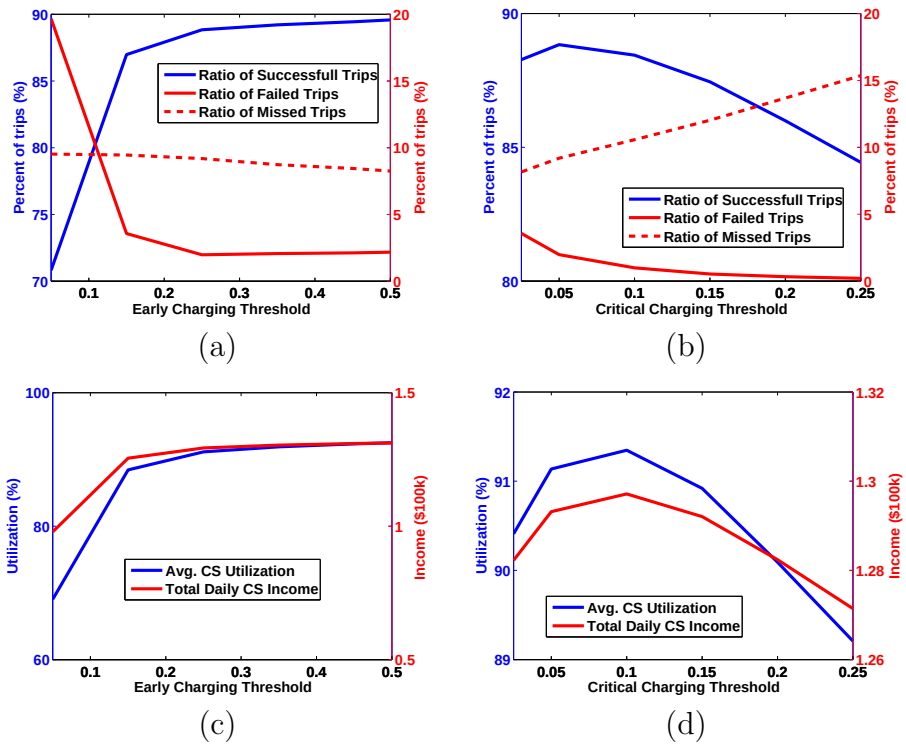


Figure 5.5: (a) Ratio of trips w.r.t. early charging threshold. (b) Ratio of trips w.r.t. critical charging threshold. (c) Utilization and income w.r.t. early charging threshold. (d) Utilization and income w.r.t. critical charging threshold.

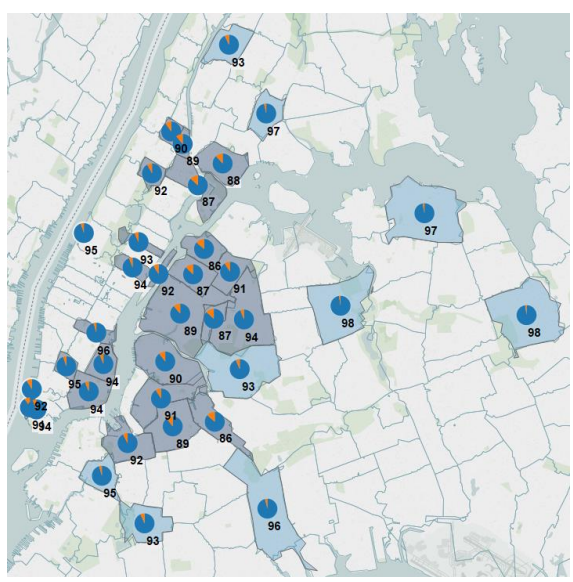


Figure 5.6: Utilization of charging stations for $M = 8000$, $K = 400$.

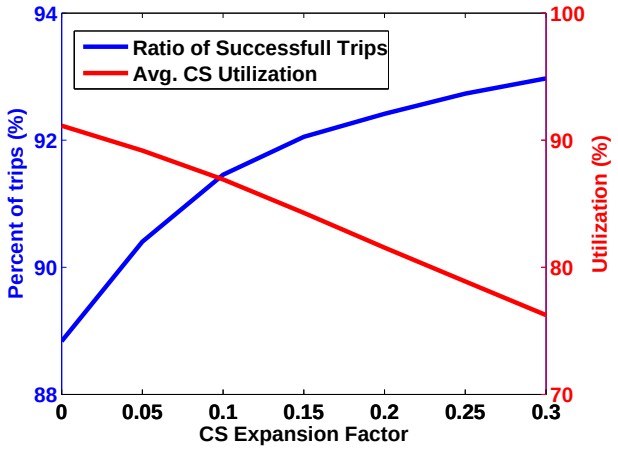


Figure 5.7: Effect of expansion factor (η).

Chapter 6

Conclusion

Due to the growing demand for transportation in large metropolitan areas, its negative impacts on health, quality of life, climate, space, and landscape are also increasing. In recent years, scientists and policy makers have observed the urgent need for a transition to smart cities and to use sustainable transportation solutions in urban areas. This transition needs to be performed precisely from socioeconomic perspective and depends on the dynamic interactions that mutually exist between various aspects of a modern city life. For example, inter-relationships between transportation, social networks, energy consumption, and economic infrastructure make urban mobility systems extremely complex [87]. Hence, we need to precisely study the current mobility issues in urban areas and provide a well-defined analytic framework to explore the possibility of transition toward having more sustainable metropolitan areas.

The goal of this dissertation is to use data analytic methods in order to tackle major issues in Sustainable urban mobility. We have provided a general analytic framework to propose effective solutions for challenging problems which exist in urban areas. We deployed various data mining techniques for the purpose of knowledge discovery. After describing our framework, we first divided the SUM issues into two main categories: mobility/transportation issues and energy issues. Due to the existence of various types of transportation in large cities, we studied the mobility issues from two perspectives: private transportation (e.g. personal vehicles) and public transportation (e.g. taxis). This dissertation specifically dealt with several challenging data analytic problems as follows:

- **Characterization of transportation in urban area:** We characterized mobility patterns underlying taxis in New York City to extract behavioral features and find locations of interest. Furthermore, using a signature graph of traffic flow, we were able to detect spatio-temporal anomalies through the year.
- **Characterization of energy networks:** We studied energy systems (e.g. microgrids) to extract the hidden relationships and inter-dependencies among observed vari-

ables in the system. We created an invariant graph that can be used to infer such relationships and also to monitor the behavior of system through the time to be able to detect outliers during system operation.

- **Charging and storage infrastructure design for electric vehicles:** With respect to the advent of new technologies such as EVs, we locate the best possible locations to install charging station infrastructure in urban areas. This study is specifically designed for personal vehicles with respect to the activity of people throughout the day. Based on people behavior and electricity consumption patterns, we proposed an integrated data-driven solution to overcome issues related to adoption of EVs. Furthermore, we investigated the electricity load imposed by EVs and proposed a storage placement method to overcome the grid capacity issues.

- **Adoption of electric taxi fleet in dense urban area:** We investigated the feasibility of electric taxi adoption in a large urban area (i.e. New York City) and proposed a data-driven solution for charging station placement for these taxis. Moreover, with respect to the rest times of drivers, we tried to determine the required number of fast chargers in proper locations. Furthermore, we developed a recommendation system to suggest appropriate charging stations and charging times to the taxi drivers.

In this research, we showed that the use of big data and data analytics methods can be beneficial in solving SUM issues. The proposed methods can provide an optimal solution for characterizing, monitoring, and investment of different components in SUM.

6.1 Deployment Considerations

While the proposed solutions in this dissertation have been examined over specific case studies and datasets, the proposed framework and approaches may be easily extended and deployed in other situations. As an example, the proposed invariant discovery approach that has been used for anomaly detection in energy systems can be easily extended to use for the control and monitoring of other types of cyber-physical systems.

Various proposed solutions in this dissertation, such as charging station placement for personal vehicles and electric taxis in Chapter 5 and Chapter 4, depend on multiple parameters and constraints that should be determined by the policy makers. As an example, in electric taxi fleet deployment, the maximum number of electric taxis and charging stations greatly depend on the dedicated budget. Furthermore, different cities may have different priorities on the land price, electricity load, and other influencing parameters. Therefore, an expert can use the proposed approach to get the desired outcome by tuning the parameters such as impact of land price on placement of fast chargers for electric taxis. Appropriate visualizations and an interactive tool can be used to facilitate the whole design process.

6.2 Emerging Trends and SUM

Transportation industry and mobility in urban areas have been tremendously affected by various technological advancements that have been made in recent years. Moreover, with the popularity of social networks and mobile applications, more interactions occur among people over the Internet and this phenomenon has also affected the transportation methods. For example, autonomous cars will be going to take the place of traditional cars where there is no need for a human driver. As another example, new trends in sharing economy such as bike-sharing can be mentioned that is now more feasible due to the use of mobile apps. In this section, we contemplate the new trends in urban areas w.r.t. SUM and address how the proposed framework may be potentially applied there.

- Autonomous cars, vehicles that can be driven without a human driver, have this capability to interact with the Internet and have access to online and up-to-date resources of information. Therefore, these vehicles can make optimal decisions regarding the location and time of fueling/charging. This optimum decisions can even be made in a centralized manner, which can more precisely consider the requirements of other vehicles, charging/fueling stations, and the whole city. In this dissertation, we developed centralized recommender systems that suggest optimal charging places to both personal EVs and electric taxis w.r.t. various parameters. The proposed solution can be modified to address the requirements of autonomous EVs in a straightforward manner, as these vehicles already have all the required networking and communication devices.

- Sharing economy, sharing of access to products or services, is an emerging trend with the main goal of enabling the optimization of resource consumption. The rise of disownership via sharing economy have several benefits such as saving costs and reducing negative environmental impacts [105]. However, important concerns must be addressed carefully such as cooperation of stakeholders, labor regulation, and consumer protection regulation. Ride-sharing and bike-sharing are two important examples that lay at the intersection of sharing economy and SUM. These sharing examples have been deployed in some metropolitan areas around the world such as New York City [112]. Similar to other SUM issues, the proposed SUM framework can be also applied to the ride and bike sharing problems. However, it should be noted that the corresponding sub-problems are different here. As an example, in the case of bike-sharing, we do not have any concern regarding the electricity consumption, while the sharing stations should be accessible through the public transportation network. Hence, similar to the other transportation issues that we studied in this dissertation, we need to precisely consider the dynamic mobility behavior of people in the area. It should be noted that due to the importance of public transportation networks in the sharing economy, this category of transportation and the way that people ride subways and use bus shuttles should be also included in such a study.

- With respect to the energy management issues of SUM, intelligent energy systems

and advanced information and communication technology (ICT) are beneficial in cities [113]. Sustainable energy production, consumption, and distribution are tied to the existing patterns in the city. For example, smart homes can change the behavior of electricity usage of city by scheduling high-consumption activities (e.g. laundry) to off-peak hours. Smart energy management systems that are deployed in smart homes have also inter-relationships with the SUM, as personal EV owners typically charge their cars at home. In this regard, understanding the detailed pattern of activities is beneficial in implementing efficient intelligent systems. On the other hand, issues such as control and automatic fault recovery of the deployed smart devices should be done automatically. We proposed a data analytic framework for the purpose of anomaly detection in complex energy systems which also facilitate the fault localization step. However, for each specific application, precise and accurate model is required with respect to the actual implementation or goals.

6.3 Future Work

This dissertation opens up many opportunities for future research both from theoretical and application perspectives. Some of the possible future works are described below:

- **Integrated solution for sustainable public transportation:** In Chapter 5, we proposed a hierarchical optimization approach for adoption of electric taxis in highly populated areas. For future work, we aim to propose an integrated solution for the electric taxi fleet utilizing other data sources (including detailed information of electricity consumption, capacity of the grid, and safety index of neighborhoods) as well as incorporating other types of public transportation to understand passengers' preferences. Moreover, information about daily activity of people can be added here similar to Chapter 4.

- **Integration of anomalies and urban planning in SUM:** Anomalies are inevitable parts of any large metropolitan area. Therefore, an efficient SUM solution should focus on the average behavior of the city and also should behave efficiently under abnormal conditions. In abnormal times, such as special game days, not only the transportation network is altered due to the high demand but also, traffic conditions may affect the availability of infrastructures such as charging stations. One way to handle this is to provide self-organizing infrastructure or to use mobile equipment. This approach has been previously used in cellular networks to overcome the capacity issues during big events (e.g. sports) [93]. For the future work, we aim to include anomaly detection into the urban and infrastructure planning solutions. A main step here is to determine the correlations and casual relationships between various types of anomalies and each of the smart city issues. As an example, to overcome the unavailability of charging stations caused by anomalies, we need to determine which city-wide anomalies may

alter the availability of CSs. Therefore, we not only need to detect anomalies in the city transportation, but also we have to determine their causal relations with the availability of urban resources and services. Graph mining techniques may be effectively deployed in this research.

- **Social media:** While we deployed various data sources for the problems we explored in this dissertation, social media datasets, such as Foursquare, Yelp, Wikipedia, and Twitter may also be incorporated for future research. These datasets can be used for a diverse set of purposes such as points of interest determination and anomaly detection in the city. Another potential use of social media datasets is to improve the decision making processes in urban planning for future smart cities. Social media datasets can be used to understand people and communities, locations, and events. Hence, these data sources can be beneficial to forecast resource requirements and perform city-wide analysis more precisely.

- **Visual analytics in SUM:** We plan to develop a visual analytic system for infrastructure investment in SUM. A demo[1] has been developed for charging station placement for personal EVs. However, an interactive application with the ability to respond with respect to user's preferences requires extensive work.

- **Use of other techniques:** Use of other techniques for electric taxi adoption and CS placement such as agent-based and game theoretic approaches can be done to compare it with the results of our data-driven framework.

[1] `http://bioinformatics.cs.vt.edu/EVchargingplacement/`

Bibliography

Bibliography

[1] A. Abur and A. G. Exposito. *Power system state estimation: Theory and implementation*. Marcel Dekker, 2004.

[2] L. Akoglu, M. McGlohon, and C. Faloutsos. Oddball: Spotting anomalies in weighted graphs. In *Proceedings of the 14th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD)*, pages 410–421, 2010.

[3] L. Akoglu, H. Tong, and D. Koutra. Graph-based anomaly detection and description: A survey. *Data Mining and Knowledge Discovery*, 29(3):626–688, May 2015.

[4] F. Alesiani and N. Maslekar. Optimization of charging stops for fleet of electric vehicles: A genetic approach. *IEEE Intelligent Transportation Systems Magazine*, 6(3):10–21, Fall 2014.

[5] S. Aman, Y. Simmhan, and V. K. Prasanna. Improving energy use forecast for campus micro-grids using indirect indicators. In *Proceedings of IEEE Workshop on Domain Driven Data Mining*, 2011.

[6] A. Arnold, Y. Liu, and N. Abe. Temporal causal modeling with graphical granger methods. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 66–75, 2007.

[7] M. Z. Austwick, O. OBrien, E. Strano, and M. Viana. The structure of spatial networks and communities in bicycle sharing systems. *PLoS ONE*, 8(9):e74685, September 2013.

[8] C. Bailey-Kellogg, N. Ramakrishnan, and M. Marathe. Spatial data mining to support pandemic preparedness. *SIGKDD Explorations Newsletter*, 8(1):80–82, Jun 2006.

[9] P. Baptista, J. Ribau, J. Bravo, C. Silva, P. Adcock, and A. Kells. Fuel cell hybrid taxi life cycle analysis. *Energy Policy*, 39(9):46834691, 2011.

[10] M. Batty, K. Axhausen, G. Fosca, A. Pozdnoukhov, A. Bazzani, M. Wachowicz, G. Ouzounis, and Y. Portugali. Smart cities of the future. *European Physical Journal - Special Topics*, 214:481–518, November 2012.

[11] I. Bayram, G. Michailidis, M. Devetsikiotis, S. Bhattacharya, A. Chakrabortty, and F. Granelli. Local energy storage sizing in plug-in hybrid electric vehicle charging stations under blocking probability constraints. In *Proceedings of IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 78–83, 2011.

[12] K. Bisset, K. Atkins, C. Barrett, R. Beckman, S. Eubank, A. Marathe, M. Marathe, H. Mortveit, P. Stretz, and V. A. Kumar. Synthetic data products for societal infrastructures and proto-populations: Data Set 1.0. Technical Report TR-06-006, Network Dynamics and Simulation Science Laboratory, Virginia Tech, Blacksburg, VA, 2006.

[13] K. S. Blake, R. L. Kellerson, and A. Simic. Measuring overcrowding in housing. U.S. Department of Housing and Urban Development, Last accessed Oct 13 2012 `http://www.huduser.org/publications/polleg/overcrowding_hsg.html`, September 2007.

[14] Z. Cao, S. Wang, G. Forestier, A. Puissant, and C. F. Eick. Analyzing the composition of cities using spatial clustering. In *Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing (UrbComp)*, 2013.

[15] P. S. Castro, D. Zhang, C. Chen, S. Li, and G. Pan. From taxi gps traces to social and community dynamics: A survey. *ACM Computing Surveys (CSUR)*, 46(2):17:1–17:34, December 2013.

[16] S. Chawla, Y. Zheng, and J. Hu. Inferring the root cause in road traffic anomalies. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pages 141–150, 2012.

[17] D. Chen, C.-T. Lu, Y. Kou, and F. Chen. On detecting spatial outliers. *Geoinformatica*, 12:455475, 2008.

[18] F. Chen, C.-T. Lu, and A. P. Boedihardjo. Gls-sod: A generalized local statistical approach for spatial outlier detection. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1069–1078, 2010.

[19] H. Chen, H. Cheng, G. Jiang, and K. Yoshihira. Exploiting local and global invariants for the management of large scale information systems. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pages 113–122, 2008.

[20] H. Chen, G. Jiang, K. Yoshihira, and A. Saxena. Invariants based failure diagnosis in distributed computing systems. In *Proceedings of the 29th IEEE Symposium on Reliable Distributed Systems*, pages 160–166, 2010.

[21] T. D. Chen, K. M. Kockelman, and M. Khan. The electric vehicle charging station location problem: A parking-based assigment method for seattle. In *Proceedings of*

the 92nd Annual Meeting of the Transportation Research Board in Washington DC, January 2013.

[22] Y. Chen, L. Xie, and P. Kumar. Dimensionality reduction and early event detection using online synchrophasor data. In *Proceedings of the 2013 IEEE Power & Energy Society General Meeting (PES)*, pages 1–5, July 2013.

[23] Z. Chen, C. Mi, J. Xu, X. Gong, and C. You. Energy management for a power-split plug-in hybrid electric vehicle based on dynamic programming and neural networks. *IEEE Transaction on Vehicular Technology*, 63(4):1567–1580, May 2014.

[24] D. Cheng, M. T. Bahadori, and Y. Liu. Fblg: A simple and effective approach for temporal dependence discovery from time series data. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 382–391, 2014.

[25] T. Cheng, J. Wang, J. Haworth, B. Heydecker, and A. Chow. A dynamic spatial weight matrix and localized space-time autoregressive integrated moving average for network modeling. *Geographical Analysis*, 46:75–97, January 2014.

[26] Coulomb Technologies, Inc. Electric vehicle charging solutions for fleets. Technical report, Coulomb Technologies, Inc, 2011.

[27] J. C. F. De Winter and D. Dodou. Common factor analysis versus principal component analysis: a comparison of loadings by means of simulations. *Communications in Statistics - Simulation and Computation*, June 2014.

[28] M. Ding, H. Chen, A. Sharma, K. Yoshihira, and G. Jiang. A data analytic engine towards self-management of cyber-physical systems. In *Proceedings of the 33rd IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW)*, pages 303–308, 2013.

[29] Eltis. Sustainable urban mobility plans: Concepts, process, content, and benefits. `http://www.eltis.org/sites/eltis/files/events/documents/presentation_4_sump_concept_rob_jeuring.pdf`, 2015. [Online; accessed 23-February-2016].

[30] J. J. Escudero-Garzas and G. Seco-Granados. Charging station selection optimization for plug-in electric vehicles: An oligopolistic game-theoretic framework. *IEEE PES Innovative Smart Grid Technologies (ISGT)*, 0:1–8, 2012.

[31] J. J. Escudero-Garzas and G. Seco-Granados. Monitoring disease trends using hospital traffic data from high resolution satellite imagery: a feasibility study. *Nature Scientific Reports*, 5(9112), March 2015.

[32] M. Ester, H. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 226–231, 1996.

[33] C. Etienne and O. Latifa. Model-based count series clustering for bike sharing system usage mining: A case study with the velib system of paris. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(3):39:1–39:21, July 2014.

[34] Y. Ge, G. Jiang, M. Ding, and H. Xiong. Ranking metric anomaly in invariant networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(2):8:1–8:30, June 2014.

[35] Y. Ge, G. Jiang, and Y. Ge. Efficient invariant search for distributed information systems. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pages 1049–1054, 2013.

[36] E. H. Gerding, S. Stein, V. Robu, D. Zhao, and N. R. Jennings. Two-sided online markets for electric vehicle charging. In *Proceedings of the 12th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2013.

[37] GM-Volt. Chevrolet Volt Specifications. Last accessed May 16 2012 `http://Gm-volt.com/full-specification`.

[38] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han. Outlier detection for temporal data: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 25(1):1–20, 2014.

[39] Y. Han and F. Moutarde. Analysis of large-scale traffic dynamics in an urban transportation network using non-negative tensor factorization. *International Journal of Intelligent Transportation Systems Research*, pages 1–14, August 2014.

[40] H. Harman. *Modern Factor Analysis*. University of Chicago Press, Chicago, 3rd edition edition, 1976.

[41] K. Henderson, B. Gallagher, T. Eliassi-Rad, H. Tong, S. Basu, L. Akoglu, D. Koutra, C. Faloutsos, and L. Li. Rolx: Role extraction and mining in large graphs. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1231–1239, 2012.

[42] A. Hess, F. Malandrino, M. B. Reinhardt, C. Casetti, K. A. Hummel, and J. M. Barcelo-Ordinas. Optimal deployment of charging stations for electric vehicular networks. In *Proceedings of the 1st Workshop on Urban Networking (UrbaNe)*, 2012.

[43] M. Hoffman, A. Sadovsky, M. Kintner-Meyer, and J. DeSteese. Analysis tools for sizing and placement of energy storage in grid applications. Technical Report PNNL-19703, Pacific Northwest National Laboratory Richland, Washington 99352 , 2010.

[44] M. S. Hossain, S. Tadepalli, L. T. Watson, I. Davidson, R. F. Helm, and N. Ramakrishnan. Unifying dependent clustering and disparate clustering for non-homogeneous data. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 593–602, 2010.

[45] C. Huang and X. Wu. Discovering road segment-based outliers in urban traffic network. In *Proceedings of Globecom 2013 Workshop - Vehicular Network Evolution*, pages 1350 – 1354, 2013.

[46] T. Ide, A. C. Lozano, N. Abe, and Y. Liu. Proximity-based anomaly detection using sparse structure learning. In *Proceedings of the 9th SIAM International Conference on Data Mining (SDM)*, 2009.

[47] G. Jiang, H. Chen, and K. Yoshihira. Discovering likely invariants of distributed transaction systems for autonomic system management. *Cluster Computing*, 9(4):385–399, Oct. 2006.

[48] G. Jiang, H. Chen, and K. Yoshihira. Efficient and scalable algorithms for inferring likely invariants in distributed systems. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 19(11):1508–1523, November 2007.

[49] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang. Catchsync: Catching synchronized behavior in large directed graphs. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 941–950, 2014.

[50] R. Jiang, R. Lu, Y. Wang, J. Luo, C. Shen, and X. S. Shen. Energy-theft detection issues for advanced metering infrastructure in smart grid. *Tsinghua Science and Technology*, 19(2):105–120, April 2014.

[51] S. Jiang, J. Ferreira, Jr., and M. C. Gonzalez. Discovering urban spatial-temporal structure from human activity patterns. In *Proceedings of the ACM SIGKDD International Workshop on Urban Computing (UrbComp)*, pages 95–102, 2012.

[52] C. Jin, J. Tang, and P. Ghosh. Optimizing electric vehicle charging: A customer's perspective. *IEEE Transaction on Vehicular Technology*, 62(7):2919–2927, Sept 2013.

[53] K. Joreskog. Some contributions to maximum likelihood factor analysis. *Psychometrika*, 38(2):183–202, 1967.

[54] M. Kahlen and W. Ketter. Aggregating electric cars to sustainable virtual power plants: The value of flexibility in future electricity markets. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI)*, 2015.

[55] B. M. Kalesar and A. R. Seifi. Optimal substation placement and feeder routing in distribution system planning using genetic algorithm. *Elixir Elec. Engg.*, 37:3908–3915, July 2011.

[56] Kema, Inc. Distributed energy storage: Serving national interests, advancing wide-scale DES in the united states. Technical Report 20130065, National Alliance for Advanced Technology Batteries, 2012.

[57] M. Khan, M. Li, P. Ashton, G. Taylor, and J. Liu. Big data analytics on pmu measurements. In *Proceedings of the 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 715–719, Aug 2014.

[58] S. Khuller, A. Malekian, and J. Mestre. To fill or not to fill: The gas station problem. *ACM Transactions on Algorithms (TALG)*, 7(36):1–15, July 2011.

[59] T. Kindberg, M. Chalmers, and E. Paulos. Urban computing. *IEEE Pervasive Computing*, pages 18–20, July-September 2007.

[60] Y. Kou, C.-T. Lu, and R. Dos Santos. Spatial outlier detection: A graph-based approach. In *Proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, volume 1, pages 281–288, Oct 2007.

[61] G. Kreisselmeier and R. Steinhauser. Systematic control design by optimizing a vector performance index. In *Proceedings of IFAC Symposium on Computer Aided Design of Control Systems*, pages 113–117, 1979.

[62] A. Lam, Y.-W. Leung, and X. Chu. Electric vehicle charging station placement: Formulation, complexity, and solutions. *IEEE Transaction on Smart Grid*, 5(6):2846–2856, Nov 2014.

[63] V. Larsson, L. Johannesson, and B. Egardt. Analytic solutions to the dynamic programming subproblem in hybrid vehicle energy management. *IEEE Transaction on Vehicular Technology*, 64(4):1458–1467, April 2015.

[64] S. Li, S. Sharkh, F. Walsh, and C. Zhang. Energy and battery management of a plug-in series hybrid electric vehicle using fuzzy logic. *IEEE Transaction on Vehicular Technology*, 60(8):3571–3585, Oct 2011.

[65] D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui. Geomf: Joint geographical modeling and matrix factorization for point-of-interest recommendation. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 831–840, 2014.

[66] S. Lihao and Y. Zaiyue. Optimal charging schedule of phev taxi with time-varying price. In *Proceedings of the 31st Chinese Control Conference (CCC)*, pages 2537–2542, July 2012.

[67] W. Liu, Y. Zheng, S. Chawla, J. Yuan, and X. Xie. Discovering spatio-temporal causal interactions in traffic data streams. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1010–1018, August 2011.

[68] X. Liu, F. Chen, and C.-T. Lu. On detecting spatial categorical outliers. *Geoinformatica*, 18:501–536, 2014.

[69] Y. Liu, A. Niculescu-mizil, A. C. Lozano, and Y. Lu. Learning temporal causal graphs for relational time-series analysis. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pages 687–694, 2010.

[70] Y. Liu, Z. Sui, C. Kang, and Y. Gao. Uncovering patterns of inter-urban trip and spatial interaction from social media check-in data. *PLoS ONE*, 9(1):e86026, January 2014.

[71] J.-L. Lu, M.-Y. Yeh, Y.-C. Hsu, S.-N. Yang, C.-H. Gan, and M.-S. Chen. Operating electric taxi fleets: A new dispatching strategy with charging plans. In *Proceedings of IEEE International Electric Vehicle Conference (IEVC)*, pages 1–8, March 2012.

[72] C. Luo, Y.-F. Huang, and V. Gupta. A consumer behavior based approach to multi-stage ev charging station placement. In *Proceedings of 81st IEEE Vehicular Technology Conference (VTC Spring)*, pages 1–6, May 2015.

[73] K. Maalej, S. Kelouwani, K. Agbossou, Y. Dube, and N. Henao. Long-trip optimal energy planning with online mass estimation for battery electric vehicles. *IEEE Transaction on Vehicular Technology*, 64(11):4929–4941, Nov 2015.

[74] Y. Makarov, D. Pengwei, M. Kintner-Meyer, J. Chunlian, and H. Illian. Sizing energy storage to accommodate high penetration of variable energy resources. *IEEE Transactions on Sustainable Energy*, 3(1):34–40, Jan. 2012.

[75] R. Martin. More than 11 million EV charging stations will be installed worldwide by 2020. Last accessed Oct 13 2012 `http://www.pikeresearch.com/newsroom/more-than-11-million-ev-charging-stations-will-be-installed-worldwide-by-2020`, October 2012.

[76] D. Mashima and A. A. Cardenas. Evaluating electricity theft detectors in smart grid networks. In *Research in Attacks, Intrusions, and Defenses*, volume 7462 of *Lecture Notes in Computer Science*, pages 210–229. Springer, 2012.

[77] G. Mauri and A. Valsecchi. Fast charging stations for electric vehicle: The impact on the mv distribution grids of the milan metropolitan area. In *Proceedings of IEEE International Energy Conference and Exhibition (ENERGYCON)*, pages 1055–1059, 2012.

[78] Mayors Long-Term NYC Electric Taxi Task Force. Take Charge: A roadmap to electric New York city taxis. Technical report, NYC Taxi and Limousine Commission, 2013.

[79] X. Min, J. Hu, Q. Chen, T. Zhang, and Y. Zhang. Short-term traffic flow forecasting of urban network based on dynamic starima model. In *Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, 2009.

[80] M. Momtazpour, P. Butler, M. S. Hossain, M. C. Bozchalui, N. Ramakrishnan, and R. Sharma. Coordinated clustering algorithms to support charging infrastructure design for electric vehicles. In *Proceedings of the ACM SIGKDD International Workshop on Urban Computing (UrbComp)*, pages 126–133, 2012.

[81] M. Momtazpour, P. Butler, M. S. Hossain, M. C. Bozchalui, R. Sharma, and N. Ramakrishnan. Charging and storage infrastructure design for electric vehicles. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(3):1–27, September 2014.

[82] M. Momtazpour and N. Ramakrishnan. Characterizing taxi flows in new york city. In *Proceedings of the International Workshop on Urban Computing (UrbComp)*, 2015.

[83] M. Momtazpour, R. Sharma, and N. Ramakrishnan. An integrated data mining framework for analysis and prediction of battery characteristics. In *Proceedings of IEEE PES Innovative Smart Grid Technologies - Asia*, pages 774–779, 2014.

[84] M. Momtazpour, J. Zhang, S. Rahman, R. Ratnesh, and N. Ramakrishnan. Analyzing invariants in cyber-physical systems using latent factor regression. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2015.

[85] N. Munro. Obama hikes subsidy to wealthy electric car buyers. Last accessed May 16 2012 `http://dailycaller.com/2012/02/13/obama-hikes-subsidy-to-wealthy-electric-car-buyers/`.

[86] L. Niu and D. Zhang. Charging guidance of electric taxis based on adaptive particle swarm optimization. *The Scientific World Journal*, 2015(354952), 2015.

[87] M. D. Paep, P. Vandenbroeck, and J. V. Reeth. New MIRA study report: Transition to sustainable mobility in cities: an analysis framework. Technical report, MIRA, 2014.

[88] T. Paul and H. Aisu. Management of quick charging of electric vehicles using power from grid and storage batteries. In *Proceedings of IEEE International Electric Vehicle Conference (IEVC)*, pages 1–8, 2012.

[89] B. Pincombe. Anomaly detection in time series of graphs using arma processes. *ASOR Bulletin*, 24(4):2–10, 2005.

[90] Portland General Electric Company. Charging on the Go. Last accessed Oct 13 2012 `http://www.portlandgeneral.com/community_environment/initiatives/electric_vehicles/charging_stations/charging_on_go.aspx`, April 2012.

[91] Portland General Electric Company. PGE's retail rate schedules and rules. Last accessed Sep 21 2012 `www.portlandgeneral.com/our_company/corporate_info/regulatory_documents/tariff`, April 2012.

[92] K. J. Preacher, G. Zhang, C. Kim, and G. Mels. Choosing the optimal number of factors in exploratory factor analysis: A model selection perspective. *Multivariate Behavioral Research*, 48:28–56, 2013.

[93] C. Prehofer and C. Bettstetter. Self-organization in communication networks: principles and design paradigms. *IEEE Communications Magazine*, 43(7):78–85, July 2005.

[94] D. Radmacher. Multidisciplinary team awarded 1.1 million dollar grant to focus on smart city infrastructure. `https://www.eng.vt.edu/news/multidisciplinary-team-awarded-11-million-grant-focus-smart-city-infrastructure`, 2015. [Online; accessed 23-February-2016].

[95] N. Ramakrishnan, C. Bailey-Kellogg, S. Tadepalli, and V. Pandey. Gaussian processes for active data mining of spatial aggregates. In *Proceedings of the 5th SIAM International Conference on Data Mining (SDM)*, 2005.

[96] N. Ramakrishnan and A. Grama. Mining scientific data. *Advances in Computers*, 55:119–169, July 2001.

[97] S. D. Ramchurn, P. Vytelingum, A. Rogers, and N. R. Jennings. Putting the 'smarts' into the smart grid: A grand challenge for artificial intelligence. *Communnications of the ACM*, 55(4):86–97, Apr. 2012.

[98] S. Ranshous, S. Shen, D. Koutra, C. Faloutsos, and N. F. Samatova. Anomaly detection in dynamic networks: a survey. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(3):223–247, 2015.

[99] S. Rayana and L. Akoglu. An ensemble approach for event detection and characterization in dynamic graphs. In *Proceedings of the 2nd ACM SIGKDD Workshop on Outlier Detection & Description under Data Diversity (ODD)*, 2014.

[100] R. Rose. A smart technique for determining base-station locations in an urban environment. *IEEE Transaction on Vehicular Technology*, 50(1):43–47, Jan 2001.

[101] R. Rossi and B. Gallagher. Role-dynamics: Fast mining of large dynamic networks. In *Proceedings of the 21st International Conference on World Wide Web (WWW)*, pages 997–1006, 2012.

[102] R. A. Rossi and B. Gallagher. Modeling dynamic behavior in large evolving graphs. In *Proceedings of the 6th ACM international conference on Web search and data mining (WSDM)*, pages 667–676, 2013.

[103] J. Rosswog and K. Ghose. Detecting and tracking coordinated groups in dense, systematically moving, crowds. In *Proceedings of the 12th SIAM International Conference on Data Mining (SDM)*, pages 1–11, 2012.

[104] P. Rozenshtein, A. Anagnostopoulos, A. Gionis, and N. Tatti. Event detection in activity networks. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1176–1185, 2014.

[105] A. Rudenko. The collaborative consumption on the rise: why shared economy is winning over the capitalism of me. Technical report, POPSOP, 2013.

[106] I. Shafer, K. Ren, V. N. Boddeti, Y. Abe, G. R. Ganger, and C. Faloutsos. Rainmon: An integrated approach to mining bursty timeseries monitoring data. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1158–1166, 2012.

[107] H. Shan, G. Jiang, and K. Yoshihira. Extracting overlay invariants of distributed systems for autonomic system management. In *Proceedings of the 4th IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*, pages 41–50, 2010.

[108] A. Sharma, H. Chen, M. Ding, K. Yoshihira, and G. Jiang. Fault detection and localization in distributed systems using invariant relationships. In *Proceedings of the 43rd IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 1–8, 2013.

[109] A. B. Sharma, F. Ivancic, A. Niculescu-Mizil, H. Chen, and G. Jiang. Modeling and analytics for cyber-physical systems in the age of big data. *ACM SIGMETRICS Performance Evaluation Review*, 41(4):74–77, Apr. 2014.

[110] S. Shekhar, C.-T. Lu, and P. Zhang. Detecting graph-based spatial outliers: Algorithms and applications (a summary of results). In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 371–376, 2001.

[111] Simply Hired, Inc. Portland Jobs. Last accessed May 16 2012 `http://www.simplyhired.com/a/local-jobs/city/l-Portland,+OR`.

[112] D. Singhvi, S. Singhvi, P. Frazier, S. Henderson, E. Mahony, D. Shmoys, and D. Woodard. Predicting bike usage for new york city's bike sharing system. In *Proceedings of AAAI Workshops at the 29th AAAI Conference on Artificial Intelligence*, 2015.

[113] T. Strasser, P. Siano, and V. Vyatkin. New trends in intelligent energy systems - an industrial electronics point of view. *IEEE Transactions on Industrial Electronics*, 62(4):2420–2423, April 2015.

[114] A. Styler and I. Nourbakhsh. Real-time predictive optimization for energy management in a hybrid electric vehicle. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI)*, 2015.

[115] D. Suhr. Principal component analysis vs. exploratory factor analysis. In *Proceedings of SUGI 30*, pages 203–30, 2005.

[116] T. Sweda and D. Klabjan. In an agent-based decision support system for electric vehicle charging infrastructure deployment. In *Proceedings of IEEE Vehicle Power and Propulsion Conference (VPPC)*, 2011.

[117] R. Takahashi, T. Osogami, and T. Morimura. Large-scale nonparametric estimation of vehicle travel time distributions. In *Proceedings of the 12th SIAM International Conference on Data Mining (SDM)*, pages 12–23, April 2012.

[118] The Engineering ToolBox. Common area per person in buildings. Last accessed: May 16 2012 `http://www.engineeringtoolbox.com/number-persons-buildings-d_118.html`.

[119] The official U.S. Government Source for Fuel Economy Information. Fuel Economy. Last accessed May 16 2012 `http://www.fueleconomy.gov/feg/`,.

[120] F. Tianheng, Y. Lin, G. Qing, H. Yanqing, Y. Ting, and Y. Bin. A supervisory control strategy for plug-in hybrid electric vehicles based on energy demand prediction and route preview. *IEEE Transaction on Vehicular Technology*, 64(5):1691–1700, May 2015.

[121] N. Tishby, F. C. Pereira, and W. Bialek. The Information Bottleneck Method. In *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377, 1999.

[122] H. Tong and C.-Y. Lin. Non-negative residual matrix factorization with application to graph anomaly detection. In *Proceedings of the 11th SIAM International Conference on Data Mining (SDM)*, pages 143–153, 2011.

[123] United Nations. World urbanization prospects. Technical report, United Nations, Department of Economic and Social Affairs, Population Division, 2014.

[124] U.S. General Services Administration. Office space use review, current practices and emerging trends, September 1997.

[125] K. Valogianni, W. Ketter, J. Collins, and D. Zhdanov. Effective management of electric vehicle storage using smart charging. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI)*, 2014.

[126] Y. Wang, Y. Zheng, and Y. Xue. Travel time estimation of a path using sparse trajectories. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 25–34, 2014.

[127] F. Wefering, S. Rupprecht, S. Buhrmann, S. Bohler-Baedeker, R. ConsultForschung, and B. GmbH. Guidelines: Developing and implementing a sustainable urban mobility plan. Technical report, European Union, 2014.

[128] J. Wirges, S. Linder, and A. Kessler. Modelling the development of a regional charging infrastructure for electric vehicles in time and space. *European Journal of Transport and Infrastructure Research*, 12(4):391–416, 2012.

[129] Y. Wu, F. Chen, C.-T. Lu, and B. Smith. Traffic flow prediction for urban network using spatio-temporal random effects model. In *Proceedings of the 91st Annual Meeting of the Transportation Research Board (TRB)*, 2012.

[130] Y.-J. Wua, F. Chenb, C.-T. Luc, and S. Yang. Urban traffic flow prediction using a spatio-temporal random effects model. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, July 2015.

[131] M. Xu, J. Wu, Y. Du, H. Wang, G. Qi, K. Hu, and Y. Xiao. Discovery of important crossroads in road network using massive taxi trajectories. In *Proceedings of the 3rd International Workshop on Urban Computing (UrbComp)*, 2014.

[132] J. Yang and J. Leskovec. Patterns of temporal variation in online media. In *Proceedings of the 4th ACM international conference on Web search and data mining (WSDM)*, pages 177–186, 2011.

[133] Z. Yang, L. Sun, J. Chen, Q. Yang, X. Chen, and K. Xing. Profit maximization for plug-in electric taxi with uncertain future electricity prices. *IEEE Transaction on Power Systems*, 29(6):3058–3068, Nov 2014.

[134] J. Yuan, Y. Zheng, and X. Xie. Discovering region of different functions in a city using human mobility and POI. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2012.

[135] J. Yuan, Y. Zheng, X. Xie, and G. Sun. Driving with knowledge from the physical world. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2011.

[136] J. Yuan, Y. Zheng, X. Xie, and G. Sun. T-drive: Enhancing driving directions with taxi drivers' intelligence. *IEEE Transaction on Knowledge and Data Engineering*, 25(1):220–232, January 2013.

[137] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, and Y. Huang. T-Drive: Driving directions based on taxi trajectories. In *Proceedings of the 18th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS)*, 2010.

[138] T. Zhang, W. Chen, Z. Han, and Z. Cao. Charging scheduling of electric vehicles with local renewable energy under uncertain electric vehicle arrival and grid power price. *IEEE Transaction on Vehicular Technology*, 63(6):2600–2612, July 2014.

[139] Y. Zheng, L. Capra, O. Wolfson, and H. Yang. Urban computing: Concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(3):38:1–38:55, September 2014.

[140] Y. Zheng, T. Liu, Y. Wang, Y. Zhu, Y. Liu, and E. Chang. Diagnosing new york city's noises with ubiquitous data. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, pages 715–725, 2014.

[141] Y. Zheng, L. Zhang, X. Xie, and W. Ma. Mining correlation between locations using human location history. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS)*, 2009.

[142] X. Zhu and D. Guo. Mapping large spatial flow data with hierarchical clustering. *Transaction in GIS*, 18(3):421–435, June 2014.