APPROVAL SHEET

Title of Dissertation: Data, Energy, and Privacy Management Techniques for Sustainable Microgrids

Name of Candidate: Zhichuan Huang
Doctor of Philosophy, 2017

Dissertation and Abstract Approved: ________________________
Ting Zhu
Assistant Professor
Department of Computer Science and Electrical Engineering

Date Approved: ________________________

ABSTRACT

Title: Data, Energy, and Privacy Management Techniques for Sustainable Microgrids

Zhichuan Huang, Doctor of Philosophy

Directed by: Assistant Professor Ting Zhu

Department of Computer Science and Electrical Engineering

The increased reliance on burning fossil fuels to generate electricity is rapidly depleting our planets finite resources and contributing to climate change. Recent studies show that buildings consume 40% of the annual energy consumption. Consequently, techniques to make buildings self-sustainable, while ensuring user well-being and comfort, are crucial for achieving a sustainable energy future in smart cities. Recently, the growing adoption of renewable energy sources has shifted the emphasis from large-scale centralized utility control of power grids to more localized energy system in microgrids, which comprises of residential and commercial buildings; and generate, store, and share electricity to balance local generation and consumption.

In this dissertation, we investigate the feasibility of microgrids by addressing three main challenges: i) how to collect and manage energy data from microgrids; ii) how to conduct energy energy management based on data analytics to minimize the total cost; iii) how to protect users' privacy in microgrids. Specifically, we first designed a low-cost hard-

ware PowerQM to enable accurate power quality monitoring. Next, to collect energy data from deployed multiple energy meters in residential homes, we present E-Sketch, a middleware for utility companies to gather data from smart meters with much less storage and communication overhead. Then, based on the energy data collected, we propose M-pred for accurate demand forecast by utilizing high granularity data collected from residential homes. With M-Pred, we propose two energy management techniques in microgrids: i) exploring energy sharing among residential homes to improve the utilization efficiency of renewable energy (e.g., solar energy); ii) scheduling demand in residential homes and generation in microgrids to minimize the total operational cost. Finally, to protect data privacy of homeowners, we leverage the unique feature of hybrid AC-DC microgrids and propose Shepherd, a privacy protection framework to effectively protect occupants privacy.

We implement and evaluate our proposed management techniques based on large energy datasets collected from more than 700 residential homes. The evaluation results show that i) our power quality meter PowerQM can achieve similar or even better accuracy than existing commercial products with much lower cost; ii) E-Sketch can significantly reduce the required data storage by 90% while preserving the data accuracy with more than 99%; iii) M-pred can achieve accurate demand forecast with negligible errors (e.g., Mean Absolute Percentage Error is 2.12%); iv) energy sharing can improve utilization efficiency of renewable energy up to 30%, while demand and generation scheduling can reduce total operational cost by 30% in microgrids; v) Shepherd can effectively protect users privacy of energy consumption information from multiple detection algorithms.

DATA, ENERGY, AND PRIVACY MANAGEMENT TECHNIQUES FOR SUSTAINABLE MICROGRIDS

by

Zhichuan Huang

Dissertation proposal submitted to the Faculty of the Graduate School of the
University of Maryland Baltimore County in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2017

Copyright © 2017 Zhichuan Huang.

ACKNOWLEDGMENTS

The process of PhD is a long journey and it would have been much more difficult to survive without vital support from everyone around me. I would like to express my sincere gratitude for everyone who helped me along this journey.

First and foremost, I would like to thank my advisor Prof. Ting Zhu for his support and guidance. This dissertation would not have been completed without his help. I have learned many key research skills from him: how to pick, motivate and define a research problem, find corresponding solutions, and how to write and present my work to other researchers.

I have enjoyed interacting with my committee members and have learned a great deal in the process. I would like to thank them for the same. I was fortunate to be able to work with Prof. David Irwin on multiple projects. He always provided me broader perspective about my work and systems research in general. I am glad to have had the opportunity to work with Prof. Yu Gu. He always provided me very relevant and insightful comments about my research problems. I am also thankful to him for his excellent career advice and encouraging words. I am also grateful to Prof. Jianwu Wang for his help and contributions in several research projects, especially in the beginning. Further, I am thankful to Prof. Nilanjan Banerjee and Chintan Patel for their inputs on my work.

My sincere thanks to my family members especially my parents. I am grateful to my parents for their unconditional love and support. Their support has been vital for me along the whole doctorate journey. I would also like to thank my friends Hongyao Luo, Yuling Niu, Li Lu, Xiaoxiong Zhang, Yilin Tao, Zicheng Chi, Tiantian Xie, Yao Yao, Wei Wang, Xin Liu, Yan Li, Michael Hammond, Haoyang Lu, Sandeep Narayanan, David Corrigan, Mengjun Xie, Jikui Su, Weigang Zhong, Ankur Sharma, Chu Huang, Zheng Li, Xiaoxia Zheng, Huseyin Aygan, Hongyu Sun, Zheng Lu, Yuping Zhang, Yao Li, Vlad Vengrenyuk for their support and help during these years. They provided a stimulating environment making my graduate student life enjoyable and truly memorable. Finally, my heartfelt thanks to Olivia and the rest of the staff at the Department of Computer Science and Electrical Engineering for their invaluable help and assistance over the years.

Zhichuan Huang
Baltimore, Maryland
April 26, 2017

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Microgrids play a important role in energy cyber-physical systems [73]. In a typical microgrid, it consists of local generators and energy storage (e.g., batteries) to provide power for a small community with commercial and residential buildings. Microgrids can provide power to places i) where the traditional power grid does not exist due to the poor economy or limited number of residences (e.g., islands); and ii) when the traditional power grid is temporarily not functioning due to severe weather conditions (e.g., storms). Therefore, microgrids have gained increasing attention recently [107]. However, due to the very limited capacities of local energy storage and energy generation, microgrids are more difficult to manage than traditional power grids.

To better manage the microgrids for sustainability, in this dissertation, we investigate the feasibility of sustainable microgrids from three perspectives: data, energy and privacy management. Specifically, we try to answer three questions: i) how to collect accurate energy data from microgrids and do data analytics; ii) how to conduct energy energy management based on data analytics to minimize the total cost of the microgrids; iii) how to protect users' privacy while collecting the data for analytics.

1.1 Background and Motivation

Buildings account for more than 75% of the United States electricity use [35], with the residential sector accounting for 54% of this total. To reduce energy consumption in buildings, renewable energy sources have been investigated to incorporate with the traditional power grid. Compared to energy supply from traditional power grid, renewable energy generation such as solar energy locates in a distributed manner, and introduces much less energy transmission loss, where energy transmission and distribution losses in power grid can be as high as 32.3% [35]. According to technical report from SunShot Initiative [108], the solar energy source will reach 14% of the total energy generation by 2030 and 27% by 2050. Meanwhile, the development of energy storage technology makes the distributed energy storage (e.g., Powerwall by Tesla) more and more affordable. With these increasing distributed energy generation and storage, buildings can generate, store and consume the energy locally, and form a microgrid. Compared to traditional power grid, microgrids can i) improve the energy efficiency with much short transmission loss; ii) provide the power supply when the traditional power grid is temporarily not functioning due to severe weather conditions (e.g., storms).

However, due to the very limited capacities of local energy storage and energy generation, microgrids are even more difficult to maintain than traditional power grids. To ensure the stability and reliability of a microgrid, we need to i) monitor the energy generation and consumption in real-time; ii) collect high granularity energy consumption data for better demand forecast; iii) conduct real-time energy management based on collected energy data.

To monitor the real-time energy data in buildings, many works have been developed [22] [64]. A high-fidelity wireless building energy auditing network has been built to analyze the energy consumption of a large building to identify energy waste [54]. The power budgeting system is proposed for virtualized infrastructures that enforces power limits on individual distributed applications to ensure actual consumption never exceeds capacity [67]. A lowest cost AC plug-load meter that measures real, reactive and apparent power is proposed in [33]. However, different from these existing works, in this dissertation, we are aiming to monitor the power quality (voltage, frequency and phase angle) of microgrids instead of power consumption in the power grid, which is more important for the stability of the microgrid.

With the collected accurate energy data from smart meters, different applications have been investigated to best utilize these data. One main application is how to improve the accuracy of energy demand forecast for better energy management. Some prediction algorithms have been proposed based on the aggregated energy consumption in the whole grid [11, 45]. However, there are two major limitation of these approaches: i) these algorithms can only provide demand forecast for the next day or a even longer time period, while real-time generation management (e.g., hourly) is required in microgrids; ii) the existing approaches is that they only predict the hourly average energy consumption, while peak demand is a more serious problem in microgrids with limited capacities of local energy storage and generation.

To minimize the energy cost in buildings, different techniques have been proposed for improving building energy efficiency by monitoring the occupancy [8] [14]. A novel optimization theoretic approach is proposed to scheduling electrical vehicle charging to avoid charging at peak time [7]. However, these work mainly focus on demand scheduling, while we need to conduct both demand and generation management in microgrids. Moreover, how to incorporate the renewable energy in microgrids is also an important issue due to the natural mismatch between energy harvesting and consumption in buildings.

Besides, with the high granularity energy data collected from smart meters, recent studies show that it is possible to detect human's activities in the house based on the energy consumption data. There are some cases where fairly exact appliance signatures are known, the edge detection can be disaggregated into its individual loads by solving a binary knapsack problem [72]. In [4], Molina-Markham et al. show how to use off-the-shelf statistical tools to detect household habits from power consumption patterns. Therefore, how to protect homeowners' privacy while collecting their energy consumption data for better energy management is also a critical problem in microgrids. Different approaches are proposed to protect the privacy by utilizing batteries or generate some fake power consumption in the buildings [120]. The privacy protection mechanism should introduce minimum energy overhead and affect on human's behaviors.

1.2 Contributions

This dissertation designs and evaluates systems to collect energy data from residential homes, conduct energy management in microgrids based on data analytics, and how to protect privacy of homeowners. To perform these tasks efficiently and in a user-friendly manner, we devise novel algorithms drawing from diverse fields of mathematical modeling, machine learning, and optimization. We evaluate the proposed algorithms and systems

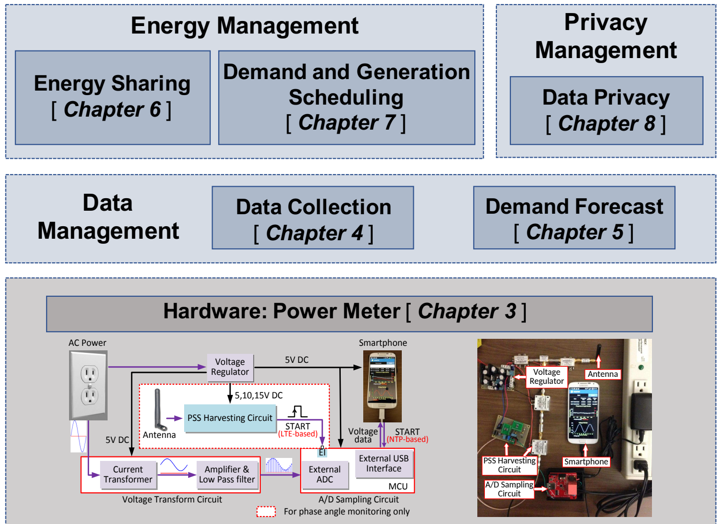


Figure 1.1: Overview of Dissertation

using simulations based on real power consumption data either collected from real homes, or obtained from smart meter companies.

The key contributions and systems developed for data, energy and privacy management in microgrids are shown in Figure 1.1 and summarized in the following sections.

1.2.1 Power Quality Meter: PowerQM

Traditional power grid is not resistant to severe weather conditions, especially in remote areas. For some areas with few people, such as islands, it is difficult and expensive to maintain their connectivity to the traditional power grid. Therefore, a self-sustainable microgrid is desired. However, given the limited local energy storage and energy generation, it is extremely challenging for a microgrid to balance the power demand and generation in real-time. To realize the real-time power quality monitoring, the power quality information of microgrid, such as voltage, frequency and phase angle in each home, needs to be collected in real-time. Furthermore, the unreliable sensing results and data collection in a microgrid make the real-time data collection more difficult. To address these challenges, we designed an accurate real-time power quality data sensing hardware to sense the voltage, frequency and phase angle in each home. A novel data management technique is also proposed to reconstruct the missing data caused by unreliable sensing. We implemented our system over off-the-shelf smartphones with a few peripheral hardware components, and realized an accuracy of 1.7 mHz and 0.01 rad for frequency and phase angle monitoring, respectively. We also show our data management technique can reconstruct the missing data with more than 99% accuracy.

1.2.2 Energy Data Collection Middleware: E-Sketch

To reduce peak demand, many utility companies are transitioning from fixed rate pricing plans to real-time pricing plans. To apply real-time pricing plans, it is crucial to collect accurate real-time power consumption readings from individual homes. Thus, utility companies are increasing the installation of smart meters in individual homes. Smart meters can record energy related data (e.g., power consumption) every second. However, power consumption data with high time granularity needs huge data storage space and generates significant communication overhead for utility companies to gather all the data for the pricing plans. In this paper, we present E-Sketch, a middleware for utility companies to gather data from smart meters with much less storage and communication overhead. E-Sketch utilizes adaptive sampling to compress power consumption changes in time domain. Then frequency compression is applied to further compress the sampled data. We conducted extensive system evaluations with 30 homes second-level power consumption data for more than 2 months. Results indicate i) our design can reduce data storage space significantly by 90% with more than 99% accuracy of second-level power consumption on average for a single home, and ii) our design can achieve even more than 99.8% accuracy on average for aggregated power consumption of 30 homes.

1.2.3 Energy Data Analytics for Demand Forecast: M-Pred

Accurate energy demand prediction is very important for smart grids to conduct demand response and stabilize the grids. In previous work, many prediction algorithms are proposed to improve the energy consumption prediction accuracy based on the aggregated energy consumption in the whole grid. Recently, with the increasing installations of smart meters in individual homes, high granularity (e.g., per minute) energy consumption data in individual homes becomes available and provides us a great opportunity for better energy consumption prediction. In this paper, we propose M-Pred to utilize the high granularity energy consumption data collected by smart meters in individual homes for better energy consumption prediction in smart grids. In M-Pred, we propose a learning algorithm to learn energy consumption patterns of individual homes from the high granularity energy consumption data. The consumption patterns we learn from homes are then applied for energy consumption prediction in smart grids. Furthermore, since not every home in a smart grid is equipped with a smart meter, we propose a matching and prediction algorithm to leverage the multi-granularity energy data for accurate consumption prediction. We conducted extensive system evaluations with 726 homes minute-level power consumption data for more than 12 months. The simulation results show that our design can provide accurate energy consumption prediction for the next hour with negligible errors (e.g., Mean Absolute Percentage Error is 2.12%).

1.2.4 Energy Sharing in Microgrids

To reduce electricity usage and peak demand, many utilities are introducing market-based time-of-use (TOU) pricing models. In parallel, government programs that in- crease the fraction of renewable energy are incentivizing residential consumers to adopt on-site renewables and energy storage. Connecting on-site renewables and energy storage between homes forms a sustainable microgrid capable of generating, storing, and sharing electricity to balance local generation and consumption in residential areas. In this paper, we investigate how to minimize the costs of electricity from a utility for a microgrid under market-based TOU pricing models. In particular, we (i) present a system architecture for an energy-sharing microgrid; and (ii) develop optimal energy-sharing algorithms for homes within the microgrid. We conduct an extensive evaluation under two typical TOU pricing models that use data from more than 40 homes. Our results indicate that our system reduces the costs of Alternating Current (AC) electricity by 20%, even for homes with similar energy usage patterns.

1.2.5 Demand and Generation Scheduling

Given the limited capacities of local energy generation and storage in such a community, it is extremely challenging for an isolated microgrid to balance the power demand and generation in real-time with dynamically changing energy demand. Meanwhile, more and more sensing devices (such as smart meters) are deployed in individual home to monitor the real-time energy data, which can be helpful for homes and microgrid to better schedule the workload and generation. However, it is still difficult to conduct real-time distributed control due to the unreliable sensing devices and delayed communication which introduce missing and delayed data. To address these issues in microgrids, we designed a novel technique for the system to i) process the collected sensing data, ii) reconstruct the missing data caused by sensing error or unreliable communication, and iii) predict the future demand for real-time distributed control with missing data in extreme situations. The control center then decides the operations of the local generator and each home decides the scheduling of the flexible workload of appliances based on the collected and predicted data. Through extensive experiments and simulations, we show that our design can recover the missing data with more than 99% accuracy and our distributed control can balance power demand and generation in real-time and reduce the operational cost by 23%.

1.2.6 Privacy Management: Shepherd

With the high granularity power consumption data, Non-Intrusive Load Monitoring (NILM) can be applied to analyze the data for revealing appliances' activities ([47]). The widely used technique is the edge detection ([68]), which looks for the sharp edges that reveal the significant changes in the steady power consumed by the household. More seriously, we demonstrated a new signature detection technique which can reveal appliances' usage more accurately than existing approaches. Appliances usage information can then be used to reveal private information of occupants. For example, usage time of certain appliances (e.g., water heater) can reveal the number of people living in the home. Furthermore, changes of appliances usage patterns can also reveal private information (e.g., health conditions). For example, if a person usually turns off all the lights when he/she sleeps, and suddenly he/she turns on and off the lights frequently in the night while other appliances' usage patterns stay the same; this indicates that he/she may be sick or has a sleeping problem. Thus, it is critical to protect power consumption information and prevent privacy leakage for occupants in individual homes.

1.2.7 Dissertation Outline

Chapter 2 includes background information needed to set context for the contributions in this dissertation. Chapter 3 presents PowerQM, an accurate power quality monitor with much lower cost compared to existing commercial products. E-Sketch, a middleware for minimizing data storage and communication overhead between smart meters and servers is presented in chapter 4. Chapter 5 talks about M-Pred, which utilizes the high granularity energy data collected by PowerQM through E-Sketch for accurate demand forecast in microgrids. Chapter 6 details our solution for energy sharing, which addresses the mismatch problem between energy harvesting and consumption in residential homes to improve the renewable energy utilization efficiency. Chapter 7 presents another approach of energy management by scheduling both demand in residential homes and generation to minimize total operational cost in microgrids. With the large scale energy data, we show the possible privacy leakage in Chapter 8 and provide corresponding privacy protection. Finally, we conclude the dissertation with a summary of findings and future work in Chapter 9.

CHAPTER 2

BACKGROUND AND MOTIVATION

This chapter presents background information on microgrids management to set the context for our contributions. More detailed related work sections are included in the relevant chapters.

2.1 Background of Microgrids

In microgrids, researchers have i) developed models based on measurement from phaser measurement units to solve wide area control problem of large-scale power systems ([1, 2, 3]), ii) investigated the integration of renewable energy into power grid ([6, 60, 78]), iii) optimized the packing size of large scale batteries to improve battery utilization in microgrids ([40]), and (iv) applied stochastic network calculus to analyze the power supply reliability with various renewable energy configurations and store that energy into very large scale batteries ([113]). Our work builds on previous works, but targets minimizing energy sharing loss over the small community level DC line.

2.2 Energy Monitoring

Energy monitoring is the basis of the research in microgrids. There are many works on energy monitoring in buildings [22] [54] . A power budgeting system is proposed for virtualized infrastructures that enforces power limits on individual distributed applications to ensure actual consumption never exceeds capacity [67]. A contactless sensing method is proposed to detect 100W loads from 10cm away [94]. A lowest cost AC plug-load meter that measures real, reactive and apparent power is proposed in [33].

2.3 Energy Data Management in Microgrids

Energy data is widely used to reduce and schedule energy generation and consumption ([121, 122]). Chen et al. explore energy consumption in everyday home environments to study the relationship between behavioral patterns ([27]) and energy consumption and investigates how this relationship can be helpful for people to act in a more energy-efficient manner. Xiang et al. design a novel data aggregation scheme that exploits compressed sensing (CS) to achieve both recovery fidelity and energy efficiency in WSNs with arbitrary topology ([118]). A system framework of data reduction is proposed to minimize energy consumption in wireless sensor network ([126]). There are also works on energy data collection. A new and severe denial of service attack is proposed for data collection in AMI network ([123]). Time-Log Tree (TL-Tree), a novel indexing structure is proposed to consider time-series as a primary characteristic for optimizing both memory and energy constraints ([66]). Zhang et al. study timely, cost-minimizing upload of massive, dynamically-generated, geo-dispersed data into the cloud, for processing using a

MapReduce-like framework ([125]).

Different from previous work that focus on data collection and aggregation, we propose to use data compression to process collected energy data. By learning from pattern of energy consumption data, we present the design to save storage space while maintain accurate data.

2.4 Energy Management in Microgrids

This research mainly focuses on (i) energy auditing ([53]) and design of control algorithms to reduce energy consumption inside a single building ([19, 50]); (ii) reducing the energy usage of building-wide heating, energy-efficient building automation, ventilation, and air conditioning ([13, 70]); (iii) investigation on the integration of renewable energy into power grid ([130, 131]); and (iv) applying stochastic network calculus to analyze the power supply reliability with various renewable energy configurations and store that energy into very large scale batteries ([82]); and v) taking model predictive control approach to schedule the workload to reduce the energy cost in the buildings ([128]). Our work takes a different approach to reduce energy cost by sharing the renewable energy. Unlike these other approaches, our work opens up new approach where energy can be gained efficiently and used smartly.

Therefore many approaches are devised to minimize building power consumption. Several control algorithms are proposed to use occupancy information to reduce energy consumption while maintaining people's comfort in ([19]). An efficient energy control system is proposed by combining optimization and scheduling ([86]). In our paper, the energy sharing among homes with renewable energy is studied and the energy consumption of the cluster of homes is reduced by energy sharing. Even enormous work has been done and still it's going on in the areas like energy-efficient building automation ([13]), or energy saving electronics ([30, 111]), energy efficient data centers ([18, 75]), optimal charging of plug-in hybrid electric vehicles ([104]) but our work opens up new approach where energy can be gained efficiently and used smartly. A lot of work has been done in increasing the efficiency of these batteries ([55]). As well as changing the harvesting patterns or effective use of TOU ([16]). But all of these have certain limitations upon their usage.

2.5 Data Privacy in Microgrids

The large-scale placement of smart meters has introduced leakage of private and valuable information about occupants' activities [29]. NILM algorithms have been widely used in the research of residential settings to reveal the usage of individual appliances with consumption data [96]. In [68], NILM algorithms are extended to evaluate the threat to individual privacy by considering the potential disclosure from smart-meter data. A statistical technique is used to develop a simple approach to discover people's life patterns [4]. In this paper, we develop a new detection technique based on the consumption signature of appliances that achieves a higher detection ratio.

To protect the privacy of energy data, Battery-based Load Hiding is to use a rechargeable battery to store and supply power to home appliances at strategic times to hide the appliances' consumption from smart meters [120]. The BE algorithm [109] tries to avoid charging the external load whenever possible, and when the actual demand is different from the external load, the battery can be charged or discharged to counteract the difference. The NILL algorithm [95] has three states and attempts to maintain a different constant load for each state.

CHAPTER 3

HARDWARE DESIGN OF POWER QUALITY MONITOR

Given the limited local energy storage and energy generation in microgrids, it is extremely important to balance the power demand and generation in real-time. To realize the real-time balance between power demand and generation, the power quality information of microgrid, such as voltage, frequency and phase angle in each home, needs to be collected in real-time. In this chapter, we designed an accurate real-time power quality data sensing hardware to sense the voltage, frequency and phase angle in microgrids.

3.1 Introduction

A microgrid enables local electricity generation, energy storage and load to operate independently of the traditional power grid. Compared to the traditional power grid, a microgrid normally coordinates energy demand and generation in a small community, thus the energy transmission and distribution losses in a microgrid are much lower and the investment for transmission and distribution infrastructures can be minimized. Furthermore, because the microgrid can be isolated from the traditional power grid, it provides the power supply i) to places where the traditional power grid does not exist due to the poor economy or limited number of residences (e.g., islands); ii) when the traditional power grid is temporally not functioning due to severe weather conditions (e.g., storms). Therefore, it has been gaining increasing attention lately.

To enable the functionality of microgrid, the key challenge is to balance the power demand and generation in real-time with dynamically changing power supply and demand. When a microgrid is connected to the traditional power grid, variations in equilibrium are resolved through the support of the traditional power grid. However, when the traditional power grid is down or not connected, the challenge to maintain stability is greater because there are fewer resources in a microgrid. In a typical microgrid, the power generation comes from local generators, renewable energy and energy stored in batteries, while the power demand comes from the loads from commercial and residential buildings. To maintain the stability of a microgrid, we need to schedule the operations of local generators, batteries, and controllable workloads of appliances to offset the dynamically changing and renewable energy generation and power demand of uncontrollable appliances. To achieve the stability of the microgrid, power quality through the power lines needs to be monitored in real-time (e.g., every a couple seconds). The power quality of microgrid is typical monitored with the data such as voltage, frequency and phase angle. However, devices that can monitor these data are normally very expensive. For example, the installation cost of one transmission-level Phasor Measurement Unit (PMU) is more than $80,000 at the Tennessee Valley Authority (TVA). The reason why PMU is so expensive is that specialized DSP chips are needed for synchrophasors for synchronizing the GPS signal. In the meanwhile, PMU requires GPS signal for synchronization and GPS does not work in indoor environment (such as homes and buildings). In this chapter, we design a low cost accurate power quality data sensing hardware to sense the voltage, frequency and phase angle in each home and utilize the existing WiFi or cellular network for data communication. Specifically, we summarize our contributions as follows:

• We designed and implemented a high accurate real-time voltage, frequency and phase angle monitoring hardware platform with a small quantity of peripheral hardware. Experimental results show that our prototypes can achieve frequency accuracy of 1.7 mHz and voltage accuracy of 0.02 V.

• The cost of our hardware is much lower compared to existing products such as PMU and Frequency Disturbance Recorder (FDR).

3.2 Background and Motivation

A microgrid is a distributed power system that can autonomously coordinate local generations and demands in a dynamic manner [63]. Microgrids can operate in either grid-connected mode or isolated mode. There have been worldwide deployments of pilot microgrids, such as the US, Japan and European.

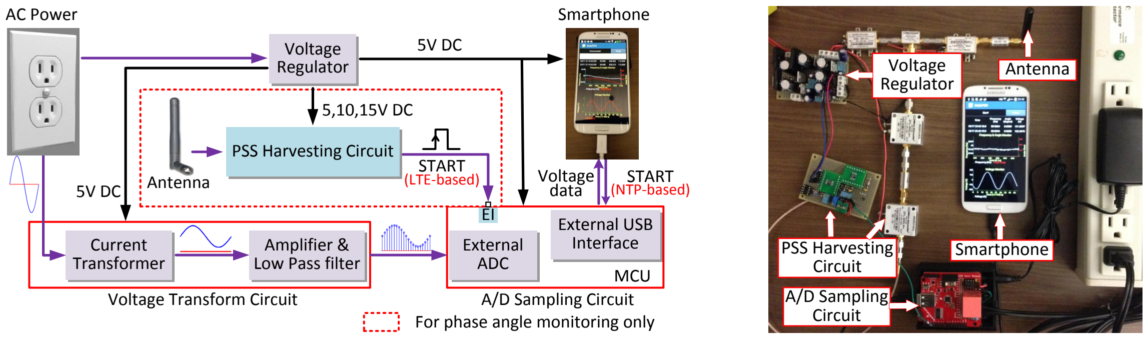
3.2.1 Background

In this chapter, we consider a modern microgrid, which consists of generation technology (e.g., renewable energy and local electricity generators) and batteries. To ensure compatibility with the traditional power grid, we adopt the microgrid architecture, which is similar to the one used in a traditional power grid. If the microgrid is built from nothing (e.g., island, where there is no electricity grid before), the microgrid can be built the same architecture as traditional grid with a distribution network across the community of homes. If the microgrid is built from a traditional grid, we only need to add local generators, batteries and control center into the microgrid. Within the microgrid, sensors are deployed in each home to collect energy related data (e.g., power, voltage, frequency and phase angle) and send to control center for power quality monitoring.

3.2.2 Motivation

Different from previous work on power monitoring devices, in this chapter, we propose a hardware to sense the voltage, frequency and phase angle of the microgrid. In this section, we demonstrate why these sensing data is important and the reliability of sensing results.

Sensing Requirements. In traditional power grid, voltage, frequency and phase angle is used in different layers to monitor the stability and power quality of power grid. Specifically, i) voltage needs to be regulated to ensure the appliances in each home are working in the proper state; ii) frequency is used to monitor the power load in the transmission lines so that utility company knows whether to generate more or less energy; iii) phase angle difference between buses is measured by PMU to indicate the system stability and stress in advance. However, devices that can monitor these data are normally very expensive. Furthermore, the distributed renewable energy can put back unpredictable amount of energy into microgrid, which makes the distributed power quality monitoring is even more important and challenging. Thus, it is crucial to develop a low cost device for power



(a) Hardware design (b) Hardware implementation

Figure 3.1: Power quality monitoring over smartphones

quality sensing in microgrids.

3.3 Sensing Hardware

In this section, we propose the detailed design of sensing hardware in homes. The main challenge of sensing hardware is the time synchronization among sensors. Because self-sustainable microgrid requires to monitor the energy data from all the homes in real-time, the readings from different homes must be accurately synchronized. To address this challenge, we propose two time synchronization methods applied in our sensing hardware for frequency and phase angle monitoring.

3.3.1 Hardware Design and Implementation

Our sensing hardware consists of a voltage regulator module, a voltage transform circuit, a microprocessor-based analog-to-digital (AD) sampling module, a PSS harvesting circuit, and an Android-based smartphone. The system design and implementation are shown in Figure 3.1(a) and 3.1(b), respectively. The voltage regulator outputs the necessary DC power to power up the whole system, including the smartphone. A voltage transform circuit takes an analog voltage signal from 120 V wall outlet and transforms the AC power into the voltage range for analog-to-digital conversion (ADC). An 8-bit microprocessor (MCU) is used to control the voltage sampling process through external ADC at the sampling frequency of 1,440 Hz, and sends the data to smartphone every 100 ms for phasor state estimation.

The communication between the microprocessor and the smartphone is conducted by the USB host controller IC MAX3421E (USB host shield) [77]. The MAX3421E host controller contains the digital logic and analog circuitry necessary to implement a full-speed host compliant to USB specification v2.0. Under this circumstance, similar as being connected to the desktop PC, the smartphone behaves as USB slave in relation to the USB host chip, and can communicate with the MCU and be charged at the meantime. The MCU communicates with the USB host through SPI bus.

The PSS harvesting circuit, shown within the dotted line in Figure 3.1(a), extracts the PSS signals and transmits them to the MCU in the form of pulses. The rising edges of the pulses will be detected through External Interrupt (EI) in the MCU, and trigger new sampling cycles.

Hardware Cost. Our implementation includes following hardwares: i) a MCU (ATmega328), which costs less than \$10; ii) multiple ADCs, which costs less than \$ 10; iii) USB host controller (MAX3421E), which costs less than \$10; iii) harvesting antenna, which costs less than \$10; iv) other our designed circuits, which also costs less than \$10; and v) a smartphone to collect data and running NTP protocol, which we use Samsung S3. Note that we only use the smartphone for the prototype, we can easily design a board running NTP protocol and store the data in the board. Therefore, our total cost for hardware

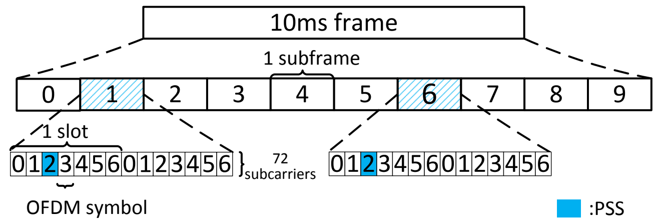


Figure 3.2: Synchronization signals in LTE FDD downlink

can be less than $50.

3.3.2 Time Synchronization

Power grid operations should be monitored in real-time using globally synchronized timestamps, so that measurements from dispersed locations can be compared on a common time reference. Being different from current synchrophasors, our system does not rely on continuous GPS reception and hence is highly accessible and applicable to heterogeneous microgrid scenarios. Instead, we develop two techniques to provide timing signals that are necessary for precise monitoring.

Frequency monitoring: Network Time Protocol (NTP) [80] is being widely used in current computing systems, such as the Windows Time Service, in order to synchronize the local clock of digital devices with Coordinated Universal Time (UTC). Due to the uncertainty of network transmission delay, the timing accuracy of NTP is in the order of 10 milliseconds [114] and is much lower than that of the GPS signal. Nevertheless, by investigating the frequency oscillation in the power grid, we found that such time precision is sufficient for detecting a frequency disturbance event. Therefore, NTP is an appropriate alternative to GPS to provide global time synchronization to frequency measurement data.

Phase angle monitoring: Compared to frequency monitoring, phase angle moni-

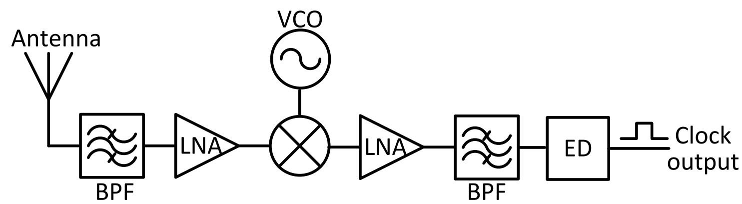


Figure 3.3: System block diagram of PSS signal harvesting circuit

toring requires a globally synchronized clock with higher accuracy and stability. Simply speaking, a 15-millisecond timing error, which is usually the upper bound of NTP timing error, corresponds to an unacceptable phase angle measurement error of 5.76 radians in a 60 Hz power system. Instead, we propose to harvest the precise timing signal from the 4G LTE cellular signal, which is widely available nationwide nowadays.

The enhanced base station (eNodeB) of LTE is strictly synchronized with GPS or the Precision Time Protocol (PTP) [51]. The cell ID in the LTE network is defined within two synchronization signals, namely Primary Synchronization Signal (PSS) and the Secondary Synchronization Signal (SSS). Figure 3.2 illustrates the LTE frame format and the location of synchronization signals under Time-Division Duplexing (TDD) mode. The PSS repeats periodically (every 5 ms) and therefore can be regarded as a time synchronization signal. Note that though the frame structure is a little different in Frequency-Division Duplexing (FDD) mode, the PSS will also repeat every 5 ms. Thus, our design can be applied with LTE under different modes.

3.3.3 PSS Harvesting Circuit Design

Since measurement of phase angle requires more accurate timing information than the frequency monitoring does. In our system design, we aim to harvest synchronization signals from 4G LTE cellular signal for time synchronization. Similar to the GPS-based system, the harvested LTE signal can be directly used to trigger a new sampling cycle.

In LTE networks, to achieve high data transmission rate, Orthogonal Frequency Division Multiple Access (OFDMA) is utilized as the physical layer technique in the downlink data transmission. The frequency of PSS signal (200 Hz) is far lower than the bandwidth of data transmitted (in the order of 1 MHz). Since our purpose of PSS detection is not to decode the signal but only to identify the arrival of the PSS signals, the PSS signal can be detected based on the scheme shown in Fig. 3.3. A Voltage-Controlled Oscillator (VCO) is used to detect the frequency band with the strongest signal strength. The signal in 1900 MHz frequency band is selected and down-converted to 200 kHz intermediate frequency (IF) output. The PSS signal would be transformed as a pulse after passing the bandpass filter with a bandwidth of 120 kHz and the envelope detector. The MCU will capture the rising edge of the PSS pulses as the trigger to start a new sampling cycle.

The cell ID can be calculated as:

$$N_{ID}^{Cell} = 3N_{ID}^{1} + N_{ID}^{2} \tag{3.1}$$

where $N_{ID}^{1} \in 0, 1, ..., 167$ is the cell identity group and is located in SSS signal, and $N_{ID}^{2} \in 0, 1, 2$ is the cell identity and is located in PSS signal. The SSS signal indicates the frame timing as they are different within a frame, while PSS signal indicates the OFDM symbol timing as they are the same within a frame. The sequence used for the PSS is generated from a frequency domain Zadoff-Chu (ZC) sequence according to [84]:

$$c_u(n) = \begin{cases} e^{-j\frac{\pi u n(n+1)}{63}} & n = 0, 1, ..., 30 \\ e^{-j\frac{\pi u(n+1)(n+2)}{63}} & n = 31, 32, ..., 61 \end{cases} \tag{3.2}$$

where the ZC root u for the PSS sequence is 25,29,34 that corresponds to the value of N_{ID}^{2}

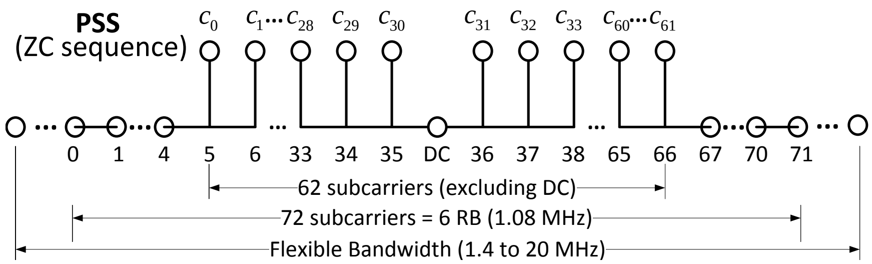


Figure 3.4: Transmission of PSS signal in the frequency domain

(a) eGauge and our prototype

(b) Synchronization experiment

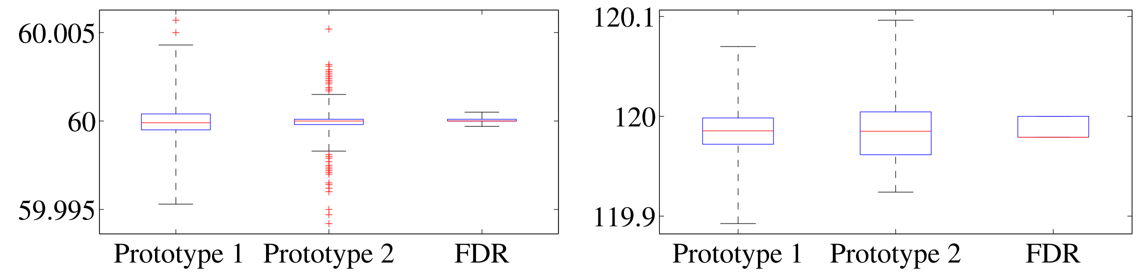
Figure 3.5: Experiment setup in residential homes

= 0,1,2, respectively.

As shown in Fig. 3.4, comprised with three Zadoff-Chu sequences in frequency domain, the PSS signal maps to the central 62 subcarriers around DC (within the central six Resource Blocks (RBs)), enabling the frequency mapping of the synchronization signals to be invariant with respect to the system bandwidth, which varies from 1.4 MHz to 20 MHz.

3.4 Evaluations

In this section, we evaluate the performance of our design from two perspectives: i) data measurement accuracy; ii) time synchronization between sensors.



(a) Frequency (Hz) (b) Voltage (V)

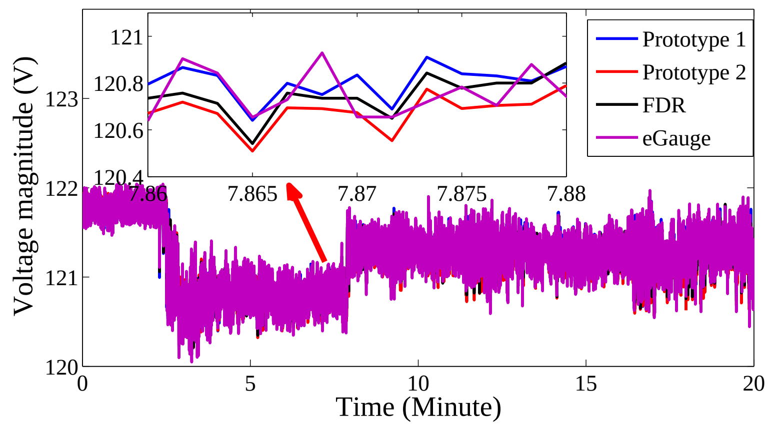


Figure 3.7: Voltage measurement from wall output

3.4.1 Frequency & Voltage & Phase Angle Measurement

We collect empirical data of voltage, frequency and phase angle from three homes. The experiment setup is shown in Figures 3.5(a) and 3.5(b). To evaluate the accuracy of our system, we test our system against the traditional Frequency Disturbance Recorder (FDR) and commercial products from eGauge. Both standard power generator and AC wall power are used in our experiments.

First, the frequency measurement result over standard 60 Hz power generator is shown in Figure 3.6, and the GPS-synchronized Doble F6150 signal generator is used to generate 60 Hz, 120V AC power. The prototypes can achieve a frequency accuracy of less than 1.7 mHz and a 0.02 V voltage magnitude accuracy. Second, the measurement

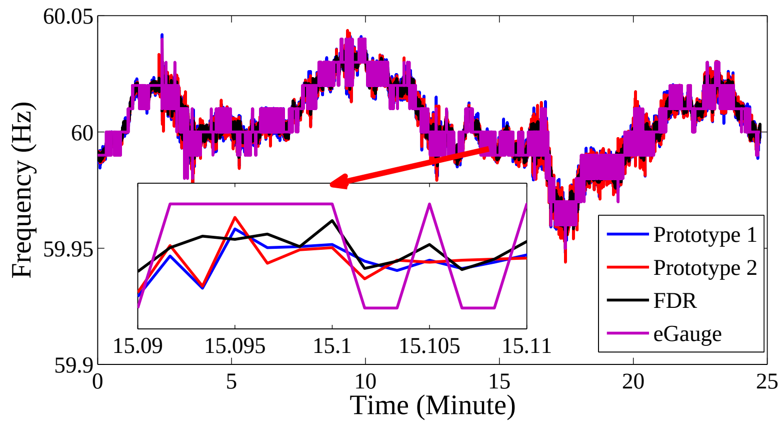
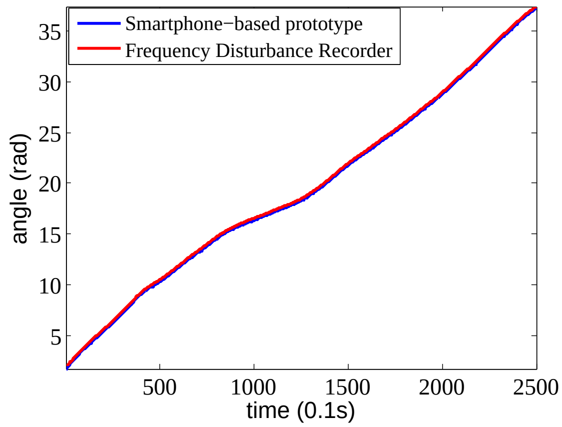


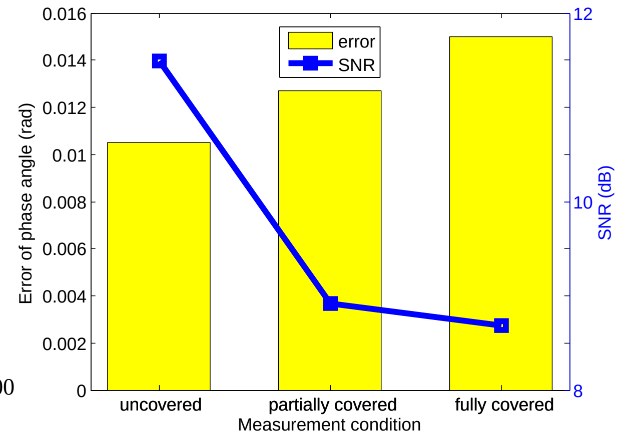
Figure 3.8: Frequency measurement from wall output

results over the 120V AC wall outlet are shown in Figure 3.7 and 3.8. We compare our design with both FDR and commercial eGauge energy monitoring system. We find our system has similar accuracy of voltage compared to eGauge, and much higher accuracy of frequency than eGauge over time.

For angle measurement, we conducted the experiments over wall power. The difference between our design and FDR is 0.011 rad (shown in Figure 3.9(a)). Compared to FDR with an accuracy of 0.0001 rad, the timing error is mainly introduced in the PSS signal harvesting process. The PSS signals are harvested in the form of pulses, and the slope of the rising edge of the PSS pulse is flatter and can be affected by the strength of the signal. To evaluate the relationship between the measurement accuracy and the quality of the LTE cellular signal reception, we emulate the environment with different signal-to-noise ratio (SNR) by using aluminum foil to cover the antenna of the PSS harvesting antenna. The performance of the phase angle measurement with respect to different SNR is depicted in Fig. 3.9(b). PSS harvesting module can achieve the highest SNR of 11.49 dB under no cover with the error of 0.01 rad on phase angle monitoring. As the coverage area of the foil increases, the error increases to 0.0127 and 0.015 rad in partially covered and totally



(a) Wall output



(b) Error

Figure 3.9: Phase angle measurement

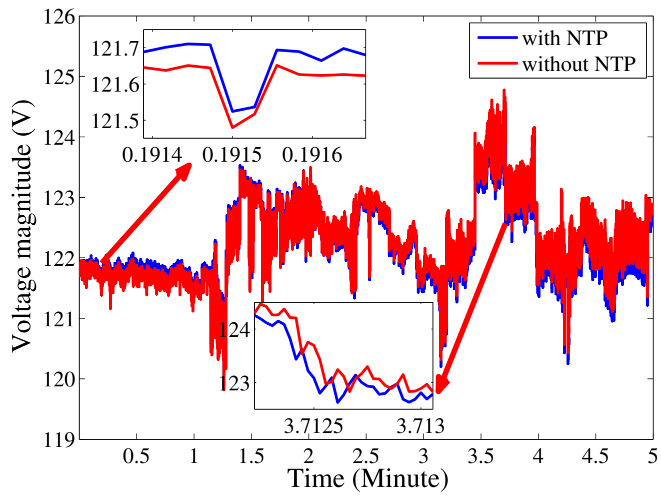


Figure 3.10: Measurement with/without NTP synchronization

covered situation, respectively. However, the accuracy of phase angle sensing is still good enough for power quality monitoring.

3.4.2 Time Synchronization

The synchronization effect of NTP approach is evaluated by comparing the measurement of two prototypes, one of which disables the WiFi network to disable NTP synchronization. The local clock of both two prototypes are calibrated before the experiment starts. The voltage measurement result is shown in Figure 3.10. It is expected that, without synchronization, the smartphone will assign timestamps to measurement points through local clock. The local clock which is vulnerable to the environmental factors, such as temperature, will continuously drift without calibration. 10 minutes after startup, the spikes of voltage measurement are aligned with each other. However, after 3 hours, there is a two-sample-interval (0.2 s) delay in the NTP-disabled device, which means its local time is drifted by 0.2 seconds and the measurement will be assigned a timestamp that is earlier than the real time. It's expected that this delay will accumulate as time goes.

3.5 Related Work

Energy Monitoring. There are many works on energy monitoring in buildings [22] [54] . A power budgeting system is proposed for virtualized infrastructures that enforces power limits on individual distributed applications to ensure actual consumption never exceeds capacity [67]. A contactless sensing method is proposed to detect 100W loads from 10cm away [94]. A lowest cost AC plug-load meter that measures real, reactive and apparent power is proposed in [33].

Different from these existing works, in this chapter, we are aiming to monitor the power quality (voltage, frequency and phase angle) of power grid instead of power consumption in the power grid, which is more important for the stability of the microgrid. Our experiment results show that we can achieve similar performance compared to PMUs by reducing the cost with three magnitudes.

Sensing Deployment. Based on different application, there are massive work on how to minimize the deployment effort of sensor networks [61] . A sensor sampling methodology is proposed for better trade-offs between sampling rate and energy consumption [88]. The energy consumption of data transmission over WiFi is investigated in [119]. An external hardware based clock tuning circuit is used to improve synchronization and reduce clock drift [21].

3.6 Conclusion

To achieve the stability of the microgrid, power quality through the power lines needs to be monitored for balancing demand and generation. However, the existing approaches (PMU) are very expensive. To address these issues, we design an accurate real-time energy data sensing hardware to sense the voltage, frequency and phase angle in each home. Through extensive experiments and simulations, we show that our design realized an accuracy of 1.7 mHz and 0.01 rad for frequency and phase angle monitoring, respectively.

CHAPTER 4

DATA MANAGEMENT

Nowadays, utility companies are increasing the installation of smart meters in individual homes. Smart meters can record energy related data (e.g., power consumption) every second. However, power consumption data with high time granularity needs huge data storage space and generates significant communication overhead for utility companies. In this chapter, we present E-Sketch, a middleware for utility companies to gather data from smart meters with much less storage and communication overhead.

4.1 Introduction

Peak demand of power grids is the main concern for utility companies because it determines how much power utility companies need to generate. Based on the government research report [87], most of the generation and distribution infrastructures are constructed to handle some extremely rare peak demands. In 2010, Energex, a distribution network in Queensland, used 13% of its $8.8 billion infrastructures for only 100 hours of the year [39]. To reduce peak demand, many utility companies intend to introduce real-time pricing plans to encourage homes to reduce high peak demand. Presently, utility companies monitor aggregated peak demand with high time granularity (e.g., every second) [5]. However, at the individual home level, traditional meters at each home monitor energy consumption every hour. Yet, in one hour, the power grid may need notably more power than average hourly power consumption for several minutes. Thus, utility companies are transitioning from hourly pricing plans to real-time pricing plans for better regulation of peak demand. To apply real-time pricing plans, utility companies need to collect power consumption of individual homes at high time granularity.

To collect accurate power consumption information at individual homes, smart meters have been rolled out in many countries [117]. Smart meters deployed at homes usually record power consumption and other energy data every second. However, it requires huge data storage for utility companies to gather power consumption readings every second from all the homes. For example, in our experiments, 127.1TB is needed to store all second-level raw data in one day for homes in New York State (detailed discussed in §4.2.2). Thus, it is critical to design an architecture to gather data from all the homes with less storage and communication overhead. Besides, power consumption data at different periods of the day is not equally valuable. For example, most appliances run at stable states with relative stable energy consumption when people are not at home. Then the power consumption data at those periods is not as valuable as power consumption when you frequently turn on/off appliances at home. A possible solution for reduction of data storage is to only record appliances' usage to recover power consumption every second. However, it requires servers to know the consumption pattern of all appliances at each home for recovery of power consumption data every second. Moreover, appliances may not consume stable power all the time, which makes it difficult to recover accurate power consumption for pricing plans only with appliance usage data.

In this chapter, we study the power consumption patterns of homes to compress second-level power consumption data. A middleware E-Sketch is proposed to work between smart meters at individual homes and central servers in the utility company to reduce data storage of power consumption data. In order to reduce communication overhead, E-Sketch is executed at each local smart meter. However, smart meters have limited storage and computing power, therefore the design should be simple and fast to reduce computation overhead at smart meters. Our design consists of three parts: i) adaptive sampling in time domain; ii) data compression in frequency domain; iii) encoding and decoding. We also show how compressed data can be used to recover power consumption in the central server and analyze the performance of our design. The main contributions of this chapter are as follows:

- To our best knowledge, this is the first work to investigate the power consumption at different time granularity. We study the necessity to record power consumption with high time granularity and investigate the relationship between high and low time granularity power consumption data.

- With the analysis of second-level power consumption data, we propose **E-Sketch** to compress the data in both time domain and frequency domain to reduce data storage and communication overhead. Our design can guarantee that power consumption error is always under the given error bound.

- To validate our design, we evaluate our work extensively with empirical traces of 30 homes' second-level power consumption for more than 2 months. The results show that our design reduces data storage significantly by 90% with more than 99% accuracy of second-level power consumption on average for a single home. For aggregated power consumption of 30 homes, our design can achieve even more than 99.8% accuracy on average by canceling errors from different homes.

4.2 Motivation

In this section, we firstly give an example from our empirical data to show the importance of high time granularity data at a single home and a small community; then we show the problem of gathering and utilizing data with high time granularity.

4.2.1 The Need for High Granularity Power Monitoring

In this section, we show that in a single hour there may be several minutes where a home consumes much more power than it does during the rest of given hour. That means hourly power consumption misses a lot of vital information on how homes consume energy. The missing information can be utilized for applying real-time pricing plans. Also, different substations have different capabilities, overlapping of peak demand from individual homes may cause blackouts at some substations within several seconds. Thus, it is important to investigate the relationship of second-level and hourly average power consumption. The second-level and hourly average power consumption of two different homes and aggregated power consumption of 30 homes are shown in Figure 4.1. For hourly average power consumption, it is defined as the average second-level power consumption within one hour.

• **Observation 1:** *Different homes with similar hourly power consumption may have sig-*

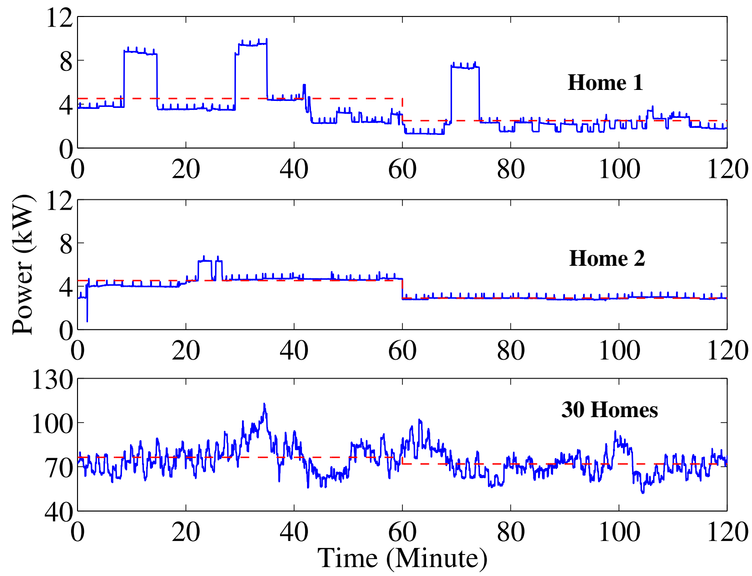


Figure 4.1: Second-level and hourly average power consumption of home 1, 2 and aggregated data of 30 homes in two hours (blue solid line is hourly average power consumption; red dashed line is second-level power consumption)

nificantly different second-level power consumption. As shown in Figure 4.1, home 1 and home 2 have similar hourly power consumption for two hours. During some periods, second-level power consumption is much larger than hourly average power consumption. However, the detailed consumption for every second of different homes is significantly different. In Figure 4.1, home 1 has peak demand around $9kW$ for 20 minutes in the first hour and another peak demand around $8kW$ for 10 minutes in the next hour. At the mean time, home 2 only has peak demand around $6kW$ in the first hour and almost keeps stable consumption of $3kW$ in the next hour. The detailed distribution of power differences between consumption per second and hour for two homes is shown in Figure 4.2. For home 1, 10% period in an hour (6 minutes), second-level power consumption is $4.2kW$ larger than hourly average power consumption; around 30 seconds in an hour, second-level power consumption is $5kW$ larger than hourly average consumption. For home 2, 10% period in an hour (6 minutes), second-level power consumption is $0.2kW$ larger than hourly average

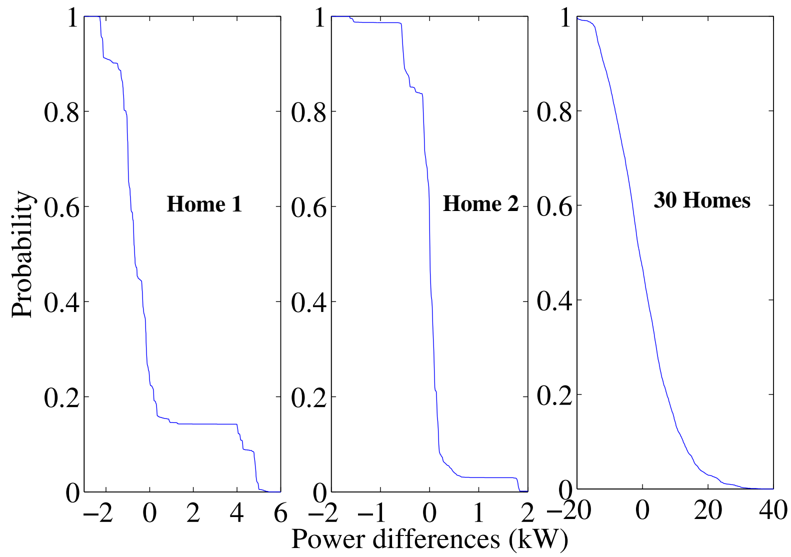


Figure 4.2: Distribution of power differences between second-level and hourly consumption for a single home and 30 homes in two hours

consumption; around 30 seconds in an hour, second-level power consumption is $1.8kW$ larger than hourly average consumption. From perspective of utility company, home 2 consumes energy more reasonable than home 1. Because even they have similar hourly power consumption, home 2 has much lower second-level peak demand and should be charged less than home 1.

• **Observation 2:** *Multiple homes can generate very high peak demand for minutes while hourly average power consumption is low.* We also show aggregated power consumption for 30 homes in Figures 4.1. Second-level power consumption still shows peak demand compared hourly average power consumption. In the first hour, from 30 to 35 minutes, it consumes more than $90kW$ while hourly power consumption is only $75kW$. For distribution of power differences between second-level and hourly average consumption for 30 homes (shown in Figure 4.2), 10% period in an hour (6 minutes), second-level power consumption is $12.4kW$ larger than hourly average consumption; around 30 seconds of an hour, second-level power consumption is $28.0kW$ larger than hourly average consumption. Thus, compared to a single home, hourly power consumption for a small community (e.g., substation) also misses detailed second-level peak demand. Moreover, peak demand of homes may overlap and create larger peak demand of $28.0kW$. Thus, it is very important to monitor second-level power consumption at individual homes for utility company designing real-time pricing plans.

4.2.2 Large Volume of Energy Data

In $\S$ 4.2.1, we showed the necessity of recording second-level power consumption at individual homes. However, second-level data requires huge data storage and causes communication overhead. In our simulations, for one single home, 16.4MB is needed to store all second-level data related to energy for one day. The data includes date, total power consumption, power consumption of two phases, voltage of two phases and frequency of two phases. Considering the number of housing units in New York State (8,123,051 in 2012 [110]), 127.1TB is needed to store each day's power consumption data. The amount of historical data that stores for a long period for backup or study of the power grid will increase with time, which becomes a huge cost for utility companies. Furthermore, to utilize the second-level power consumption data for real-time pricing plans, transmission of such large amount of data from smart meters to the respective substations also generates huge communication overhead. Thus, it is important to design a middleware for utility companies to gather data from all the homes with less data storage and communication overhead. Besides, the computing resources at the smart meter are limited, thus the design for gathering data at individual homes should be simple and efficient.

4.3 Problem Formulation

In this section, we give an overview of E-Sketch, describe how power consumption is collected and present our design goal.

4.3.1 Overview of E-Sketch

The real-time data collected at the smart meters needs to be sent to utility company for billing calculation. However, we already show in § 4.2.2 that the amount of real-time data is huge, thus it is not possible for smart meters to send out the raw real-time data to the utility company. Thus, in this chapter, we propose a middleware ***E-Sketch*** to work between the smart meters and the central server in the utility company (shown in Figure 4.3). Each home collects raw real-time power consumption data and then runs E-Sketch middleware to reduce data storage for every window size N, then the central server in the utility company decompressed the data from smart meters for billing calculation. E-Sketch includes three components: i) adaptive sampling that only samples power consumption change that is larger than certain threshold (detailed in § 4.4.1); ii) frequency compression that compresses adaptive sampled data in frequency domain (detailed in § 4.4.2); and iii) encoding that encodes compressed data (detailed in § 4.4.3). The compressed data is then sent to central server for every window.

In the central server, when it receives compressed data from smart meters, it first stores the data in the database directly for persistent storage. Then for every time interval (e.g., day or month), the utility company can recover the compressed data in the database to calculate billing details for consumers. The recovery of the compressed data is the inverse

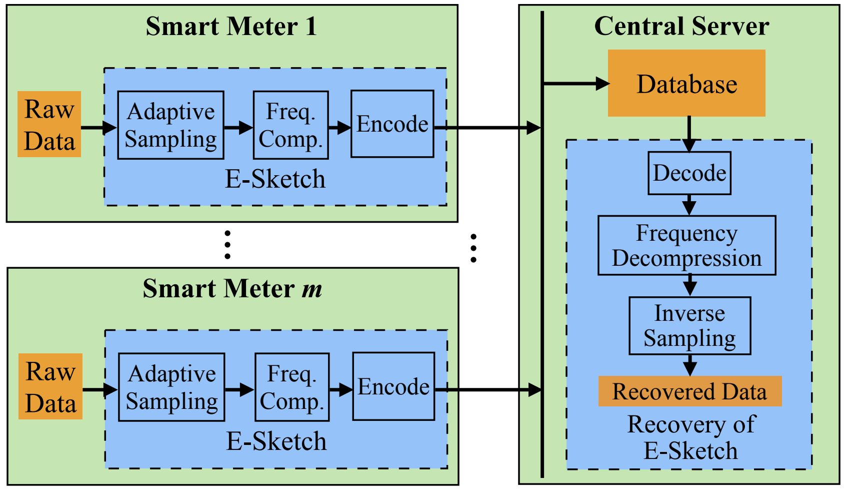


Figure 4.3: Overview of system architecture

process of E-Sketch. It first decodes the compressed data, utilizes the frequency data to conduct frequency decompression and obtains data in time domain. Then inverse sampling is processed to get recovered data. The detailed algorithm for recovery of E-Sketch is discussed in § 4.4.4.

4.3.2 Design Goal

To record power consumption with high time granularity with less storage, we propose middleware E-Sketch to work between smart meters and central server in the utility companies. Because E-Sketch needs to be run at local smart meters, the computation and implementation complexity of E-Sketch should be low due to limited computation and energy resource in smart meters.To achieve these design goals, we formulate the problem as follows:

Let $\mathbf{P} = \{p(1), \cdots, p(n)\}$ be the original power consumption series. Given the specific boundaries of $e(t)$ ($|e(t)| \leq \theta$), our design goal is to minimize the communication overhead and data storage for storing power consumption data every second in the central server. Assume $\mathbf{Q} = \{q(1), \cdots, q(m)\}$ be the data stored in the central server and $\mathbf{R} =$

Notations	Definitions
N	Window size of E-Sketch
$p(t)$	Original power consumption of home at t
$\tilde{p}(t)$	Sampled power consumption of home at t
$d(t)$	Power change from t to $t+1$
$d_c(t)$	Compressed high power change from t to $t+1$
$\tilde{d}(t)$	Sampled power change from t to $t+1$
I_c	Index of compressed high power change
$\tilde{I}$	Index of sampled power change
$f(k)$	Frequency value of $\tilde{d}(t)$
$f_q(k)$	Quantization results of $f(k)$
$e_{\tilde{d}}(t)$	Error caused by quantization in time domain
$e_f(k)$	Error caused by quantization in frequency domain
$r(t)$	Recovered power consumption from E-Sketch
θ	Desired error bound
$e(t)$	Power difference between $p(t)$ and $r(t)$

Table 4.1: Notations for Energy Data Collection

$\{r(1), \cdots, r(n)\}$ be the recovered power consumption series in the central server. We can formulate our problem as

$$
\begin{aligned}
\min \quad & H(\mathbf{Q}) \\
\textit{s.t.} \quad & g_1 : \mathbf{P} \rightarrow \mathbf{Q} && (a) \\
& g_2 : \mathbf{Q} \rightarrow \mathbf{R} && (b) \\
& e(t) = |p(t) - r(t)| \leq \theta, t \in [1, n] && (c)
\end{aligned}
$$

g_1 is the function that maps from **P** to **Q** and g_2 is the function that maps from **Q** to **R**. $H(\mathbf{Q})$ is the information content of **Q**. Then our goal is to find the functions g_1 and g_2 to minimize the information content of stored data **Q**. And our proposed design E-Sketch can be considered as g_1 and recovery algorithm of Sketch can be considered as g_2. Because g_1 is run in smart meters, thus the complexity of g_1 should be relative low. Note that $m = n$ is not required, which means the data points stored in the central server and data points of original power consumption may not be equal.

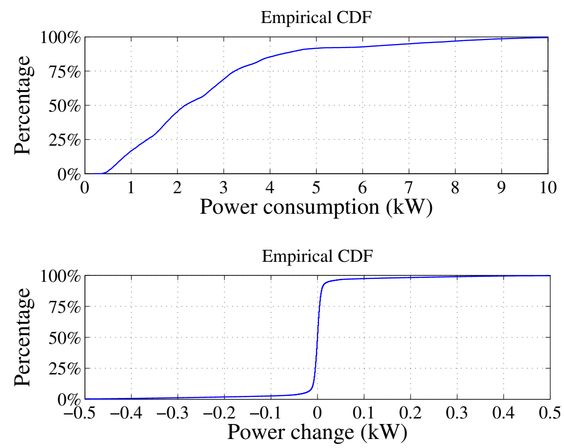


Figure 4.4: Empirical CDF of power consumption and power change

4.4 System Design

In this section, we introduce the main design of E-Sketch. Our design consists of three parts: i) adaptive sampling in time domain; ii) data compression in frequency domain; iii) encoding and decoding. We also show how compressed data can be recovered and analyze the performance of our design.

4.4.1 Adaptive Sampling

In this section, we introduce adaptive sampling to reduce data storage while keeping the valuable data of power consumption. To design adaptive sampling, we first analyze the distribution of power consumption. The empirical Cumulative Distribution Function (CDF) of power consumption $p(t)$ and power change $d(t)$ in a window (window size N for the experiment is 2 hours) is shown in Figure 4.4. And we have:

$$d(t) = p(t+1) - p(t) \tag{4.1}$$

In Figure 4.4, though power consumption varies from $1kW$ to $10kW$, power change is mostly very small. For example, 90% of power change is less than $0.05kW$, which provides great opportunity to reduce storage space by only storing the power consumption change that is valuable. We divide $d(t)$ into $\tilde{d}(t)$ and $\upsilon(t)$. When power consumption change at t is valuable, we have $\tilde{d}(t) = d(t)$; when power consumption change at t is not valuable, we have $\tilde{d}(t) = 0$ to save the storage. $\upsilon(t)$ is 0 when $\tilde{d}(t) = d(t)$, $\upsilon(t) = d(t)$ when $\tilde{d}(t) = 0$. Then we have

$$\mathbf{D} = \tilde{\mathbf{D}} + \mathbf{\Upsilon} \tag{4.2}$$

D, $\tilde{D}$ and Υ are vectors of $d(t)$, $\tilde{d(t)}$ and $\upsilon(t)$. If the data of $\tilde{D}$ is stored, then we calculate $\tilde{p}(t)$ with $\tilde{p}(t) = \tilde{p}(t-1) + \tilde{d}(t-1)$. Then at t, the error of power consumption data can be calculated as

$$p(t) - \tilde{p}(t) = \sum_{i=1}^{i=t-1} d(i) - \sum_{i=1}^{i=t-1} \tilde{d}(i) = \sum_{i=1}^{i=t-1} \upsilon(i) \tag{4.3}$$

Assume that $\upsilon(t)$ is a random variable with mean of zero and variance of σ^2. Then the error of power consumption data at t will be a random variable with mean of zero and variance of $s_\upsilon(t-1) \cdot \sigma^2$. $s_\upsilon(t-1)$ is the number of $\upsilon(t) > 0$ for $t \in [1, t-1]$. With the increase of time, the variance of power consumption error increases, which means the error $p(t) - \tilde{p}(t)$ can be very high when t is high. To overcome this problem, we present an adaptive sampling algorithm shown in Algorithm 1.

The basic idea of adaptive sampling is to keep track of power consumption when we decide whether power consumption change at t is valuable or not. Given the error

Algorithm 1: Adaptive Sampling Algorithm

Input: Power consumption data $p(t)$ and desired error bound θ
Output: Compressed power consumption data.

1: $err = 0, count = 0;$
2: **for** t = 1 to N **do**
3: $\quad d(t) = p(t+1) - p(t);$
4: $\quad$ **if** $|err + d(t)| < \theta$ **then**
5: $\qquad err = err + d(t);$
6: $\quad$ **else**
7: $\qquad count + +;$
8: $\qquad d_c(count) = d(t) + err$, $I_c(count) = t;$
9: $\qquad err = 0;$
10: $\quad$ **end if**
11: **end for**

bound θ, we iteratively check the power consumption error at $t+1$ based on $d(t)$ and power consumption error at t if ignoring power consumption change $d(t)$ (Lines 1-3). If power consumption error at $t+1$ is less the error bound θ, then we can ignore the $d(t)$ and keep track of power consumption error at $t+1$ (Lines 4-5). Otherwise, we rewrite $\tilde{d}(t) = err + d(t)$ and power consumption error at $t+1$ will be zero (Lines 6-11). Let t_i be the ith time that $\tilde{d}(t) = err + d(t)$ is rewritten. Because power consumption error is fixed by $\tilde{d}(t)$, we always have $p(t_i + 1) = \tilde{(p)}(t_i + 1)$. Then the maximum error we get is at $t = t_1, \cdots, t_i, \cdots$. The power consumption error of $\tilde{p}(t_i)$ can be calculated as

$$p(t_i) - \tilde{p}(t_i) = \sum_{j=t_{i-1}}^{j=t_i} \upsilon(j), \qquad i = 1, 2, \cdots \tag{4.4}$$

We know the power consumption error is always less than bound θ. We can calculate the mean of error is zero and the variance of error is $E[t_{i+1} - t_i] \cdot \sigma^2$. $E[t_{i+1} - t_i]$ is the expected mean of $t_{i+1} - t_i$. Based on the probability theory, we have

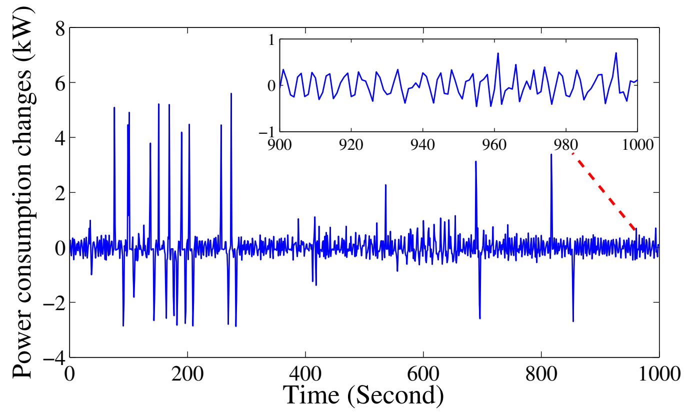


Figure 4.5: Results of adaptive sampling

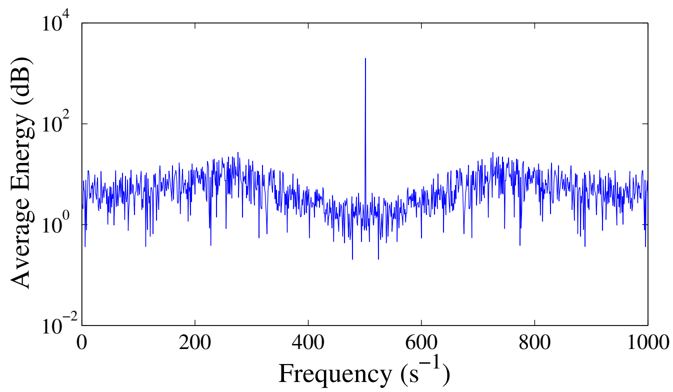


Figure 4.6: Spectrum analysis of sampled data over 30 days

$$E[t_{i+1} - t_i] = P_1 + \sum_{j=2}^{\infty} j * P_j \prod_{k=1}^{k=j-1} (1 - P_k) \tag{4.5}$$

P_i is the probability that the sum of i $v(t)$ is larger than θ. With the probability distribution of $v(t)$, we can calculate P_i based on probability theory. We will not do the detailed calculation since the result is too complicated. But it can be proved that $E[t_{i+1} - t_i] < \frac{\alpha}{(1-\alpha)^2}$ and α is a constant determined by probability distribution of $v(t)$.

4.4.2 Data Compression in Frequency Domain

With adaptive sampling, we can remove most of ignorable power consumption changes. And an example of power consumption changes after adaptive sampling is shown in Figure 4.5. Most of the power consumption changes are still quite small (less than $0.5kW$) and few of the power consumption changes are more than $4kW$. The detailed power consumption changes from time 900 to 1000 shown in zoom-in figure is frequently fluctuated. This is because that power consumption increase caused by turning on appliances will be followed by power consumption decrease caused by turning off appliances. The highly fluctuated signal is better analyzed in the frequency domain but not time domain. We use Discrete Fourier Transform (DFT) to transfer sampled data from time domain to frequency domain. The discrete time signal $\tilde{d}(t)$ can be equivalently represented by its DFT:

$$f(k) = \sum_{t=1}^{N} \tilde{d}(t) \cdot e^{-i2\pi kt/N} \tag{4.6}$$

For example, Figure 4.6 shows the frequency spectrum measured over 30 days of power consumption change after adaptive sampling. It is clear that most of the "energy" in the signal is stored in low frequency components. The mechanism to keep the signals in the time domain is through filtering and downsampling. To compress the data from frequency domain, we use Discrete Cosine Transform (DCT) instead of DFT, which outputs real number and is more suitable for data compression. The formula of DCT is as follow:

$$f(k) = \sqrt{\frac{1}{N}}\tilde{d}(1) + \sqrt{\frac{2}{N}} \sum_{t=2}^{N} \tilde{d}(t) \cdot cos\left[\frac{\pi}{N}(t+\frac{1}{2})k\right] \tag{4.7}$$

To simplify the calculation, we revise the parameters of standard DCT so that DCT matrix is real unitary transform. The elements of the DCT matrix $\mathbf{H} = \{H\,[k,t]\}$ are

$$H[k,t] = \begin{cases} \sqrt{\frac{1}{N}}, & t = 1, k \in [1,N] \\ \sqrt{\frac{2}{N}} cos\left[\frac{\pi}{N}(t+\frac{1}{2})k\right], & t \in [2,N], k \in [1,N] \end{cases} \tag{4.8}$$

Let $\mathbf{F}$ be $\{f(1),\cdots,f(N)\}$, then we have $\mathbf{F} = \mathbf{H}\tilde{\mathbf{D}}$. Since the DCT is a real unitary transform, $\mathbf{H}^{-1} = \mathbf{H}^T$ and the inverse DCT (IDCT) is described by $\tilde{\mathbf{D}} = \mathbf{H}^T\mathbf{F}$.

When compressing a signal, the DCT coefficients are typically quantized rather than the actual signal $\tilde{\mathbf{D}}$. The quantized DCT coefficients are denoted as $\mathbf{F}_q$, with $f_q(k) = Q[f(k)]$, where $Q[\cdot]$ is the quantization operator. Quantization is a non-linear operation that results in a loss of information; only scalar quantization is considered here, where each element of $f(k)$ is quantized individually. Scalar quantization is a many-to-one mapping that transforms intervals of real numbers $[q_i^k, q_{i+1}^k)$ to single real numbers. The superscript "k" accounts for the possibility of different quantization intervals for different frequency coefficients, and the subscript "i" indicates the ith quantization level. Transform coefficients that are in these intervals are typically mapped to the midpoint of the interval, so that $f_q(k) = \frac{1}{2}(q_i^k + q_{i+1}^k)$, for $q_i^k \leq f(k) \leq q_{i+1}^k$.

The recovered signal $\tilde{\mathbf{D}}_q$ is obtained by performing the IDCT on the quantized frequency values, $\tilde{\mathbf{D}}_q = \mathbf{H}^T\mathbf{F}_q$. Two quantities of interest in this chapter are the quantization errors in both the spatial and the frequency domains. Spatial-domain error is represented by $\mathbf{e}_{\tilde{d}} = \tilde{\mathbf{D}}_q - \tilde{\mathbf{D}}$, and frequency-domain error by $\mathbf{e}_f = \mathbf{F}_q - \mathbf{F}$. Note that the quantization error in the spatial domain can be expressed as

$$\mathbf{e}_{\tilde{d}} = \mathbf{H}^T \mathbf{e}_f = \sum_{k=1}^{N} h_k (f_q(k) - f(k)) \tag{4.9}$$

To compress the signal, we need to carefully select q_i^k to maximize the compression ratio while fulfilling the error bound θ. Different from traditional DCT which applies in image data compression, the high frequency component of the signal we need to compress is not ignorable and the error should be within the error bound θ. Thus, we present a frequency compression algorithm to decide quantization results of frequency data $f(k)$ that maximize the compression ratio and fulfill the error bound. The basic idea is to decide quantization results based on transferring quantization errors in frequency domain to time domain. The detailed description is in Algorithm 2. We first calculate $f(k)$ based on our defined DCT Equation (4.7) (Line 1). Then $f(k)$ is sorted in ascending order and initialize $f_q(k)$ and current error bound $b(t)$ (Line 2). For each $f(k)$, we first check when $f(k)$ is ignored whether the error caused by $f(k)$ still less than current error bound $b(t)$. If $|e_{\tilde{d}}(t)| < b(t)$, we can ignore $f(k)$ and set $f_q(k) = 0$ and update $b(t)$ (Lines 4-8). Otherwise, we divide $f(k)$ by λ, and check the error bound again until we find the maximum $f_s(i)$ that fulfill the error bound (Lines 10-15). Then we can update $f_q(k)$ and current error bound $b(t)$ (Lines 16-19).

To analyze the performance of our frequency compression algorithm, we calculate the variance of error after quantization. Because the quantization process is symmetry, the DCT-domain quantization errors are uncorrelated random variables. The covariance matrix $\mathbf{K}_{e_f} = E(\mathbf{F}_q - \mathbf{F})(\mathbf{F}_q - \mathbf{F})^T$ is then diagonal, with its N non-zero elements equal to the

Algorithm 2: Frequency Compression Algorithm

Input: Power consumption changes after adaptive sampling $\tilde{d}(t)$ and desired error bound θ

Output: Quantization results $f_q(k)$ of frequency data $f(k)$ for $k \in [1, N]$.

Calculate $f(k)$ with $\tilde{d}(t)$ based on Equation (4.7);
Sort $f(k)$ in ascending order;
$b(t) = \theta$ for $t \in [1, N]$, $f_q(k) = -1$ for $k \in [1, N]$;
for i = 1 to N **do**
    $e_{\tilde{d}}(t) = \sum_{k=1}^{k=N} h_k(t) * (-f(i))$;
    if $|e_{\tilde{d}}(t)| < b(t)$ for $t \in [1, N]$ **then**
        $f_q(i) = 0$;
        $b(t) = b(t) - e_{\tilde{d}}(t)$;
    else
        $f_s(i) = f(i)$;
        $e_{\tilde{d}}(t) = \sum_{k=1}^{k=N} h_k(t) * (f_s(i)/\lambda)$;
        while $|e_{\tilde{d}}(t)| > b(t)$ for $t \in [1, N]$ **do**
            $f_s(i) = f_s(i)/\lambda$;
            $e_{\tilde{d}}(t) = \sum_{k=1}^{k=N} h_k(t) * (f_s(i)/\lambda)$;
        end while
        $f_q(k) = f_{(}i) - f_s(i)$;
        $b(t) = b(t) - e_{\tilde{d}}(t)$;
    end if
end for

quantization noise of the individual frequency-domain coefficients, $\sigma^2_{e_f}[k]$. With $\mathbf{K}_{e_f}$, we can calculate the covariance of the error in time domain as

$$\mathbf{K}_{e_{\tilde{d}}} = E(\tilde{\mathbf{D}}_q - \tilde{\mathbf{D}})(\tilde{\mathbf{D}}_q - \tilde{\mathbf{D}})^T = \mathbf{H}^T \mathbf{K}_{e_f} \mathbf{H} \tag{4.10}$$

Equation 4.10 includes information about the correlation of the time domain error sequence, but another quantity of interest is the variance of the individual time domain errors. The variance of $e_{\tilde{d}}(t)$ is found as $\sigma^2_{e_{\tilde{d}}}[t] = \mathbf{K}_{e_{\tilde{d}}}[t, t]$, or in summation notation

$$\sigma^2_{e_{\tilde{d}}} = \sum_{k=1}^{N} H^2[k, n] K_{e_f}[k, k] = \sum_{k=1}^{N} H^2[k, n] \sigma^2_{e_f} \tag{4.11}$$

4.4.3 Encoding

After frequency compression, we can further reduce data storage by encoding the compressed data. Because the range of compressed data is much less than the range of original power consumption, the storage space for compressed data can be reduced.

The first step of encoding is to generate the probability distribution of compressed data. Then we can encode compressed data based on their probability to minimize the data storage. The encoding works by creating a binary tree of nodes. These nodes can be stored in a regular array, the size of which depends on the number of symbols. A node can be either a leaf node or an internal node. Initially, all nodes are leaf nodes, which contain the symbol itself, the weight (frequency of appearance) of the symbol and optionally, a link to a parent node which makes it easy to read the code (in reverse) starting from a leaf node. For example, with number of n values $[v_1, \cdots, v_n]$, we can calculate their probability to generate n nodes $[(v_1, pr_1), \cdots, (v_n, pr_n)]$. Internal nodes contain symbol weight, links to two child nodes and the optional link to a parent node. As a common convention, bit '0' represents following the left child and bit '1' represents following the right child. The process essentially begins with the leaf nodes containing the probabilities of the symbol they represent, then a new node whose children are the two nodes with smallest probability is created, such that the new node's probability is equal to the sum of the children's probability Assume (v_1, pr_1) and (v_2, pr_2) are merged to (v_{12}, pr_{12}), then we have $pr_{12} = pr_1 + pr_2$. With the previous two nodes merged into one node (thus not considering them anymore), and with the new node being now considered, the procedure is

Algorithm 3: Recovery of Power Consumption

Input: $p(1)$, N, $\mathbf{I}_c$, $\mathbf{D}_c$,$\tilde{\mathbf{I}}_c$ and $\mathbf{F}_q$
Output: Recovered power consumption $r(t)$

$r(1) = p(1)$;
$index_s = 2$, $index\_e = 2$;
for i = 1 to $|\mathbf{I}_c|$ **do**
  $index\_e = I_c(i)$;
  for t = $index\_s$ to $index\_e$ **do**
    $r(t) = r(t-1) + d_c(i)$;
  end for
end for
for t = 1 to N **do**
  Calculate $\tilde{d}(t)$ with $f_q(k)$ and N based on IDCT;
end for
$index_s = 2$, $index\_e = 2$;
for i = 1 to $|\tilde{\mathbf{I}}_c|$ **do**
  $index\_e = \tilde{I}_c(i)$;
  for t = $index\_s$ to $index\_e$ **do**
    $r(t) = r(t-1) + \tilde{d}(i)$;
  end for
end for

repeated until only one node remains. Finally, based on the constructed tree, we can assign each value a code to be stored.

4.4.4 Recovery of Power Consumption

In previous sections, we present *E-Sketch* to compress power consumption with high time granularity. In this section, we show how the compressed data can be used to recover original power consumption.

Since we encode compressed data, the first step of recovery of original data is to decode by simply translating the stream of prefix codes to individual byte values. Then we have initial power consumption $p(1)$, length of power consumption data N, indexes and values of high power consumption changes $\mathbf{I}_c$ and $\mathbf{D}_c$, and indexes and frequency values of power consumption changes $\tilde{\mathbf{I}}_c$ and $\mathbf{F}_q$. Then we show how power consumption can be recovered with above data in Algorithm 3. First, we can recover the power consumption data with the indexes and values of high power consumption changes (Lines 1-8). Because we store the power consumption changes but not original power consumption, we need to calculate original power consumption based on the compressed data. Then we recover time domain data of power consumption change with quantization results in frequency domain based on IDCT (Lines 9-11). The recovered power consumption changes $\tilde{\mathbf{D}}_q$ can be obtained by performing IDCT on quantized frequency values $f_q(k)$, $\tilde{\mathbf{D}}_q = \mathbf{H}^T\mathbf{F}_q$. With the indexes and time domain values $\tilde{\mathbf{I}}_c$ and $\tilde{\mathbf{D}}$, we can also recover the power consumption data from 1 to N (Lines 12-18). Algorithm 3 can recover the original power consumption from 1 to N. If we only need power consumption of a specific time, we can also calculate the original power consumption as follows:

$$\tilde{p}(t) = p(1) + \sum_{i=1}^{I_c(i)<t} d_c(i) + \sum_{i=1}^{\tilde{I}(i)<t} \sum_{j=1}^{j=N} H[i,j] f_q(j) \tag{4.12}$$

Then we can reduce the computation overhead of central server and obtain original power consumption of a specific time faster.

4.4.5 Time Complexity and Storage Analysis

First, we analyze the complexity and data storage of E-Sketch at local smart meters in the following three stages.

i) *Adaptive sampling*. Basic power consumption curve is sketched in time domain. In adaptive sampling, Algorithm 1 is applied to generate sampled power consumption changes and time complexity of Algorithm 1 is $O(N)$. For window size N, we need to store the high power consumption changes and indexes in this window. Thus the storage cost should be related to the number of high power consumption changes N_h.

ii) *Frequency compression*. Sampled data of power consumption changes are transferred into frequency domain and quantize frequency data. In frequency compression, we need to conduct DCT for $\tilde{d}(t)$ and run Algorithm 2. The time complexity for DCT is $O(NlogN)$ with fast Fourier transform (FFT) and time complexity for Algorithm 2 is $O(N)$. For output of frequency compression, we need to store $f_q(k)$ and $\tilde{I}$, which is related to the number of sampled consumption changes N_s.

iii) *Encoding*. The compressed data in adaptive sampling and frequency compression is encoded to further reduce storage space. Time complexity of encoding is $O(NlogN)$ and encoding introduces no storage cost.

In total, the time complexity of our design is still $O(NlogN)$, which means our design is simple for smart meters. And the total data storage cost will be related to N_h and N_s.

Even after data compression, the amount of data in the central server is still a huge number because the number of homes is large. Thus, we also analyze the time complexity of decompression in the central server. The decompression algorithm includes IDCT calculation and two loops with complexity of $O(N)$. Thus the complexity of decompression is also $O(NlogN)$.

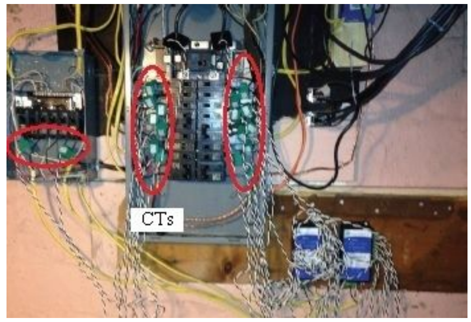


Figure 4.7: eGauge deployment in a home

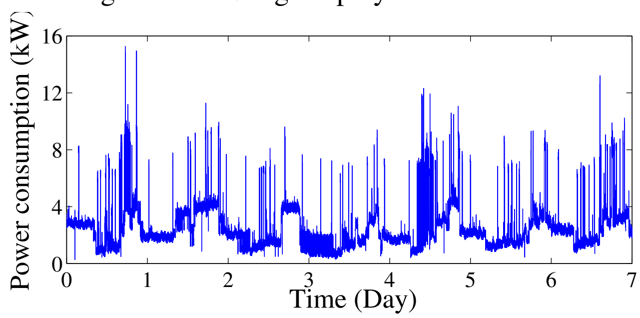


Figure 4.8: Power consumption of a home in seven days

4.5 Implementation and Evaluation

In this section, we evaluate the performance of our proposed design. We deploy eGauge power meters at individual homes to collect the energy consumption related data (e.g., power, voltage, frequency, etc.) every second ([38]). One of the experiment setup is shown in Figure 4.7. We add current transducers (CTs) around each leg of the home's split-phase power input from the grid to monitor all the circuits inside the home every second. In our simulation, we use the power consumption traces that we collected from 30 homes in eGauge website for two months. To make the figure easy to follow, we only show the power consumption of a home for seven days in Figure 4.8. In a day, the power consumption is mostly in the afternoon and evening. And the power consumption for different days varies significantly.

4.5.1 Evaluation Baseline and Metrics

Baseline. To verify the efficiency of our approach, we compare our design with three approaches: i) **Sparse Data**, which utilizes lower time granularity power consumption to estimate high time granularity power consumption; ii) **Polynomial Fitting**, which applies polynomial fitting to estimate the power consumption; iii) **Zip**. We realize deflate algorithm, which is most commonly used compression method for zip files ([97]).

Metrics. We use two metrics to evaluate the performance of our approach: i) **storage space**: the amount of storage space used to store data; ii) **average error of power consumption**: average power differences between original and recovered data.

4.5.2 Basic Evaluation Results

We evaluate the effectiveness of our proposed E-Sketch compression algorithm, which includes the storage space, error of power consumption for a single home and a community of 30 homes. All results are simulated with the two months empirical data of energy consumption. And the selection of parameters are: i) error bound $\theta = 0.05$; ii) window size $N = 3600s$. The impacts of these parameters are investigated in latter sections.

Error of power consumption for a single home. The errors of two approaches over time are shown in Figure 4.9. Our proposed E-Sketch can recover data with at most $0.05kW$ power error. The average power error of E-Sketch is $0.02kW$. As shown in Figure 4.1, the home consumes more than $3kW$ power at most time; thus E-Sketch recovers data with more than 99% accuracy. We also show the power errors of sparse data and polynomial fitting. We use average power consumption data for every 5 seconds to recover power

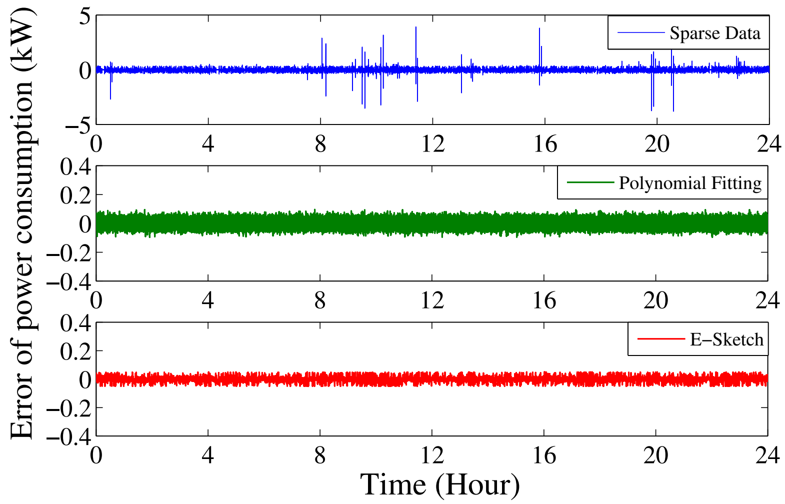


Figure 4.9: Error of power consumption of different approaches compared to original power of a single home

consumption for every second. The average errors of power consumption for sparse data and polynomial fitting is also $0.02kW$. However, at some time, the power errors of sparse data and original data is more than $2kW$. And for polynomial fitting, the maximum error is around $0.1kW$, which is much better than sparse data but worse than E-Sketch. Because Zip is lossless compression, there would be no error of power consumption.

Error of power consumption for 30 homes. The high time granularity power consumption data of each home is sent to the utility companies for calculating the price of electricity. To further verify the performance of E-Sketch, we apply E-Sketch for power consumption data in 30 homes. The power errors of aggregated power consumption for 30 homes are shown in Figure 4.10. Compared to power errors for a single home, the average power error for 30 homes increases from $0.02kW$ to $0.1kW$. While for sparse data, average power error becomes $0.52kW$. The average power error for polynomial fitting is close to E-Sketch, however, the standard deviation (STD) of polynomial is much higher than E-Sketch. And maximum error for polynomial fitting can reach $2kW$ while maximum error for E-Sketch

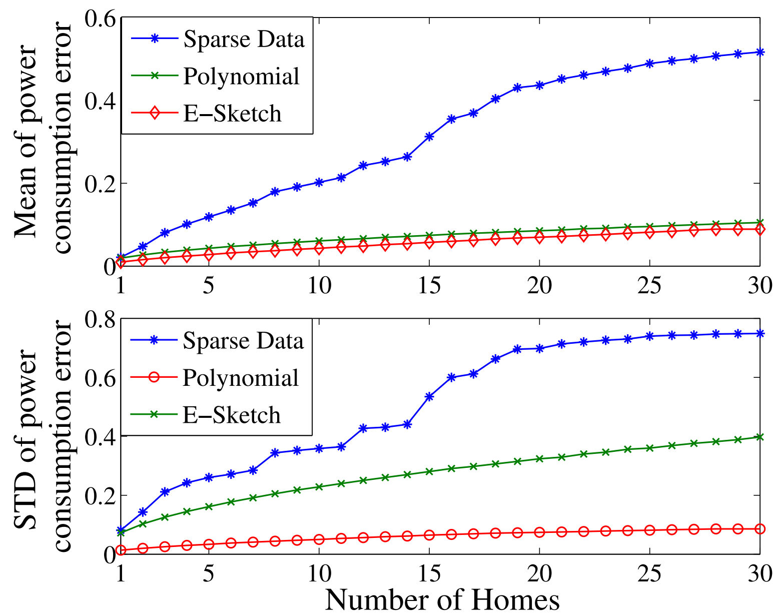


Figure 4.10: Error of power consumption of different approaches compared to original aggregated power of 30 homes

is only $0.3kW$. The reason that E-Sketch performs better than sparse data is E-Sketch randomizes the errors, then power errors of different homes can be cancelled.

4.5.3 Advanced Evaluation Results

In the basic evaluation results, we show E-Sketch works well for a single home and 30 homes. In this section, we investigate the impact of different parameters in E-Sketch to verify the robustness of our design.

Impact of error bound θ. The results of tradeoff of θ between storage and accuracy are shown in Figure 4.12. In this simulation, the setting of window size N is 1 hour. When θ increases, the storage space increases linearly and average power error decreases slowly. With higher error bound θ, adaptive sampling can sample less data and frequency compression can take advantage of more aggressive quantization, thus the data storage space decreases significantly.

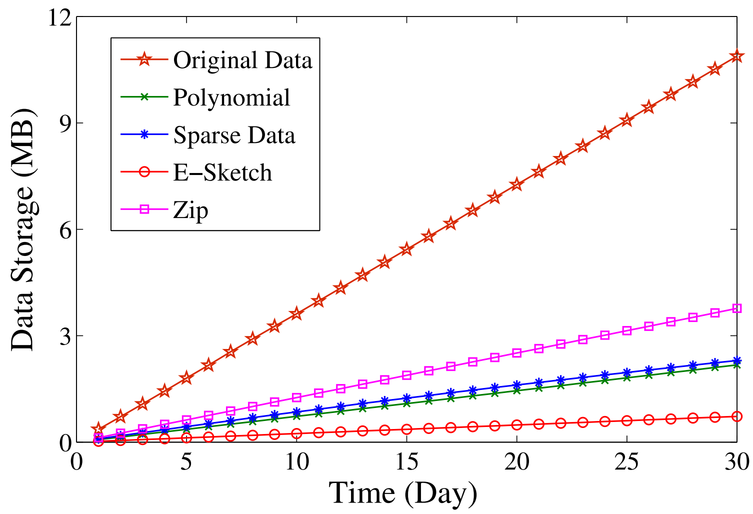


Figure 4.11: Storage space of different approaches

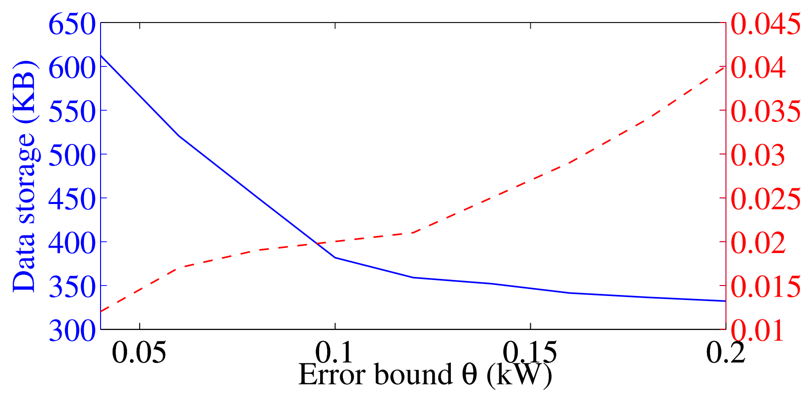


Figure 4.12: Impact of error bound θ

Impact of window size N. We show the impact of window size N in Figure 4.13. In this simulation, the setting of θ is $0.05kW$. With shorter period of window size N, the adaptive sampling samples less data for a window. However, it makes frequency compression harder to compress the sampled data, which introduces more data storage. Thus, data storage decreases with larger window size. For the average power error, the adaptive sampling and frequency compression still works to keep errors under error bound, thus average power error almost stay the same.

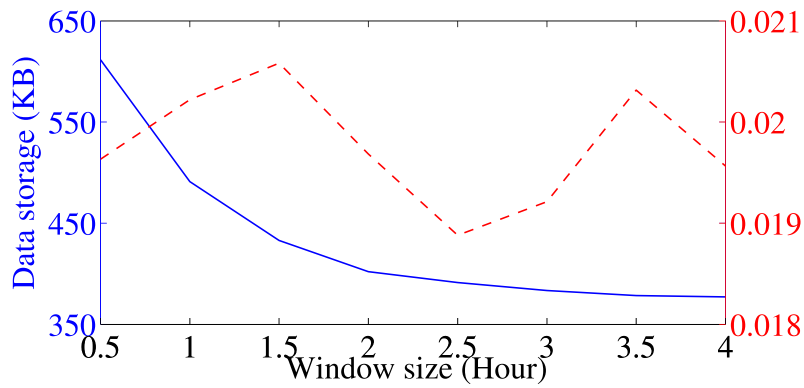


Figure 4.13: Impact of window size N

4.6 Conclusion

In this chapter, we present E-Sketch to save the space and keep high accuracy of energy data. In order to reduce communication overhead, E-Sketch is executed at each local smart meter. However, smart meters have limited storage and computing power, therefore the design should be simple and fast to reduce computation overhead at smart meter. Our design consists of three parts: i) adaptive sampling in time domain; ii) data compression in frequency domain; iii) encoding and decoding. We also illustrate how compressed data can be used to recover power consumption in the central server and analyze the performance of our design.

We conducted extensive system evaluations with 30 homes' second-level power consumption data for more than 2 months. Results indicate i) our design can reduce data storage space significantly by 90% with more than 99% accuracy of second-level power consumption on average for a single home, and ii) our design can achieve even more than 99.8% accuracy on average for aggregated power consumption of 30 homes.

CHAPTER 5

DEMAND FORECAST IN MICROGRID

Accurate energy demand prediction is very important for smart grids to conduct demand response and stabilize the grids. In previous work, many prediction algorithms are proposed to improve the energy consumption prediction accuracy based on the aggregated energy consumption in the whole grid. Recently, with the increasing installations of smart meters in individual homes, high granularity (e.g., per minute) energy consumption data in individual homes becomes available and provides us a great opportunity for better energy consumption prediction. In this chapter, we propose M-Pred to utilize the high granularity energy consumption data collected by smart meters in individual homes for better energy consumption prediction in smart grids.

5.1 Introduction

Compared to traditional power grid, smart grids i) are expected to be robust against grid disturbance or outage; ii) can use more environmental friendly renewable resources; and iii) can utilize the rich information from homes for better energy management. However, to better utilize energy generation in smart grids, one main challenge is how to accurately predict the energy demand (i.e., consumption). This is because generators in smart grids need to generate enough power for energy usage to avoid power outage. To improve the accuracy of energy demand prediction, some prediction algorithms [11, 45] have been proposed based on the aggregated energy consumption in the whole grid. One major limitation of these approaches is that they are predicting energy consumption for the next day or a even longer time period. To achieve faster demand response in smart grids, energy demand prediction for the immediate near future (e.g., the next hour) is more desirable. Another limitation of the existing approaches is that they only predict the hourly average energy consumption. However, energy consumption over an hour can change dynamically. If generators only generate the energy based on the average energy consumption, it is highly possible that the peak power demand will be higher than the power generated in smart grids. Thus, it is very important to predict not only average hourly energy consumption but also the peak demand within an hour. To address these limitations, we propose to utilize both the energy consumption in individual homes and aggregated energy consumption in smart grids for more accurate energy consumption prediction. With the increasing installations of smart meters in individual homes nowadays, high granularity (e.g., per minute) energy consumption data in individual homes becomes available [117]. Because high granularity energy consumption in individual homes provide more information of consumption patterns, high granularity energy consumption data from some of the individual homes with smart meters provides us a great opportunity for better energy consumption prediction.

To utilize the high granularity energy consumption data from some of individual homes, there are three big data related challenges: i) the huge amount of high granularity energy consumption data collected from homes introduces the severe data storage issue;

ii) the energy consumption prediction operation should have low computation overhead for faster demand response; and iii) not every home in the power grid has a smart meter for monitoring high granularity energy consumption data, thus we need to predict energy consumption and peak demands based on different granularities of energy consumption data from different homes. To address these three challenges, we propose M-Pred for accurate energy consumption and peak demand prediction in smart grids. In M-Pred, to reduce the large amount of data storage, we propose to store the energy consumption patterns instead of the whole energy consumption data. Thus, a learning algorithm is proposed to learn energy consumption patterns of individual homes from their energy consumption data. To reduce the computation complexity and communication overhead for energy consumption prediction, we design a distributed energy consumption prediction algorithm to utilize the consumption patterns of homes. Furthermore, considering not every home in power grids has a smart meter, we propose a matching algorithm to cope with different granularity of the collected energy consumption patterns from individual homes and aggregated energy consumption in the grids. Then the matching results can be applied to utilize the energy consumption patterns collected from homes that have smart meters. The main contributions of this chapter are as follows:

• To the best of our knowledge, this is the first work to utilize the spatial and temporal features of the detailed power consumption in individual homes for more accurate power consumption prediction in smart grids.

• With the analysis of massive minute-level power consumption data, we propose **M-Pred** to learn energy consumption patterns of individual homes from their energy consumption data and then utilize these patterns to predict the power consumption in smart grids. Considering that not every home in smart grids has a smart meter, we also propose a matching and prediction algorithm for accurate energy consumption prediction in such smart grids.

• To validate our design, we evaluate our work extensively with more than 12 months' empirical minute-level power consumption data from 726 homes'. The evaluation results show that our design can provide accurate energy consumption prediction for the next hour with Mean Absolute Percentage Error (MAPE) of 2.12%.

5.2 Motivation

In this section, we first explain that why peak demand forecast is needed compared to hourly average power consumption in smart grids. Then we give an example from our empirical data to demonstrate that the power consumption patterns of high time granularity data at individual homes can be beneficial for better prediction. Finally, we summarize the opportunities and challenges of utilizing high time granularity energy consumption data in individual homes.

5.2.1 The Need for Peak Demand Forecast

Peak demand is crucial in real-time demand response applications in smart grids. In this section, we investigate the relationship between hourly average power consumption and peak demand during the hour. We collect the minute-level power consumption data from 726 homes within a city. For hourly average power consumption, it is defined as the average minute-level power consumption within one hour and peak demand is defined as the maximum power consumption in each hour. The relationship between hourly average power

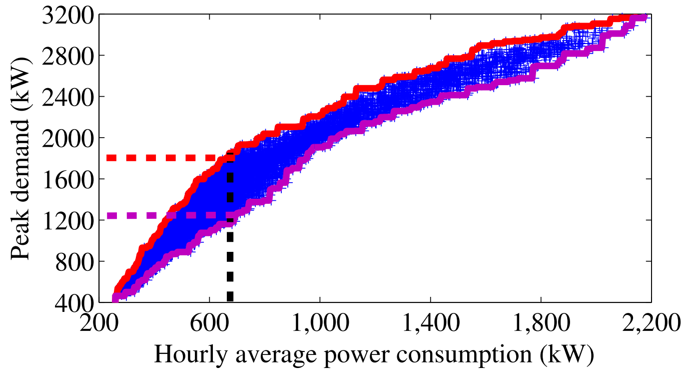


Figure 5.1: Average VS peak hourly energy consumption

consumption and peak demand of 726 homes is shown in Figure 5.1. For the same hourly power consumption in these homes, the peak demand during the hour can be quite diverse. For example, when hourly average power consumption is around 672kW, the difference between peak demand and hourly average power consumption varies from 1.16MW to 1.85MW, which is almost 700kW. Furthermore, with the higher hourly average power consumption, the diversity of the peak demand is more significant. This indicates hourly power consumption misses a lot of vital information on how homes consume energy. The missing information can be crucial for energy generation control and scheduling in smart grids. For example, if we only predict the hourly power consumption in future, the generators may generate either too much energy or not enough energy to cause power outage. Therefore, it is important to not only predict the hourly average power consumption but also peak demand within an hour.

5.2.2 The Need for Power Consumption in Individual Homes

One main limitation of existing prediction approaches is that they only utilize the aggregated power consumption in smart grids to predict hourly power consumption in short-term. Unfortunately, because not every home in smart grids has the same consumption pattern,

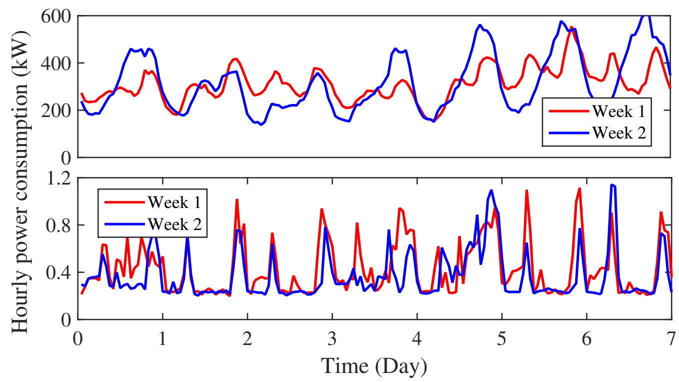


Figure 5.2: Hourly power consumption over two weeks (the top figure shows the aggregated power consumption in the smart grid, and the bottom figure shows the power consumption data from a single home)

the aggregated power consumption in smart grids does not show strong correlation over time. The aggregated hourly average power consumption in a smart grid over two weeks is shown in top figure of Figure 5.2. We can find that the power consumption patterns in each day are quite different. Besides, the power consumption pattern in two consecutive weeks are also different. Therefore, if we utilize previous day's power consumption to forecast current day's power consumption with aggregated data, the forecast accuracy would be low. In the meanwhile, from the bottom figure of Figure 5.2, we find that power consumption patterns of a single home in consecutive two weeks are quite similar. The reason is that for a single home, the weekly activities of homeowner are normally fixed with minor varieties. While in a smart grid, the minor varieties of different homes aggregate together, which causes that the aggregated power consumption patterns of different periods are different. Therefore, if we can utilize the power consumption patterns in individual homes, power consumption in smart grids can be better predicted.

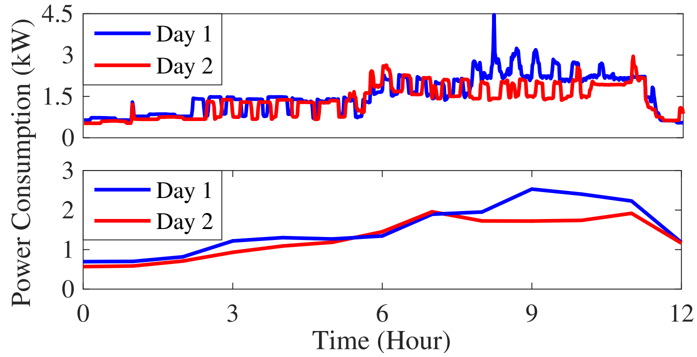


Figure 5.3: Power consumption pattern in a single home

5.2.3 The Need for High Time Granularity Data

In previous sections, we show hourly energy consumption in smart grids has weak correlation over time. Thus the energy consumption prediction based on aggregated energy consumption is not accurate. In this section, we investigate the correlation of high granularity energy consumption data in a single home. Hourly energy consumption and minute-level energy consumption data of a single home are shown in Figure 5.3. The top figure is the minute-level energy consumption in a single home for 12 hours. We can find the energy consumption pattern of two days are quite similar except there are several minutes delay between two days. This is because homeowner in a single home has relatively stable behavior pattern, which consumes similar amount of energy. However, the hourly energy consumption does not show the same phenomenon in the bottom figure. This is because even though homeowner has similar energy consumption pattern, the small delay of energy consumption makes the energy consumption pattern disappear in hourly energy consumption. Thus, to realize real-time energy consumption prediction, it is important to utilize the minute-level energy consumption data in individual homes.

5.2.4 Challenges of Utilizing Energy Consumption Patterns

From the empirical results of previous sections, we find out energy consumption patterns of homeowner can be explored with high granularity energy consumption data in individual homes. Then the explored energy consumption pattern in individual homes can be beneficial for energy consumption prediction in smart grids. However, there are many challenges for utilizing energy consumption pattern in individual homes. First of all, high granularity energy consumption data that reveals energy consumption pattern are large amount of data especially for large amount of homes in smart grids. Thus, we need an efficient algorithm to first learn energy consumption patterns in individual homes and then use the learned consumption patterns for energy consumption prediction. Secondly, in reality, not every home in the smart grids is deployed with a smart meter for monitoring high granularity energy consumption data. Thus, it is important to investigate how to utilize partial of the high granularity energy data from homes in smart grids to realize real-time energy consumption prediction. To address these challenges, we propose M-Pred, which utilizes different granularity of energy consumption data from individual homes for energy consumption prediction in smart grids.

5.3 Problem Formulation

In this section, we provide an overview of M-Pred, describe how energy consumption in individual homes can be utilized and present our design goal.

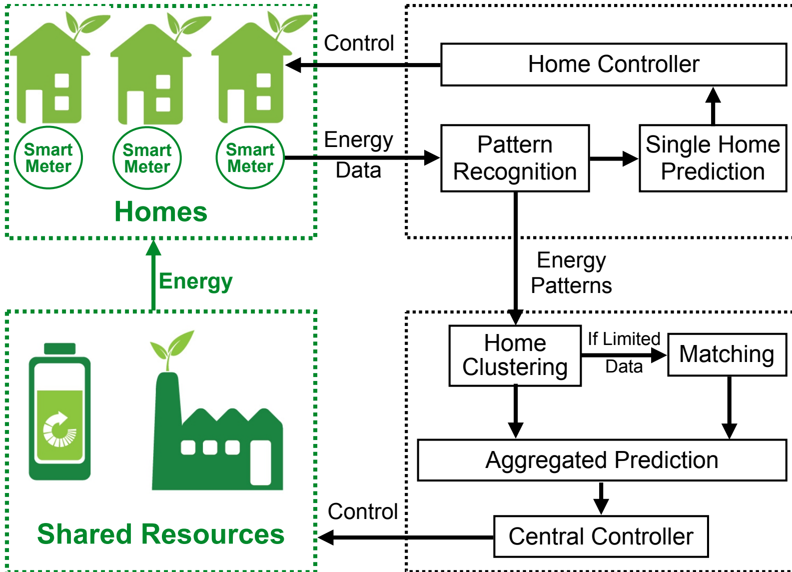


Figure 5.4: System Overview of M-Pred

5.3.1 Overview of M-Pred

To realize real-time energy consumption prediction in smart grids, we propose M-Pred, which utilizes the high granularity energy consumption data from individual homes to predict energy consumption in smart grids. The system overview of our design is shown in Figure 5.4. In smart grids, some homes are deployed with smart meters for monitoring high granularity energy consumption data. The energy consumption data is then processed locally for pattern recognition (detailed in § 5.4.1). The patterns recognized can be first utilized for energy consumption prediction in a single home. Then, to reduce the computation complexity and communication overhead for energy consumption forecast, a distributed energy consumption prediction algorithm is proposed to utilize the consumption patterns learned from homes. The learned energy consumption patterns from different homes will be clustered and then applied for energy consumption prediction in smart grids (detailed in § 5.4.2). Considering that not every home in smart grids is deployed with a smart meter, an

Notations	**Definitions**
N	Number of home in a smart grid
$p_i(t)$	Original power consumption of home i at t
$d_i(t)$	Predicted power consumption of home i at t
$e_i(t)$	Prediction error of home i at t
$h_i(t)$	Predicted peak demand of home i at t
S	Power consumption pattern set S
$sdist(X_1, X_2)$	Distance between two vectors X_1 and X_2
c_{ij}	Consumption correlation between home i and j

Table 5.1: Notations for Demand Forecast

energy matching algorithm is proposed for smart grids in which only partial homes are deployed with smart meters (detailed in § 5.4.3). Finally, the energy consumption prediction can be latter used for generation scheduling and control to avoid power outage and improve the energy efficiency in smart grids.

5.3.2 Design Goal

To predict power consumption with high time granularity in short term, we propose a middleware M-Pred, which is designed to be run at both local smart meters and central servers. Due to the large amount of high granularity energy consumption data, the computation and implementation complexity of M-Pred should be low. Let $\{p_i(1), \cdots, p_i(t)\}$ be the original minute-level power consumption series for home i, $\{p(1), \cdots, p(t)\}$ be the aggregated energy consumption in smart grids, and $\{\hat{d}(1), \cdots, \hat{d}(m)\}$ and $\{\hat{h}(1), \cdots, \hat{h}(m)\}$ be the peak demand and hourly average power consumption in smart grids, respectively. Then we have

$$h(i) = \sum_{j=1}^{j=N} \sum_{k=i*60-59}^{k=i*60} p_j(k) \tag{5.1}$$

$$d(i) = \max_{k=i*60-59}^{k=i*60} \sum_{j=1}^{j=N} p_j(k) \tag{5.2}$$

Our design goal is to minimize the prediction error of both hourly average power consumption $\sum_{i=1}^{i=m}\{h(i)-h(i)\}$ and peak demand $\sum_{i=1}^{i=m}\{d(i)-d(i)\}$, where $h(i)$ and $d(i)$ are predicted hourly average and peak demand. Note that our proposed design M-Pred is used to provide accurate power consumption (hourly average and peak demand) prediction in both a single home and smart grids. Also, due to different applications, the energy power prediction should also be able to conducted for different prediction window size (e.g., next 1 hour, 4 hours or 1 day).

5.4 System Design

In this section, we introduce the main design of M-Pred. Our design consists of three parts: i) power consumption patterns recognition in a single home; ii) power consumption prediction in smart grids; iii) power consumption prediction with limited data from individual homes. In the end, we analyze the performance and complexity of M-Pred.

5.4.1 Consumption Pattern Recognition

Different from previous prediction model based on historical data, we need to predict the power consumption in the immediate future (e.g., next minute) for real-time control because of the limited energy storage units in smart grids. Thus the power consumption data of yesterday or last month can be much less useful. And because there are different power consumption signatures for different loads, we can predict the data based on the detected power consumption signatures. To evaluate our idea, we use trace data of one year for one home to investigate the correlation between power consumption of different time gaps.

We first run a power consumption pattern detection algorithm on the data set to recognize the power consumption patterns. In this chapter, we use a Euclidean distance-based function to quantify the similarity between two vectors. The distance between two vectors can be calculated as:

$$\rho_{i,j} = \frac{1}{l(S_i)} \sum_{t=1}^{l(S_i)} (S_i(t) - S_j(t))^2 \tag{5.3}$$

If the distance of two vectors calculated through Equation (5.3) is small, then the similarity of two vectors is high. Then we go through the whole data set to find the possible consumption patterns. To simplify the algorithm, we use fixed length of energy consumption patterns. The algorithm we use is shown in Algorithm 4. At the beginning, the consumption pattern set S is empty. For $t < T$, we calculate similarity between power consumption data and recognized consumption patterns based on Equation (5.3). If we find the similarity between current power consumption and existing power consumption pattern is higher than current maximum similarity, we reassign maximum similarity and mark $index = i$. Then we compare the maximum similarity we find to the threshold of minimum similarity ρ_{min}. If $\rho_{max} > \rho_{min}$, we then detect a new consumption pattern S_{new} and add it to consumption pattern set S. Then, we update $t = t + l(S_{new})$ for further recognition. Otherwise, we update $t = t + 1$ to continue the recognition process.

To evaluate the performance of our consumption pattern recognition algorithm, we show some of the recognized consumption patterns from one home in Figure 5.5. For

Algorithm 4: Pattern Recognition Algorithm

$S = \emptyset$;
while $t < T$ **do**
  $\rho_{max} = 0, index = -1$;
  for detected consumption pattern S_i **do**
    Calculate $\rho_i(t)$ based on Equation (5.3);
    if $\rho_i(t) > \rho_{max}$ **then**
      $\rho_{max} = \rho_i(t), index = i$,
    end if
  end for
  if $\rho_{max} > \rho_{min}$ **then**
    Detect a new consumption pattern S_{new} and add to S;
    $t = t + length(S_i)$;
  else
    $t = t + 1$;
  end if
end while

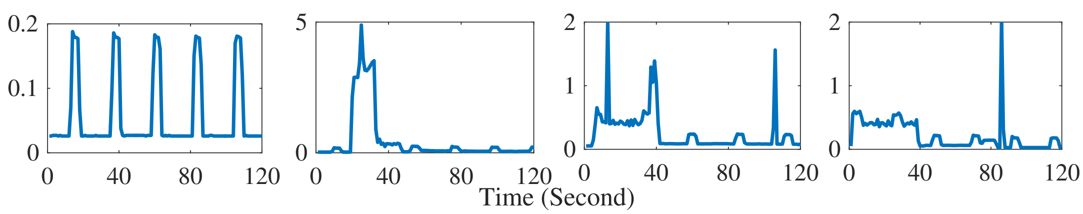


Figure 5.5: Some examples of signatures

example, the left top figure is the periodical activity of refrigerators while other three consumption patterns are the combinations of several appliances activities.

Consumption Pattern Fitting

With the recognized consumption patterns in individual homes, future power consumption can be predicted. However, to enable power consumption prediction, the length of energy consumption pattern have to be long enough (e.g., 120 minutes in Figure 5.5). To save the storage in smart meter and communication overhead for future energy consumption prediction in central server, we introduce polynomial curve fitting to sketch the power consumption patterns. To fit power consumption patterns, we first consider the general form for a polynomial of degree n:

$$p(t) = a_0 + a_1 x(t) + a_2 x(t)^2 + a_3 x(t)^3 + \cdots + a_n x(t)^n \tag{5.4}$$

The curve that gives minimum error between real power consumption pattern and the fitted curve is best. In our case, we use least squares error to find the best fitted curve of power consumption patterns. The general expression for any error using the least squares approach is:

$$err = \sum_{t=1}^{T_p} (d(t) - p(t))^2 \tag{5.5}$$

We then find the A to minimize err where A is $[a_0, a_1, \cdots, a_n]^T$. To minimize Equation (5.5), take the derivative with respect to each coefficient set to zero:

$$\frac{\partial err}{\partial a_j} = -2 \sum_{t=1}^{T_p} (d(t) - p(t)) x(t)^j \tag{5.6}$$

Then we have to solve $n+1$ equations to find A to minimize err:

$$XA = B \tag{5.7}$$

where

$$X = \begin{bmatrix} 1 & \sum x(t) & \sum x(t)^2 & \dots & \sum x(t)^n \\ \sum x(t) & \sum x(t)^2 & \sum x(t)^3 & \dots & \sum x(t)^{n+1} \\ \sum x(t)^2 & \sum x(t)^3 & \sum x(t)^4 & \dots & \sum x(t)^{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum x(t)^n & \sum x(t)^{n+1} & \sum x(t)^{n+2} & \dots & \sum x(t)^{n+n} \end{bmatrix} \tag{5.8}$$

$$B = \begin{bmatrix} \sum d(t) \\ \sum x(t)d(t) \\ \sum x(t)^2 d(t) \\ \dots \\ \sum x(t)^n d(t) \end{bmatrix} \tag{5.9}$$

Then we can get A from $X^{-1}B$. With A, we use Equation (5.4) to calculate fitted curve. After fitting, we only need to store A instead of $\{p(1), \cdots, p(t)\}$ for future power consumption prediction.

Power Consumption Prediction in A Single Home

With the fitted power consumption pattern A, we can easily recover the power consumption pattern with Equation 5.7. With the recognized power consumption patterns and historical data, we can predict the energy consumption in future. The process is similar to pattern recognition. For each recognized power consumption pattern S_i, we calculate the similarity between power consumption pattern and historical data $P = \{p(1), \cdots, p(t)\}$ based on Equation 5.3. Then based on the distance between historical data and consumption pattern, we predict the future power consumption as

$$d(t+k) = s_i(t+k) + sdist(S_i, P) \tag{5.10}$$

$sdist(S_i, P)$ is the Euclidean distance between two vectors S_i and P, which can be calculated with Equation 5.3.

5.4.2 Power Consumption Prediction in Smart Grids

In previous section, we investigate the power consumption patterns in individual homes. In this section, we describe how we can utilize the recognized consumption patterns to predict future aggregated power consumption in smart grids for real-time demand response.

Individual Homes Clustering

A simple solution to utilize recognized consumption pattern for aggregated power consumption prediction is to conduct predictions in each home and then send the prediction results from each smart meters to the central server. However, with large number of homes in smart grids, it is not scalable because this solution would introduce high storage need and communication overhead between smart meters and central server. Because homes in the same area may have the similar power consumption patterns, in this section, we describe how to use the spatial correlation among power consumption of homes for power consumption prediction. However, different homes will not have different correlations at different time. Thus, we need to keep updating the correlations among homes for prediction. To evaluate our idea, we use trace data of 726 homes for one month to investigate the correlation among power consumption of different homes. The spatial correlation between

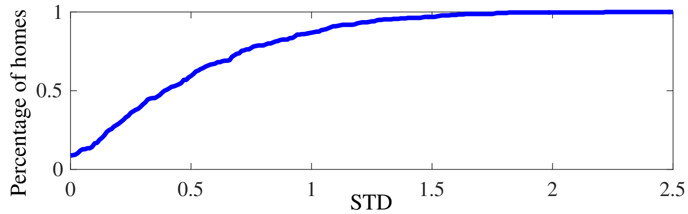


Figure 5.6: Spatial correlation of different homes over time

726 homes and one single home i is shown in Figure 5.6. X-axis is the standard correlation between two vectors. The correlation between two homes can be calculated as:

$$c_{ij}(t) = \frac{1}{l}\sum_{k=t-l}^{t}(p_i(k) - p_j(k) - \frac{1}{l}\sum_{k=t-l}^{t}(p_i(k) - p_j(k)))^2 \tag{5.11}$$

Y-axis is the CDF of given correlation. In Figure 5.6, most of the homes have similar consumption patterns with home i and 20% of the homes have correlation less 0.1 with home i. Thus, it is possible to infer power consumption of multiple homes based on the prediction results of a single homes.

Updating Power Consumption Patterns

Based on analysis of § 5.4.1, we show how to generate consumption pattern from real power consumption and polynomial fitting results. The key idea is to discover valuable power consumption data by comparing differences between real power consumption and polynomial fitting results. The detail of the algorithm is described in Algorithm 5. For each second t from 1 to T, it checks if the fitting error is larger than threshold of high power consumption d_h (Lines 1-2). Note that d_h needs to be selected carefully. If d_h is too small, it has better performance but cost too much storage; if d_h is too large, the error of polynomial fitting will be too high. Then if $|e(t)| > d_h$, it finds the endtime of high power consumption and add $\{power, start, end\}$ to P (Lines 3-7). If the fitting error is less than the threshold,

Algorithm 5: Consumption Pattern Updating Algorithm

Input: Fitting errors $e(t)$ between real secondly power consumption and polynomial fitting curve $p(t) - d(t)$.
Output: Consumption pattern set S.

1: **for** $t = 1$ to T **do**
2:   **if** $|e(t)| > d_h$ **then**
3:     $power = e(t)$, $start = t$, $i = t$;
4:     **while** $i < T$ & $|e(i)| > d_h$ **do**
5:       $i++$;
6:     **end while**
7:     $end = i$, Add $\{power, start, end\}$ to P;
8:   **else**
9:     **for** $j = 1$ to t_p **do**
10:       **if** $|e(t) - e(t+j)| \leq \theta$ **then**
11:         $per = j$;
12:         **for** $k = 1$ to t_k **do**
13:           **if** $|e(t+k) - e(t+k+j)| \geq \theta$ **then**
14:             $time = k$;
15:           **end if**
16:         **end for**
17:         Add $\{power, start, end, per, time\}$ to S;
18:       **end if**
19:     **end for**
20:   **end if**
21: **end for**

then the algorithm checks if there is periodical power consumption (Line 8-10). If yes, it finds the period and last time of periodical power consumption (Lines 11-16), then adds $\{power, start, end\}$ to P (Lines 17-21).

Utilization of Consumption Correlation

Based on the analysis of § 5.4.2, power consumption of different days may have similar patterns and thus only one day's consumption needs to be stored for reducing storage space. For example, instead of storing new power consumption data at Day 3, we only need to store the power consumption differences between Day 1 and Day 3. However, we need to carefully determine whether we can reduce storage space based on the correlation between two days. The format for storing correlation information is as follows: $\{b_c,\ t_1,\ e_1,\ \cdots,\ t_m,\ e_m\}$. b_c is the bit used to store whether or not we utilize the correlation of power consumption in storing the data. t_i ($i\ \in\{1, \cdots, m\}$) is used to store the time slot that power consumption of two days are different and e_i ($i\ \in\{1, \cdots, m\}$) is used to store the power consumption differences. As long as $m < T/2$, the data used to store time and power consumption differences ($2*m$ data points) would be less than directly storing fitting errors (T data points). Algorithm 6 gives a detailed description on how to make the decision. For each second of t from 1 to T, it checks if the difference of power consumption for two days is less than threshold d_c and counts the number of time slot (Lines 1-6). Note that d_c needs to be selected carefully. If d_c is too small, it has better performance but cost more storage to store high power consumption; if d_c is too large, the error of polynomial fitting will be too high. Then algorithm checks if correlation between two days can be used to save space (Lines 7-11). In server, when power consumption data of a new day is received from a smart meter, it runs Algorithm 6 using the power consumption of past several days to find a day to be used to save space. If it does not find, it stores the power consumption data of that day from smart meter to database.

Power Consumption Prediction with Low Computation

Based on the correlation, we can predict $p(t)$ based on readings from other homes:

$$p(t) = \sum_{i=1}^{N} p_i(t) = \sum_{i=1}^{N} \sum_{j=1}^{N_h} \frac{p_j(t) * c_{ij}(t)}{\sum_{j=1}^{N} c_{ij}(t)} \tag{5.12}$$

If $c_{ij}(t)$ does not exist, then we replace $c_{ij}(t)$ as $c_{ij}(t_k)$ where t_k is the latest time for updating correlation between home i and j. N_h is the number of homes selected for con-

Algorithm 6: Correlation of Power Consumption

Input: Fitting errors $e_1(t)$ and $e_2(t)$ of two days.
Output: Decision of correlation of two days.

$count = 0$;
for t = 1 to T **do**
  if $|e_1(t) - e_2(t)| < d_c$ **then**
    $count + +$;
  end if
end for
if $count \geq T/2$ **then**
  Correlation can be used to update pattern;
else
  Correlation can not be used to update pattern;
end if

ducting power consumption prediction based on correlation among homes. The prediction accuracy is highly dependent on the selection of homes for power consumption prediction. Here we give a detailed description on how to make the selection. We first add home 1 as one of the selected homes. For each home i from 2 to N, it checks if the correlation of power consumption between home i and any selected homes is smaller than threshold d_c. If the correlation between home i and one of the selected homes is smaller than d_c, we skip the home i. If the correlation between home i and any selected homes is larger than d_c, then we add home i as one of the selected homes. We continue this process until all homes are either skipped or selected one of the homes.

5.4.3 Energy Consumption Forecast with Limited Data

In previous design, we consider that every home in smart grids are deployed with smart meters. However, it may not be true in reality. Thus, we also propose an algorithm to improve energy consumption forecast in smart grids when only partial of homes in smart grids are deployed with smart meters. The key idea is to first conduct the energy matching with existing individual homes' energy consumption data we collect in the central server. Then based on the matching results, we know the information of how many homes of different power consumption patterns exist in smart grid. Based on the energy consumption prediction in individual homes with smart meters and aggregated energy consumption in smart grids, we utilize the matching results to predict the energy consumption of smart grids in future.

The high-level idea of our algorithm is that it searches for a shapelet which can separate and remove a subset of time series from the rest of the dataset, then iteratively repeats this search among the remaining data until no data remains to be separated.

As discussed before, an ideal shapelet has the ability to divide a dataset **D** into two groups of time series, D_A and D_B. D_A consists of the time series that have subsequences similar to while D_B contains the rest of the time series in **D**. Simply stated, we expect the mean value of $sdist(S, D_A)$ to be much smaller than the mean value of $sdist(S, D_B)$. Since we ultimately use a distance map that contains distance vectors to cluster the dataset, the larger the gap between these two means of these distances vectors, the better. We use the algorithm to extract shapelets. In essence, this algorithm can be seen as a greedy search algorithm which attempts to maximize the separation gap between two subsets of D. This separation measure is formally encoded in the following equation:

$$gap = \mu_B - \sigma_B - (\mu_A + \sigma_A) \tag{5.13}$$

In Equation 5.13, μ_A and μ_B represent $mean(sdist(S, D_A))$ and $mean(sdist(S, D_B))$ respectively, while σ_A and σ_B represent $std(sdist(S, D_A))$ and $std(sdist(S, D_B))$, respectively. In our algorithm, we consider all subsequences of the time series as candidate shapelets and compute their distance vectors. We can represent a distance vector as a schematic line. Then we search these lines for the location that maximizes the gap function introduced in Equation 5.13. We refer to this point as dt. Points to the left of dt represent $sdist(S, D_A)$, while points to the right correspond to $sdist(S, D_B)$

Once we know the gap scores for all the subsequences of a time series, we add the subsequence with maximum gap score in the set of shapelets. Given that we have selected a u-shapelet, we do not want subsequences similar to it to be selected as shapelets in subsequent iterations. Thus we remove the time series that have subsequences similar to the shapelet from the dataset and use only the remaining dataset to search for the next shapelet.

The detailed design of energy consumption prediction is shown in Algorithm 7. For polynomial fitting, we can calculate gap with $\{T_p, a_0, \cdots, a_n\}$ based on Equation 5.7 and 5.13. For energy consumption pattern abstraction, we can calculate high energy consumption patterns in a short time and periodical energy consumption over a long time with $\{power2, starttime2, endtime2, period, time\}$. For utilization of consumption correlation, we can use b_c to detect whether we utilize correlation of energy consumption. For the limited data, we apply energy matching algorithm for energy consumption pattern from given individual homes and aggregated energy consumption data. Finally, with the data from four components, we calculate the predicted energy consumption in smart grids.

Algorithm 7: Energy Consumption Prediction with Limited Data

for $t = 1$ to T_p **do**
  Calculate gap based on Equation 5.7 and 5.13;
end for
for $t = starttime1$ to $endtime1$ **do**
  $d(t) = d(t) + power1$;
end for
for $t = starttime2$ to $endtime2$ **do**
  if $t\%(time + period) = 0$ **then**
    for $i = 1$ to $time$ **do**
      $d(t) = d(t) + power2$;
    end for
  end if
end for
if $b_c = 1$ **then**
  for $i = 1$ to m **do**
    $d(t) = d(t) + e(t)$;
  end for
end if

5.4.4 Time Complexity Analysis

In this section, we analyze the complexity of M-Pred at local smart meters in the following three stages.

i) *Learning Energy Consumption Patterns*. Basic power consumption curve is sketched in time domain. In learning algorithm, Algorithm 4 is applied to generate sampled power consumption changes and time complexity of Algorithm 4 is $O(N)$.

ii) *Energy Consumption Prediction*. Sampled data of power consumption changes are transferred into energy consumption pattern data. In energy consumption prediction, we need to conduct $\tilde{d}(t)$ and run Algorithm 5. The time complexity is $O(NlogN)$ and time complexity for Algorithm 5 is $O(N)$.

iii) *Energy Prediction for Smart Grids with Limited Data*. The limited data from individual homes is used for energy matching and energy consumption prediction in smart grids. Be-

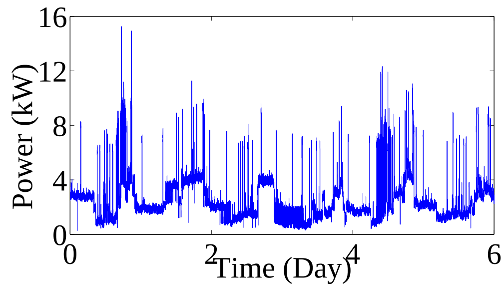


Figure 5.7: Aggregated power consumption over 12 months

cause the number of homes available is limited, thus the communication overhead between homes and central server is low. Time complexity of matching and energy consumption prediction is $O(NlogN)$.

In total, the time complexity of our design is still $O(NlogN)$, which means our design is simple for smart meters.

5.5 Implementation and Evaluation

In this section, we evaluate the performance of our proposed design. We deploy eGauge power meters at individual homes to collect the energy consumption data every minute. In our simulation, we use the power consumption traces that we collected from 726 homes for more than one year. To make the figure easy to follow, we only show the aggregated power consumption for 12 months in Figure 5.7. It can be found that the power consumption for different days varies significantly and is higher in summer while lower in winter.

5.5.1 Evaluation Baseline and Metrics

Baselines. To verify the prediction accuracy of our approach, we compare our design with three existing approaches: i) NYISO: New York ISO [85], which is the standard of energy consumption prediction in New York State; ii) CASCE: Southern California Edison ISO

[105], which is the standard of energy consumption prediction in South California; and iii) CAISO: California ISO [24], which is the standard of energy consumption prediction in California.

For our design, to verify the prediction accuracy of pattern recognition and home clustering, we also compare to i) our design with only pattern recognition (**PR**), which only utilizes aggregated power consumption in smart grids for prediction; ii) our design with perfect home clustering (**PR+All**), which assumes each home is a cluster. The prediction accuracy of **PR+All** should be better than our design, however, it is not scalable when the number of homes are huge. In our simulations, we consider **PR+All** as optimal algorithm for prediction accuracy.

Metrics. Because we predict both peak demand and hourly average power consumption in smart grids. Thus, we use two metrics to evaluate the performance of our approach: i) **MAPE of peak demand** and ii) **MAPE of hourly average power**.

5.5.2 Energy Consumption Prediction in A Single Home

To enable real-time demand response in individual homes, energy consumption prediction in a single home must be very accurate. In our simulations, we run the pattern recognition algorithm with six months data and conduct the prediction with another six months. The prediction window size in these series of simulations are set as 24 hours.

Hourly Average Power Consumption and Peak Demand Prediction

To make the results easy to follow, we only show the prediction results of one typical home for one week in Figure 5.8. The prediction of hourly average power consumption matches

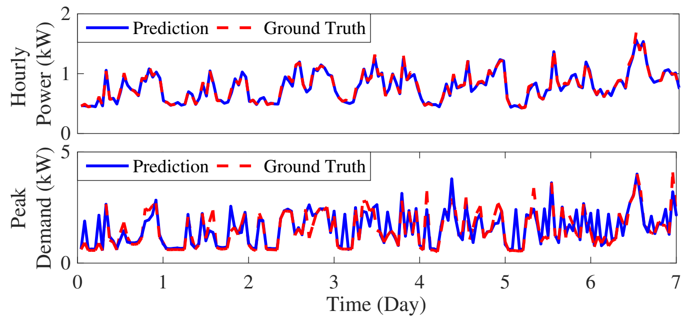


Figure 5.8: Prediction accuracy of hourly average power and peak demand in a typical home

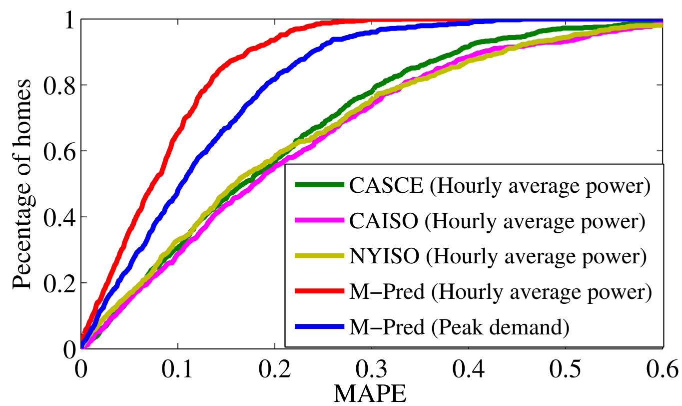


Figure 5.9: CDF of MAPE for different homes with different methods

very well with the ground truth. The prediction of peak demand is also very accurate for the most of time. However, the prediction of peak demand is less accurate compared to hourly average power consumption. This is because hourly average power consumption in a single is typical more stable; in the meantime, peak demand is highly dependent on the accurate time of energy consumption events. Thus, it is much more difficult to predict peak demand accurately.

Prediction Results for Different Homes

The prediction results of different homes compared to existing approaches are shown in Figure 5.9. Because existing approaches can only predict hourly average power consumption, we only show the prediction results of peak demand with our design. For CASCE, CAISO and NYISO, the prediction accuracy is similar and varies from different homes, which is consistent with the results from [41]. The prediction results of hourly average power with our design is much better than existing approaches, 95% of the homes can be predicted with MAPE less than 0.2 and average MAPE for all the homes is 0.08. For the prediction of peak demand in different homes, the accuracy is also very good. 83% of the homes can be predicted with MAPE less than 0.2 and average MAPE for all the homes is 0.11.

5.5.3 Energy Consumption Prediction in Smart Grids

Because smart grids need accurate aggregated power consumption prediction, thus we provide the prediction results of aggregated peak demand and hourly power consumption in this section.

Prediction with Different Methods

To verify the detailed performance of our design, we compare prediction results with **PR** and **PR+All**. For **PR**, we applied our pattern recognition algorithm with aggregated power consumption in smart grids. No minute-level energy data from individual homes are used for **PR**. For **PR+All**, we assume all the minute-level energy data from individual homes are available in the central server. In reality, it is not practical because large amount of energy data needs to be transmitted from smart meters to central server especially when the number of homes in a smart grid is huge. In this section, we consider **PR+All** as the optimal prediction accuracy we can achieve by utilizing minute-level energy data from individual homes. The MAPE of different methods is shown in Figure 5.10. We can find that for all three methods, the prediction accuracy of hourly average power is always better than prediction

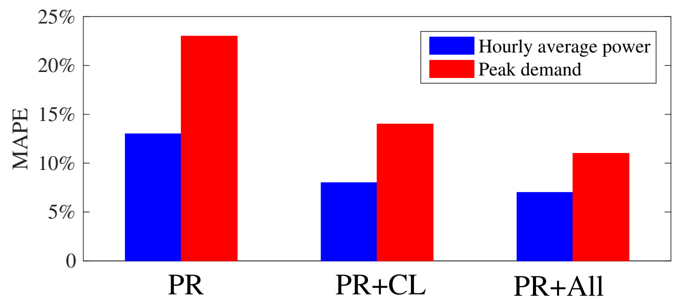


Figure 5.10: MAPE of aggregated hourly average power and peak demand prediction in smart grids

of peak demand, which is similar to prediction accuracy in a single home. Compared to **PR**, the prediction accuracy of hourly average power consumption and peak demand with M-Pred is both around 40% better and very close to optimal results of **PR+All**. Therefore, our design is well balanced between prediction accuracy and communication overhead.

Prediction with Limited Available Homes

Considering not every home in smart grids is deployed with smart meters, we evaluate the performance of our design with minute-level energy data from different number of homes. The results are shown in Figure 5.11. The X-axis is the percentage of homes that are deployed with smart meters in a smart grid; and Y-axis is the MAPE value of prediction results. We can find that the prediction accuracy of both hourly average power consumption and peak demand increase with minute-level energy data from more homes. For prediction of hourly average power consumption, the accuracy increases more slowly with percentage of available homes. This is because in residential homes, energy consumption is mostly high in the morning, low in the day time and high in the evening. Therefore, the curve of hourly average power consumption over time is similar for different homes. At the mean time, prediction accuracy of peak demand only increases significantly when the percentage of available homes reaches 40%. This is because peak demand in different homes are

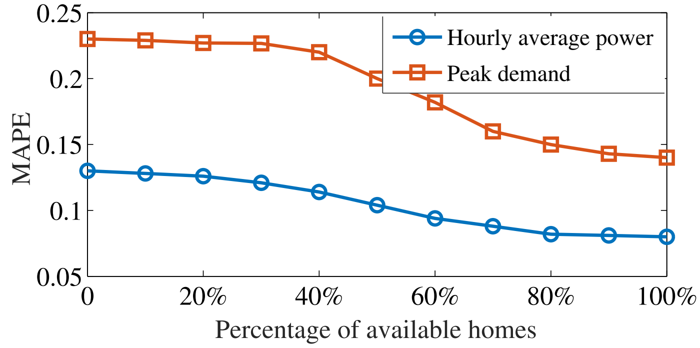


Figure 5.11: MAPE of aggregated hourly average power and peak demand prediction with different percentages of homes with minute-level energy data

usually more dependent on homeowners' behavior patterns, thus, only with enough energy data from individual homes, the prediction accuracy of peak demand will be improved.

5.5.4 Impact of Window Size

In this section, we investigate the impact of window size on the prediction accuracy of hourly average power consumption and peak demand. We apply our design with window size of 1 hour, 4 hours, 8 hours, 12 hours and 24 hours. The results are shown in Figure 5.12. With larger window size, the consumption events of homeowners' are more unpredictable. Thus, the prediction accuracy of both hourly average power consumption and peak demand decreases with larger window size. For 1 hour window size, the prediction accuracy of our design is extremely high with MAPE around 0.02. Therefore, our design can provide very accurate power consumption prediction for real-time demand response in smart grids. Similar to results in previous sections, prediction accuracy of peak demand decrease faster with large window size because the peak demand is more unpredictable.

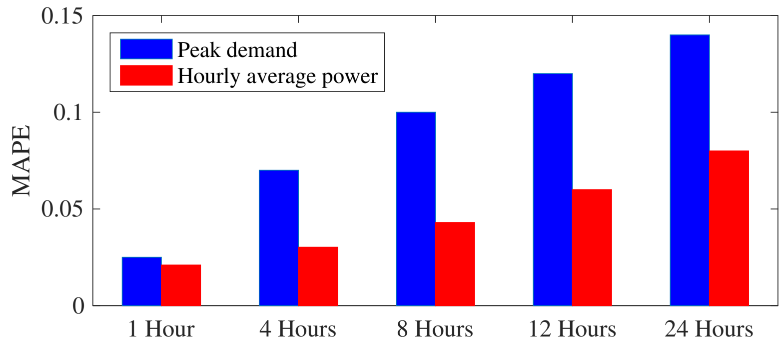


Figure 5.12: MAPE of aggregated hourly average power and peak demand prediction with different window size

5.6 Related Work

Our work is related to two areas of previous work: demand forecast and peak demand:

Demand Forecast. Research on electricity demand forecast includes long-term and medium-term prediction for utility planning and maintenance purposes, and short-term forecast for economic scheduling [9]. In this chapter, we focus on the short-term demand forecast. Related work on demand forecast includes three types of methods: simple averaging models [24, 85, 105]; statistical models (e.g., regression [57] and time series [25, 46]); and machine learning techniques(e.g., Artificial Neural Networks (ANNs) [48, 90] and pattern matching [76, 102]). However, existing forecast techniques only conduct forecast with aggregated power consumption in smart grids. In this chapter, we show that with detailed power consumption in individual homes collected from smart meters, power consumption patterns in each home can significantly help the demand forecast in smart grids.

Peak Demand. There are many works on modifying the elastic load components of common household appliances to reduce peak demand [49]. In [106], a novel demand response mechanism is proposed to exploits appliance elasticity to decrease peak loads. A real-time distributed deferrable load control algorithm is proposed to reduce the peak load by shifting the power consumption of deferrable loads to periods with high renewable generation [42]. Batteries are deployed at homes to supply energy when peak demand and store energy when energy consumption is low [82]. To support different approaches on flattening peak demand in smart grids, we present peak demand forecast in this chapter for the first time. The simulation results show that our design can significantly improve prediction accuracy of peak demand in smart grids.

5.7 Conclusion

To the best of our knowledge, this is the first work to utilize the detailed power consumption in individual homes to help power consumption prediction in smart grids. We show that the detailed power consumption patterns in each home can significantly improve prediction accuracy of power consumption in smart grids. In this chapter, we propose M-Pred to learn energy consumption pattern of individual homes from their energy consumption data and then utilize these patterns to predict the power consumption in smart grids. Our design consists of three parts: i) energy consumption patterns recognition in a single home; ii) energy consumption prediction in smart grids; iii) energy consumption prediction with limited data from individual homes. Finally, we analyze the performance and complexity of M-Pred. We conducted extensive system evaluations with 726 homes' minute-level power consumption data for more than 1 year. The results show that our design can provide accurate real-time energy consumption with negligible errors (e.g., Mean Absolute Percentage Error is 2.12%).

CHAPTER 6

ENERGY SHARING IN MICROGRID

To reduce electricity usage and peak demand, many utilities are introducing market-based time-of-use (TOU) pricing models. In parallel, government programs that in- crease the fraction of renewable energy are incentivizing residential consumers to adopt on-site renewables and energy storage. Connecting on-site renewables and energy storage between homes forms a sustainable microgrid capable of generating, storing, and sharing electricity to balance local generation and consumption in residential areas. In this chapter, we investigate how to minimize the costs of electricity from a utility for a microgrid under market-based TOU pricing models.

6.1 Problem Formulation

To minimize the AC energy costs of the microgrid, we propose the system architecture for energy-sharing and describe the system components and interactions between these components. Then we analyze the model of energy-sharing to formulate the problem.

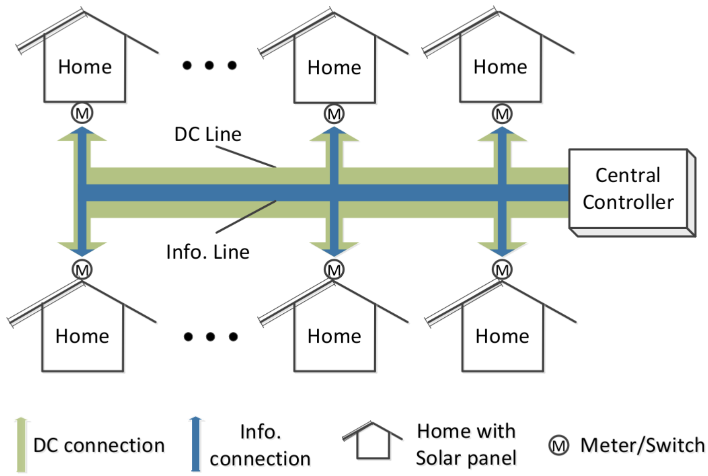


Figure 6.1: Architecture of Microgrids: Interconnected homes with renewable energy supply (e.g., solar panels)

6.1.1 System Overview

To ensure compatibility with the traditional power grid, we adopt the microgrid architecture (shown in Figure 6.1), which is similar to the one used in a traditional power grid. Just as the traditional grid has a distribution network, our microgrid employs a similar but separate distribution network across the community of homes comprising the microgrid. Within this network, there is a power meter and a switch between every home and the central controller. The power meter is used to measure energy harvesting and consumption, while the switch is used to control energy sharing with other homes. Batteries are also deployed in each home to supply energy when there is not enough energy sharing. A centralized design is chosen to optimize AC energy costs of the microgrid and save most computation in homes. The central controller collects energy-related data from homes, arranges energy transmissions and determines the price of sharing energy among homes. Presently, our microgrid distribution network assumes a DC-based network for reasons of convenience and efficiency (e.g., to reduce conversion losses). Our system architecture and design are also compatible with an AC distribution network by considering the energy conversion loss from AC to DC and vice versa. For the sake of clarity, in the rest of the chapter, we use DC distribution network to share renewable energy among homes and use traditional AC power line for distributing energy from the utility company to homes. The design goal is to minimize the cost from the AC line by optimally sharing renewable energy under different pricing models.

To realize optimal energy sharing, we propose the system design as shown in Figure 6.2, which includes two components: home controllers and a central controller. The home controllers have two planes: an energy sensing plane and an execution plane. The energy sensing plane senses the home's real-time energy data and makes a prediction, then forwards those data to the central controller, which includes (i) current and future energy consumption data, (ii) current and predicted energy harvesting data, and (iii) current amount of energy in the battery. For prediction of energy harvesting, we focus on solar energy here as it is the predominant renewable energy source in residential DG deployment. To predict energy harvesting, a weather forecast based prediction model similar to Sharma's approach ([101]) is adopted. At any time t, based on the sky condition percentage $C(t)$ released by the National Weather Service (NWS), we predict the solar panel's energy harvesting rate. For prediction of home's consumption, we use historical consumption data to predict future energy consumption based on an Exponentially Weighted Moving Average (EWMA). The EWMA exploits the diurnal nature of a home's consumption, while it also adapts to seasonal variations. Note that more sophisticated models that consider changing weekend activity patterns and weather conditions can be used to improve our work. However, this is not our main contribution and from evaluation results, the following prediction models can already provide enough accuracy for our system.

In addition, homes convey their battery and solar panel capacities and cost data when they join the system. The above data will be used by the central controller for energy allocation and price decision purposes. After receiving energy-sharing instruction from central controller, the home controller's execution plane toggles the power meter and a switch to transmit a certain amount of energy. The switch controls energy flow from the following options: (i) use energy from AC line to power appliances; (ii) charge battery from AC line; (iii) charge battery from DC line; (iv) discharge battery to DC line; and (v) discharge battery to power appliances.

The central controller contains only the energy-sharing plane, which includes the spatial energy sharing, temporal energy sharing and AC allocation.

The energy-sharing plane processes data needed during energy sharing. It has three major modules: (i) the spatial energy-sharing module that uses the individual home's energy data and sharing efficiency to find energy-sharing home pairs (detailed in §6.2.1); (ii) the temporal energy-sharing module that yields the optimal solution on minimizing the AC energy costs of the microgrid based on TOU prices and energy-sharing pairs of spatial sharing module (detailed in §6.2.2); and iii) the AC allocation module that optimally calculates the amount of energy each home should get from the AC line based on data from both the spatial energy sharing and temporal energy sharing module (detailed in §6.2.3).

In summary, our system works as follows: (i) home controllers gather consumption, harvesting and battery information and send it to central controller; (ii) central controller then decides energy-sharing sequences and sharing price; (iii) homes discharge battery to share energy to others based on central controller's instruction; and (iv) if a home still needs energy after sharing, it gets energy from AC line.

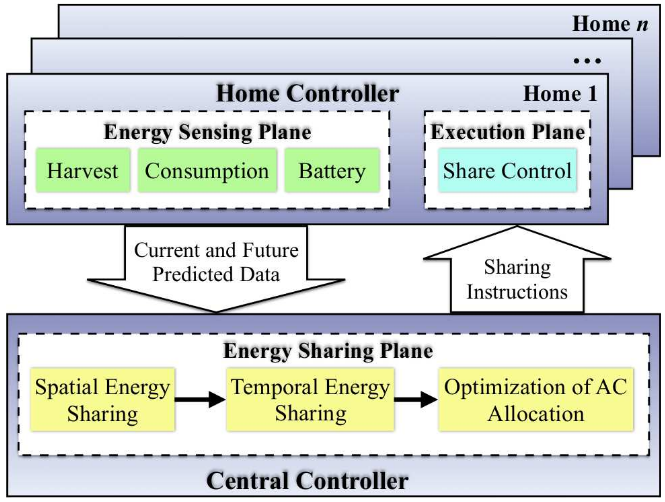


Figure 6.2: Overview of system architecture

6.1.2 Problem Definition

With the proposed system for energy sharing, we model the energy-sharing process and formulate the problem. Because energy sharing takes time to transmit energy from one home to another home, in our system, time is divided into time slots, and the size of a slot is referred to as **window size** w. Then we can do energy sharing at window n based on energy consumption and harvesting at window $n+1$ to reduce electricity cost. Let $\triangle E_i(nw) = EH_i(nw) - EC_i(nw)$ be the difference between harvested energy $EH_i(nw)$ and consumed energy $EC_i(nw)$ for home i in the time interval $[nw, (n+1)w]$ and $n \in [1, N]$. To simplify the notation, we will use ***n*** to represent ***nw*** in rest of the chapter. Let $E_{i \to j}(n)$ be the amount of energy transmitted from home i to j in window n; and η_{ji} be the transmission efficiency between homes i and j. The amount of surplus energy $ES_i(n)$ home i can get during energy sharing is as follows:

$$ES_i(n) = \sum_j (E_{j \to i}(n) \cdot \eta_{ji} - E_{i \to j}(n)) \tag{6.1}$$

$\sum_j E_{j \to i}(n) \cdot \eta_{ji}$ is the amount of energy home i receives from other homes and

$\sum\limits_{j} E_{i \to j}(n)$ is the amount of energy that home i provides to other homes. Let $B_i(n)$ and C_i be the battery level (amount of energy in battery) of window n and battery capacity of home i respectively. Table 6.1 summarizes the definition of parameters. Based on information of window n, we can calculate $B_i(n+1)$ as follows:

$$B_i(n+1) = \begin{cases} 0 & B_i(n) + \triangle B_i(n) < 0 \\ C_i & B_i(n) + \triangle B_i(n) > C_i \\ B_i(n) + \triangle B_i(n) & \text{otherwise} \end{cases} \tag{6.2}$$

where $\triangle B_i(n)$ represents the total amount of energy gap at home i in window n, which includes the difference between harvested and consumed energy, energy obtained from the AC line and energy transmission between home i and other homes in the microgrid. $\triangle B_i(n)$ can be calculated as follows:

$$\triangle B_i(n) = \triangle E_i(n) + EA_i(n) + ES_i(n) \tag{6.3}$$

where $\triangle E_i(n)$ is the difference between harvested and consumed energy at home i in window n; $EA_i(n)$ and $ES_i(n)$ is the amount of energy home i gets from the AC line and energy sharing in window n respectively; Based on the definition above, we can have the following lemma:

Lemma 6.1.1. *Home i does not have enough energy to consume if $B_i(n) + \triangle B_i(n) < 0$.*

Proof. The equation in lemma can be rewritten by using Equation 6.3 to separate the energy consumption and the energy sources as follows:

$$EC_i(n) > B_i(n) + EH_i(n) + EA_i(n) + ES_i(n) \tag{6.4}$$

In Equation 6.4, the energy consumption is larger than the energy obtained from all sources,

Notation	Definition
w	Window size
$B_i(n)$	Amount of energy in battery of home i in window n
C_i	Battery's capacity of home i
r_i^c	Battery charging rate of home i
r_i^d	Battery discharging rate of home i
η_{ij}	Energy transmission efficiency between i to j
$C(t)$	Sky condition percentage at time t
$EC_i(n)$	Consumed energy of home i in window n
$EH_i(n)$	Harvested energy of home i in window n
$\widehat{EC}_i(n)$	Predicted consumed energy of home i in window n
$\widehat{EH}_i(n)$	Predicted harvested energy of home i in window n
$Y_{AC}(n)$	Price of AC line based on TOU in window n
$\triangle E_i(n)$	$EH_i(n) - EC_i(n)$
$\triangle B_i(n)$	Amount of energy gap of home i in window n
$E_{i\rightarrow j}(n)$	Energy transferred from i to j in window n
$EA_i(n)$	Energy from AC line of home i in window n
$ES_i(n)$	Energy surplus by sharing of home i in window n

Table 6.1: Notations for Energy Sharing

which will cause power shortages. Thus that situation should always be avoided. □

Similarly, we have the following lemma:

Lemma 6.1.2. *Home i wastes energy if $B_i(n) + \triangle B_i(n) > C_i$.*

Lemma 6.1.2 happens when the battery in home i cannot store extra energy due to its capacity limit. This situation should be avoided, but may not be eliminated. Consider the case when in the middle of a day, the harvesting overwhelms the consumption, the battery may be charged to full capacity and extra harvested energy would be wasted.

Let $Y_{AC}(n)$ be the price of the AC line based on TOU in time interval $[n, (n+1)]$. Given $\triangle E_i(n)$ and $B_i(n)$, we can formulate our design goal of minimizing the microgrid electricity cost from the AC line as follows:

$$
\begin{aligned}
\min \quad & \sum_n \Big(Y_{AC}(n) \cdot \sum_i EA_i(n) \Big) \\
\textit{s.t.} \quad & 0 \leq B_i(1) \leq C_i && (a) \\
& EA_i(n) \geq 0 && (b) \\
& \sum_j E_{i \to j}(n) - \triangle E_i(n) - EA_i(n) \leq r_i^d \cdot w && (c) \\
& \sum_i E_{i \to j}(n) \cdot \eta_{ij} + \triangle E_j(n) + EA_j(n) \leq r_j^c \cdot w && (d) \\
& B_i(n) + \triangle B_i(n) \geq 0 && (e)
\end{aligned}
$$

Constraint (a) ensures the initial battery energy level will always be no less than zero and not greater than the battery capacity. Constraint (b) means a home can only get energy from AC line, but not sell energy to utility company. Constraints (c) and (d) mean that amount of energy can be transmitted from i to j is determined by discharging rate of i, charging rate of j and window size. Constraint (e) ensures every home can have enough energy during window n. $\triangle E_i(n)$ and $B_i(n)$ are determined by users' power consumption. To minimize the total AC energy cost, we can adjust $ES_i(n)$ by choosing proper energy-sharing home pairs (i.e., home i shares $E_{i \to j}(n)$ amount of energy with j) and allocate the amount of energy $EA_i(n)$ each home gets from the AC line over time. Thus we can rewrite Constraint (e) by plugging in Equation (6.3) as follows:

$$ES_i(n) + EA_i(n) \geq -B_i(n) - \triangle E_i(n) \tag{6.5}$$

The problem is a linear programming problem. However, for constraints (b) to (e), it needs to be valid for all windows n, thus total number of constraints is huge when number of homes and total time increases. Further, the objective function value for different windows are correlated and cannot be decomposed. Thus in this chapter, we propose a spatial-temporal energy sharing design and prove that our solution is a local optimal solution.

6.2 System Design

In this section, we describe the detailed system design, mainly focusing on the central controller part in §6.2.1, §6.2.2 and §6.2.3.

6.2.1 Spatial Energy Sharing

The first part of central controller spatial energy sharing is introduced in this section. The goal of spatial energy sharing is to minimize AC transmission in a single window. Because in a single window, $EA_i(n)$ is needed unless home i cannot get enough energy from energy sharing, we have

$$EA_i(n) = \triangle B_i(n) - ES_i(n) - \triangle E_i(n) \tag{6.6}$$

In window n, $B_i(n)$ and $\triangle E_i(n)$ are fixed values, the optimization problem can be rewritten as follows:

$$\begin{aligned}
\max \quad & \sum_i ES_i(n) \\
\textit{s.t.} \quad & \sum_j E_{i\to j}(n) - \triangle E_i(n) - EA_i(n) \le r_i^d \cdot w \qquad (a)\\
& \sum_i E_{i\to j}(n)\cdot \eta_{ij} + \triangle E_j(n) + EA_j(n) \le r_j^c \cdot w \qquad (b)\\
& B_i(n) + \triangle B_i(n) \ge 0 \qquad (c)\\
& ES_i(n) \le \triangle B_i(n) - \triangle E_i(n) \qquad (d)
\end{aligned}$$

$\sum_i ES_i(n)$ is total amount of energy sharing in window n. To maximize $\sum_i ES_i(n)$, we need to address the following two challenges:

• **Transmission Conflict**. Take Figure 6.3 as an example in which homes H1 and H3 need to provide energy to H2 and H4. Because all the homes are connected to same distribution network, if we do the energy sharing simultaneously, we cannot control the amount of energy from H1 to H2. Then we cannot know what amount of energy is transmitted from home i to j and cannot calculate how much j should pay to i for energy it receives. Thus, in our system, only one to multiple and multiple to one energy transmission is allowed at a time. If more transmissions are needed in microgrid, multiple one-to-multiple or multiple-to-one transmissions can be executed one by one.

• **Transmission Efficiency**. Because distances between homes are different, energy transmission losses between homes are also different. Besides, energy sharing between homes may be discharged from battery and charged to battery, which introduces battery conversion loss. Thus we need to consider different transmission efficiency between homes when designing the energy sharing algorithm. For example, transmission efficiency between $H1$, $H3$ and $H2$, $H4$ given in Figure 6.3 (a) includes both transmission loss and battery conversion loss.

To address the above challenges, we introduce maximum transmission speed of homes. We divide homes into an energy supplier set S and a demander set D according to whether the energy difference is positive or negative. Then we consider maximum transmission speed for two types of energy-sharing home pairs: (i) one demander with multiple suppliers in which the transmission speed is limited only by the demander's battery charging rate when there are enough suppliers; and (ii) one supplier with multiple demanders in which the transmission speed is determined by not only the supplier's discharging rate, but also by transmission loss between demanders and the supplier.

Assuming home i shares energy with home j, the discharging rate for i is r_i^d, the charging rate for j is r_j^c, and the transmission efficiency is η_{ij}, then the energy transmission rate r_{ij} during energy sharing is as follows:

$$r_{ij} = min(r_i^d \cdot \eta_{ij}, r_j^c) \tag{6.7}$$

If multiple suppliers share energy to home j, the maximum transmission speed is determined by minimum value of its charging rate and total energy transmission rate available from suppliers.

$$r_j = min(\sum_k r_{kj}, r_j^c) \tag{6.8}$$

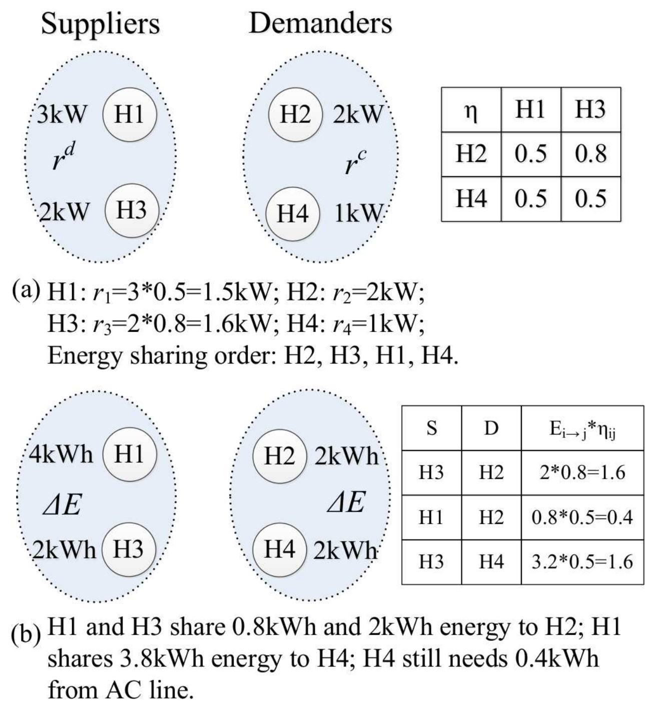


Figure 6.3: An example of energy sharing between demander set D $\{$H1, H3$\}$ and supplier set D $\{$H2, H4$\}$.

If home i shares energy with multiple demanders, the maximum transmission speed is determined by discharging rate and transmission efficiency between i and demanders. To maximize transmission speed, we select demander with highest transmission efficiency until we reach discharging rate of i. Assume the demander set for maximum transmission speed is $M = \{m_1, m_2, ...\}$, $\eta_{im_i} > \eta_{im_j}$ if $i < j$. Then we have maximum transmission speed of i as follows:

$$r_i = \sum_{k \in M} r_{ik}, \quad s.t. \quad \sum_{k \in M} r_{ik}/\eta_{ik} \leq r_i^d \tag{6.9}$$

An example of spatial energy sharing is shown in Figure 6.3. Homes are divided into an energy supplier set S and a demander set D according to whether the energy difference is positive or negative. Homes H2 and H4 are in the demander set. Figures 6.3(a) and 6.3(b) show two steps of energy sharing, respectively. Figure 6.3(a) shows charging rate of demanders, discharging rate of suppliers and transmission efficiency between demanders and suppliers. For example, discharging rate of supplier $H1$ r_1^d is 3kW, charging rate of demander $H2$ r_2^c is 2kW, and transmission efficiency from $H1$ to $H2$ is 0.5. Then maximum transmission speed for four homes is calculated and energy-sharing order is determined. Because H2 has highest energy intake speed, H2 will be the first home to do the sharing. The sharing process follows the sharing order of Figure 6.3(a). Figure 6.3(b) shows energy difference of four homes and energy-sharing results.

Then we propose our spatial energy-sharing algorithm based on maximum transmission speed to maximize total energy transmission in a single window. The detail of algorithm is described as follows: The maximum transmission speed for every home is calculated at first. For demander i it is r_i^c, and for suppliers it is obtained from 6.9 by iterating over a list of supplier or demanders sorted by transmission efficiency η_{ij}. We then fetch home i with highest transmission speed for energy sharing (Line 2). If $i \in S$, we start energy sharing with home $j \in D$, which maximizes transmission efficiency η_{ij} (Lines 3-5). Otherwise, we start energy sharing with home $j \in S$, which maximizes transmission efficiency η_{ji} (Lines 6-9). Then energy difference of home i and j will be updated (Line 10). The energy-sharing process will continue until all homes finish energy sharing (Line 11).

The **time complexity** for Algorithm 8 is as follows: To calculate the maximum transmission rate, we need $O(n^2)$ where n is the number of homes. Then it costs $O(nlgn)$ to sort according transmission rate. For the energy-sharing process, the time complexity is at most $O(n^2)$. Note that the sorting of η_{ij} can be done once at a cost of $O(n^2)$, and we will reuse the result in the following algorithms. So all altogether, the time complexity is about

Algorithm 8: Spatial Energy Share Algorithm

Input: Supplier set S and demander set D with homes' r_i^c, r_i^d, $B_i(n)$ and $\triangle E_i(n)$; Transmission efficiency η_{ij}.
Output: Energy sharing results $E_{i \to j}(n)$.

1: Calculate the maximum transmission rate for every home;
2: Fetch home i from $S \bigcup D$ that has the maximum r_i;
3: **if** $i \in S$ **then**
4:     Fetch home j that $\eta_i = \max\limits_{j \in D}(\eta_{ij})$;
5:     $E_{i \to j}(n) = \min(|\triangle E_j(n) + B_i(n)|, |(\triangle E_i(n) + B_i(n)) \cdot \eta_{ij}|, r_j^c \cdot w)$;
6: **else**
7:     Fetch home j that $\eta_i = \max\limits_{j \in D}(\eta_{ji})$;
8:     $E_{j \to i}(n) = \min(|\triangle E_i(n) + B_i(n)|, |(\triangle E_j(n) + B_i(n)) \cdot \eta_{ij}|, r_i^c \cdot w)$;
9: **end if**
10: Update $\triangle E_i(n)$, $\triangle E_j(n)$;
11: Go to Line 2 if $\triangle E_i(n) == 0$, otherwise go to Line 3.

$O(n^2)$.

Remark. Here we give a brief description to demonstrate that Algorithm 8 maximizes amount of energy sharing in a single window. Because we allow only one-to-multiple and multiple-to-one energy transmission, the energy transmission sequences will be like $K = \{s_1, d_2, ..., s_k\}$, s_i and d_j are the only supplier or demander in transmission. Suppose there is a optimal sequence $K^{'} \neq K$ with maximum amount of energy sharing. According to Algorithm 8 s_k has lowest maximum transmission speed, there must be a home $s_i \in K^{'}$, $s_i \notin K$ with higher transmission speed. Then Algorithm 8 does not select s_i but s_k, which contradicts that it always selects the home with highest transmission speed.

6.2.2 Temporal Energy Sharing

With the spatial energy-sharing results, TOU model and battery capacity data, we introduce the temporal energy sharing algorithm in this section. The temporal energy sharing algorithm gathers the energy difference data and makes its charging decision in the current

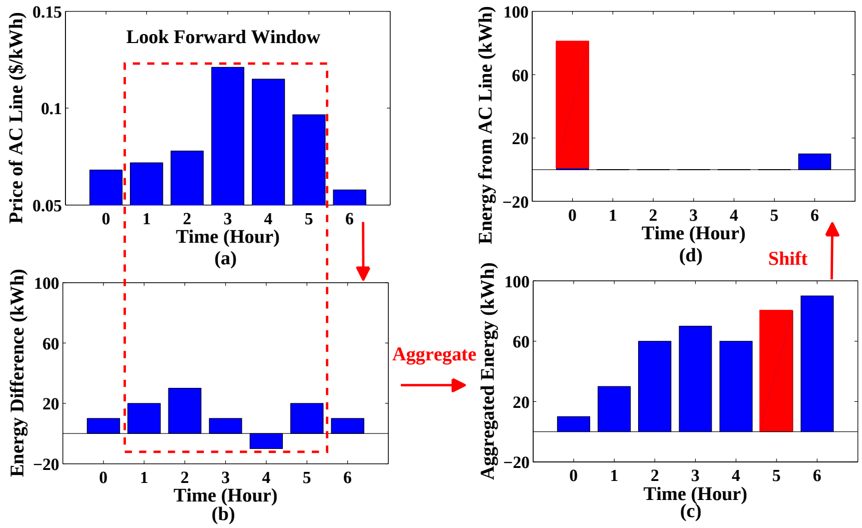


Figure 6.4: An example of using a look forward window to shift an energy request to a low price period to reduce cost: Energy needed from window 1-5 is shifted to window 0.

window to store energy for future higher price window usage. Based on the spatial energy-sharing results, for home i and window n, we can easily calculate the energy difference of i after energy sharing as $\triangle E_i^{'}(n)$:

$$\triangle E_i^{'}(n) = B_i(n) + \triangle E_i(n) + ES_i(n) \tag{6.10}$$

If $\triangle E_i^{'}(n) < 0$, it means home i still does not have enough energy for its usage after energy sharing, then it has to obtain energy from AC line. Then the goal of temporal energy sharing can be formulated as follows:

$$\begin{aligned} \min \quad & \sum_n \Big(Y_{AC}(n) \cdot EA_i(n)\Big) \\ s.t. \quad & 0 \leq B_i(1) \leq C_i && (a) \\ & B_i(n) + \triangle E_i^{'}(n) + EA_i(n) \geq 0 && (b) \\ & EA_i(n) \geq 0 && (c) \end{aligned}$$

To minimize the cost of AC energy over time, the key idea is to charge battery at windows with lower AC price and discharge battery at windows at higher AC price. To anticipate the exact amount of energy needed at window n, it is important to know how many windows to look forward. Therefore, we propose an approach called **Look Forward Window**. The next window that has a lower TOU price is the end of the look forward window, as getting energy at that time can reduce costs more than at the current window.

An example of the look forward window is shown in Figure 6.4. First, it looks forward to the windows in the future until it finds a window whose price of the AC line is lower than the current price (window 6) in (a). Then it aggregates energy differences of homes before that window in (b)-(c). Finally, it shifts all the energy needed from the AC line in the future to the current window. Aggregate energy $AGG_i(k)$ can be calculated by summing up $\triangle E_i^{'}(n)$:

$$AGG_i(k) = \sum_{n \leq l \leq k} \triangle E_i^{'}(l) \tag{6.11}$$

With $AGG_i(k)$, we then find the maximum energy needed between current window and look forward window, which is window 5 in Figure 6.4 (c). Note that $AGG_i(k)$ are normally negative value because homes normally need energy. In Figure 6.4 (c), we show the absolute value of $AGG_i(k)$, which is amount of energy needed. Then we have maximum energy we need to charge for future at window n:

$$E_{max}^{-}(i) = -\max_{k \in [n,m]} (-AGG_i(k)) \tag{6.12}$$

However, we also need to consider two other factors: limited battery capacity and peak energy demands. If the energy obtained from the AC line after shift exceeds the

Algorithm 9: Temporal Energy Sharing Algorithm

Input: Spatial energy share results $E_{i \to j}(t)$ and $\triangle E_i(t)$, $t \in [n, n+24]$;
battery level $B_i(n)$; TOU AC price.
Output: Energy from AC line $EA_i(n)$.

1: **for** each home i **do**
2: Find first window $m(m > n)$ that $Y_{AC}(m) < Y_{AC}(n)$
3: **for** every window t from current window n to m **do**
4: Get the $AGG_i(t)$
5: **end for**
6: Find window k that $E^{-}_{max}(i) = -\max\limits_{k \in [n,m]}(-AGG_i(k))$;
7: $E^{+}_{max}(i) = C_i - \max\limits_{l \in [n,k]}(AGG_i(l))$;
8: Find the number of windows num that have same price with current window
9: $EA_i(n) = min(E^{+}_{max}(i)/num, |E^{-}_{max}(i)|/num, r^c_i \cdot w)$.
10: **end for**

battery capacity, part of the energy will be wasted. Thus, our algorithm will allow the home to obtain energy only from the AC line to fill the battery. Then the maximum energy we can charge to avoid energy waste at window n is as follows:

$$E^{+}_{max}(i) = C_i - \max_{l \in [n,k]}(AGG_i(l)) \tag{6.13}$$

In a TOU model that same price spans over multiple windows; if the system requests all the energy it needs in the first window, there may be a peak energy demand from the AC line. To address this issue, our algorithm will distribute the energy requests evenly among these same price windows, so the energy peaks from the AC line can be smoothed out, which results in minimum impact to the grid.

The detail of the temporal energy-sharing algorithm is shown in Algorithm 9. First, it finds the look forward window m according to the TOU price (Lines 1-2). Then it computes the aggregated energy difference for windows between n and m (Lines 3-5). The maximum energy needed $E^{-}_{max}(i)$ is given (Line 6) and the maximum energy that the homes can charge due to battery capacity is calculated in window n (Line 7). Then it finds the number of windows with same price of AC after window n (Line 8). At last, energy from AC line $EA_i(n)$ for home i is calculated by minimum value specified by constraint (b) in the problem formulation (Lines 9-10).

The **time complexity** for Algorithm 9 is quite straight forward. Given the fact that we look forward at most 24 hours, the loop body in Lines 2-9 can be regarded as a constant time. Then multiplying the number n of homes, we get the total time complexity as $O(n)$.

Theorem 6.1. *Algorithm 2 minimizes cost of AC energy over time for a single home.*

Remark. For a single home, based on Equation (10) and (11), Algorithm 2 enables homes to charge as much energy as possible to battery for future windows. Look forward window ensures that homes only charge battery at low AC price for energy usage at windows with higher AC price. Thus Algorithm 2 minimizes cost of AC energy over time for a single home.

6.2.3 Optimization of AC Allocation

In §6.2.2, we determined an initial amount of energy needed from the AC line. The reason we need to do optimization of energy from the AC line is that spatial sharing and temporal sharing are executed separately, which may cause some unnecessary energy sharing among homes and energy requests from the AC line. In this way, we can reduce the microgrid level energy costs from the AC line.

An example is shown in Figure 6.5. Assume the transmission efficiency between home i and j is 0.5. At the beginning, both i and j have no energy in the battery (shown in

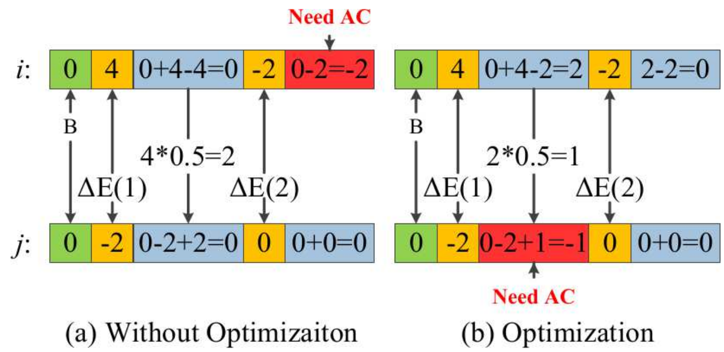


Figure 6.5: An example of optimization of AC allocation

green box); at window 1 and 2, the energy difference (shown in yellow box) between harvested and consumed is calculated for energy sharing (shown in blue box); if after energy sharing, the home still does not have enough energy, it needs to get energy from the AC line (shown in yellow box). Without optimization, home i will first transmit 4kWh energy to home j at window 1 with Algorithm 8. Then home i needs to get 2kWh energy from the AC line at window 2. With Algorithm 9, because the look forward window always has a higher AC price than the current window, home i will get 2kWh energy from the AC line at window 1. With optimization, home i will transmit only 2kWh energy to home j at window 1. Then only home j needs to get 1kWh energy from the AC line at window 1. Overall, this optimization approach can save 1kWh energy from the AC line.

Here we give the condition that we should do optimization of AC allocation for home i.

Lemma 6.2.1. *Optimization of Home i at window n can save the cost of AC energy if* $\exists j$, $E_{i \to j}(n) > 0$, $EA_i(n+1) > 0$ *and* $\eta_{ij} < Y_{AC}(n+1)/Y_{AC}(n)$.

Proof. Before optimization, home i needs $EA_i(n+1)$ energy from AC line and the cost of AC energy is $EA_i(n+1) * Y_{AC}(n+1)$. After optimization, home i does not need energy from AC line, but ; home j needs $EA_i(n+1) \cdot \eta_{ij}$ energy from AC line and cost is $EA_i(n+$

$1)*\eta_{ij}*Y_{AC}(n)$. Then if $\exists j$, $E_{i\to j}(n) > 0$, $EA_i(n+1) > 0$ and $\eta_{ij} < Y_{AC}(n+1)/Y_{AC}(n)$, then we have $EA_i(n+1)*\eta_{ij}*Y_{AC}(n) < EA_i(n+1)*(Y_{AC}(n+1)/Y_{AC}(n))*Y_{AC}(n) = EA_i(n+1)*Y_{AC}(n+1)$. Thus, optimization of home i at window n can save the cost of AC energy. □

With the Lemma 6.2.1, we only need to find all the scenarios that fulfill Lemma 6.2.1 in sharing results from temporal and spatial energy sharing to minimize the electricity cost. The detail of algorithm is described in Algorithm 10. For each home $i \in S$, it checks if it both shares energy with its neighbors and gets energy from the AC line at window n (Lines 1-2). If yes, it finds home $j \in D$ that with the smallest energy transmission efficiency η_{ij} (Line 3). Then it checks if there is energy transmitted from i to j (Line 4). If yes, it cancels energy transmission that causes redundant AC transmission of i (Lines 5-13).

The **time complexity** for Algorithm 10: If we implement the data structure for storing the energy-sharing results properly, either as a dictionary or as a union-set or individual

Algorithm 10: AC Allocation Optimization Algorithm

Input: Energy share and AC transmission results $E_{i\to j}(n)$ and $EA_i(n)$ from Algorithm 8 and 9.
Output: Optimized $E_{i\to j}(n)$ and $EA_i(n)$.

1: **for** each home $i \in S$ **do**
2:   **if** $EA_i(n) > 0 \quad \& \quad \exists j \quad E_{i\to j}(n) \neq 0$ **then**
3:     Find home j that $\min\limits_{j\in D}(\eta_{ij})$
4:     **if** $E_{i\to j}(n) > 0$ & $\eta_{ij} > Y_{AC}(n+1)/Y_{AC}(n)$ **then**
5:       **if** $EA_i(n) > E_{i\to j}(n)$ **then**
6:         $EA_i(n) -= E_{i\to j}(n)$, $E_{i\to j}(n) = 0$;
7:       **else**
8:         $E_{i\to j}(n) -= EA_i(n)$, $EA_i(n) = 0$;
9:       **end if**
10:       remove j from D;
11:     **end if**
12:   **end if**
13: **end for**

lists by using home as key/index, then the checking for whether home i shares energy to other homes just takes constant time. We could reuse the sorting result in the previous algorithm for Line 3, so that Lines 3-10 also have constant time. Then the total time is $O(n)$.

Summary. In the above three sections §6.2.1-§6.2.3, we first balance energy usage of different homes in the microgrid by a spatial energy-sharing algorithm. Then based on a TOU model, energy needed from the AC line is balanced by a temporal energy-sharing algorithm. Finally, combined with two-dimensional energy-sharing results, total AC energy costs in the microgrid are minimized over time. The total time complexity for all the three algorithms is around $O(n^2)$.

Theorem 6.2. *The solution obtained from the above algorithms is a local optimal solution.*

Remark. As in our algorithms, $EA_i(n)$ and $E_{i \to j}(n)$ are determined at every window. According to Constraint (b), only $EA_i(n)$ and $E_{i \to j}(n)$ can be adjusted to reduce the total AC costs. Thus, we can prove that if any $EA_i(n)$ decreases, it will be always more expensive to fulfill Constraints (d). The detailed proof is provided in the Appendix.

6.2.4 Energy-Sharing Price

In order to incentivize energy sharing, we introduce the energy-sharing price in this section. The design of the energy-sharing price needs to take into consideration the energy generation cost and the energy transmission efficiency as well as the TOU price. The net result is that the energy-sharing price will vary between different energy-sharing home pairs and also will vary between the same energy-sharing home pairs in different TOU windows. In our design when a home joins the system, the home shares its basic data with the central controller (price of the battery (Y_B), price of the solar panel (Y_S), size of the battery and energy in it). The system calculates the lower-bound price as follows:

$$Y_{lb} = Y_B * \gamma_B + Y_S * \gamma_S \tag{6.14}$$

where γ_B and γ_S are the depreciation rate of the battery and solar panel respectively. The energy-sharing price should not be higher than the current window's AC price, which is the higher-bound price. As energy transmission causes transmission loss, the energy-sharing price is determined using the lower-bound divided by the energy transmission efficiency η_{ij} between homes i and j and the higher bound as shown in the following equation:

$$SharingPrice = \beta * Y_{lb}/\eta_{ij} + (1 - \beta) * Y_{AC}(n). \tag{6.15}$$

Here β is a parameter to calculate the price of the energy that can vary from 0 to 1. If the value of β is approaching 0, it means the energy-sharing price is closer to the AC price; on the other hand, if the value of β is approaching 1, then it means the current price of the system is closer to the lower bound.

6.2.5 Energy Harvesting and Consumption Prediction

In previous design, we assume accurate future energy consumption and harvesting information are available, which is impossible in reality. In this section, we introduce prediction model of energy consumption and harvesting used in our implementation. Note that more sophisticated models that consider changing weekend activity patterns, weather conditions can be used to improve our work. However, this is not our main contribution and from evaluation results, the following prediction models can already provide enough accuracy for our system.

Harvesting Prediction: We focus on solar energy in this chapter as it is the predominant renewable energy source in residential DG deployment. To predict energy harvesting, a weather forecast based prediction model similar to Sharma's approach ([100]) is adopted. At any time t, based on the sky condition percentage $C(t)$ released by the National Weather Service (NWS), we predict the solar panel's energy harvesting rate $P_i(t)$ as:

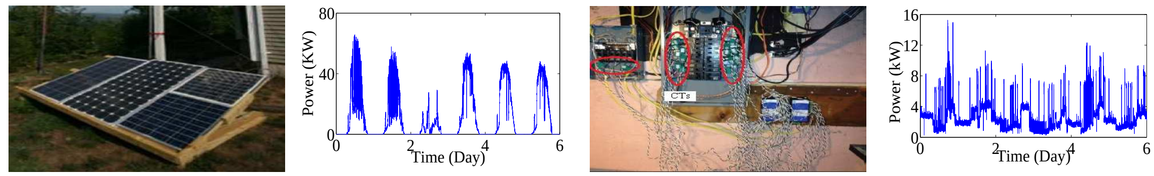
$$P_i(t) = P_{max} \cdot (1 - C(t)) \tag{6.16}$$

where P_{max} is the solar panel's maximum harvesting power. Sharma et al. show that Equation 6.16 provides a more accurate prediction than existing techniques that use the past to predict the future. Thus, based on Equation (6.16), at any time $t = n$, we predict the harvested solar energy within the next energy-sharing window as follows:

$$\widehat{EH}_i(n+1) = \int_n^{n+1} P_i(\tau) d\tau \tag{6.17}$$

Consumption Prediction: To predict the home's consumption from historical consumption data, we use a model based on an Exponentially Weighted Moving Average (EWMA). The EWMA exploits the diurnal nature of a home's consumption, while it also adapts to seasonal variations. Let $EC_i(n)$ denote the amount of energy consumed in $[n, n+1]$ and $\widehat{EC}_i(n+1)$ denote the predicted energy consumed in $[n+1, n+2]$, which is given by:

$$\widehat{EC}_i(n+1) = \alpha \cdot \widehat{EC}_i(n) + (1 - \alpha) \cdot EC_i(n) \tag{6.18}$$



(a) Solar Panel (b) Harvesting Power (c) Energy Measurement (d) Consumption Power

(e) Battery (f) Inverter (g) Battery Measurement

Figure 6.6: Experiment setup

The value of α is chosen by using the method in ([55]), which is dynamically changed based on the observed prediction error of the previous prediction.

6.3 Implementation and Evaluation

In this section, we evaluate the performance of our system. We collect empirical data of (i) energy harvesting from solar panels, (ii) energy consumption from 40 homes, and (iii) charging and discharging power of a battery. We evaluate our system under two types of real world TOU price models in §6.3.3; we also validate that our system can work with homes with similar harvesting and consumption models.

6.3.1 Experiment Setup

We collect energy-harvesting data from solar panels. The solar panels we use are Grape Solar 75-Watt Monocrystalline PV Solar Panels (shown in Figure 6.6(a)). We collect six days' energy-harvesting data shown in Figure 6.6(b). In a day, the solar panel begins to

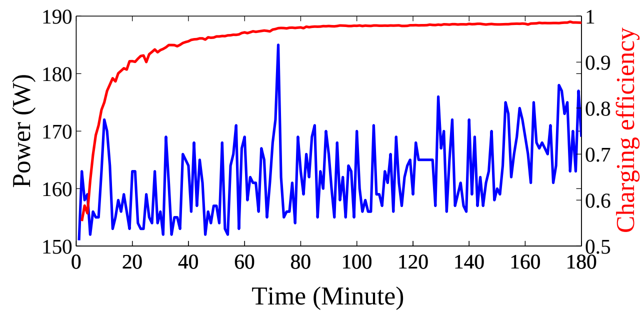


Figure 6.7: Battery Charging

harvest energy at around 7 a.m., the energy peaks around 12 p.m., and the harvesting ends around 8 p.m. However, the harvested energy on different days varies, which may be due to the varying weather conditions. Because the energy-harvesting pattern from solar panels is similar in a single area, we use the trace to produce energy-harvesting data of other homes with some randomness. Harvesting data is collected hourly. The weather forecast data we use is from the NWS (National Weather Service). The consumption data of homes consist of energy information collected every minute over six days. With empirical data, we calculate the predicted energy harvesting and consumption data over six days for our simulations.

We also collect the energy consumption data of 40 homes ([38]). We add current transducers (CTs) around each leg of a home's split-phase input power from the grid (shown in Figure 6.6(c)) to monitor all the circuits inside a home every second. Figure 6.6(d) shows the aggregated energy consumption data within six days in a deployed home.

The energy storage unit we deploy is UB12100-S Universal Battery and Xantrex PowerHub 84053 shown in Figure 6.6(e) and 6.6(f), which is a combination of an inverter/charger module capable of delivering up to 1800 watts of household power. It can work as a backup power solution to operate with solar inputs. We use iMeter Solo (an IN-

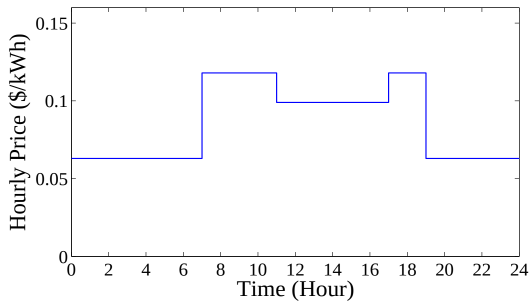
STEON power meter) to measure the battery energy charging and discharging rate in real time (shown in Figure 6.6(g)). The power consumption for charging a battery is shown in Figure 6.7. The average power for charging the battery is around 160W, which implies that within a one-hour window, only a limited amount of energy can be transmitted. Therefore, our design addresses the challenge of the limited energy transmission speed in § 6.2.1. We also verify the charging efficiency of battery. At the beginning of charging, the efficiency is relatively low. However, efficiency increases quickly with time and after 30 minutes, the efficiency is more than 95%.

6.3.2 Evaluation Baseline

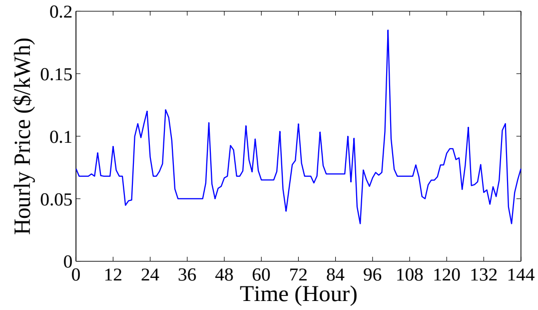
To verify the efficiency of our system, we compare our design, which is referred to as **GSC (Global Sharing and Charging)** in latter evaluation results, with (i) **Oracle**, which uses the same energy charging and sharing algorithm as GSC but assumes real energy consumption and harvesting data in the future is available; (ii) **Individual smart charge(ISC)** ([81]), which only allows homes to take advantage of TOU individually with no energy sharing; and (iii) **Collective sharing (GES)** ([129]), which aims to share energy among homes, but not take advantage of TOU.

6.3.3 Evaluation Results

In this section, we will evaluate the effectiveness of our system, which includes the efficiency of our system under two kinds of TOU models. All results are simulated with the six days' empirical data of energy harvesting and consumption introduced in Section 6.3.1.



(a) TOU in Ontario



(b) TOU in New England

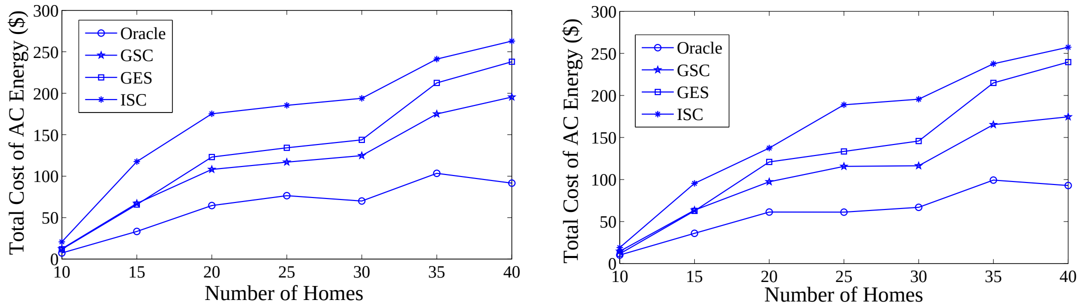
Figure 6.8: Different TOU models

The battery loss rate we use is 15% ([99]); the average AC and DC transmission loss rate is around 22.6% and 7.6%, which varies with different distances among homes ([62]).

Different TOU Models: We ran our system under two different TOU models: TOU in Ontario and TOU in New England, as shown in Figures 6.8(a) and 6.8(b), respectively. These two models are carefully selected to represent a wide range of TOU models: (i) a higher price for daytime use per day (shown in Figure 6.8(a) for TOU in Ontario), and (ii) price dynamic changes every hour based on demand, which is also referred to as Real-time Pricing Model (RTP) in other papers (shown in Figure 6.8(b) for TOU in New England).

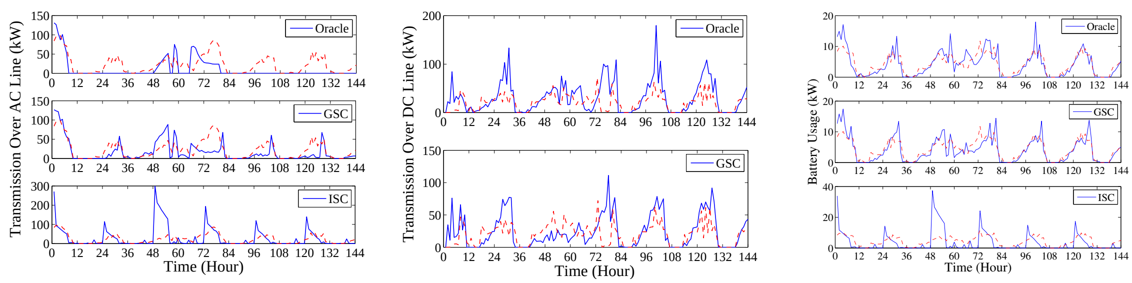
Total Cost of AC Energy: Figure 6.9(a) shows total cost of AC line for four different algorithms under TOU in Ontario. In all four algorithms, the total cost of the AC line generally increases with the number of homes, which is quite obvious. However, in Oracle, the total cost decreases when the number of homes increases from 25 to 30 and 35 to 40, which is due to those five additional homes having more energy surplus. Because the other algorithms do not have accurate energy information, prediction error causes the increase in total cost. Our algorithm outperforms GES by 22% and is less than Oracle by only 9.2%. ISC performs worst, which shows the importance of energy sharing among homes.

Figure 6.9(b) shows total cost of AC line for four different algorithms under TOU



(a) TOU in Ontario (b) TOU in New England

Figure 6.9: Total AC energy cost of different TOU models



(a) AC Transmission (b) DC Transmission (c) Battery Usage

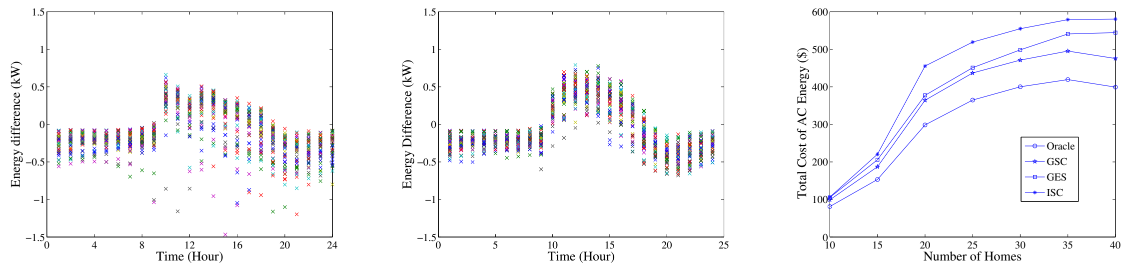
Figure 6.10: Transmission over AC, DC line and battery usage in Ontario (red dashed line in all figures is the transmission or battery usage of GES)

in New England. Similar to the previous TOU model, the total cost of the AC line generally also increases with the number of homes. Our algorithm outperforms GES by 37.9% and is less than Oracle by only 6.6%, which is even better than the TOU model in Ontario. The main reason for the better performance is that with the higher dynamic of the AC price, our temporal-sharing algorithm can take advantage of the TOU model more efficiently. Because the AC price changes vary frequently, the looking forward window could be relatively small, which does not need prediction for a long period. Thus, our algorithm is closer to Oracle.

Transmission Over AC Line: We also show the detailed energy transmission over the AC line per hour under TOU in Ontario in Figure 6.10(a). All three energy-sharing algorithms are compared to GES. For Oracle, homes seldom need energy from the AC line except when the harvested energy from a solar panel is not enough in day 3 (Hour 48 to 72). Also, because the initial battery is not fully charged, homes need to get energy from the AC at the beginning. Our algorithm is close to Oracle, in which for nearly 10 hours of one day, homes do not need to obtain energy from the AC line. However, homes need to get more energy from the AC line with ISC. For some particular time, transmission over the AC line has large peaks. That is because homes take advantage of TOU individually. If the price of the AC line is the same for every home, then all homes will try to charge at the time with the lowest TOU price. Because energy information of all homes can be achieved with GSC, central controller can simply avoid this phenomenon to reduce the peak of AC line.

Transmission Over DC Line: Transmission over the DC per hour under TOU in Ontario is shown in Figure 6.10(b). Oracle and our algorithm share more energy among homes to reduce AC energy cost when the price of the AC line is relatively high. Thus, their transmission of DC would be higher. Because ISC does not allow energy sharing among homes, there is no transmission over DC line.

Battery Charging and Discharging: Battery usage includes battery charging from AC or DC line and discharging to DC line and appliances' usage. Battery usage per hour under TOU in Ontario is shown in Figure 6.10(c). Even Oracle and our algorithm share more energy through DC line, the total battery usage of Oracle, GSC and GES is close. That means the main difference among three algorithms is the way of utilizing TOU and renewable energy but not battery. Because ISC does not allow energy sharing among homes, battery usage is only for charging from AC line and discharging to appliances' usage. Thus the curves of battery usage and AC transmission in ISC are similar. The peak demand from AC line is mainly to charge energy to battery, but not for current appliances' usage.



(a) Homes with different consumption patterns (b) Homes with similar consumption patterns (c) Cost of AC energy

Figure 6.11: Cost of AC energy with similar energy consumption pattern

Summary of Different TOU Models: We have evaluated our system under two TOU models. Due to the limited space, we only show AC, DC transmission and battery usage under Ontario TOU model while the results under New England TOU model is similar. In all these scenarios, our system outperforms GES and ISC. The key observations are as follows: (i) Oracle and GSC need less AC transmission and cost compared to GES and ISC; and (ii) Oracle and GSC may cause some little peaks of AC transmission if periods with lowest price of AC are short. However, peaks of Oracle and GSC are much lower than ISC.

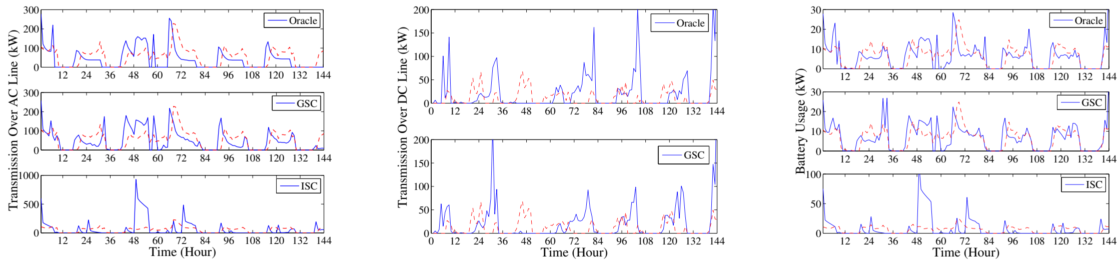
6.3.4 Advanced Evaluation Results

Because our system is designed for different environments, such as different consumption patterns of homes, it is crucial to investigate the system's behavior and sensitivity under diverse settings.

Impact of Similar Consumption Pattern

The consumption data we used in the above simulations come from 40 different homes with different patterns, which provides opportunity to balance energy transmission over the AC line by consuming energy at different times. In this section, we use data of 40 homes that have similar consumption pattern to validate that our system still works under this scenario. Energy differences of homes over time is shown in Figure 6.11(a) and 6.11(b). The upper figure displays energy differences of homes which are used for previous section. And the lower figure displays energy differences of homes with similar consumption pattern. In general, both two microgrids need energy in early morning; have surplus energy at noon and need energy in evening. However, in microgrid 1, homes have different energy difference at noon while homes nearly share same energy difference in microgrid 2. The variances of energy differences of two microgrid are 0.0448 and 0.0017 which also shows homes in microgrid 2 have more similar energy consumption pattern.

Figure 6.11(c) shows the total cost of the AC line for four algorithms. When homes have similar consumption pattern, it is hard to find homes to share energy since they may also need energy. Thus, the gap among the four algorithms is much smaller especially when number of homes are small. However, similar to previous results, Oracle and GSC are still better than GES and ISC, which means that our system can still work well under homes with similar energy consumption pattern. We also show the detailed energy transmission over AC line per hour in Figure 6.12(a). All three energy sharing algorithms are compared to GES. Because we use TOU in Ontario, Oracle and GSC don't have peaks of AC transmission. However, ISC produce huge peak (929.7kW). That is because when the consumption pattern of homes are similar, homes will decide to charge from AC at similar period and cause the peak. Note that in the simulation, we assume there is no transmission limit for distribution network. In reality, peak demand of 929.7kW may cause blackout in the community. The detailed energy transmission over DC line and battery usage is also



(a) Transmission over AC line (b) Transmission over DC line (c) Battery Usage

Figure 6.12: Transmission over AC, DC line and battery usage with similar energy consumption pattern

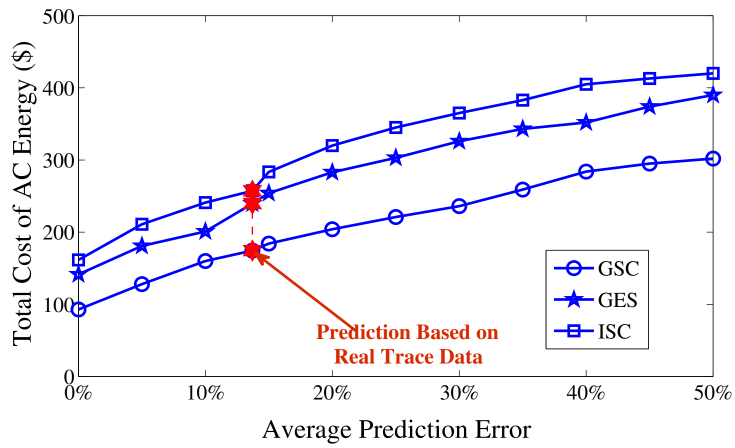


Figure 6.13: Total cost for different average prediction error

shown in Figure 6.12(b) and 6.12(c). Similar results are obtained compared to homes with different consumption pattern, which means our design also works well for homes with similar consumption patterns.

Impact of Prediction Accuracy

In our previous section, we make use of weather forecast and energy consumption history to predict future energy differences. However, prediction accuracy can vary under different environments. Thus in this section, we evaluate our system with different prediction accuracy. Since it is difficult to attain prediction accuracy of different environments, we artificially generate prediction results with different prediction errors. And the prediction

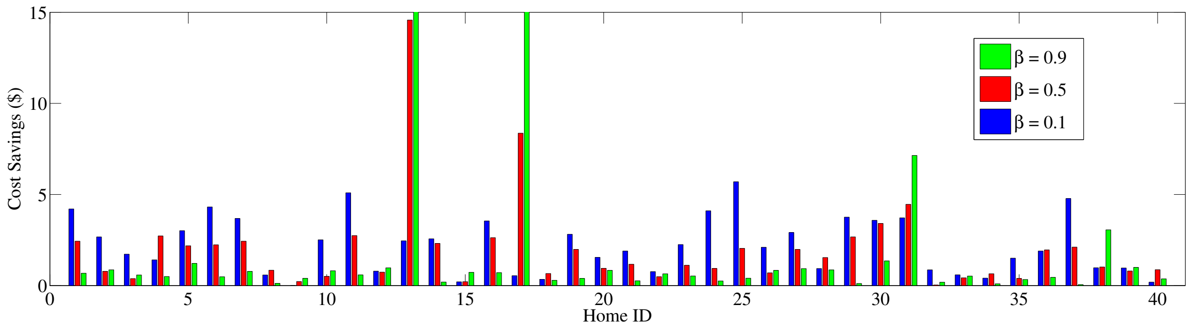


Figure 6.14: Cost savings of different price models for 7 days

β	0.1	0.3	0.5	0.7	0.9
Average Saving ($)	1.65	1.65	1.66	1.65	1.64
RSD of Saving	0.51	0.70	0.93	1.24	1.46

Table 6.2: RSD of Energy Savings

errors follows normal distribution. The detailed results are shown in Figure 6.13. The red stars are results with real trace data with 13.7% average prediction error of all homes at all time. Prediction error is defined as $|\triangle E_i(n) - \triangle \widehat{E}_i(n)|/\triangle E_i(n)$. When GSC with 0% prediction error, it is the same as Oracle. In Figure 6.13, total cost of AC energy increases with larger prediction error. This is because that with higher prediction error, homes may share more energy to others, and then they do not have enough energy for their own usage and have to obtain energy from AC line. However, with different prediction error, our proposed GSC is always better than GES and ISC.

Impact of Price Model

As introduced in §6.2.4, we use β to control price model. In this section, we investigate the impact of price model with different β.

Figure 6.14 shows the cost savings per home of different price models. Cost savings of a home is calculated by AC cost with ISC minus AC cost with GSC. With higher β, cost savings of homes varies more significantly, which may reduce the incentives for homes to join in. We also show statics of cost savings in Table 6.2. The average energy saving for homes are nearly same with different β. With higher β, Relative Standard Deviation (RSD) of energy saving is higher. The main reason is when β is higher, sharing price is closer to lower bound. Then the homes needs more energy can benefit more from low price energy. Thus, to achieve greater fairness in system, the sharing price should be close to real time price of electricity from AC line. However, sharing price can not exceed the price of AC line, otherwise homes have no incentives to get energy from community.

6.3.5 Summary of Evaluation

With the simulations results, we easily find our system can work under different TOU models and homes with similar consumption patterns. We also investigate impact of price models and find the sharing price should be close, but not exceed the price of AC line.

6.4 Cost-Benefit Analysis

The previous section shows that our system can reduce AC energy cost of the whole microgrid by more than 20%. In this section, we discuss our system's return on investment.

TOU	Ontario			New England		
Algorithm	Oracle	GSC	GES	Oracle	GSC	GES
Cost ($\$10^5$)	3.51			3.51		
Benefit ($\$10^5/yr$)	0.79	0.71	0.59	0.77	0.72	0.52
Years for Return	4.44	4.94	5.95	4.56	4.86	6.75

Table 6.3: Cost and Benefit

In many instances, homes already have the necessary infrastructure to implement energy sharing. More and more homes will be equipped with solar panels and batteries to generate renewable energy. To implement energy sharing microgrid, the main expense is to construct lines for distribution network, use solar panels and a larger battery to harvest and store energy. For the battery, the price is around $\$200/$kWh. For solar panels, the price is around $\$0.6/$Watt. The price of other equipment, such as inverter, cabling and energy monitor, is also included in total investment cost. Finally, we estimate two weeks' labor at $4000 for installation. The benefit realized with our system design is mainly due to the savings of energy transmission over AC lines. We use two types of empirical TOU price models (i.e., Ontario and New England). Based on the above pricing data, our analysis of benefit and cost is shown in Table 6.3. In general, our system can return the investment in less than five years. We note that the AC price is based on current TOU prices. Given the increase in electricity price, we expect that the number of years for return will be even fewer.

Centralized versus Distributed. Our current system design is a centralized control and a centralized cluster controller that needs to collect energy harvesting and consumption information of all the homes in a cluster. However, because the number of homes in a cluster is limited, the computation and storage consumption would not be too high. After the energy sharing in a cluster, the cluster controllers need only to send the energy information of homes that still have energy surplus or shortage to a higher layer controller. Thus, the total computation and storage cost is under control. However, we also plan to develop distributed control in future to allow homes to collect information from their neighbors for energy sharing to further reduce the computation and storage cost of controllers.

6.5 Related Work

Our work is related to three areas of previous work: energy harvesting and energy efficient systems and building energy.

• **Energy harvesting.** The renewable energy sources have become an alternative way to consume power and reduce electricity bills. However, they have limits in some instances when harvested energy availability typically varies with time in a non-deterministic manner and power systems surpass the consumption or vice-versa, which results in a mismatch [65]. To manage renewable energy, Deborah et al. [89] propose a method to exploit robotic mobility by having energy producers be mobile nodes. In [52], the authors designed perpetual environmentally powered sensor networks. Our work follows the simple idea where we build an energy sharing microgrid system in an entire community to reduce the AC energy cost, which uses the energy sensing data and market-based TOU price to decide when and who to share energy with.

• **Energy-efficient systems.** Our work is also related to energy-efficient systems [12] [**yi2014renewable**]. In energy efficient systems, researchers mainly focus on i) energy management in data centers [44] and leveraging renewable energy with carbon-aware in data centers [34], (ii) developing models to balance performance measures and energy consumption in wireless networks [36]; iii) energy management in web search by understanding the query complexity and its implications for energy-efficient web search [91]; (iv) mobile devices by empowering developers to estimate app energy, end to end energy management [83]; (v) energy-aware dispatching of parallel queues, efficient virtual machine scheduling in computer architecture [74].

• **Building energy.** This research mainly focuses on (i) energy auditing [53] and design of control algorithms to reduce energy consumption inside a single building [50]; (ii) reducing the energy usage of building-wide heating, energy-efficient building automation, ventilation, and air conditioning [70]; (iii) investigation on the integration of renewable energy into power grid [131] [130]; and (iv) applying stochastic network calculus to analyze the power supply reliability with various renewable energy configurations and store that energy into very large scale batteries [82]; and v) taking model predictive control approach to schedule the workload to reduce the energy cost in the buildings [128]. Our work takes a different approach to reduce energy cost by sharing the renewable energy. Unlike these other approaches, our work opens up new approach where energy can be gained efficiently and used smartly.

• **Economics and network communication.** Allocation optimization and fair allocation mechanisms are important factors for workload scheduling. Many complex and stochastic approaches have been proposed in economics and network communication area, where allocation optimization approach is used for flow control to optimize a global measure of network performance [15] [112]. [32] investigates a class of pricing mechanisms that both induce deep customer participation and enable efficient management of their end-use devices. Our approach is built on previous approaches, where the energy-sharing price reflects the supply-demand relationship and investment of each home.

Our work is built on previous works, but homes with renewable devices and small batteries are the main research focus. Most related work is [129], which tries to minimize energy transmission loss in microgrid. In this paper, we propose a holistic approach to minimize community AC cost under different TOU models. Specifically, we designed spatial and temporal energy sharing algorithms, and developed optimal AC allocation algorithm to minimize electricity cost.

6.6 Conclusion

In this chapter, we attempt to investigate how to minimize AC energy costs in a sustainable microgrid under different market-based TOU price models by exploring three types of energy sensing data: (i) sensing data of solar panels' energy-harvesting rate; (ii) sensing data of individual homes' energy consumption rate; and (iii) sensing data of battery charging and discharging patterns. Specifically, we build an energy-sharing microgrid system, which decides the energy-sharing home pairs and when to share energy based on the sensing data and market-based TOU price so that AC energy costs in the whole microgrid are minimized. We evaluate our system using empirical traces of harvested solar energy and home energy consumption. Through extensive simulations, we verify that our system can reduce AC energy costs of the whole microgrid by more than 20% under different TOU price models and can still reduce AC energy costs even when homes have similar energy consumption patterns.

CHAPTER 7

DEMAND AND GENERATION SCHEDULING IN MICROGRIDS

Given the limited capacities of local energy generation and storage in such a community, it is extremely challenging for an isolated microgrid to balance the power demand and generation in real-time with dynamically changing energy demand. Meanwhile, more and more sensing devices (such as smart meters) are deployed in individual homes to monitor real-time energy data, which can be helpful for homes and microgrid to better schedule the workload and generation. However, it is still difficult to conduct real-time distributed control due to the unreliable sensing devices and communications between sensing devices and controllers. To address these issues in microgrids, we designed a novel approach for the system to conduct both demand scheduling in residential homes and generation scheduling to minimize the total operation cost in microgrids.

7.1 Introduction

Microgrids play a important role in energy cyber-physical systems [73]. In a typical microgrid, it consists of local generators and energy storage (e.g., batteries) to provide power for a small community with commercial and residential buildings. Microgrids can provide power to places i) where the traditional power grid does not exist due to the poor economy or limited number of residences (e.g., islands); and ii) when the traditional power grid is temporarily not functioning due to severe weather conditions (e.g., storms). Therefore, microgrids have gained increasing attention recently [107]. Due to the very limited capacities of local energy storage and energy generation, microgrids are more difficult to maintain than traditional power grids. To ensure the stability and reliability of a microgrid, we need to conduct real-time scheduling and control the operations of local generators, batteries, and controllable workloads of appliances to offset the dynamically changing power demands of uncontrollable appliances.

Therefore, it is extremely important to collect the power consumption and generation data for distributed control in real-time, which becomes possible with the recent rapid development of smart meters. However, the sensors in smart meters and wireless communications between smart meters and controllers are not 100% reliable. The microgrid controller may encounter missing data and delayed data when conducting the scheduling for generators and distributed control operations in each home. Furthermore, to cope with the dynamically changing demand, the microgrid controller needs information to predict the future power demand to decide the operations of local generators. Meanwhile, each home also needs to predict its own future demand to schedule workloads. The existing techniques on energy consumption forecasting are mainly for long term offline forecast for large generators [11, 45]. However, to realize real-time distributed control in a microgrid, real-time data processing and short term prediction is needed. In this chapter, we propose a novel data management technique for distributed control in a microgrid to process the received data, reconstruct the missing data, and predict the future power demand with existing data. The key idea is to utilize the correlation between power, voltage and frequency data of different homes in the microgrid. Then, the misses or delayed data can be reconstructed with a portion of received data or data from other homes. Because energy consumption patterns in one home are limited due to the limited number of appliances, it provides us opportunity to reconstruct the missing data and predict future data in a short term based on existing data and detected energy consumption patterns. With the reconstructed and predicted data, the controller decides the scheduling of workloads in each home and the operations of generators to maintain the stability of the microgrid.

While the workload is scheduled in each home to avoid power failures in the microgrid, the behaviors and comfort of users should not be affected. Therefore, we choose the flexible and controllable workloads of appliances for scheduling. For example, water heaters are flexible and controllable loads because we only need to make sure that there is enough hot water in water heaters when people use hot water. Our approach can be easily extended to support any other types of flexible and controllable workload (e.g., HVAC). For the local generator's scheduling, we adopt a widely used generator model and propose an optimal algorithm to minimize the operational cost. Specifically, we summarize our major contributions as follows:

- We conducted a systematic investigation on the correlation between power, volt-

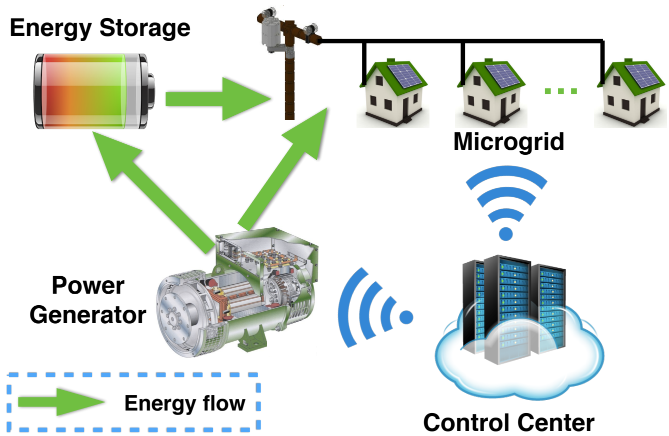


Figure 7.1: Architecture of a microgrid

age, and frequency in a microgrid and developed a holistic sets of correlations models (i.e., power-voltage, frequency-voltage, temporal, and spatial correlation). Through extensive experiments and simulations, we show that our design can recover the missing data with more than 99% accuracy for the short term prediction.

• To reduce the operational cost of isolated microgrids, we present holistic real-time scheduling algorithms for both local generators and controllable loads in individual homes even when there exists communication failures between control center and individual homes.

• Utilizing the empirical energy consumption data from 100 residential homes, we conducted extensive simulations. The results indicate that our proposed distributed control can reliably balance power demand and generation in real-time and reduce operational cost by 23%.

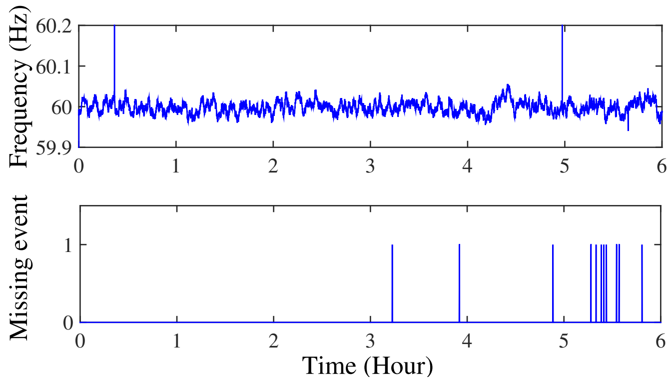


Figure 7.2: Examples of data faults in energy monitoring

7.2 Background and Motivation

A microgrid is a distributed electric power system that can autonomously coordinate local generations and demands in a dynamic manner [63]. Microgrids can operate in either grid-connected mode or isolated mode, some of which are now deployed in the US, Japan and European countries [79].

Background. In this chapter, we consider a modern microgrid, illustrated in Figure 7.1, consists of generation technology (e.g., local electricity generators) and batteries. To ensure compatibility with the traditional power grid, we adopt the microgrid architecture, which is similar to the one used in a traditional power grid. If the microgrid is built from nothing (e.g., island, where there is no electricity grid before), the microgrid can be built the same architecture as traditional grid with a distribution network across the community of homes. If the microgrid is built from a traditional grid, we only need to add local generators, batteries and a control center into the microgrid. Within the microgrid, sensors are deployed in each home to collect and send energy related data (e.g., power, voltage and frequency) to the control center. The control center decides the workload scheduling in each home and the generations to balance the power demand and supply.

Motivation. To realize the real-time control, it is very important to collect the energy related data from homes and send back the control instructions in real-time. However, based on our more than 6 years' experiences of energy monitoring in residential homes, the data collection from homes may suffer from different types of faults: i) data point missing; ii) sensing error; iii) communication delay; and iv) communication loss. The first two faults are caused by the low reliability of sensors due to the long-term monitoring. The latter two faults are caused by unreliable wireless communication. We show some examples of faults in Figure 7.2. The first one is caused by sensing errors, which generate peaks but do not happen very frequently (we observe average 1.5 seconds sensing error in 12 hours). The second one is whether we receive readings in the cloud server. The Y-axis value is set to be 1 if there is a data missing event. We can see the missing events are very bursty, which means once we have a missing event, there will be high probability there would be missing events in near future.

With the demand of real-time data collection and reality of multiple different faults in monitoring, it is crucial to manage the real-time collected data and reconstruct the missing data for real-time control in a self-sustainable microgrid.

7.3 System Overview

To ensure the reliability of the microgrid, we propose the system design as shown in Figure 7.3, which includes three main components: data management, central scheduler, and local scheduler. In summary, our system works as follows: i) power meters deployed in homes monitor the power consumption, voltage and frequency in the power line, then send collected data to control center; ii) control center receives the collected data from homes and processes the data for missing data reconstruction and future data prediction; iii) the central scheduler decides the control instructions for each home and generators based on the processed data; iv) individual homes and the power generator execute the instructions from central scheduler if control instructions are received; v) if control instructions are not received by the individual homes and the power generator, local scheduler will conduct local control in these homes and the power generator will maintain the same amount of power generation.

Data Management. Due to the sensing errors and unreliable communication, it is highly possible we will miss important energy data from sensors. To reconstruct and predict the sensing data, we investigate the correlation models among energy data for recovery. The received data will be used both for data reconstruction and update for correlation model. Specifically, we investigate i) correlation between power and voltage for homes under the same transformer; ii) correlation between frequency and voltage at individual homes; iii) temporal and spatial correlation for power consumption of all homes. Based on these correlations, the missing data can be reconstructed and future data can be predicted for real-time control.

Central and Local Scheduler. Based on recovered and predicted data, the central scheduler decides control instructions for both controllable workload in homes and the generation from the power generator. The key idea of workload control in each home is to turn on some appliances when power consumption is low and turn off some appliances when power consumption is high. In this chapter, we use the workload of the water heater as an example because its workload is flexible and it is commonly installed in residential homes. Our design can easily be extended to support the scheduling of HVAC systems. The gen-

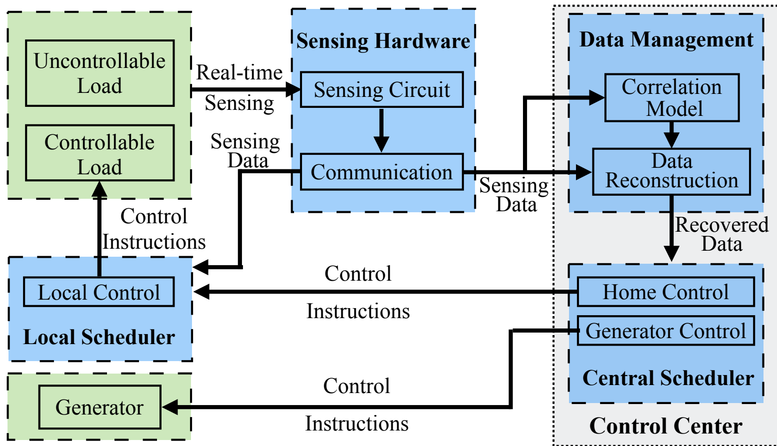


Figure 7.3: System overview

eration control is to decide the power supply from generators. Because the power supply of generators cannot be changed as fast as workload at homes, we schedule the generators based on prediction of future power demand in the microgrid. Batteries are used as buffer to offset the prediction errors of the future power demand. Due to the unreliable communication, control instructions may not be received at the local home or generator, then the local scheduler conducts local control based on the local sensing data.

7.4 Data Management

To reconstruct and predict the sensing data, in this section, we introduce four correlation models for reconstructing the missing data and predicting the future data. While most of existing works focus on missing data reconstruction of a single time series data [69, 124], we investigate the correlations among multiple time series data (power consumption from multiple homes, voltage and frequency) in microgrids and utilize the correlation models to reconstruct the missing data.

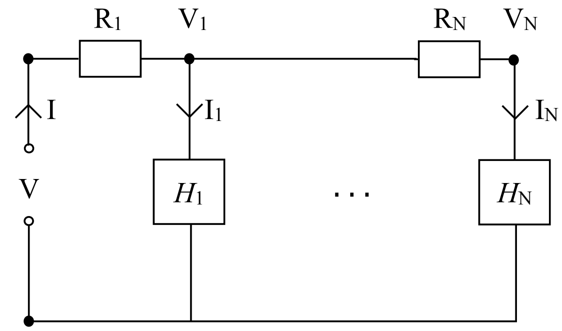


Figure 7.4: The topology of homes and the transformer

7.4.1 Correlation Models

The most important part of data management is to build and utilize the correlation models among the collected data. Specifically, we identified and built four correlation models: i) correlation between power and voltage; ii) correlation between frequency and voltage; iii) temporal correlation of power consumption data in a single home; and iv) spatial correlation between power consumption data from multiple homes.

Power Voltage Correlation

Without loss of generality, we assume that N homes are connected under the same transformer (shown in Figure 7.4). According to the Electric Power Distribution Handbook [103], a transformer can be considered as a constant kVA device for a voltage from 100% to 105%. If the power consumption of one home increases, the total current I increases and voltage V drops. We find that home H_{i-1}'s voltage value depends on i) the transformer's output voltage (V); ii) the current from the transformer to H_{i-1}; and iii) resistances of the power line from the transformer to H_{i-1}. For example, H_1's voltage value only depends on the transformer's output voltage (V), the current (I_1) through H_1, and the resistance (i.e., R_1). Based on the above analysis, the voltage values at homes H_{i-1} and H_i can be

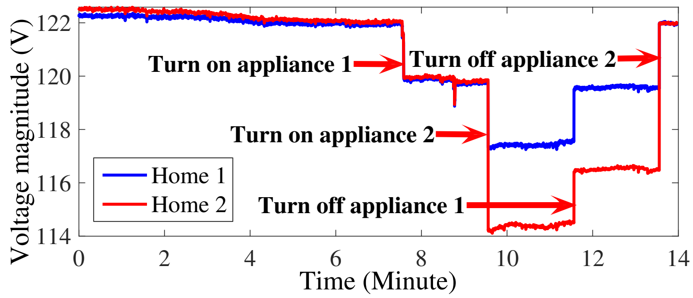


Figure 7.5: Relationship between power and voltage

calculated by using Equation (7.1).

$$V_{i-1} = V - \sum_{j=1}^{i} \sum_{k=j}^{N} I_{k-1} R_{j-1} \quad i = 1, 2, ..., N \tag{7.1}$$

To verify it, we conduct experiments with 2 homes under the same transformer and keep power consumption at home 2 stable to study the power voltage relationship. The measured voltage in both homes are both related to the power consumption in home 1 (shown in Figure 7.5). Thus, based on Equation (7.1) and evaluation results, the voltage drop from transformer to each home is in linear relationship of currents going through the power line.

Frequency Voltage Relationship

A typical microgrid may contain multiple transformers. Therefore, it is also important to investigate the other features in a microgrid. According to the Electric Power Distribution Handbook [103], frequency is a good indicator of the relationship between power supply and demand. If the power demand surpasses the power supply, then the frequency decreases because the generator can not generate enough power. Thus, the frequency should be related to the total power consumption of the microgrid. Because voltage is related to power supply and demand, thus, the frequency value has a linear relationship with the voltage

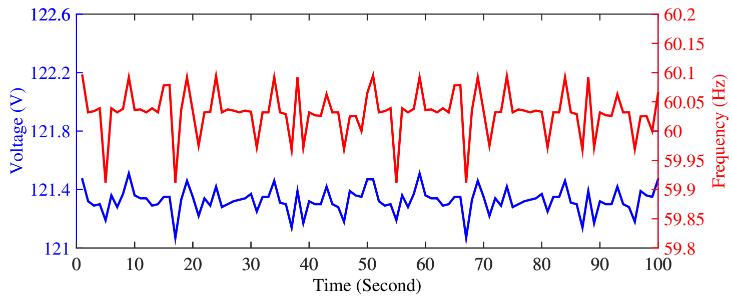


Figure 7.6: Relationship between frequency and voltage

value. This relationship can be modeled as follows:

$$\triangle F = \triangle V * \lambda_1 \tag{7.2}$$

To verify it, we conduct experiments with 3 homes with one month data. Two of them are under the same transformer while the other home is under a different transformer. To make the relationship clear to see, a typical example of the measured frequency and voltage relationship is shown in Figure 7.6. The frequency value is well synchronized with the voltage value.

Based on Equation (7.2) and experimental results, the frequency change in each home is in linear relationship of voltage change. Therefore, we can utilize the frequency voltage relationship to recover the missing data.

Temporal Correlation

In a microgrid, we need to reconstruct the missing data and predict the power consumption in the very near future (e.g., next 1 second) for real-time control. Thus the power consumption data of yesterday or last month is much less useful. To address this problem, we leverage the power consumption signatures of appliances to reconstruct the missing data and predict the short term power consumption. This is because there are limited number of power consumption signatures for different loads. To evaluate this approach, we use the empirical data collected for 2 years from a home to investigate the temporal correlation between power consumption data. We run a power consumption signature detection algorithm on the data set to find the signatures. Specifically, the similarity between two vectors is calculated by using a Euclidean distance-based function as shown below:

$$\rho_{i,j} = \frac{1}{|S_i|} \sum_{t=1}^{|S_i|} (S_i(t) - S_j(t))^2 \tag{7.3}$$

$|S_i|$ is the length of signature S_i. If the distance of these two vectors is small, then the similarity of two vectors is high. Then we go through the data set to find the possible signatures. To simplify the algorithm, we use a fixed length of energy consumption patterns which achieves very effective results (detailed in Section 7.6). As shown in Algorithm 11, the signature set S is empty initially. When $t < T$, we calculate the similarity between power consumption data and signatures of each appliances based on Equation (7.3). If we find the similarity between current power consumption and existing signatures is higher than the current maximum similarity, we reassign the maximum similarity and mark $index = i$. Then we compare the maximum similarity we find to the threshold of the minimum similarity ρ_{min}. If $\rho_{max} > \rho_{min}$, we then detect a new signature S_{new}, add it to the signature set S and update $t = t + |S_{new}|$. Otherwise, we update $t = t + 1$ and continue the detection process.

Based on the detection signatures, we can reconstruct the missing data and predict the near future data as:

Algorithm 11: Signatures Detection Algorithm

$S = \emptyset$;
while $t < T$ **do**
  $\rho_{max} = 0, index = -1$;
  for detected signature S_i **do**
    Calculate $\rho_i(t)$ based on Equation (7.3);
    if $\rho_i(t) < \rho_{max}$ **then**
      $\rho_{max} = \rho_i(t), index = i$,
    end if
  end for
  if $\rho_{max} > \rho_{min}$ **then**
    Detect a new signature S_{new} and add to S;
    $t = t + |S_i|$;
  else
    $t = t + 1$;
  end if
end while

$$P_i(t+k) = S_j(|S_j|/2 + k) + \sum_{x=1}^{|S_j|/2} \frac{2 \cdot (P_i(t + x - |S_j|) - S_j(x))}{|S_j|} \tag{7.4}$$

where S_j is the detected signature that has shortest distance to $\{P_i(t-l(S_j)+1), \cdots, P_i(t)\}$.

Spatial Correlation

Because homes in the same area may have the similar power consumption pattern, we can use the spatial correlation among power consumption of homes for power consumption prediction. However, different homes will have different correlations at different time. Thus, we need to keep on updating the correlations among these homes for prediction. To evaluate our idea, we use empirical power consumption data collected from 100 homes for one month to investigate the spatial correlation of power consumption among different homes. The spatial correlation between homes 1 and 3, and homes 2 and 3 is shown in Figure 7.7. X-axis is the time, Y-axis is the correlation between either homes 1 and 3 or

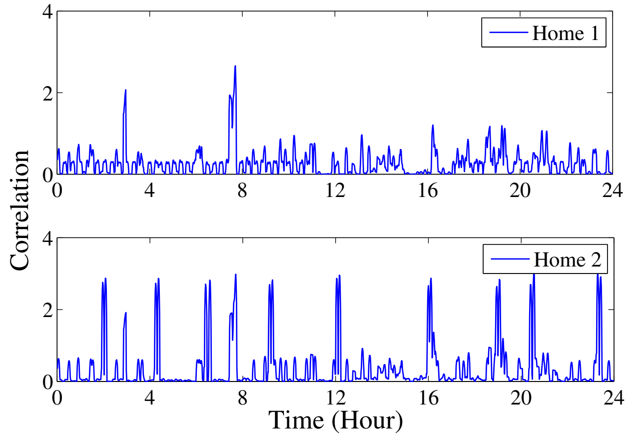


Figure 7.7: Spatial correlation of different homes over time

homes 2 and 3. For most of the time, home 1 and home 2 are quite similar to home 3. However, from hour 2 to hour 4, home 1 is closer to home 3 while from hour 4 to 6, home 2 is closer to home 3. Thus, we build a model to predict the power consumption based on historical correlations among homes. The correlation between two homes can be calculated as:

$$c_{ij}(t) = \frac{1}{l}\sum_{x=t-l}^{t}(P_i(x) - P_j(x) - \frac{1}{l}\sum_{x=t-l}^{t}(P_i(x) - P_j(x)))^2 \tag{7.5}$$

From the above equation, we can predict $p_i(t)$ based on readings from other homes:

$$P_i(t) = \sum_{j=1}^{N}\frac{P_j(t) * c_{ij}(t)}{\sum_{j=1}^{N} c_{ij}(t)} \tag{7.6}$$

If $c_{ij}(t)$ does not exist because of missing data, then we replace $c_{ij}(t)$ with $c_{ij}(t_k)$. Where t_k is the latest time for updating the correlation between homes i and j.

7.4.2 Data Reconstruction and Prediction

With the above four correlation models, we can reconstruct the missing data and predict the future data for real-time control. In order to schedule the controllable workload in each home and maintain the stability of the microgrid, the control center needs to collect the data of real-time power consumption, voltage, and frequency. Then the reconstruction process is executed as follows: i) if only part of the data is missing in one home, the power voltage relationship can be used to reconstruct the power consumption or voltage; ii) if only frequency data is collected from one home, frequency voltage correlation can be used to reconstruct the voltage and then apply power voltage relationship to recover power; iii) if no data is collected from one home, we can utilize the temporal and spatial correlation to reconstruct the data; iv) if no data is collected from any homes, only temporal correlation can be used to reconstruct the data; and v) if all the data is collected from one home, the collected data is applied to update the correlation weight for temporal and spatial correlation models.

The prediction process is the same as the scenario that no data is collected from any homes. Note the prediction with temporal correlation is only accurate for short-term data missing. Because generators can not be turned on/off very frequently and the generation control needs long-term power consumption prediction, other traditional consumption prediction algorithms can be applied in this scenario.

7.5 Central and Local Schedulers

With the reconstructed data and predicted future data, central and local schedulers need to schedule the future generation of generators and the controllable workload in each home. The design goal is to balance the power demand and generation with minimum operation cost in the microgrid.

7.5.1 Design Goal

Assume there are N homes in the microgrid and the power consumption of home i at time t is $g(t)$. The microgrid has M units of homogeneous local generators, each has a maximum power output capacity L. Based on a common generator model [56], we denote β as the startup cost of turning on a generator. Startup cost β typically involves the heating up cost (in order to produce high pressure gas or steam to drive the engine) and the time-amortized additional maintenance costs resulted from each startup (e.g., fatigue and possible permanent damage resulted by stresses during startups). We denote y_m as the sunk cost of maintaining a generator in its active state per unit time, and y_o as the operational cost per unit time for an active generator to output an additional unit of energy. Note that our design is not limited to any generator model. Table 7.1 summarizes the definition of parameters. Our design goal is to balance the power demand and generation while minimizing the operational cost of power generators. The problem can formulated as follows:

$$
\begin{aligned}
\text{Min} \quad & \sum^{M}(y_o g(t) + y_m g_o(t) + \beta n_o(t)[g(t) - g(t-1)]) \\
\textit{s.t.} \quad & 0 \le b_i(1) \le B_i; \quad \forall i && (a) \\
& 0 \le b_i(t) + b_i^g(t) - b_i^u(t) \le B_i; \quad \forall i, t && (b) \\
& d_i(t) \le b_i^u(t) + e_i(t); \quad \forall i, t && (c) \\
& \sum_{i=1}^{N}(e_i(t) + b_i^g(t)) \le g(t) \le L; \quad \forall t && (d) \\
& g_o(t) - g_o(t - t_o) \le p_u; \quad \forall t && (e) \\
& g_o(t - t_o) - g_o(t) \le p_d; \quad \forall t && (f) \\
& \sum_{i=t+1}^{i=t+t_o} g_o(i) = t_o, \sum_{i=t-t_o}^{i=t-1} g_o(i) = 0; \forall t, n_o(t) = 1 && (g)
\end{aligned}
$$

Constraints (a) and (b) ensures the battery energy level is always not less than zero and not greater than the battery capacity. Constraint (c) means a home consumes less energy than the amount of energy it obtains from generator and battery. Constraint (d) limits the output power of generator. Constraints (e) and (f) mean that the speed of increasing and decreasing generator power. Constraint (g) ensures that the minimum time for generator to change output power is t_o. The object function and constraints are all linear functions, thus the problem is a mixed integer programming problem, which is NP-complete. In reality, it is not possible to obtain optimal solutions in real-time for generation and workload scheduling. Therefore, the central scheduler uses a heuristic approach to solve this problem (detailed in §7.5.2). Furthermore, because of the unreliable communication, the home controller may not receive the control message from the control center. Therefore, a distributed algorithm is proposed for the local scheduler to schedule workload in each home (detailed

Notation	Definition
$b_i(t)$	Amount of energy in battery of home i at time t
B_i	Battery's capacity of home i
$d_i(t)$	Consumed power of home i at time t
β	Cost of changing output power of generator
L	Maximum power output of generator
y_m	Sunk cost of maintaining generator per time
y_o	Operational cost of generator for output power
p_u	Maximum ramping-up rate
p_d	Maximum ramping-down rate
$g(t)$	Output power of generator at time t
$g_o(t)$	On/off status of generator at time t
$n_o(t)$	Equals 1 if output of generator changes at time t
t_o	Minimum time for generator to change output power
$e_i(t)$	Power from generator to home i at time t
$b_i^u(t)$	Power discharged from battery to home i at time t
$b_i^g(t)$	Power from generator to battery at time t

Table 7.1: Notations for Demand and Generation Scheduling

in §7.5.3).

7.5.2 Central Scheduler

The central scheduler has the power consumption prediction in next few seconds and next minimum on/off time of generator. Thus, central scheduler can schedule controllable workload in homes to fulfill constraint (c) and schedule generation for next minimum on/off time. Therefore, the optimization problem can be decomposed into two subproblems: i) generation scheduling to fulfill the demand and minimize the operational cost; and ii) workload scheduling in each home to stabilize the aggregated demand.

Generation Scheduling

The key idea of the generation scheduling is to turn on the generator when power demand is low and turn off the generator when the power demand is high. Note that it is not able to change the output power of generator due to the minimum on/off time. Thus, we should decide whether to change the output power of generator based on power demand in the next minimum on/off time. At time t, the power demand from time $t+1$ to $t+t_o$ can be calculated as

$$\triangle E(t+1, t+t_o) = \sum_{k=t+1}^{k=t+t_o} \sum_{i=1}^{N} d_i(k) \tag{7.7}$$

With the given demand, we need first to ensure the power generation and energy storage in battery is higher than the power demand. Assume the power generation at time t is $g(t)$, we have three options: i) increasing generation with $g^u(t)$; ii) decreasing generation with $g^d(t)$; and iii) maintaining the same generation $g(t)$ in next minimum on/off time. The algorithm of decision making for generation is shown in Algorithm 12. If power demand is higher than energy storage in battery and energy generation $g(t)$ in next t_o, we have to increase the power generation to avoid power outage (Lines 1-3). To minimize the operation cost. The amount of generation increase can be calculated as:

$$g^u(t) = \frac{\triangle E(t+1, t+t_o) - b(t+1) - g(t) \cdot t_0}{t_o} \tag{7.8}$$

Otherwise, we can either decrease power generation or maintain the same generation. Note that there is extra cost for changing output power of generator, thus, we only decrease the power generation when the power demand is too low that the battery can not store the extra power generation (Lines 4-5). The amount of generation decrease can be calculated as

Algorithm 12: Generation Scheduling Algorithm

1: Calculate demand in next minimum on/off time $\triangle E(t+1, t+t_o)$ based on Equation (7.7);
2: **if** $\triangle E(t+1, t+t_o) > b(t+1) + g(t) \cdot t_o$ **then**
3: Increase generation $g^u(t)$ based on Equation (7.8);
4: **else if** $\triangle E(t+1, t+t_o) < b(t+1) + g(t) \cdot t_o - B$ **then**
5: Decrease generation $g^d(t)$ based on Equation (7.9);
6: **else**
7: Maintain the same generation $g(t)$.
8: **end if**

$$g^d(t) = \frac{b(t+1) + g(t) \cdot t_o - B - \triangle E(t+1, t+t_o)}{t_o} \tag{7.9}$$

Otherwise, we maintain the same power generation $g(t)$ (Lines 6-8).

Workload Scheduling

The goal of workload scheduling is to avoid power outage and minimize the extra cost for changing power generation. The key idea is to turn off the controllable workload when aggregated power demand is high to avoid power outage and turn on the controllable workload when power demand is low to maintain stable power demand. Note that the demand in each home can be divided by controllable workload $d_i^c(t)$ and uncontrollable workload $d_i^u(t)$:

$$d_i(t) = d_i^u(t) + d_i^c(t) \tag{7.10}$$

Based on the prediction of short time power consumption in next time slot, our central scheduler decides the scheduling the controllable workload demand in each home. At time t, we can calculate power demand at time $t+1$ as:

$$\triangle E(t+1) = \sum_{i=1}^{N} d_i(t+1) \tag{7.11}$$

If power demand is higher than power generation and energy storage in battery $\triangle E(t+1) > b(t+1) + g(t+1)$, we need to turn off some controllable workload. Otherwise, if $\triangle E(t+1) < b(t+1) + g(t+1)$, we need to turn on some controllable workload. In our simulation, we utilize water heater as controllable workload. Thus, If $\triangle E(t+1) > b(t+1) + g(t+1)$, we turn off some water heaters in homes with less hot water demand until $\triangle E(t+1) = b(t+1) + g(t+1)$. If $\triangle E(t+1) < b(t+1) + g(t+1)$, we turn on some water heaters in homes with most hot water demand until $\triangle E(t+1) = b(t+1) + g(t+1)$.

7.5.3 Local Scheduler

However, if the communications between the control center and homes are unreliable, the instructions for each home may be lost or arrive late. Thus, we also provide a distributed control when the control instructions from central controller are not available. The key idea is to schedule controllable workload based on power-voltage model proposed in §7.4.1 because the local voltage in each home can be used to infer the aggregated power demand in the microgrid. However, the problem is that every home may decide to turn on/off controllable workload simultaneously to balance power supply and demand because they can not communicate with each other. Then it may cause an endless loop for each home to turn on/off controllable workload at the same time, which is not helpful to balance power generation and demand.

In our design, we take an adaptive feedback control to enable homes to stabilize the

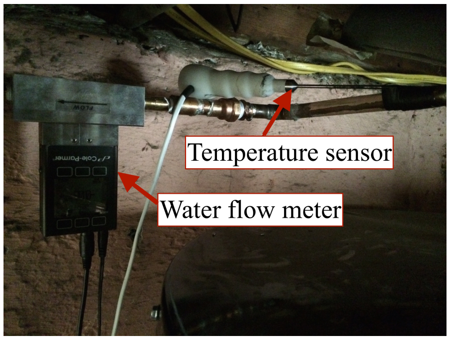
Algorithm 13: Local Scheduler Algorithm

```
1: Calculate △V_i, d_i^c(t);
2: if △V_i ≠ 0 & t_i^b = 0 then
3:     Calculate t_i^b based on Equation 7.12;
4: else if t_i^b ≠ 0 then
5:     t_i^b = t_i^b − 1;
6:     if △V_i > 0 then
7:         Increase controllable workload;
8:     else if △V_i < 0 then
9:         Decrease controllable workload.
10:    end if
11: end if
```

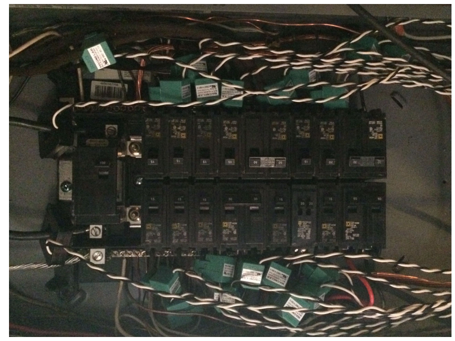
power demand cooperatively. When each home detects the power demand change (power generation is relative stable since there exist minimum on/off time for generators), it does not turn on/off controllable workload immediately but with some backoff time. The detailed algorithm is shown in Algorithm 13. At time t, each home calculates the consumption of its controllable workload and estimate the change of power demand $\triangle d$ in the microgrid by utilizing power voltage correlation shown in § 7.4.1 (Line 1). When the power demand is stable, each home keeps controllable workload with previous state. When it detects either high or low demand and there is no backoff timer, each home calculates backoff time t_i^b (Lines 2-3). t_i^b can be calculated as:

$$t_i^b = \frac{\triangle d}{N \cdot d_i^c(t)} \tag{7.12}$$

If $t_i^b \neq 0$, home updates the timer (Lines 4-5). Then each home check the timer again, if the timer expires, local controller immediately turns on the controllable workload when demand is low and turn off the controllable workload when demand is high (Lines 6-10).



(a) Water flow & temperature monitor



(b) Energy monitoring

Figure 7.8: Experiment setup in residential homes

7.6 Experimental Evaluations

In this section, we evaluate the performance of our design. We collect empirical data of i) total power consumption and water heater power consumption from 100 homes; ii) hot water usage from three homes; and iii) voltage and frequency data from three homes. Note that we only have control access of three homes (in Binghamton, New York), in which we collected hot water usage, power consumption of water heaters, voltage and frequency data and all the experiments are conducted in these three homes. For the rest 97 homes (in Austin, Texas), we collected the power consumption data for simulations. Because we do not have the hot water usage, voltage and frequency data from these 97 homes, we use water heater power consumption to generate the hot water usage data and apply correlations among power, voltage and frequency obtained from the experiments to generate voltage and frequency data for simulations. The hot water usage and power consumption are measured by water flow sensors and eGauge sensors every second. The experiment setup of one home is shown in Figures 7.8(a) and 7.8(b). The power consumption in one year and water flow data in two months are shown in 7.9 and Figure 7.10, respectively.

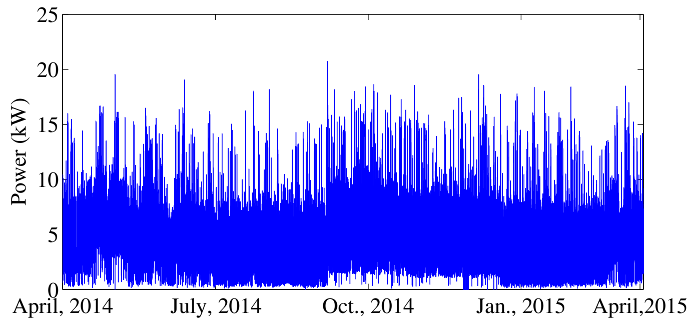


Figure 7.9: Power consumption in one year

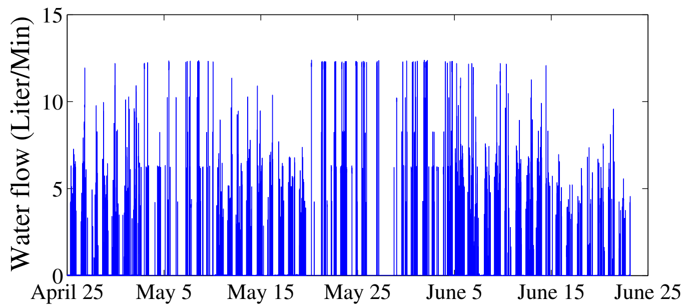


Figure 7.10: Water flow in two month

7.6.1 Basic Evaluation Results

In this section, we evaluate the effectiveness of our system, which includes three metrics: i) data reconstruction accuracy; ii) total operation cost in the microgrid; and iii) the impact on homeowners' hot water usage. All results are simulated with the seven days' empirical data of hot water usage and power consumption. Because our design goal is to minimize the operation cost of generator, we refer our design as MOC in the latter description. The baseline we compared with is original power consumption in individual homes without workload scheduling. In simulations of MOC, we also use baseline in the first day since the correlation model needs to be trained based on historical data. Thus we mainly compare the performance of MOC and baseline for the rest of six days.

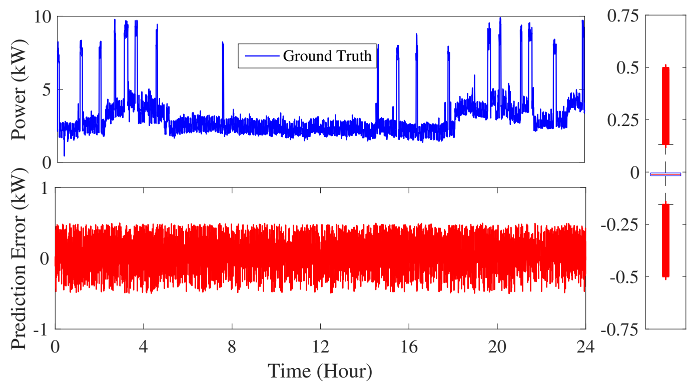


Figure 7.11: Prediction accuracy with temporal correlation model, the right figure shows the average, 25 percentile and 75 percentile of the prediction error

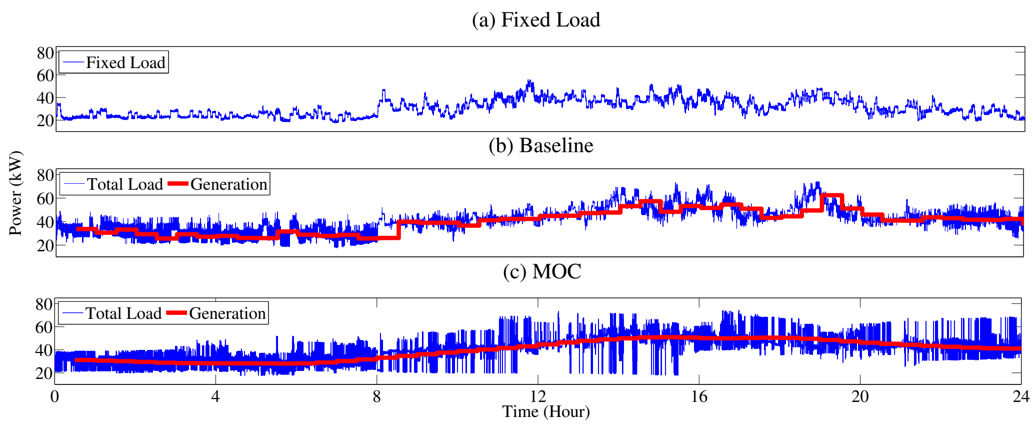


Figure 7.12: Power consumption and generation for one day

Reconstruction Accuracy

We run the detection algorithm with one month data and predict the missing data for the next month. The results are shown in Figure 7.11. The prediction matches well with the ground truth. The maximum error of prediction we observe is 0.498kW and the average error of prediction is 0.0289kW.

We run spatial reconstruction algorithm to recover one home's energy data from other 99 homes. The results are shown in Figure 7.13. The prediction overall is very close to ground truth. The maximum error of prediction and average error of prediction are 3.836kW and 0.131kW, respectively.

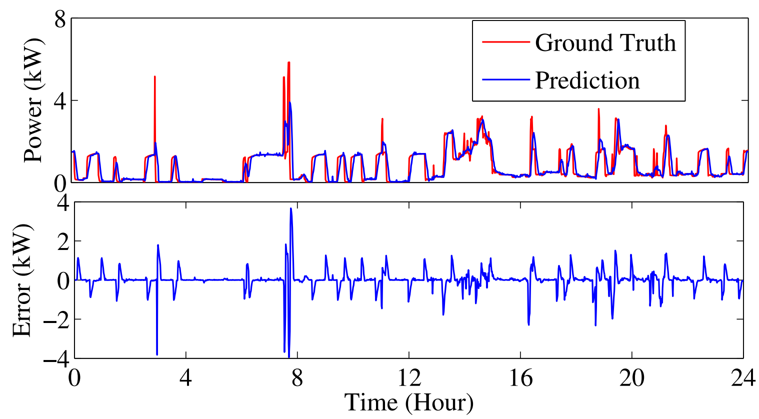


Figure 7.13: Prediction accuracy with spatial correlation model

Generation and Consumption

To clearly illustrate the difference between baseline and our design, we only show the power consumption of baseline and our design for one day in Figure 7.12. For power consumption without water heater, the peak demand is mainly from 8am to 12am and from 6pm to 8pm. In the mean time, hot water usage is also during the similar time. For baseline, when it detects hot water usage, it turns on water heater immediately. Furthermore, because different homes are highly possible to use hot water in similar time of evening, the peak demand rises from 50.31kW to 75.45kW (at around 7pm and 11pm). The power of the water heater in our simulation is 5.29kW, thus at least 5 homes turn on water heaters at the same time. For the generation, because it does not predict the future power consumption to smooth the generation, the hourly generation changes very quickly, which introduces more operation cost. For MOC, because it predicts short-term and long-term power consumption in future, the hourly generation is stable.

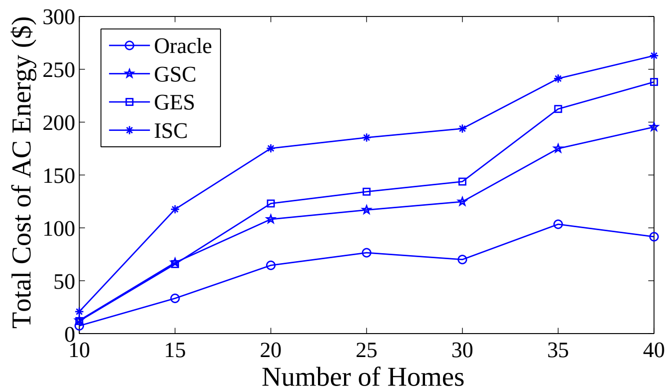


Figure 7.14: Total operation cost for generators

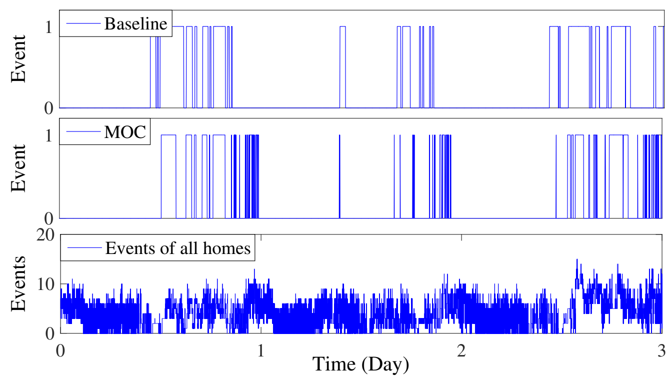


Figure 7.15: Water heater events over three days

Total Cost

The total operation cost for generators for a typical day are shown in Figure 7.14. In the beginning, because the power consumption is low, the cost for different approaches is similar. However, from 8am, power demand increases quickly because most of the people wake up. Thus, the operation cost increases quickly in baseline and the operation cost for MOC and offline optimal still increases linearly. For six days' simulations, the average daily operation cost in baseline is \$292.5 while daily operation cost for MOC is \$224.6, which is 23% lower.

Water Heater Scheduling

To better understand how homes schedule water heater event, we show detailed water heater energy consumption events of 15 homes in Figure 7.15. To show the detailed difference

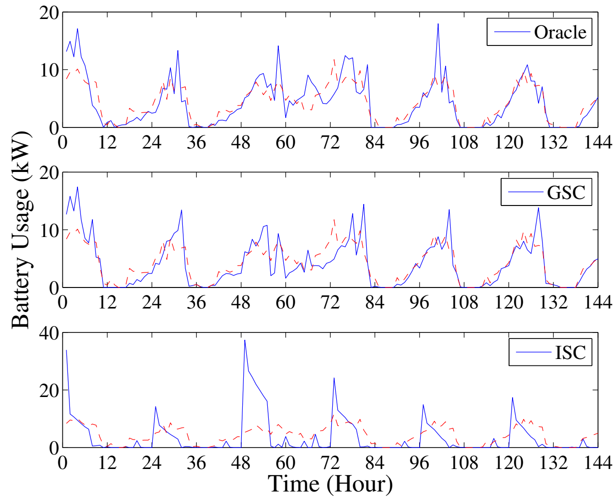


Figure 7.16: Power consumption and generation for different battery capacities

between baseline and MOC, we show the water heater events for 3 days. The upper figure is the water heater events of the baseline in one home. For the baseline, most of the water heater energy consumption events last for longer periods. This is because the water heaters are turned on right after hot water usage. After people take a shower or bath for ten minutes, the water heater will be turned on for around 1 hour to reach the high temperature threshold. The middle figure shows the water heater events of MOC in one home. Compared to baseline, the water heater events are more sparsely distributed to reduce the overlap of events from different homes. The bottom figure shows the total events of MOC in all 15 homes.

Hot Water Temperature

Though MOC allows homes to turn on water heater earlier or later, MOC can also fulfill users' hot water usage efficiently. In Figure 7.17, we show the distribution of difference between the targeted temperature and the hot water temperature when there exists hot water usage events. The targeted temperature of hot water in our experiment is set as $50°C$. For

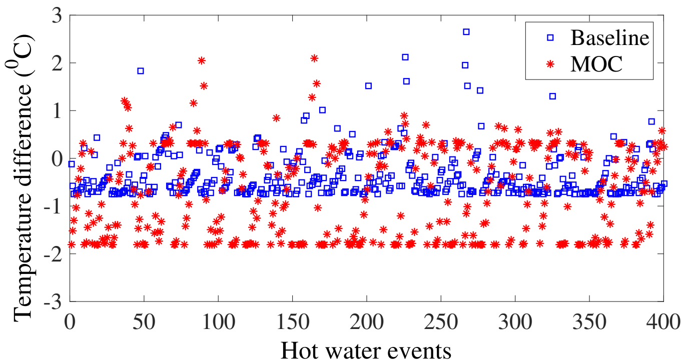


Figure 7.17: Temperature distribution of hot water events

the baseline, it turns on water heater immediately after hot water usage, then the temperature of hot water is always a little lower than the targeted hot water usage (mainly $1°C$ lower than the targeted temperature in Figure 7.17). For MOC, when a home predicts future hot water usage, it can turn on water heater earlier to better fulfill the hot water usage, thus the hot water temperature can be higher than the targeted temperature for some time. In Figure 7.17, the hot water temperature in the water tank is at most $2°C$ lower than the targeted temperature. Thus the impact of our design on people's hot water usage is very low.

7.6.2 Advanced Evaluations

Because our system is designed for severe environments, such as islands, it is crucial to investigate the system's sensitivity under diverse settings.

Impact of Battery Capacity

Because battery is expensive and has limited life-time, it is important to investigate the system benefit with different capacities of batteries. We show the power consumption and generation with three different battery capacities in Figure 7.18. The top figure shows

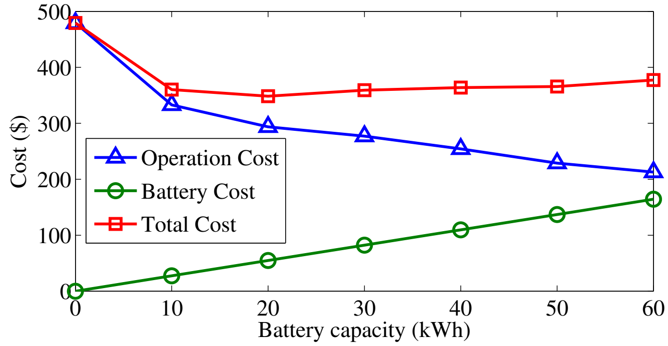


Figure 7.18: Operation, battery and total cost for different battery capacities

the scenario with no battery in the system. Because there is no battery, the generators needs to generate the maximum power in its working period to avoid power outage, which introduces high energy waste. The middle figure shows the results with $30kWh$ battery capacity. With the battery, the difference between generation and consumption can be offset, thus the overall generation is reduced. For the bottom figure, with very high battery capacity, the generator can only generate the average power consumption in the next hour, thus the overall generation is minimized. The operation, battery and total cost for different battery capacities are shown in Figure 7.18. With higher battery capacity, the operation cost decreases especially from no battery to $10kWh$ battery. However, the decrease slows down with higher battery capacity. For the total cost, we find that microgrid with $20kWh$ battery performs best since it balances cost between generation and batteries.

Impact of Data Missing Rate

With different environments, the data missing rate can be quite different. Thus, it is important to study the reconstruction accuracy of our data management design under different scenarios. In these sets of simulations, we simulated the accurate data to generate missing data with different missing rate from 4% to 20%. The results of reconstruction accuracy are shown in Figure 7.19. With the increase of data missing rate, reconstruction accuracy

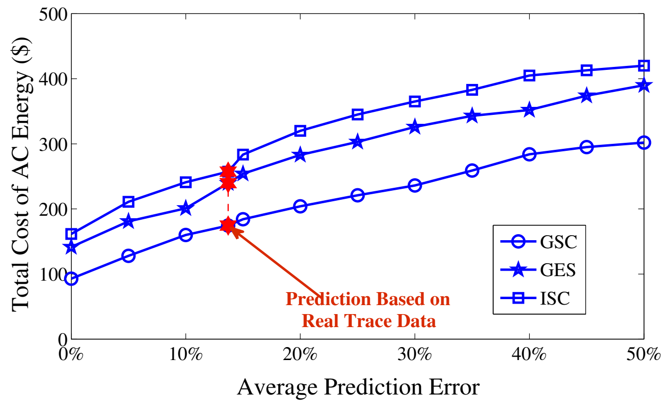


Figure 7.19: Reconstruction accuracy with different r_d

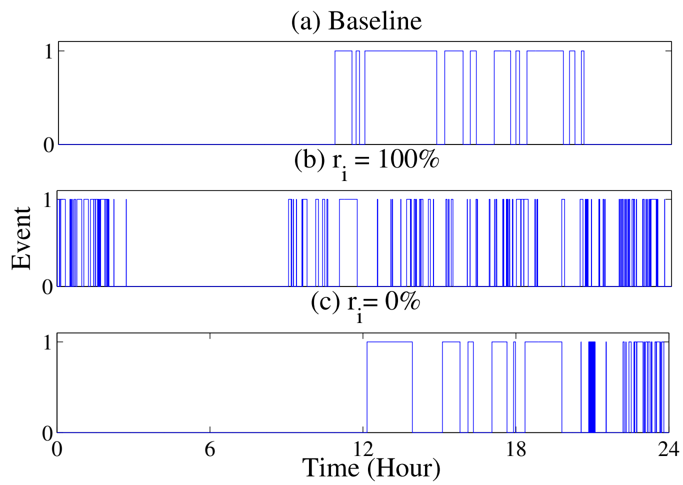


Figure 7.20: Water heater events with different r_i

decreases slowly. Even with 20% data missing rate, the average of reconstruction accuracy is above 80%. Thus, our design is robust in situations with high data missing rate.

Impact of Instruction Missing Rate

In the mean time, the instruction missing rate can also be quite different under different environments. Thus, we study the performance of centralized control (instruction missing rate r_i is 0%) and distributed design (instruction missing rate r_i is 100%). We simulated the different instruction missing rate for two extreme cases (shown in Figure 7.19). The water heater is turned on and off more frequently when r_i is 100%. That is because each home lacks the knowledge of behaviors of other homes, thus they may collide to turn on/off water heater and then immediately turn off/on water heater. The frequent on/off operations will decrease the lifetime of water heater. However, total cost for different instruction missing rate is similar, which is not included due to the limited space.

7.7 Related Work

Our work is related to the following areas of previous work:

Energy Management. Different techniques are proposed for energy management in either demand side or generation [71, 115]. In [127], a decentralized optimal load control mechanism is proposed to provide contingency reserve in the presence of sudden demand-supply mismatch. In [116], a model predictive control algorithm is proposed to co-schedule HVAC control, EV scheduling and battery usage to reduce the building energy consumption. In [28], stochastic and robust optimization are applied for real-time price based demand response management. In [58], an optimal solution is provided to trade off between quantity and quality of variable renewable energy source in smart grid. Different from existing work, our design presents a holistic approach of real-time scheduling for both demand in individual homes and generation of the local generators in microgrids.

Missing Data Management. There have been various works in the research community to investigate how to manage missing data in cyber-physical systems [59]. Traditionally the approach to obtaining missing values for linear time series has involved the use of curve fitting [26]. Autoregressive integrated moving average (ARIMA) model is fitted to time series data to predict future points in the time series data [124]. Maximum likelihood based approach is applied to estimate the missing data [69]. Different from existing work, we investigate the correlation between different energy data in microgrids and utilize the correlation models to cope with unreliable sensors and wireless communication.

7.8 Summary

The biggest challenge of maintaining a self-sustainable microgrid is to balance the power demand and generation in real-time with dynamically changing power demand. Furthermore, the unreliable data collection and communication between homes and the control center in a microgrid makes the real-time control even harder. To address these issues, we propose a novel data management technique to process the collected data, reconstruct the missing data caused by sensing error or unreliable communication, and predict the future demand for real-time control with missing data in extreme situations. The control center then decides the scheduling of the workload of appliances in each home and the operations of the local generator based on the collected and predicted data. Through extensive experiments and simulations, we show that our design can recover the missing data with 99% accuracy and our distributed control can balance power demand and generation and reduce operation cost by 23%.

CHAPTER 8

PRIVACY MANAGEMENT

Though microgrids can improve renewable energy utilization efficiency and reduce the operational cost, there can be privacy leakage issue in a hybrid AC-DC microgrid because power consumption information of each home can be exposed through the power lines or compromised neighbors in the microgrid. Power consumption data then can be used to reveal precise information about appliances' activities with non-intrusive load monitoring algorithms. In this chapter, instead of using batteries, we propose to leverage the unique features of hybrid AC-DC microgrids to hide power consumption information. Specifically, we design *Shepherd*, a privacy protection framework to hide power consumption information from different types of power consumption detection techniques.

8.1 Introduction

With the increasing demand of energy consumption and the desire to reduce carbon dioxide emissions, renewable energy has become an important alternative choice. The government is encouraging the utilization of renewable energy and expects the renewable energy can reach up to 33% of total energy supply by 2020 [23] [37]. However, renewable energy that can be harvested in residential homes is typically DC power (e.g., solar energy), while the power grid nowadays is only providing AC power. In fact, many appliances in residential homes are operated using DC power, such as TVs, computers, DC water heaters and lighting. According to the government survey, these DC appliances consume around 20% to 30% energy in residential homes [92]. Furthermore, with the popularity of the electrical vehicles, the DC appliances will consume much more energy in residential homes. To utilize the renewable energy in existing AC power grid, DC power from renewable energy must be converted to AC and then converted back to DC again to power DC appliances. The conversion loss of DC-AC-DC can be as high as 50% [43]. Therefore, instead of using the existing AC grid, researchers are studying the possibility of the hybrid AC-DC microgrids, in which homes obtain AC power from existing AC power grid to power AC appliances (e.g., air conditioners, compressors, etc.) and utilize DC power from renewable energy (e.g., solar energy) and batteries to power DC appliances (e.g., TVs, computers, DC water heaters, etc.). The advantages of the hybrid AC-DC microgrids are: i) higher energy efficiency for DC appliances because DC appliances can directly use DC power, which reduces energy conversion from renewable energy of DC power to AC and conversion from AC power to DC to power DC appliances; ii) lower conversion loss for batteries because they can be charged and discharged in DC power; and iii) lower cost of utilizing renewable energy because with higher energy efficiency for DC appliances and lower conversion loss for batteries, the amount of renewable energy needed is smaller and the investment cost of renewable energy (e.g., solar panels) can be lower. Recently, system architecture of co-existence of AC and DC power lines has been proposed [129] and we show that the homes in a microgrid can utilize DC power line to share renewable energy to minimize the energy cost in Chapter 6. It is highly possible that in the near future we will witness a paradigm shift from a centralized AC power grid to a hybrid AC-DC microgrids in residential communities. Therefore, it is essential to explore this frontier in advance.

Although hybrid AC-DC microgrids have many advantages, they impose a major challenge on privacy leakage. This is because homes are connected to both AC and DC power lines, which provides vulnerability for malicious users to reveal power consumption information of neighboring homes in power lines. For example, illegal eavesdropping on the wireless communication of smart meters is investigated in [93]. In this chapter, the authors discover two novel possible vulnerabilities for malicious users to obtain the accurate power consumption of individual homes under the infrastructures of the hybrid AC-DC microgrids: i) high accuracy power consumption leakage via voltage based on the power-voltage relationship; and ii) monitoring energy sharing from compromised homes in DC power line to obtain power consumption information of neighbors. Therefore, malicious users or third-parties can easily utilize these vulnerabilities to obtain the high granularity power consumption data of homes in the hybrid microgrids.

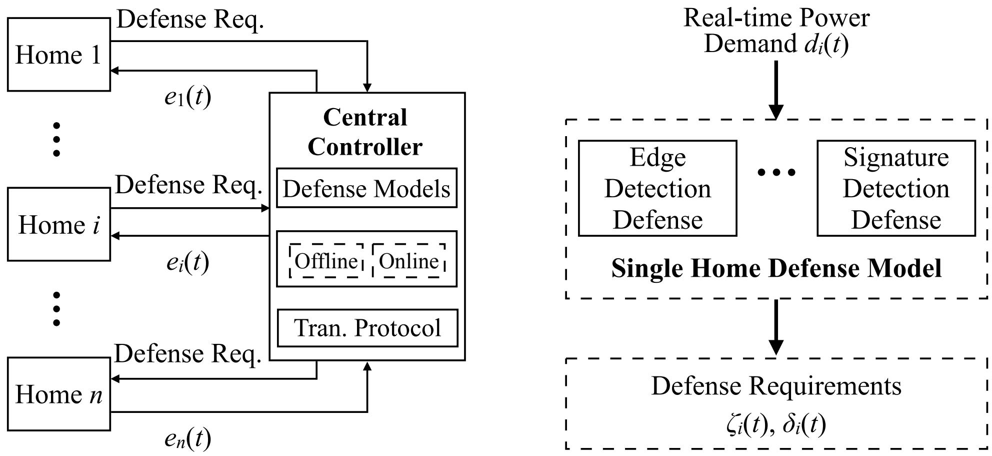
With the high granularity power consumption data, Non-Intrusive Load Monitoring (NILM) can be applied to analyze the data for revealing appliances' activities [47]. The widely used technique is the edge detection [68], which looks for the sharp edges that reveal the significant changes in the steady power consumed by the household. More seriously, we demonstrated a new signature detection technique which can reveal appliances' usage more accurately than existing approaches. Appliances usage information can then be used to reveal private information of occupants. For example, usage time of certain appliances (e.g., water heater) can reveal the number of people living in the home. Furthermore, changes of appliances usage patterns can also reveal private information (e.g., health conditions). For example, if a person usually turns off all the lights when he/she sleeps, and suddenly he/she turns on and off the lights frequently in the night while other appliances' usage patterns stay the same; this indicates that he/she may be sick or has a sleeping problem. Thus, it is critical to protect power consumption information and prevent privacy leakage for occupants in individual homes.

To achieve this, researchers proposed battery-based load hiding (BLH) algorithms in [95] [120], which utilize batteries to partially supply the net demand load from the home to alter the external load as seen by the smart meter. The battery is charged and discharged at a specific time to hide the power consumption. However, battery-based algorithms have three limitations: i) they have to cope with limited battery capacity and discharge rates or need batteries with large capacities; ii) they need to charge and discharge batteries frequently, which will significantly decrease the battery's lifetime; and iii) they lack a generic model for privacy preserving under different types of attacks. To overcome these limitations of BLH, we leverage the unique features of hybrid AC-DC microgrids and propose *Shepherd*, a privacy protection framework to effectively protect occupants' privacy. In Shepherd, we provide a generic model for energy consumption hiding from different types of detection techniques. We also propose a novel approach of coordinating AC and DC power lines to hide energy consumption in individual homes. Specifically, each home obtains partial energy from neighboring homes to power its DC appliances and hide its own power consumption information while protecting the actual power consumption information from its neighbors. Because power consumption collected by the smart meter of each home is different from the actual amount of energy consumed by its own appliances, the power consumption information of each home can be protected. To ensure that every home is correctly billed based on the amount of energy consumed by its own appliances instead of shared or obtained energy, we propose the energy sharing control protocol for control and billing. The main contributions of the chapter are as follows:

• We study the privacy leakage problem in hybrid AC-DC microgrids and discover two novel vulnerabilities for malicious users to obtain power consumption information of individual homes without occupants' authentication.

• We leverage the unique features of hybrid AC-DC microgrids and propose Shepherd, a privacy protection framework to allow homes in a microgrid to coordinate with each other to hide power consumption information. Because different homes need to coordinate with each other, we also analyze how compromised neighbors in a microgrid can be used to provide energy consumption information to malicious third-parties. The corresponding defense models are proposed and we present an optimal offline solution and an efficient heuristic online algorithm so that the transmission loss is minimized.

• We conduct real-world experiments by deploying energy meters in multiple homes to collect the consumption signatures of individual appliances. We also run large-scale simulation with the empirical power consumption traces from 40 homes. Results show that Shepherd can i) significantly reduce the detection ratio from 33% to 13% compared to BLH, and ii) effectively hide consumption information even with 25% compromised neighbors.



(a) Architecture of Shepherd

(b) Home Controller

Figure 8.1: Overview of Shepherd

8.2 Overview of Shepherd

In this chapter, we leverage the unique feature of hybrid AC-DC microgrids to enable homes to help their neighbors hide the power consumption information from the malicious third parties on the traditional power grid. To hide power consumption information for homes in a microgrid, we propose $Shepherd$, a privacy protection framework in hybrid AC-DC microgrids. The overview of our design is shown in Figure 8.1(a), which contains two components: a home controller at each home and a central controller.

The detailed design of the home controller is shown in Figure 8.1(b). It collects real-time power demand from smart meter measurements. Then we analyze the single home adversarial model based on different detection techniques of power consumption (§ 8.3.1). To defend from the adversarial model, we propose a generic single home defense model to calculate the amount of power required to defend from the single home adversarial model (detailed discussion in § 8.3.2). The defense requirements would be sent to central controller.

The central controller collects defense requirement from homes in the community in order to generate energy sharing solution for homes to defend from single home adversarial model. We also analyze the adversarial model with compromised neighbors in hybrid AC-DC microgrids (detailed discussion in § 8.4.1) and propose corresponding defense model to protect privacy of occupants (detailed discussion in § 8.4.2). The defense model is then illustrated as a convex optimization problem. To solve the optimization problem, we propose an optimal solution for energy sharing so that the transmission loss can be minimized (detailed discussion in § 8.4.4). To further reduce the computation complexity, we also propose an efficient heuristic online algorithm (detailed discussion in § 8.4.5). The generated energy sharing solution will return to each home controller through transmission protocol (detailed discussion in § 8.4.6).

8.3 Security Model in A Single Home

In this section, we analyze the generic security models for power consumption information hiding in residential homes. We provide the adversarial model in a single residential home and propose the corresponding defense model. All the notations used in this chapter are summarized in Table 8.1.

8.3.1 Single Home Adversarial Model

The single home adversarial model is to detect appliances' activities based on the real-time power consumption of a single home. The detection techniques are widely studied [96] and the key idea is to match detected power consumption with labelled power consumption of

Notations	Definitions
$d_i(t)$	Power demand of home i at t
$e_i(t)$	Real power consumption home i at t
c_i	Power capacity of home i
$\zeta_i(t)$	Min power increase to avoid both detections
$\delta_i(t)$	Min power decrease to avoid both detections
$p_i(t)$	Power difference between $d_i(t)$ and $e_i(t)$
$\eta_i(t)$	Energy transmission efficiency of home i at t
$m_j(t)$	Modelled power of appliance j at t
g_{ij}	Power of appliance j for edge detection at home i
$\rho_{ij}(t)$	Similarity between appliance signature j and $e_i(t)$

Table 8.1: Notations for Privacy Management

appliances. We present case studies for two representative detection techniques.

Edge Detection

Edge detection technique looks for significant changes in the energy being consumed by the household [120]. Such changes are characterized by sharp edges in the energy consumed by the appliances. These edges are then clustered and matched against known appliance profiles. Let power consumption of appliance k at home i for edge detection be g_{ik}, We define the appliance j detected by edge detection with power consumption $e_i(t)$ at home i as follows:

$$|g_{ij} - e_i(t)| = \min_k |g_{ik} - e_i(t)| \tag{8.1}$$

For instance, if someone turns on/off a $20W$ lamp, then the net power consumption increases/decreases by $20W$. The algorithm detects the pair of edges with equal magnitude and opposite direction, and matches them against the electric profile for a $20W$ lamp.

Signature Detection

While edge detection methods are simple, they are often inaccurate, because they fail to capture the complex power usage patterns of different loads. Recently, researchers revealed the empirical power consumption signature of different electrical loads [17]. Electrical loads are categorized into five consumption signature models. With the energy consumption signatures of different appliances, we design a more efficient method than edge detection to reveal appliances' usage patterns with a home's energy consumption data. The key idea is to detect appliances' usage by the similarity between real power consumption and appliances' consumption models. If consumption model of an appliance a_i is most similar to real power consumption, then it is highly possible that appliance a_i is working but not other appliances. In this chapter, we propose a Euclidean distance-based function to quantify the similarity between two vectors. Let $e(t)$ be the real energy consumption and $m_i(t)$ be energy consumption data generated by models at time t, where $length(a_i)$ is the signature length of appliance a_i. The similarity between two vectors can be calculated as:

$$\rho_i = \frac{1}{1 + l_{(e,m_i)}} \tag{8.2}$$

where

$$l_{(e,m_i)} = \frac{1}{length(a_i)} \sum_{t=1}^{length(a_i)} (e(t) - m_i(t))^2 \tag{8.3}$$

Equation (8.3) is used to calculate the distance between two vectors. Because different appliances' models have different lengths of signature sequences, we use $1/T$ to normalize the distance of two vectors. For example, the signature sequences of a lamp are short due to the on-off model; while the signature sequences of TV is long due to dynamic power consumption during usage. Equation (8.2) is used to transfer distance to similarity within range of [0, 1].

Based on the similarity between consumption models of different appliances and real consumption data, we detect the appliances' usage patterns. Suppose that appliance i has the highest similarity with real power consumption from time t, we then consider that appliance i is working. Because several appliances can be working at the same time, we can remove the detected appliance's model from real data and then repeat the detection process again. When the similarity between rest of appliances and real consumption is low, we end the detection process for time t and continue the detection process from the time when detected appliances stop working. An example of detection results are shown in Figure 8.2. In our experiment results, our proposed consumption signature detection method can detect appliances' activities 30% more accurate than edge detection method and can still detect 70% appliances' activities when the power consumption is processed with BLH.

Note that there are many other detection techniques too [4]. However, the detection techniques are mostly based on matching between detected power consumption and labelled power consumption, which provides us opportunities to alter power consumption to hide from these detection techniques.

8.3.2 Single Home Defense Model

To defend from the single home adversarial model, each home can increase or decrease its power consumption by sharing energy with neighbors in a microgrid. We define the

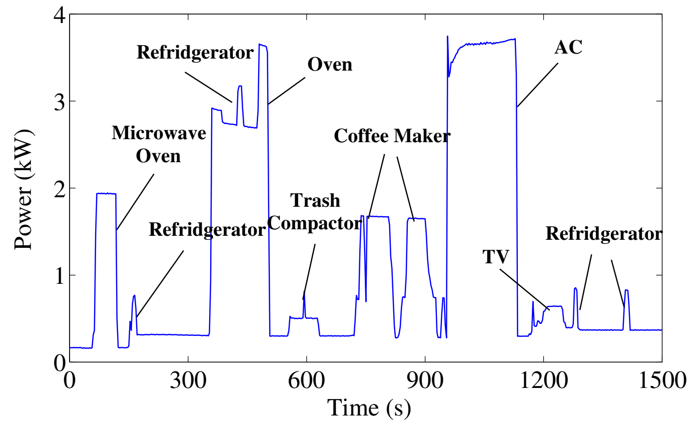


Figure 8.2: An example of detection results

minimum change of power consumption ζ (increase) or δ (decrease) as follows:

$$\zeta_i(t) = \min\{p \in \mathbb{R}_{>0} | A_i(e_i(t)) \neq A_i(e_i(t) + p)\} \tag{8.4}$$

$$\delta_i(t) = \min\{p \in \mathbb{R}_{>0} | A_i(e_i(t)) \neq A_i(e_i(t) - p)\} \tag{8.5}$$

$A_i(e_i(t))$ is the detected appliance based on power consumption $e_i(t)$ at home i with a given adversarial model. Equations (8.4-8.5) show that ζ and δ are the minimum change of power consumption to avoid detection from a given adversarial model. Here we present two case studies for the calculation of the minimum change of power consumption. Note that our defense model is generic. The minimum changes of power consumption for specific detection methods can be calculated based on Equations (8.4-8.5).

• For edge detection, we assume the appliance's modeled data $m_{i(j-1)}(t)$ (($m_{i(j-1)}(t) < d_i(t)$) with minimum $d_i(t) - m_{i(j-1)}(t)$ and appliance's modeled data $m_{i(j+1)}(t)$ ($m_{i(j+1)}(t) > d_i(t)$) with minimum $m_{i(j+1)}(t) - d_i(t)$. Based on Definition (1), we have the minimum change of power consumption $\zeta_i(t) = [m_{i(j+1)}(t) - d_i(t)]/2$ and $\delta_i(t) = [d_i(t) - m_{i(j-1)}(t)]/2$.

• For signature detection, we not only need to change power consumption based on current power consumption, but also power consumption in history because signature detection can detect appliances usage by their unique consumption patterns. The consumption signature is based on similarity of real consumption and model, thus we can hide power consumption based on minimizing the probability of detection. The key idea is to let the probability of detection decrease with new power consumption. Let $e_i(t)$ be the real power consumption, $d_i(t)$ be the power demand of the appliance, and $\rho_i(t)$ be the similarity of consumption and demand data, where $t = 1, 2, ..., T$. Then at time T, we need to make sure with $e_i(T)$ that $\rho_i(T) \leq \rho_i(T-1)$. To find minimum change of power change $\zeta_i^s(T)$ or $\delta_i^s(T)$, we first solve the equation $\rho_i(T) = \rho_i(T-1)$. With definition of similarity in Equation (8.2) and (8.3), we can rewrite the equation as:

$$\frac{1}{T}\sum_{t=1}^{T}[e_i(t) - d_i(t)]^2 = \frac{1}{T-1}\sum_{t=1}^{T-1}[e_i(t) - d_i(t)]^2 \tag{8.6}$$

Theorem 8.1. *There exist two solutions $e_i^1(T)$ and $e_i^2(T)$ of Equation (8.6), and $e_i^1(T) < d_i(T) < e_i^2(T)$.*

The detailed proof is in the Appendix. After solving Equation (8.6), we can calculate minimum power change. Let $e_i(T) < e_i^1(T)$, because $e_i^1(T) < d_i(t)$, with Equation (8.2) and (8.3), we have $\rho_i(T) \leq \rho_i(T-1)$. Thus $\zeta_i^s(T) = \sqrt{l_{(e_i,d_i)}(T-1)}$. Similarly, we can also have $\delta_i^s(T) = \sqrt{l_{(e_i,d_i)}(T-1)}$.

For the homes that need to defend against different detection techniques, we select the minimum change of power consumption as the maximum of minimum change of power consumption for all detection techniques. Note that our defense model is generic to any detection techniques. To defense a new detection technique, we only need to analyze the technique to obtain $\zeta_i(t)$ and $\delta_i(t)$, then our defense model can be applied to defend from the detection technique.

8.4 Security Model with Compromised Neighbors

With the minimum power change calculated, we can ensure that the power consumption pattern cannot be detected by power consumption data collected by smart meter in a single home. However, in a hybrid AC-DC microgrid, homes can share extra energy from renewable energy through DC line. Thus, it is possible that some compromised homes can use their power consumption change to detect their neighbors' power consumption. In this section, we analyze the adversarial model with compromised neighbors and propose the defense model. Then we illustrate the defense model as a convex optimization problem and provide optimal and heuristic solutions.

8.4.1 Adversarial Model with Compromised Neighbors

With the compromised neighbors in hybrid AC-DC microgrids, it is possible that some homes can use their power consumption change to detect their neighbors' power consumption. For example, home i turns on the lamp for $20W$. To hide from edge detection, it finds the appliance with closest power consumption is laptop with $100W$. Then based on single home defense model, we have $\zeta_i = 40W$. In a microgrid, home i can share energy to home j and k for $20W$ to defense single home adversarial model. However, if home j is compromised by third-parties and provides them the information that home i shares $20W$ energy to home j. Then based on single home adversarial model, malicious third-parties can know the power consumption increase of home i is $40W$ instead of $60W$. Although it is not totally accurate as $20W$, malicious third-parties can detect lamp is turned on at home i but not laptop based on edge detection. We define the condition that compromised homes can reveal real appliances' activities of other homes.

Definition 8.4.1. *Let home set that home i shares energy to be D_i and compromised home set be C_i, the condition for adversarial model with compromised neighbors to work at time t is*

$$\sum_{j \in D_i \& j \notin C_i} p_j(t) \geq \zeta_i(t) \tag{8.7}$$

Clearly, if enough homes are compromised by malicious third-parties, the third-parties can get enough information of energy sharing among homes. Then with the energy sharing information, the appliances' activities can be with the power consumption of home i.

8.4.2 Defense Model with Compromised Neighbors

To defend from adversarial model with compromised neighbors, we propose that each home's power change should not be balanced by only one home, but several homes to hide power consumption from neighbors. Because homes do not know that which homes are compromised, we propose the defense model to avoid detection from given number of compromised neighbors. For practical solutions, we also propose an online solution in §8.4.5 to work under the scenario that the number of compromised neighbors is unknown. To avoid detection from k compromised neighbors for home i, we need to ensure that any k neighbors are compromised, the appliances' activities are still protected. We propose two approaches for defense model with compromised neighbors: i) sharing energy with more homes to reduce detection probability. Because it reduces the amount of energy shared to one single home, it also reduces the probability of detection from multiple homes adversarial model. ii) sharing the same amount of energy to other homes. If homes i shares more energy to home j and less energy to home k, when home j is compromised, home i may not be protected. If home i shares same energy to home j and k, any single home of j or k would not affect the protection of home i.

8.4.3 Problem Formulation of Defense Models

Because we need to hide power consumption from detection techniques, the amount of power consumption to be hidden is determined by detection techniques. We already gave the formulation to calculate the minimum power change to avoid those detection techniques. A simple approach is to generate random power consumption for each home to avoid those detection techniques. Then the power consumption to be hidden can be calculated by real power consumption and generated random power consumption. The problem is to generate random power consumption for each home, which would require homes to either i) use large batteries to randomize power consumption; or ii) exchange a lot of energy among homes. Solution i) is limited because large batteries cost lots of money and the capacity of batteries decreases with frequent charging and discharging operations. Solution ii) may be limited because each home has its own maximum power consumption from the power grid and energy transmission introduces some transmission loss. Thus we try to minimize the energy to be transferred to hide the power consumption information of each home.

We categorized homes in a hybrid AC-DC microgrid and define two terms as follows:

• **Supplier set *S***: A set of homes in a microgrid that need to hide power consumption information by decreasing their power consumption.

• **Demander set *D***: A set of homes in a microgrid that need to hide power consumption information by increasing their power consumption.

Based on the above definitions and defense models, we theoretically formulate the problem and illustrate it as a convex optimization problem. The design goal is to minimize energy transmission in alternative local power lines while hiding power consumption information of homes from both the utility company and their neighbors:

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{N} |p_i(t)| && (3.1)\\
s.t. \quad & \sum_{i=1}^{N} p_i(t) \cdot \eta_i(t) = 0 && (a)\\
& p_i(t) \geq \gamma \cdot \zeta_i(t), \;\; i \in S && (b)\\
& p_i(t) \leq \gamma \cdot \delta_i(t), \;\; i \in D && (c)\\
& p_i(t) + p_j(t) \geq \gamma \cdot \zeta_i(t), \;\; i \in S; \;\; j \neq i && (d)\\
& p_i(t) + p_j(t) \leq \gamma \cdot \delta_i(t), \;\; i \in D; \;\; j \neq i && (e)\\
& d_i(t) + p_i(t) \leq c_i, \;\; i = 1, ..., N && (f)
\end{aligned}
$$

$\eta_i(t)$ is the energy efficiency of home i at time t. If home i supplies energy to other homes, then $\eta_i(t) = 1$; if home i demands energy from other homes, then $\eta_i(t)$ is the transmission efficiency between home i and its suppliers. Constraints (b) and (c) indicate that the power change of each home is larger than the minimum power change to defend against detection models. Constraints (d) and (e) indicate that even with one neighbor's real power consumption data, the power consumption data of other homes still can be protected.

γ is used to control power consumption to hide. To hide appliances' usage patterns, γ should be larger than 1. Constraint (f) indicates that the real power consumption of each home should not exceed its own maximum power consumption. Because all the constraints are linear functions, which are always convex; and the objective function is also convex, our problem is a convex optimization problem.

8.4.4 Optimal Solutions

With the formulation of defense models, in this section, we develop an optimal solution with convergence and complexity analysis.

Barrier Method

To solve the convex optimization problem, we use the barrier method to provide an optimal solution. The key idea of the barrier method is to make the inequality constraints implicit in the optimization objective and convert the original problem into a sequence of linear equality constrained minimization problems. The solutions of these linear equality constrained minimization problems are called central points in the central path related to the original problem. The central point will be more accurately approximated to the optimal solution as the parameter s increases. For the minimization problem (3.1), we first need to remove all inequality constraints into a logarithmic barrier function $\phi(\boldsymbol{p})$:

$$\begin{aligned}\phi(\boldsymbol{p}) &= -\sum_{i=1}^{N} log(c_i - p_i(t) - d_i(t)) \\ &- \sum_{i \in S}(log(p_i(t) - \gamma \cdot \zeta_i(t)) + \sum_{j \neq i} log(p_i(t) + p_j(t) - \gamma \cdot \zeta_j(t))) \\ &- \sum_{i \in D}(log(-p_i(t) + \gamma \cdot \delta_i(t)) + \sum_{j \neq i} log(-p_i(t) - p_j(t) + \gamma \cdot \delta_j(t)))\end{aligned} \tag{8.8}$$

Then we write $f(\boldsymbol{p}) = \sum_{i=1}^{N} |p_i(t)|$ and rewrite the minimization problem with a certain parameter s as:

$$\begin{aligned} \min \quad & \psi(\boldsymbol{p}) = -s \cdot f(\boldsymbol{p}) + \phi(\boldsymbol{p}) && (4.1) \\ \textit{s.t.} \quad & \boldsymbol{Ap} = 0 && (a) \end{aligned}$$

where

$$A_{i,j} = \begin{cases} 1 & i = j \\ 0 & otherwise \end{cases} \tag{8.9}$$

The optimal solution to problem (4.1) is an approximation of the original problem. As s increases, the approximation is much closer to the optimal solution. At the centering step of the barrier method, Newton's method is employed to compute the central point.

The details of algorithm are described in Algorithm 14. First, we need to find a feasible starting point that satisfies the constraint of Equation (Line 1). Then we select proper α and β to apply Newton's method (Lines 2-3). With Newton's method, we calculate centering path until $\lambda^2/2 \geq \varepsilon_n$ is fulfilled (Lines 4-11). Then we update p and $p^\star(s)$ (Lines

Algorithm 14: Barrier Method

Input: Home's d, c and δ and ζ

Output: Home's p.

1: Find strictly feasible point p, $s \geq 0$, tolerance $\varepsilon \geq 0$, $\mu \geq 1$;
2: Centering path: Compute $p^{\star}(s)$;
3: Starting point p, subject to $Ap = 0$, tolerance $\varepsilon_n \geq 0$,$\alpha \in (0, 1/2)$,$\beta \in (0, 1)$;
4: Compute Δp and $\lambda = - \bigtriangledown \psi_s(p)\Delta p$;
5: **if** $\lambda^2/2 \geq \varepsilon_n$ **then**
6:     Go to Line 4;
7: **end if**
8: Backtracking line search on $\psi_s(p)$ and $h = 1$;
9: **while** $\psi_s(p + h\Delta p) \geq \psi_s(p) - \alpha h\lambda^2$ **do**
10:     $h = \beta h$;
11: **end while**
12: Update $p = p + h\Delta p$;
13: Update $p^{\star}(s) = p$;
14: **if** $(N + 2)/t \geq \varepsilon$ **then**
15:     Increase $s = \mu s$;
16:     Go to Line 2;
17: **end if**

12-13). Finally, we check if threshold ε is fulfilled, if not, increase t by μ; otherwise, the algorithm ends (Lines 14-17).

Solution Analysis

With the barrier method, it is guaranteed that we can achieve any desired accuracy we need. In this section, we analyze the number of iterations to converge to our desired accuracy and computation complexity.

Convergence Result. Given the desired accuracy $\varepsilon \geq 0$, the convergence speed can be calculated by using Theorem 8.2.

Theorem 8.2. *The centering steps to achieve a desired accuracy ε is:*

$$I = \frac{log(m/\varepsilon s_{(0)})}{log\mu} \tag{8.10}$$

where $s_{(0)}$ is the original s we choose and m is the number of inequality constraints which in our case is $N + 2|S| + 2|D|$, Convergence analysis for the barrier method is straightforward. Assuming that $sf_0 + \phi$ can be minimized by Newton's method for $s = \{s_{(0)}, \mu s_{(0)}, \mu^2 s_{(0)}, \cdots\}$, the duality gap after the initial centering step, and k additional centering steps, is $m/(\mu^k t^{(0)})$. Thus, the centering steps to achieve ε are $\frac{log(m/\varepsilon s_{(0)})}{log \mu}$. The detailed proof can be found in [20].

Algorithm Complexity. The computational complexity of the barrier method is mainly for the computation of Newton's method that needs matrix inversion with the complexity of $O(N^3)$. However, because we don't know whether a home should be a supplier or demander to minimize the total energy transmission, we need to try every combination of the homes' status, which can be 2^N combinations, then the complexity of the offline solution will be $O(2^N \cdot N^3)$.

8.4.5 Online Solutions

To reduce the complexity of the offline solution, we propose an efficient heuristic algorithm. Furthermore, the offline solution does not require the information of the number of compromised neighbors. The key idea is that when a home shares power to another home, the amount of power should be larger than its minimum power change, and less than enough to detect other homes' power consumption.

Theorem 8.3. *If k homes are selected to share energy to home i, to avoid detection from l compromised neighbors, $l + 1$ homes should at least share $\zeta_i(n)/(k - 1)$ to home i.*

Proof. k homes are selected to share energy with home i. To avoid detection from l com-

Algorithm 15: Heuristic Algorithm

Input: Home's δ and ζ
Output: Home's p.

Fetch home i with largest value of δ_i or ζ_i;
Calculate minimum energy transmission needed r_i and number of homes needed n_i in Algorithm 16;
if $i \in \mathbf{D}$ **then**
  for Home j in **F** and **S** **do**
    Fetch home j with largest value of ζ_j;
    $\zeta_j = \zeta_j - r_i/n_i$; $\delta_i = \delta_i - r_i/n_i$;
    Add (j, i) to energy sharing pair;
  end for
else
  for Home j in **D** **do**
    Fetch home j with largest value of δ_j;
    $\delta_j = \delta_j - r_i/n_i$; $\zeta_i = \zeta_i - r_i/n_i$;
    Add (i, j) to energy sharing pair;
  end for
end if

promised neighbors, any $k - l$ neighbors should share more energy than $\zeta_i(n)$. Then for first $k - l$ neighbors, at least one home k_1 shares more energy than $\zeta_i(n)/(k - l)$. Similarly, $k - l$ neighbors without home j should also share more energy than $\zeta_i(n)$ to avoid detection, therefore, there would be another l homes that share more energy than $\zeta_i(n)/(k - l)$. Finally, at least homes k_1 and other l homes share more energy than $\zeta_i(n)/(k - l)$ with home i. $\square$

Based on Theorem 8.3, we can have a basic idea of the total energy transmission with k homes shared to home i. This is because there are l homes which share more than $\zeta_i(n)/(k - l)$ and any $k - 1$ neighbors share more than $\zeta_i(n)$. Thus, for k homes, the total energy shared to home i should be larger than $\zeta_i(n) \cdot k/(k - l)$. Because $k/(k - l)$ decreases with the increase of k, it would be better to hide the real power consumption with more homes to reduce energy transmission. Based on the result, we propose an online solution

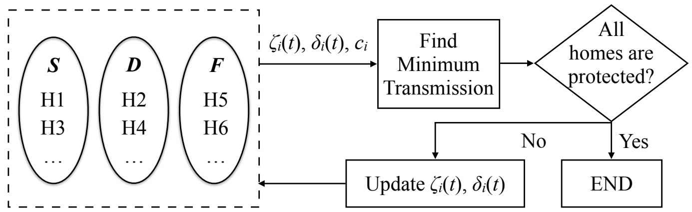


Figure 8.3: Overview of online algorithms (***S*** and ***D*** is the set of homes that need to hide consumption by increasing and decreasing their power consumption; ***F*** is the set of homes that do not need to hide power consumption)

for calculating real-time energy sharing pairs. The overview of online solution is shown in Figure 8.3. Because sharing energy with more homes can reduce energy transmission, we find the maximum number of homes to share energy with minimum transmission. Let r_i be the minimum transmission of n_i homes to protect home i, we have

$$r_i = \frac{n_i}{n_i - 1}\zeta_i + \sum_{\delta_j < \frac{1}{(n_i-1)}\zeta_i} (\frac{1}{(n_i - 1)}\zeta_i - \delta_j) * (n_i - 1) \tag{8.11}$$

Then we check if all the homes are protected, if yes, then our solution ends, otherwise, it assigns energy sharing pairs and update $\zeta_i(t)$ and $\delta_i(t)$ and then continue the process again until all the homes are protected.

Algorithm 15 is proposed to calculate the amount of real-time shared energy. First, we fetch the home with the largest value of δ_i or ζ_i (Line 1). We then use Algorithm 16 to calculate the minimum energy transmission needed and number of homes needed (Line 2). If home i is a demander, then we find a match in supplier set S and free set F, and update the power change they need (Lines 3-8). If home i is a supplier, then we find a match in demander set D, and update power changes they need (Lines 9-15).

Because we need to minimize energy transmission, another problem is to find the minimum energy transmission for each home. The detailed algorithm is described in Algorithm 16. Because a single home cannot provide protection by itself, the first step is to

Algorithm 16: Calculation of Minimum Energy Transmission

Input: A homes' d_i, c_i, δ_i and ζ of homes in **F** and **S**
Output: Minimum energy transmission r_i and number of homes needed n_i.

```
1:  Fetch two homes in F and S with largest ζ_j and ζ_k;
2:  r_i = δ_i; n_i = 2;
3:  for Home j in F and S do
4:      Fetch n_i + 1 homes j with largest value of ζ_j;
5:      Calculate r'_i for n_i + 1 homes based on Equation (8.11);
6:      if r'_i > θ · r_i then
7:          break;
8:      else
9:          r_i = r'_i, n_i = n_i + 1;
10:     end if
11: end for
```

calculate the minimum energy transmission with two homes (Lines 1-2). Then we increase the number of homes to hide the power consumption of home i (Lines 3-5). If the energy transmission increases, we select two homes for energy transmission; otherwise, we continue to find the minimum energy transmission by increasing the number of homes (Lines 6-11). θ is used to control the number of homes to hide the power consumption of home i.

8.4.6 Energy Sharing Control

With the solutions described in above, we can calculate how much energy each home should share to its neighbors to protect privacy. However, energy sharing in the microgrid introduces billing issues among homes. Thus, we present how homes only pay the utility company for their actual power consumption which does not include energy sharing.

To ensure homes share energy based on results generated by our solution, we develop a transmission protocol to schedule energy transmission. The detailed communication protocol is shown in Protocol 17. The controller first collects energy data for every interval w and runs Algorithm 15 and 16 to get the sharing results (Line 1). Then

Protocol 17: Energy Transmission Protocol

For controller

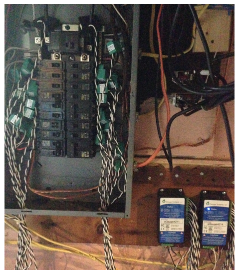
1: Collect energy data from homes and execute Algorithm 15 and 16 for every interval w;
2: Send TRANS_START and energy consumption instruction to homes;
3: If receive TRANS_END from home i, store energy consumption for home i;
4: If time w runs out, send TRANS_END to homes.

For every home

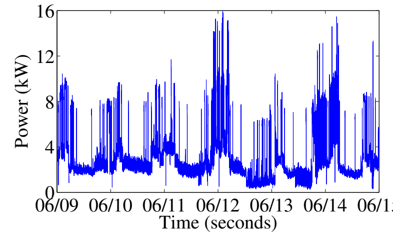
1: Send energy data to controller;
2: If receive TRANS_START, consumes energy according to instruction from controller.
3: If receive TRANS_END, send back energy consumption details.

it sends TRANS_START and power consumption results to the homes (Line 2). It monitors TRANS_END signal from homes to ensure power consumption for every home is correctly stored in order to calculate bills for each home (Line 3). The last thing for the central controller is to send TRANS_END to all homes after interval w (Line 4). For every home, it sends energy data to the controller at a new window (Line 1). It then waits for TRANS_START signal to start power consumption (Line 2) and sends back power consumption details to the controller after a TRANS_END signal. The controller needs consumption details to calculate bills for each home. The TRANS_START signal should contain the home id and amount of energy while TRANS_END signal should contain the home id and amount of energy each home consumes.

We then show that the utility company can still charge homes for their actual power consumption without energy sharing. The current price model of the utility company charges consumers based on power consumption at every window (for example every hour). The controller can add the amount of energy home i shared to other homes and get from other homes the aggregated amount of energy each home consumes in the previous window. The utility company can charge homes with the readings from smart meters for



(a) Setup of energy measurement



(b) Power consumption of one home in six days

Figure 8.4: Experiment setup and data collection

every window. Because the controller has the differences between real power consumption without sharing and readings from smart meters, it can charge homes with their real power consumption instead of reading from smart meters. Note in our chapter, we consider that the data in controller is not publicly available and the controller also deletes power consumption every window after billing calculation.

8.5 Implementation and Evaluation

In this section, we evaluate the performance of Shepherd. We collect the empirical data of power consumption from 40 homes and load events at one home. Then we evaluate the detection ratio and energy transmission of our solutions compared to existing approaches. Finally, we verify that our approach also works well using the microgrids with homes of similar power consumption patterns.

8.5.1 Data Collection

We deploy eGauge power meters at individual homes to collect the total power consumption data every one second. Experiment setup at one home is shown in Figure 8.4(a). In our simulation, we use the power consumption traces that we collected from 40 homes. We also

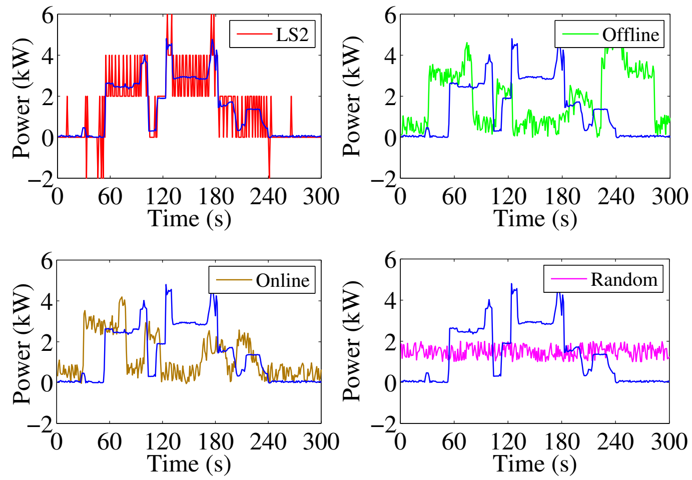
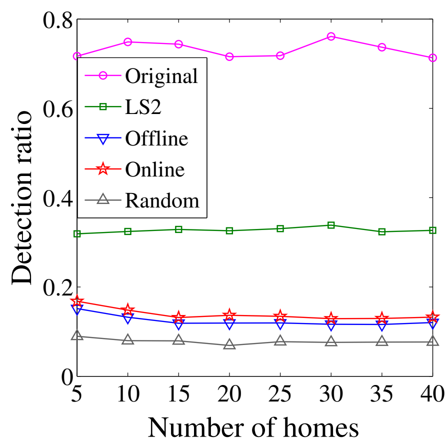
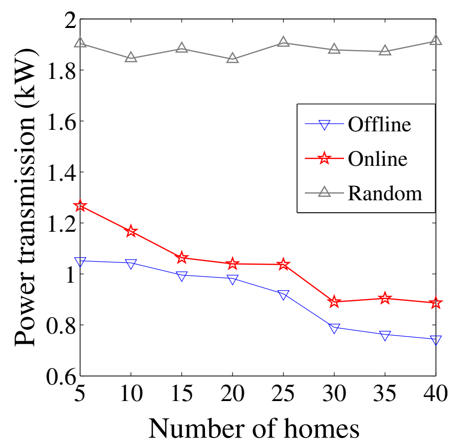


Figure 8.5: Original load and hidden load (blue lines in figure are original load)



(a) Detection Ratio



(b) Power transmission

Figure 8.6: Detection ratio, power transmission for 40 homes

collect the load events of one home to get the consumption signature of all the electrical loads (e.g., TV, oven, etc.). With the collected consumption signature, other homes' load events are detected as ground truth.

8.5.2 Evaluation Baseline and Metrics

Baselines. To verify the efficiency of Shepherd, we compare Shepherd with two baselines. i) Battery-based stepping algorithm (LS2) [120]. Yang et al. proposed four battery-based algorithms to hide power consumption. In our chapter, we select LS2 because LS2 performs the best in most scenarios. ii) Random energy sharing. Each home aims to randomize its power consumption by energy sharing.

Metrics. We use two metrics to evaluate the performance: i) **detection ratio**: the number of events detected divided by the total number of events; ii) **power transmission**: average power transmission over the additional AC line.

8.5.3 Basic Evaluation Results

In this section, we evaluate the effectiveness of our proposed offline and online solutions. All results are simulated with six days empirical data of power consumption. The battery we use to implement LS2 algorithm has $1kWh$ capacity and $2kW$ maximum charging rate. The parameter γ and θ are both selected as 1 in this set of simulations.

Power Consumption. We show the power consumption of four algorithms with comparison of the original loads in Figure 8.5. To make the difference between four algorithms' consumption visible, we show only 300 seconds of consumption data in one home. The LS2 tries to maintain power consumption at certain levels, thus its consumption can be only -2kW, 0kW, 2kW, 4kW and 6kW. However, we can still find that the shape of LS2 is similar to the original load. For offline and online solutions, we can find their consumption is totally different. Because their consumption is either sharing energy with other homes or shared by other homes. Most of the time, power consumption for the online and offline solutions are similar. For the random algorithm, each home tries to consume random amount of energy at any time, thus it has no relationship with the original load.

Detection Ratio. With power consumption results of four algorithms, we then use both edge and signature detection methods to detect load events. The average detection ratio of 40 homes is shown in Figure 8.6(a). Because with LS2, the power consumption shape is still similar to the original load, it can be detected by signature detection method. Because LS2 is adjusting power consumption at each home, detection ratio is not relevant to the number of homes. For offline and online solutions, the detection ratio gradually decreases with the increase in the number of homes. This is because with more homes in a microgrid, it is more likely you can find some homes to share energy. Power consumption with the random algorithm is not relevant with original consumption, thus detection ratio is low.

Power Transmission. Because we propose energy sharing to hide power consumption for homes, we also evaluate the amount of energy transmission for offline, online, and random algorithms. Even though random algorithm can achieve lower detection ratio, we show in Figure 8.6(b) that it costs nearly two times of energy transmission than online and offline solutions, which increases the burden of DC line and produces more transmission loss. This means that for one day, the random algorithm needs to transfer energy 20.47kWh more than offline and 15.28kWh more than online solutions. Assuming that energy transmission loss through AC line is as low as 1$\%$, it wastes 5-6kWh in a month.

8.5.4 Advanced Evaluation Results

In this section, we evaluate our design for homes with compromised neighbors in the community, similar power consumption patterns and different parameter settings to verify the robustness of our design.

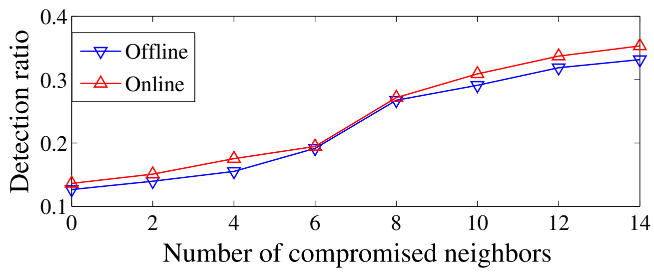


Figure 8.7: Detection ratio with compromised neighbors

Impact of Compromised Neighbors

Because homes share energy with each other to hide power consumption, homes can reveal a portion of their power consumption information to their neighbors in a microgrid. Thus we also evaluate if neighbors are compromised, whether homes in the microgrid can still hide power consumption. Because homes randomly share energy in random algorithms, neighbors do not reveal much information, we only evaluate online and offline solutions for 40 homes in the microgrid. The results are shown in Figure 8.7. With more compromised homes, the detection ratio of two solutions increases. However, even with 10 compromised homes (25% of total homes), the detection ratio of Shepherd is still lower than LS2.

Impact of similar consumption pattern

The consumption data we used in the above simulations comes from 40 different homes. Thus, their consumption patterns can be different and provide us an opportunity to balance energy transmission over AC line by consuming energy at different times. However, when homes have similar consumption patterns, homes may not be available for hiding power consumption for other homes. In this section, we use data of 40 homes with similar consumption pattern to verify that our design also works in this scenario.

We show the power consumption of online and offline algorithms in comparison

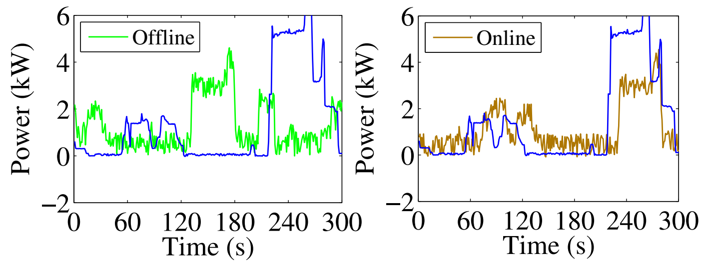
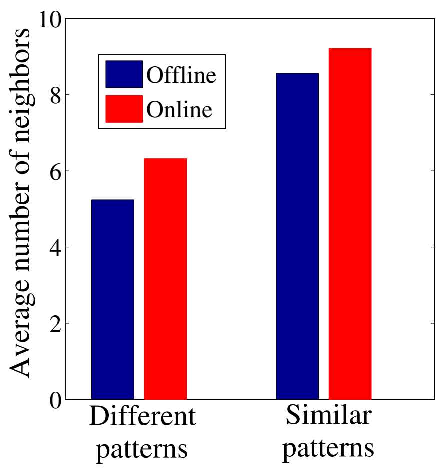
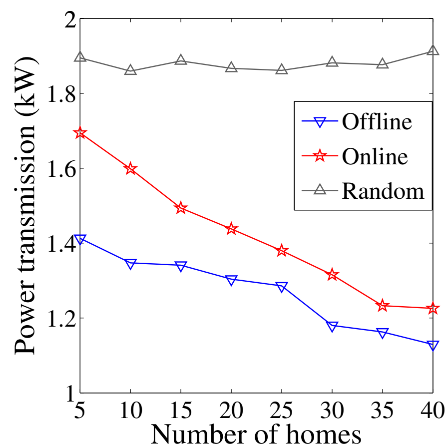


Figure 8.8: Original load and hidden load with similar consumption pattern (blue lines are original load)



(a) Avg. number of neighbors

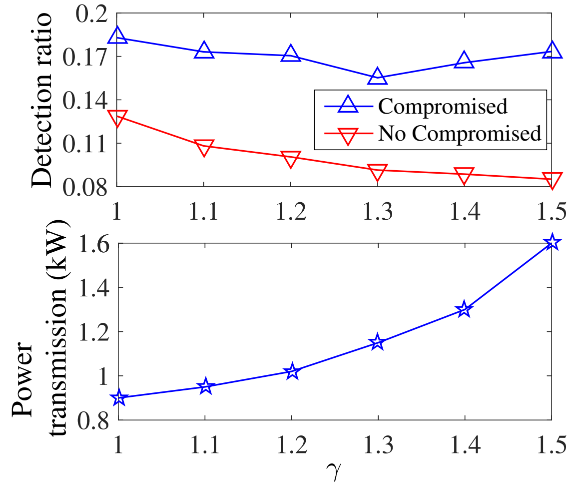


(b) Power transmission

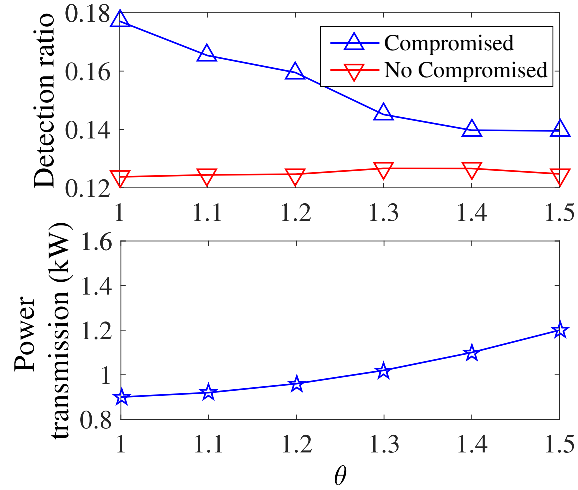
Figure 8.9: Average number of neighbors and power transmission with similar consumption pattern

to the original loads in Figure 8.8. For LS2 and random algorithm, the results are the similar because LS2 and random algorithm do not take advantage of neighbors' power consumption. However, for offline and online solutions, we can find their consumptions are quite similar to the consumption that shifts the original load over some time. This is because when homes have similar consumption patterns, our solutions shift power consumption for some time to avoid detection.

The results are almost the same for detection ratio of online and offline solutions. Thus, only average power transmission of 40 homes are shown in Figure 8.9(b). The average power transmission of online and offline solutions increases only with $0.1kW$ and $0.16kW$ and the gap between two solutions also increases. Overall, even in scenario



(a) Impact of γ



(b) Impact of θ

Figure 8.10: Impact of different parameters

where homes have similar consumption patterns in a microgrid, our proposed approach can achieve relatively low detection ratio and power transmissions. We also show the average number of neighbors for energy sharing for similar and different energy patterns in Figure 8.9(a). For homes with similar energy patterns, each home needs to find more homes to share energy because many homes have similar sharing needs. Thus, average number of neighbors for energy sharing increases from around 5-6 (different energy patterns) to 8-9 (similar energy patterns).

Impact of different parameters

In basic evaluation results, the detection ratio with or without compromised neighbors is still around 15%. In reality, some homes especially some business buildings may need to protect their information better. In our design, we allow the user to tune the parameter γ to achieve even lower detection ratio and the parameter θ to achieve lower detection ratio with one compromised neighbor.

The detailed results for impact of γ and θ are shown in Figure 8.10(a) and 8.10(b).

With a larger γ, homes try to hide more energy from real power consumption, thus the detection ratio decreases. However, it does not help to decrease the detection ratio with compromised neighbors. This is because that with larger γ, every home hides more energy but still with the same neighbors. Then with compromised neighbors, the detection ratio is still high. For average power transmission, it increases since more energy transmission is needed to hide more energy.

With a larger θ, homes try to hide energy with more homes, thus even with compromised neighbors, the detection ratio decreases. However, it does not try to hide more energy for any home, thus the detection ratio without compromised neighbors is stable. For average power transmission, it increases with larger θ. This is because it needs more energy transmission when hiding energy with more homes. However, the increase of average power transmission for a larger θ (1.32kW for $\theta = 1.5$) is much less than with larger γ (1.68kW for $\gamma = 1.5$). However, combined with larger γ and θ, our design can achieve low detection ratio both with and without compromised neighbors.

8.6 Related Work

This work aims to protect the privacy of power consumption data in a microgrid. The related work includes:

• **Non-Intrusive Load Monitoring**. The large-scale placement of smart meters has introduced leakage of private and valuable information about occupants' activities [29]. NILM algorithms have been widely used in the research of residential settings to reveal the usage of individual appliances with consumption data [96]. In [68], NILM algorithms are extended to evaluate the threat to individual privacy by considering the potential disclosure from smart-meter data. A statistical technique is used to develop a simple approach to discover people's life patterns [4]. In this chapter, we develop a new detection technique based on the consumption signature of appliances that achieves a higher detection ratio.

• **Battery-Based Load Hiding**. The basic idea of BLH is to use a rechargeable battery to store and supply power to home appliances at strategic times to hide the appliances' consumption from smart meters [120]. The BE algorithm [109] tries to avoid charging the external load whenever possible, and when the actual demand is different from the external load, the battery can be charged or discharged to counteract the difference. The NILL algorithm [95] has three states and attempts to maintain a different constant load for each state.

• **Privacy in Sensing Systems**. With the large deployment of different types of sensors, the privacy issue of sensing systems becomes an important problem [10]. In [98], a theoretical framework is proposed to allow users to quantify the utility-privacy tradeoff in smart meter data. The protocols for processing smart meter readings while preserving user privacy is designed in [31] .

Instead of using batteries, we propose a battery-free approach, which addresses the limitations of the above approaches. By leveraging the alternative local power line built in a microgrid, our online and offline solutions can enable homes to share energy with their neighbors to hide the real power consumption from the malicious third parties.

8.7 Conclusion

In this chapter, we study the privacy leakage problem in hybrid AC-DC microgrids and discover two novel vulnerabilities for malicious users to obtain power consumption information of individual homes without occupants' authentication. To protect the occupants' privacy, we leverage the unique feature of hybrid AC-DC microgrids and propose Shepherd, a privacy protection framework, to allow homes in a microgrid to coordinate with each other to hide power consumption information. We analyze the adversarial models in a single home and with compromised neighbors and propose corresponding defense models to defend from these two models. With the empirical data from more than 40 homes, we conduct extensive system evaluations. Results show that Shepherd can i) significantly reduce the detection ratio from 33% to 13%, and ii) effectively hide consumption information even with 25% compromised neighbors.

CHAPTER 9

SUMMARY AND FUTURE WORK

9.1 Dissertation Summary

This dissertation has explored to address three challenges: data and energy and privacy management in microgrids. We have proposed a set of techniques to address these challenges by a data-driven approaches without active user involvement and user inconvenience, while improving energy efficiency in microgrids.

Data Acquisition. To achieve the stability of the microgrid, power quality through the power lines needs to be monitored for balancing demand and generation. However, the unreliable data collection makes the control very hard and existing approaches (PMU) are very expensive. To address these issues, we design an accurate real-time energy data sensing hardware to sense the voltage, frequency and phase angle in each home. We propose a novel data management technique to reconstruct the missing data caused by sensing error. Through extensive experiments and simulations, we show that our design realized an accuracy of 1.7 mHz and 0.01 rad for frequency and phase angle monitoring, respectively. Our data management can reconstruct the missing data with more than 99% accuracy. With the hardware PowerQM, we present a middleware E-Sketch to to collect high accurate data while minimize the data storage and communication overhead. Results indicate i) our design can reduce data storage space significantly by 90% with more than 99% accuracy of second- level power consumption on average for a single home, and ii) our design can achieve even more than 99.8% accuracy on average for aggregated power consumption of 30 homes.

Data Analytics. To the best of our knowledge, this is the first work to utilize the detailed power consumption in individual homes to help power consumption prediction in smart grids. We show that the detailed power consumption patterns in each home can significantly improve prediction accuracy of power consumption in smart grids. In this paper, we propose M-Pred to learn energy consumption pattern of individual homes from their energy consumption data and then utilize these patterns to predict the power consumption in smart grids. We conducted extensive system evaluations with 726 homes minute-level power consumption data for more than 1 year. The results show that our design can provide accurate real-time energy consumption with negligible errors (e.g., Mean Absolute Percentage Error is 2.12%).

Data Driven Computation and Control. To achieve better energy management in microgrids, we investigated two approaches: i) energy sharing among homes, and ii) demand and generation scheduling. To ensure the efficiency of the energy sharing, we i) created a novel energy sharing system, ii) developed a greedy matching algorithm, and iii) designed a practical transmission scheduling method. We evaluated our system using empirical traces of harvested solar energy and home energy consumption in Amherst, MA. Through extensive simulations, we verified that our system can reduce AC energy costs of the whole microgrid by more than 20% under different TOU price models and can still reduce AC energy costs even when homes have similar energy consumption patterns. We also propose scheduling algorithm of the workload of appliances in each home and the operations of the local generator based on the collected and predicted data. Through extensive experiments and simulations, we show that our design can balance power demand and generation and reduce operation cost by 23%.

Data Privacy Protection. Finally, we study the privacy leakage problem in hybrid AC-DC microgrids and discover two novel vulnerabilities for malicious users to obtain power consumption information of individual homes without occupants authentication. To protect the occupants privacy, we leverage the unique feature of hybrid AC-DC microgrids and propose Shepherd, a privacy protection framework, to allow homes in a microgrid to coordinate with each other to hide power consumption information. We analyze the adversarial mod- els in a single home and with compromised neighbors and propose corresponding defense models to defend from these two models. With the empirical data from more than 40 homes, we conduct extensive system evaluations. Results show that Shepherd can i) significantly reduce the detection ratio from 33% to 13%, and ii) effectively hide consumption information even with 25% compromised neighbors.

9.2 Future Work

Here we present some of the future research directions that have emerged from the work in this dissertation.

Energy Data Visualization. In order to incentivize the homeowners to participate our designed program to minimize both their utility bills and operation cost of utility companies, we plan to conduct energy data visualization for homeowners. The main goal of the data visualization is to provide homeowners direct information about i) how they consume energy everyday; ii) which appliances in homes contribute most of the electricity bills; iii) is there any abnormal behaviors of appliances in homes? iv) how much electrical bills can be reduces with our demand scheduling design?

Generic Data Management for IoT Devices. With these millions of IoT devices deployed in the environment and connected to Internet, the enormous volumes of sensor data will be generated, and trigger an era of big data for IoT. Many IoT applications are built using a data-driven approach. However, to deal with sensing data for different applications, it requires much domain specific knowledge. To ensure the sensing data quality inconstancy overtime and specific application Quality of Service (QoS) requirements, we shall investigate how to dynamically provide feedback to correlation mining by evaluating the latest results and non-functional properties of the application. The feedback controller interprets the relationship between the selected data and the application performance using the selected data. For deterministic applications where ground truth exists, we investigate the correlations between data errors and application performance to provide feedback. For probabilistic applications, because there is no ground truth for us to evaluate the performance of application, we investigate the correlations between data selection changes and application performance changes to provide feedback. Other or multiple QoS requirements could be supported with extensions.

Integration of Electrical Vehicles in Power Grids. Nowadays, with the rapid development of electrical vehicles, there would be high impact on existing power grids.

Meanwhile, transportation problem is not only limited to the route planning but also energy scheduling and allocation. Therefore, in the next step, I plan to study the interconnections between different cyber-physical systems in smart cities, such as energy, transportation. First, I will try to investigate how existing power grids can cope with the large number of electrical vehicles. Secondly, I will study how the properties of electrical vehicles would affect route planning in transportation system.

BIBLIOGRAPHY

[1] A. Chakrabortty, "Wide-area control of large power systems using dynamic clustering and tcsc-based redesigns," in *IEEE Transctions on Smart Grid, Vol. 3, No. 3, 2012*, 2012.

[2] A. Chakrabortty and A. Salazar, "Building a dynamic electro-mechanical model for the pacific ac intertie using distributed synchrophasor measurements," in *European Transactions on Electric Power: Special Issue on PMU Applications, Vol. 21, No. 4*, 2011, pp. 1657–1672.

[3] A. Chakrabortty, J. H. Chow and A. Salazar, "A measurement-based framework for dynamic equivalencing of power systems using wide-area phasor measurements," in *IEEE Transactions on Smart Grid, Vol. 1, No. 2*, 2011, pp. 68–81.

[4] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin, "Private memoirs of a smart meter," in *BuildSys*, 2010.

[5] "A Smart Grid Strategy for Assuring Reliability of the Western Grid . U.S. Department of Energy," 2010.

[6] A. Thatte, and L. Xie, “Towards a unified operational value index of energy storage in smart grid environment,” in *IEEE Transactions on Smart Grid, Vol. 3, No. 3*, 2012, pp. 1418–1426.

[7] M. A. Adnan, B. Narayanaswamy, and R. K. Gupta, “Online reservation and deferral of EV charging tasks to reduce energy use variability in smart grids,” in *SmartGridComm*, 2014.

[8] Y. Agarwal, B. Balaji, S. Dutta, R. Gupta, and T. Weng, “Duty-cycling buildings aggressively: The next frontier in HVAC control,” in *IPSN*, 2011.

[9] H. K. Alfares and M. Nazeeruddin, “Electric load forecasting: Literature survey and classification of methods,” *International Journal of Systems Science*, vol. 33, no. 1, pp. 23–34, 2002.

[10] M. Ali, E. Al-Shaer, and Q. Duan, “Randomizing ami configuration for proactive defense in smart grid,” in *SmartGridComm*, 2013.

[11] S. Aman, Y. Simmhan, and V. K. Prasanna, “Holistic measures for evaluating prediction models in smart grids,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 2, pp. 475–488, 2015.

[12] M. Arlitt, C. Bash, S. Blagodurov, Y. Chen, T. Christian, D. Gmach, C. Hyser, N. Kumari, Z. Liu, M. Marwah, *et al.*, “Towards the design and operation of net-zero energy data centers,” in *IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*, 2012.

[13] A. Aswani, N. Master, J. Taneja, A. Krioukov, D. Culler, and C. Tomlin, “Energy-efficient building HVAC control using hybrid system LBMPC,” in *Proceedings of IFAC Conference on Nonlinear Model Predictive Control*, 2012.

[14] B. Balaji, J. Xu, A. Nwokafor, R. Gupta, and Y. Agarwal, “Sentinel: Occupancy based HVAC actuation using existing WiFi infrastructure within commercial buildings,” in *SenSys*, 2013.

[15] P. Balister, B. Bollobas, A. Anandkumar, and A. Willsky, “Energy-latency tradeoff for in-network function computation in random networks,” in *INFOCOM*, 2011.

[16] S. Barker, A. Mishra, D. Irwin, P. Shenoy, and J. Albrecht, “Smartcap: Flattening peak electricity demand in smart homes,” in *PerCom*, 2012.

[17] S. Barker, S. Kalra, D. Irwin, and P. Shenoy, “Empirical characterization and modeling of electrical loads in smart homes,” in *IGCC*, 2013.

[18] R. Basmadjian, H. D. Meer, R. Lent, and G. Giuliani, “Cloud computing and its interest in saving energy: The use case of a private cloud,” *Journal of Cloud Computing: Advances, Systems and Applications*, 2012.

[19] M. Behl, M. Aneja, H. Jain, and R. Mangharam, “Enroute: An energy router for energy-efficient buildings,” in *Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on*, 2011, pp. 125–126.

[20] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University, 2004.

[21] M. Buevich, N. Rajagopal, and A. Rowe, “Hardware assisted clock synchronization for real-time sensor networks,” in *RTSS*, 2013.

[22] M. Buevich, D. Schnitzer, T. Escalada, A. Jacquiau-Chamski, and A. Rowe, "Fine-grained remote monitoring, control and pre-paid electrical service in rural micro-grids," in *IPSN*, 2014.

[23] CA.GOV, *State of California Executive Order S-21-09*, http://gov.ca.gov/executive-order/13269, 2009.

[24] "California ISO. CAISO demand response resource user guide," *Tech. Rep.*, 2007.

[25] S.-C. Chan, K. M. Tsui, H. Wu, Y. Hou, Y.-C. Wu, and F. F. Wu, "Load/price forecasting and managing demand response for smart grids: Methodologies and challenges," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 68–85, 2012.

[26] C. Chatfield, *The analysis of time series: An introduction*. CRC press, 2016.

[27] C. Chen, D. J. Cook, and A. S. Crandall, "The user side of sustainability: Modeling behavior and energy usage in the home," *Pervasive and Mobile Computing*, vol. 9, no. 1, pp. 161–175, 2013.

[28] Z. Chen, L. Wu, and Y. Fu, "Real-time price-based demand response management for residential appliances via stochastic optimization and robust optimization," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1822–1831, 2012.

[29] S. Clark, H. Mustafa, B. Ransford, J. Sorber, K. Fu, and W. Xu, "Current events: Identifying webpages by tapping the electrical outlet," *Lecture Notes in Computer Science*, vol. 8134, pp. 700–717, 2013.

[30] J. Crowcroft, "Cutting the energy cost of tv content distribution by 5, by understanding the popularity of the top ten programs," in *E-Energy*, 2012.

[31] G. Danezis, C. Fournet, M. Kohlweiss, and S. Zanella-Béguelin, "Smart meter aggregation via secret-sharing," in *Proceedings of the First ACM Workshop on Smart Energy Grid Security*, 2013.

[32] P. De Martini, A. Wierman, S. Meyn, and E. Bitar, "Integrated distributed energy resource pricing and control," 2012.

[33] S. DeBruin, B. Ghena, Y.-S. Kuo, and P. Dutta, "Powerblade: A low-profile, true-power, plug-through energy meter," in *Sensys*, 2015.

[34] N. Deng, C. Stewart, and J. Li, "Concentrating renewable energy in grid-tied datacenters," in *IEEE International Symposium on Sustainable Systems and Technology*, 2011. DOI: 10.1109/ISSST.2011.5936855.

[35] DOE, *Building Energy Data Book*, http://buildingsdatabook.eren.doe.gov, 2014.

[36] Y. Dong, D. Yau, and J.-S. Lui, "An opportunistic scheduler to balance performance measures and energy consumption in wireless networks: Design and implementation," in *First International Conference on Wireless Internet*, 2005. DOI: 10.1109/WICON.2005.8.

[37] DSIRE, *Database of State Incentives for Renewables and Efficiency*, 2010. [Online]. Available: http://www.dsireusa.org.

[38] Egauge, *Http://www.egauge.net*, 2014.

[39] "Energex. https://www.energex.com.au."

[40] Fangjian Jin, Kang G. Shin, “Pack sizing and reconfiguration for management of large-scale batteries,” in *Proceedings of 2012 International Conference on Cyber-Physical Systems*, 2012.

[41] M. Frincu, C. Chelmis, M. U. Noor, and V. Prasanna, “Accurate and efficient selection of the best consumption prediction method in smart grids,” in *IEEE International Conference on Big Data*, 2014.

[42] L. Gan, A. Wierman, U. Topcu, N. Chen, and S. H. Low, “Real-time deferrable load control: Handling the uncertainties of renewable generation,” in *E-Energy*, 2013.

[43] K. Garbesi, “Catalog of dc appliances and power systems,” 2012.

[44] I. Goiri, W. Katsak, K. Le, T. D. Nguyen, and R. Bianchini, “Parasol and greenswitch: Managing datacenters powered by renewable energy,” in *Proceedings of the Eighteenth International Conference on Architectural Support for Programming Languages and Operating Systems*, Houston, Texas, USA, 2013, ISBN: 978-1-4503-1870-9. DOI: 10.1145/2451116.2451123. [Online]. Available: http://doi.acm.org/10.1145/2451116.2451123.

[45] S. Haben, J. Ward, D. V. Greetham, C. Singleton, and P. Grindrod, “A new error measure for forecasts of household-level, high resolution electrical energy consumption,” *International Journal of Forecasting*, vol. 30, no. 2, pp. 246–256, 2014.

[46] M. T. Hagan and S. M. Behr, “The time series approach to short term load forecasting,” *IEEE Transactions on Power Systems*, vol. 2, no. 3, pp. 785–791, 1987.

[47] G. Hart, “Residential energy monitoring and computerized surveillance via utility power flows,” 2, vol. 8, 1989, pp. 12–16. DOI: 10.1109/44.31557.

[48] H. S. Hippert, C. E. Pedreira, and R. C. Souza, “Neural networks for short-term load forecasting: A review and evaluation,” *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, 2001.

[49] H. Hlavacs, K. Hummel, R. Weidlich, A. Houyou, A. Berl, and H. Meer, “Energy efficiency in future home environments: A distributed approach,” *Home Networking*, vol. 256, pp. 69–84, 2008.

[50] J. Hsu, P. Mohan, X. Jiang, J. Ortiz, S. Shankar, S. Dawson-Haggerty, and D. Culler, “Hbci: Human-building-computer interaction,” in *BuildSys*, 2010.

[51] “IEEE standard for a precision clock synchronization protocol for networked measurement and control systems,” *IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002)*, 2008.

[52] X. Jiang, P. Joseph, and D. Culler, “Perpetual environmentally powered sensor networks,” in *IPSN/SPOTS*, 2005.

[53] X. Jiang, M. Van Ly, J. Taneja, P. Dutta, and D. Culler, “Experiences with a high-fidelity wireless building energy auditing network,” in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, Berkeley, California, 2009, pp. 113–126, ISBN: 978-1-60558-519-2.

[54] X. Jiang, M. Van Ly, J. Taneja, P. Dutta, and g. Culler, “Experiences with a high-fidelity wireless building energy auditing network,” in *SenSys*, 2009.

[55] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, “Power management in energy harvesting sensor networks,” in *ACM Transactions on Embedded Computing Systems*, vol. 6, 2007.

[56] S. A. Kazarlis, A. Bakirtzis, and V. Petridis, “A genetic algorithm solution to the unit commitment problem,” *IEEE Transactions on Power Systems*, vol. 11, no. 1, pp. 83–92, 1996.

[57] J. Z. Kolter and J. Ferreira Jr, “A large-scale study on predicting and contextualizing building energy usage,” 2011.

[58] F. Kong, C. Dong, X. Liu, and H. Zeng, “Quantity versus quality: Optimal harvesting wind power for the smart grid,” *Proceedings of the IEEE*, vol. 102, no. 11, pp. 1762–1776, 2014.

[59] L. Kong, M. Xia, X.-Y. Liu, M.-Y. Wu, and X. Liu, “Data loss and reconstruction in sensor networks,” in *INFOCOM, 2013 Proceedings IEEE*, IEEE, 2013, pp. 1654–1662.

[60] L. Xie, Y. Gu, A. Eskandari, and M. Ehsani, “Fast mpc-based coordination of wind power and battery energy storage systems,” in *Journal of Energy Engineering, Vol. 138, No. 2*, 2012, pp. 43–53.

[61] T.-T. Lai, W.-J. Chen, K.-H. Li, P. Huang, and H.-H. Chu, “Triopusnet: Automating wireless sensor network deployment and replacement in pipeline monitoring,” in *IPSN*, 2012.

[62] D. Larruskain, I. Zamora, A. Mazon, O. Abarrategui, and J. Monasterio, “Transmission and distribution networks: Ac versus dc,” in *9th Spanish-Portuguese Congress on Electrical Engineering*, 2005.

[63] R. H. Lasseter and P. Paigi, “Microgrid: A conceptual solution,” in *IEEE Power Electronics Specialists Conference*, 2004.

[64] S. Lee, D. Ahn, S. Lee, R. Ha, and H. Cha, "Personalized energy auditor: Estimating personal electricity usage," in *PerCom*, 2014.

[65] F. Li, T. Xiang, Z. Chi, J. Luo, L. Tang, L. Zhao, and Y. Yang, "Powering indoor sensing with airflows: A trinity of energy harvesting, synchronous duty-cycling, and sensing," in *SenSys*, 2013.

[66] H. Li, D. Liang, L. Xie, G. Zhang, and K. Ramamritham, "Tl-tree: Flash-optimized storage for time-series sensing data on sensor platforms," in *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, 2012.

[67] H. Lim, A. Kansal, and J. Liu, "Power budgeting for virtualized data centers," in *USENIX ATC*, 2011.

[68] M. Lisovich, D. Mulligan, and S. Wicker, "Inferring personal information from demand-response systems," *IEEE Security Privacy*, vol. 8, no. 1, pp. 11–20, 2010, ISSN: 1540-7993. DOI: 10.1109/MSP.2010.40.

[69] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2014.

[70] J. Lu, T. Sookoor, V. Srinivasan, G. Gao, B. Holben, J. Stankovic, E. Field, and K. Whitehouse, "The smart thermostat: Using occupancy sensors to save energy in homes," in *SenSys*, 2010.

[71] L. Lu, J. Tu, C.-K. Chau, M. Chen, and X. Lin, "Online energy generation scheduling for microgrids with intermittent energy sources and co-generation," in *Proceedings of the ACM SIGMETRICS*, 2013.

[72] M. LeMay, J. J. Haas, and C. A. Gunter, “Collaborative recommender systems for building automation,” in *Hawaii International Conference on System Sciences*, 2009.

[73] C. A. Macana, N. Quijano, and E. Mojica-Nava, “A survey on cyber physical energy systems and their applications on smart grids,” in *IEEE PES Conference on Innovative Smart Grid Technologies*, 2011.

[74] K. Malladi, I. Shaeffer, L. Gopalakrishnan, D. Lo, B. Lee, and M. Horowitz, “Rethinking dram power modes for energy proportionality,” in *IEEE/ACM International Symposium on Microarchitecture*, 2012.

[75] V. Mann, A. Kumar, P. Dutta, and S. Kalyanaraman, “Vmflow: Leveraging vm mobility to reduce network power costs in data centers,” *NETWORKING*, 2011.

[76] F. Martinez Alvarez, A. Troncoso, J. C. Riquelme, and J. S. Aguilar Ruiz, “Energy time series forecasting based on pattern sequence similarity,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 8, pp. 1230–1243, 2011.

[77] MAXIM Integrated, “MAX3421E, USB peripheral/host controller with SPI interface,” 2007, Copyright Maxim Integrated Products.

[78] M.D. Ilic, L. Xie and J. Joo, “Efficient coordination of wind power and price-responsive demand part i: Theoretical foundations,” in *IEEE Transactions on Power Systems Vol. 26, No. 4*, 2011, pp. 1875–1884.

[79] “Microgrid deployment tracker 4Q15,” *Navigant Research*, 2015.

[80] D. L. Mills, “Internet time synchronization: The network time protocol,” *IEEE Transactions on Communications*, vol. 39, no. 10, pp. 1482–1493, 1991.

[81] A. Mishra, D. Irwin, P. Shenoy, J. Kurose, and T. Zhu, “Smartcharge: Cutting the electricity bill in smart homes with energy storage,” in *E-Energy*, 2012.

[82] A. Mishra, D. Irwin, P. Shenoy, and T. Zhu, “Scaling distributed energy storage for grid peak reduction,” in *E-Energy*, ACM, 2013, pp. 3–14.

[83] R. Mittal, A. Kansal, and R. Chandra, “Empowering developers to estimate app energy consumption,” in *MOBICOM*, 2012.

[84] 3. G. P. P. (T. S. G. R. A. Network, “Evolved universal terrestrial radio access (E-UTRA); physical channels and modulation (release 10) TS 36.211 (version 10.0.0),” Tech. Rep., 2010.

[85] “New York ISO. Emergency demand response program manual,” *Tech. Rep.*, 2013.

[86] T. Nghiem, M. Behl, G. Pappas, and R. Mangharam, “Green scheduling: Scheduling of control systems for peak power reduction,” in *IGCC*, 2011.

[87] “Ontario’s Power Authority. Ontario’s long term energy plan,” 2010.

[88] K. K. Rachuri, M. Musolesi, and C. Mascolo, “Energy accuracy trade-offs in querying sensor data for continuous sensing mobile systems,” in *UbiComp*, 2010.

[89] M. Rahimi, H. Shah, G. Sukhatme, J. Heidemann, and D. Estrin, “Studying the feasibility of energy harvesting in mobile sensor networks,” in *IEEE ICRA*, 2003.

[90] S. D. Ramchurn, P. Vytelingum, A. Rogers, and N. R. Jennings, "Putting the "smarts" into the smart grid: A grand challenge for artificial intelligence," *Communications of the ACM*, vol. 55, no. 4, pp. 86–97, 2012.

[91] V. J. Reddi, B. C. Lee, T. Chilimbi, and K. Vaid, "Mobile processors for energy-efficient web search," *ACM Transactions on Computing Systems*, vol. 29, no. 3, 9:1–9:39, 2011, ISSN: 0734-2071. DOI: 10.1145/2003690.2003693. [Online]. Available: http://doi.acm.org/10.1145/2003690.2003693.

[92] *Residential Energy Consumption Survey (RECS)*, http://www.eia.gov/consumption/residential, 2013.

[93] I. Rouf, H. Mustafa, M. Xu, W. Xu, R. Miller, and M. Gruteser, "Neighborhood watch: Security and privacy analysis of automatic meter reading systems," in *CCS*, 2012.

[94] A. Rowe, M. Berges, and R. Rajkumar, "Contactless sensing of appliance state transitions through variations in electromagnetic fields," in *BuildSys*, 2010.

[95] S. McLaughlin and P. McDaniel and W. Aiello, "Protecting consumer privacy from electric load monitoring," in *CCS*, 2011.

[96] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd, "At the flick of a switch: Detecting and classifying unique electrical events on the residential power line," in *Ubicomp*, 2007.

[97] D. Salomon, G. Motta, and D. Bryant, *Data Compression: The Complete Reference*. Springer, 2007.

[98] L. Sankar, S. Rajagopalan, S. Mohajer, and H. Poor, "Smart meter privacy: A theoretical framework," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 837–846, 2013.

[99] S. M. Schoenung, "Energy storage systems cost update [a study for the doe energy storage systems program]," *Tech. Rep. SAND2011-2730, Sandia National Laboratories*, 2011.

[100] N. Sharma, J. Gummeson, D. Irwin, and P. Shenoy, "Cloudy Computing: Leveraging Weather Forecasts in Energy Harvesting Sensor Systems," in *SECON*, 2010.

[101] N. Sharma, J. Gummeson, D. Irwin, T. Zhu, and P. Shenoy, "Leveraging weather forecasts in renewable energy systems," *Sustainable Computing: Informatics and Systems*, 2014.

[102] W. Shen, V. Babushkin, Z. Aung, and W. L. Woon, "An ensemble model for day-ahead electricity demand time series forecasting," in *E-Energy*, 2013.

[103] T. Short, *Electric Power Distribution Handbook*. Taylor & Francis, 2003.

[104] S. Sojoudi and S. H. Low, "Optimal charging of plug-in hybrid electric vehicles in smart grids," in *Proceedings of the 2011 IEEE Power & Energy Society General Meeting*, 2011.

[105] "Southern California Edison. 10 day average baseline and "day-off" adjustment," *Tech. Rep.*, 2011.

[106] P. Srikantha, S. Keshav, and C. Rosenberg, “Distributed control for reducing carbon footprint in the residential sector,” in *IEEE Third International Conference on Smart Grid Communications*, 2012.

[107] W. Su and J. Wang, “Energy management systems in microgrid operations,” *The Electricity Journal*, vol. 25, no. 8, pp. 45–60, 2012.

[108] SunShot, *Sunshot vision study*, https://energy.gov/eere/sunshot/sunshot-2030, 2016.

[109] T. Carpenter, S. Singla, P. Azimzadeh, and S. Keshav, “The impact of electricity pricing schemes on storage adoption in ontario,” in *E-Energy*, 2012.

[110] USCB, *Http://www.census.gov*, 2012.

[111] N. Vallina-Rodriguez, P. Hui, J. Crowcroft, and A. Rice, “Exhausting battery statistics: Understanding the energy demands on mobile handsets,” *MobiHeld*, 2010.

[112] G. Wang, M. Negrete-Pincetic, A. Kowli, E. Shafieepoorfard, S. Meyn, and U. Shanbhag, “Real-time prices in an entropic grid,” in *IEEE Innovative Smart Grid Technologies*, 2012.

[113] K. Wang, F. Ciucu, C. Lin, and S. H. Low, “A stochastic power network calculus for integrating renewable energy sources into the power grid,” *IEEE Journal on Selected Areas in Communications*, 2012.

[114] L. Wang, J. Fernandez, J. Burgett, R. W. Conners, and Y. Liu, “An evaluation of network time protocol for clock synchronization in wide area measurements,” in *IEEE PES General Meeting*, 2008.

[115] T. Wei, Q. Zhu, and N. Yu, “Proactive demand participation of smart buildings in smart grid,” *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1392–1406, 2016.

[116] T. Wei, Q. Zhu, and M. Maasoumy, “Co-scheduling of hvac control, ev charging and battery usage for building energy efficiency,” in *Proceedings of the 2014 IEEE/ACM International Conference on Computer-Aided Design*, IEEE Press, 2014, pp. 191–196.

[117] M. Weiss, A. Helfenstein, F. Mattern, and T. Staake, “Leveraging smart meter data to recognize home appliances,” in *IEEE International Conference on Pervasive Computing and Communications*, 2012.

[118] L. Xiang, J. Luo, and C. Rosenberg, “Compressed data aggregation: Energy-efficient and high-fidelity data collection,” *IEEE/ACM Transactions on Networking*, vol. 21, no. 6, pp. 1722–1735, 2013.

[119] Y. Xiao, Y. Cui, P. Savolainen, M. Siekkinen, A. Wang, L. Yang, A. Ylä-Jääski, and S. Tarkoma, “Modeling energy consumption of data transmission over Wi-Fi,” *IEEE TMC*, vol. 13, no. 8, pp. 1760–1773, 2014.

[120] W. Yang, N. Li, Y. Qi, W. Qardaji, S. McLaughlin, and P. McDaniel, “Minimizing private data disclosures in the smart grid,” in *CCS*, 2012.

[121] P. Yi, T. Zhu, B. Jiang, B. Wang, and D. Towsley, “An energy transmission and distribution network using electric vehicles,” in *ICC*, 2012.

[122] P. Yi, T. Zhu, G. Lin, and Q. Zhang, “Routing renewable energy using electric vehicles in mobile electrical grid,” in *MASS*, 2013.

[123] P. Yi, T. Zhu, Q. Zhang, Y. Wu, and J. Li, “A denial of service attack in advanced metering infrastructure network,” in *ICC*, 2014.

[124] G. P. Zhang, “Time series forecasting using a hybrid arima and neural network model,” *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[125] L. Zhang, Z. Li, C. Wu, and M. Chen, “Online algorithms for uploading deferrable big data to the cloud,” in *INFOCOM*, 2014.

[126] Q. Zhang, T. Zhu, Y. Ping, and Y. Gu, “Cooperative data reduction in wireless sensor network,” in *GLOBECOM*, 2012.

[127] C. Zhao, U. Topcu, and S. Low, “Optimal load control via frequency measurement and neighborhood area communication,” *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 3576–3587, 2013.

[128] W. Zhong, Z. Huang, T. Zhu, Y. Gu, Q. Zhang, P. Yi, D. Jiang, and S. Xiao, “Ides: Incentive-driven distributed energy sharing in sustainable microgrids,” in *IGCC*, 2014.

[129] T. Zhu, Z. Huang, A. Sharma, J. Su, D. Irwin, A. Mishra, D. Menasche, and P. Shenoy, “Sharing renewable energy in smart microgrid,” in *IEEE ICCPS*, 2013.

[130] T. Zhu, A. Mishra, D. Irwin, N. Sharma, P. Shenoy, and D. Towsley, “The case for efficient renewable energy management in smart homes,” in *BuildSys*, ACM, 2011, pp. 67–72.

[131] T. Zhu, S. Xiao, Y. Ping, D. Towsley, and W. Gong, “A secure energy routing mechanism for sharing renewable energy in smart microgrid,” in *SmartGridComm*, IEEE, 2011, pp. 143–148.