Learning and Control for Energy Systems under Uncertainty

Yuanyuan Shi

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2020

Reading Committee:

Baosen Zhang, Chair

Daniel S. Kirschen

Lillian J. Ratliff

Program Authorized to Offer Degree:

Electrical Engineering

University of Washington

Abstract

Learning and Control for Energy Systems under Uncertainty

Yuanyuan Shi

Chair of the Supervisory Committee:

Baosen Zhang

Electrical Engineering

Future energy system operation faces various sources of uncertainties, including both supply-side uncertainties associated with the variability and intermittency of renewables, and demand-side uncertainties related to active load management and demand response. Due to the increasing uncertainty, power system operation is facing a growing risk of potential safety issues and economic loss. Research efforts on mitigating these uncertainties, therefore, are much-needed and of great importance.

The goal of this thesis is to take a step towards future energy system design under significant uncertainties, from the algorithmic perspective. In particular, we leverage tools that interface between machine learning, optimization, and control to address three types of uncertainties: environmental uncertainty, model uncertainty, and uncertainty from users' interactions.

The first part of the dissertation considers energy system control under environmental uncertainty, by focusing on battery control in providing pay-for-performance services, such as frequency regulation and renewable integration. We derive an optimal real-time control

algorithm with accurate battery degradation cost modeling, that achieves near-optimal performance compared to the offline optima which have complete future information.

The second part of the dissertation addresses energy system control under model uncertainty. Specifically, we propose a novel type of neural network architecture called input convex neural networks (ICNNs), for both system identification and controller design. We show that ICNNs significantly outperform existing purely data-driven or linear control methods with applications in building energy management and distribution system voltage control.

Finally, we consider uncertainties from user interactions. We use the Cournot competition model to characterize the collective behavior of learning agents in energy markets. We prove the convergence of policy gradient dynamics to the Nash equilibrium and provide insights on the effect of pricing functions and information feedback to the convergence behavior.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I have spent five wonderful years at the University of Washington and the city of Seattle. First and foremost, I would like to thank my advisor Professor Baosen Zhang, for his great support and guidance. I consider myself extremely lucky to work under his supervision. He taught me how to do research, from determining the right question for study, approaching a complex problem from the simplest setting to gain insights, to writing and presenting our results. His advice extends well beyond research. There were many moments during my graduate study (as well as in my job search) that I was frustrated and doubted whether I am able to pursue an academic career, but I know my advisor is always on my side, with his endless encouragement. Words can not express my appreciation and Baosen will always be an extraordinary role model for my future career.

Next, I would like to thank Professor Daniel Kirschen, who in many ways also acted as an advisor for my research, teaching, and life. I learned a lot from Prof. Kirschen. His great vision, expertise in power systems, enthusiasm in teaching and education, as well as his way of interacting with people and leading a big research group, are lifelong inspirations.

I also want to thank my committee member, Professor Lillian Ratliff, who provided many insightful feedbacks for my research and helped me expand my connection within and outside of UW. In the same breath, I would like to thank Professor Archis Ghate for serving as the Graduate School Representative on my final exam committee and Professor Sam Burden for being on my qualify exam committee. I extend my sincere thanks to Professor Radha Poovendran, Professor Truong Nguyen at UCSD, Professor Anima Anandkumar, and Professor

Adam Wierman at Caltech, and Professor Zhengyuan Zhou at NYU for their invaluable career suggestions. In addition, I would like to acknowledge all the professors and mentors I've taken classes and interacted with, which shaped my understanding in many aspects.

I have had an amazing group of collaborators from UW over my graduate study. Firstly, I would like to express my gratitude to Bolun Xu. Bolun taught me a lot about battery energy storage and electricity market, and many of the results in Chapter 2 and 3 are our joint work. He has been a great friend and mentor, and I learned a tremendous amount from him from writing my first paper, numerous discussions, career advice, and more. Yize Chen also deserves a big thank you, and results in Chapter 4 are our joint work. His enthusiasm for work, innovative thinking, and the talent to learn new things quickly inspired me a lot. Besides, I appreciate his good sense of humor, which brightens our office time. I also want to thank Yushi Tan and Liyuan Zheng for all the stimulating conversations and suggestions.

I was also fortunate to collaborate with many people outside of UW, through my internships at Doosan Gridtech, JD.com, and DeepMind. The internship at Doosan is my very first working experience, and I thank my mentor Tess Williams for teaching me all the good habits about being a professional. My internship at JD.com was amazingly fruitful thanks to my mentors Prof. Zuojun (Max) Shen, Dr. Rong Yuan, Dr. Di Wu, and collaborator Meng Qi. I give them and all members in the JD-Y team my most sincere gratitude. Last but not least, I really appreciate Todd Hester and Daniel Mankowitz gave me the opportunity to spend a summer at DeepMind in London, which not only led to many interesting collaborations but also provided me the opportunity to experience a different culture. It turns out to be the fastest growth period of my life to date, thanks to everyone I've met there - who are not only exceptional scientists but also truly kind and thoughtful people. Besides, I was lucky to collaborate with Dr. Di Wang from Microsoft Research, Dr. Hao He, and Sarang Amirtabar from Centrica. I appreciate their insights in bridging research and real-world applications.

Graduate school would not have been as fun without all my friends around. I want to thank all the REAL Ladies, Atinuke (Tinu) Abolaji Ademola-Idowu, Abeer Almainouni, Ahlmahz Negash, Kelly Kozdras, Anna Edwards, Mareldi Ahumada, Nina Vincent, Jackie Baum, Trisha Ray, Agustina Gonzalez, Pan Li, Yao Long, and Wenqi Cui. Through the five years, it has been (and will continue to be) a unique and sweet home to share each other's pains, and celebrate the slightest and every achievement together. I also want to give me special thanks to Tinu - we started our graduate school at the same time, and prepared for our quality exam, job interviews, thesis defense, almost all the critical stages together. Thanks for your company along the journey, and I am so happy that we are both going to start exciting new life chapters. I want to thank all my colleagues in EE433 and the REAL lab, in no particular order: Ling Zhang, Tanner Fiez, Chase Dowling, Shreyas Sekar, Daniel Calderone, Kun Su, Jimin Kim, Yue Sun, Zhipeng Liu, Yishen Wang, Zeyu Wang, Yury Dvorkin, Ricardo Fernández-Blanco, Mushfiq Sarker, Ahmad Milyani, Hao Wang, Yi Wang, Jingkun Liu, Chenghui Tang, Yaohua Cheng, Qingchun Hou, Hao Li, Jesus Contreras, Ryan Elliott, Daniel Olsen, Lane Smith, Daniel Tabas, Chanaka Keerthisinghe, Rahul Mallik, Soham Dutta, Minghui Lu, Gord Stephen, and all lab alumni we've met and worked together. In particular, I would like to thank Tanner Fiez and Ryan Elliott for always taking time from their busy schedule, listening to my questions, and offering many helpful suggestions.

Of course, life did not end at the walls of ECE building, and I want to acknowledge Yaxuan Zhou, Chen Gong, Weisi Xie, Fan Qi, Dan Guo, Chuchuan Hong, Chen Zou, Wei Zuo, and many more, who have been great friends and we shared all the wonderful moments together.

Finally, I would like to thank my family. I am most grateful to my parents for their unconditional love and support, at all times. Furthermore, I would like to thank my fiance Junmin. He has truly been a partner with love and patience, who has accompanied me throughout my undergraduate and graduate study, and encouraged me to pursue my dream.

DEDICATION

to my parents and my fiance Junmin

whose love and support made this thesis possible

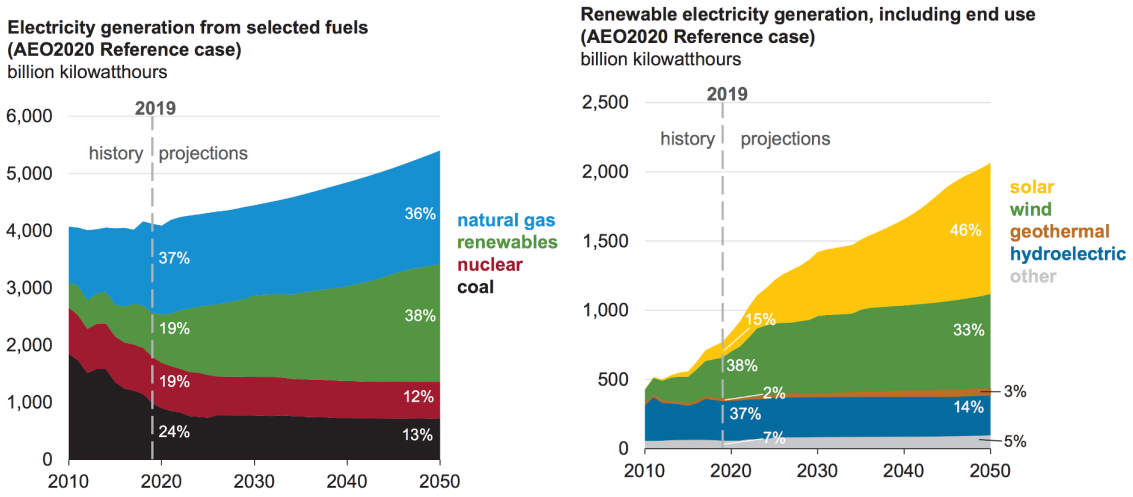
Chapter 1

INTRODUCTION

1.1 Motivation

The emergence of renewable resources, ranging from offshore wind farms to rooftop solar units, are having a transformative effect on energy systems. Historically, fossil fuel has been the main resource for generating electricity. However, due to the environmental concerns of fossil generators such as the emissions of greenhouse gases that contributes to global warming, renewable resources (e.g., solar, wind) that are less polluting and having less environmental impact, are becoming more an important part of the grid. Figure 1.1 shows that energy generation in the U.S. by source, with predictions to 2050. We can see that the share of renewable generation is expected to grow to 38% (in 2050), which will contribute to the largest share of electricity production. In fact, many regions in the U.S. actually have more ambacious goals than the average projection. For example, Connecticut and New Jersey aim to generate about 50% of their electricity from renewables by 2030, and both California and Washington aim for 100% renewable energy by 2050.

The question boils down to how can we develop an economically competitive and technologically feasible strategy for the entire energy system transition to renewable energy. One of the biggest problems for high levels of renewable penetration is to mitigate the significant amount of *uncertainties*. Unlike fossil fuel generators which we have precise control over the amount of energy being produced, we do not have full control of the renewable generation since the wind does not always blow and the sun does not always shine. How to guarantee the security of



(a) Electricity generation from selected fuels (b) Renewable electricity generation

Figure 1.1: Source: U.S. Energy Information Administration, based on projection of data in 2020. `https://www.eia.gov/outlooks/aeo/`

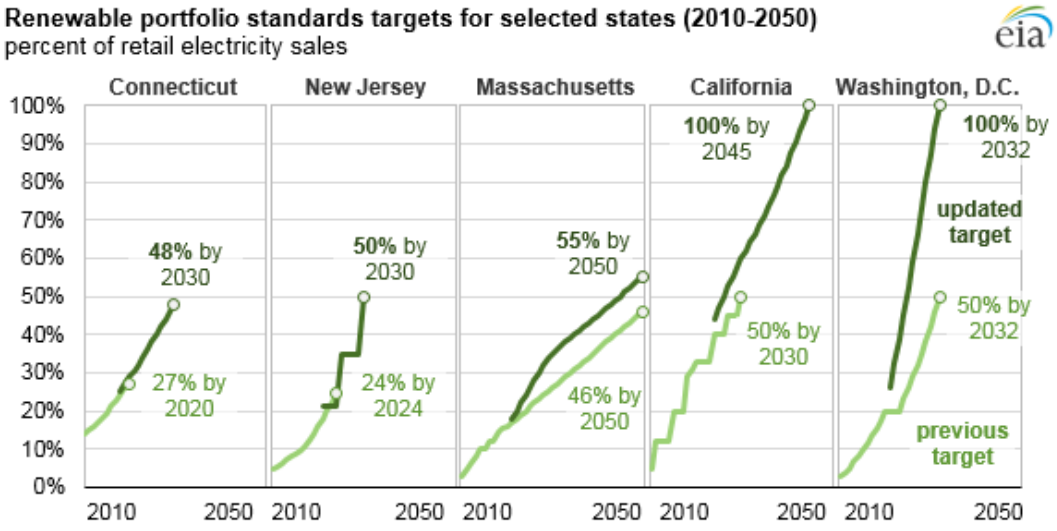


Figure 1.2: Source: U.S. Energy Information Administration, based on states' renewable portfolio standards. `https://www.eia.gov/todayinenergy/detail.php?id=38492`

power supply under renewable intermittency? It requires a holistic approach that integrates new devices/technologies, new control algorithms, and new market organization mechanisms, all together to pave the way from the current grid to the more sustainable future.

Energy storage is considered as an important technology bridge between the intermittent renewable power produced by nature, and a 24×7 reliable supply of renewable electricity. There are abundant of roles that storage can provide in the grid, which includes but not limits to, support renewables intermittency, spatial and temporal energy shifting and arbitrage, ancillary services, alternatives for peaking generation/transformers/line upgrades, voltage and frequency support, microgrid supply, electric vehicle charging support, and etc [1]. However, managing energy storage at the power system level remains an open question. Due to the fast response time requirement, any storage control algorithm being proposed should be computational-efficient, while having performance guarantee under future uncertainties and complex battery degradation characteristics. Energy storage alone, despite its versatile and powerful functionality, is impossible to solve all the uncertainties introduced by the large penetration of renewables. Even with the very optimistic prediction of energy storage growth 120 gigawatts by 2050[1], it is not enough to provide all the flexibility that is needed. Therefore, aside from using energy storage, other resources of flexibility are indispensable.

An additional key contributor to handle uncertainties would be demand-side participation. Consumers (both commercial and residential) in today's grid usually play passive roles where they just receive power from the generators. And the power flow in the tody's power distribution network is mostly unidirectional, that is from the generators to customers. However, because of many renewable energy resources will be installed in distribution networks (e.g., rooftop solar), it would be desired that if the network could be redesigned to allow for bidirectional operation such that consumers can play a more active role in the system. For instance, they can adjust their demand levels in response to the real-time supply level in

[1] `https://cleantechnica.com/2020/01/03/120-gigawatts-of-energy-storage-by-2050-we-got-this/`

the system. In addition, they can sell the energy surplus from local generations back to the grid. Active demand-side participation can not only help the grid to mitigate the renewable uncertainties, but also help users reduce their electricity costs via various participation subsidy programs.

With all these new technologies and new resources into the grid, the challenge remains is how to realize their full potential. The key to solving these challenges lies in a better understanding of the system and through the interrogation of increasingly available operations data. For example, more than 2,500 networked phasor measurement units and millions of smart meters are connected to the U.S. grid receiving and sharing data in real-time. This explosion of data provides opportunities for us to better understand the system and forecast the future, thus making better decisions. A good control and market organization framework would enable a seamless integration of hundreds of thousands of distributed energy resources in future energy systems, from energy storage, renewables, to active load management and demand response.

1.2 Contributions

The goal of this thesis is to handle challenges faced by energy system operation under significant uncertainties, from the algorithm design perspective. Our results span all the three aspects we have described so far, including handling environmental uncertainty, model uncertainty, and uncertainty from users' interactions. In particular, the contribution of this dissertation could be summarized in three themes.

1.2.1 Cost modeling and optimal control of energy storage

Energy storage offers great flexibility to the power system operation under environmental uncertainties. However, managing these assets at the system level remains an open question. One of the key challenges in energy storage operation is the degradation cost accounting.

Faithful electrochemical degradation models have always been thought of as impractical to use in real-time due to their complexity. In Chapter 2, we show that this cost is actually convex [2], resolving this long-standing algorithmic challenge of supposed impracticality in an elegant fashion. Based on this cost model, an optimal online control algorithm is proposed for energy storage in a general "pay-for-performance" market to track an uncertain instruction signal (e.g., frequency regulation service or supporting renewable integration). Via a novel change of basis, we derive a simple online control policy that minimizes the operational cost of storage that includes both the dispatch violation cost and battery degradation cost. Simulation results with real-world data from the PJM market show that we could double the frequency regulation service profit with only 20% of the degradation cost, by using the proposed control [3] and capacity bidding algorithm [4].

In Chapter 3, we further show that batteries can achieve *super-linear* economic benefits by exploring the diversity and mutual benefit of multifaceted applications [5], such as providing frequency regulation service and peak shaving at the same time. Using real-world data from Microsoft datacenters, we demonstrate that the joint optimization of energy storage can not only help reduce the electricity bill of commercial users up to 12% (totaling millions of dollars in savings each year), but also bring greater benefit to grid operation.

1.2.2 Data-driven control for energy systems with unknown dynamics

Following this work on energy storage under external uncertainties, Chapter 4 focuses on system internal uncertainties and how to design optimal control algorithms subject to them. It turns out the key is to carefully design machine learning architectures that leverage the physics of the systems. We leverage the data obtained from sensor measurements to construct an *input convex neural network*(ICNN) [6] for system dynamics modeling, where the weights between neurons are constrained to be positive and some direct "passthrough" layers are added for better representation power. Without loss of generality, we prove that ICNNs can

represent all convex system dynamics and are exponentially more efficient in fitting non-convex systems than piecewise affine functions. Using the proposed ICNN, we show that many interesting control problems in energy systems (e.g., building energy management and distribution system voltage control) can be cast as convex optimization problems, leveraging the prior knowledge that the underlying physics is in fact convex.

Experiment results show that our data-driven control framework can achieve 20% more building energy reduction compared with linear control methods and reduce the modeling time from years to minutes [7]. In addition, the same framework can be used for distribution system voltage control, by modeling the voltage dynamics via ICNN [8]. Simulation results on IEEE 13-bus and 123-bus systems show the proposed method achieves effective voltage regulation performances (e.g., maintain over 98.3% of nodal voltages within 5% deviations for the 123-bus system), which is much better than control linear fitted models. This framework bridges machine learning and closed-loop control by representing system dynamics using ICNNs, which obtain both good predictive accuracy and tractable computational complexity.

1.2.3 Multi-agent learning dynamics in energy market

Energy system is more than a physical infrastructure: it connects millions of participants with different roles and objectives. The interactions (both competitions and collaborations) of individuals also introduce great uncertainty to the system operation, as each of the individuals may have different roles and objectives. In Chapter 5, we formulate the strategic interactions between different energy suppliers as a Cournot game, where each player is self-interested and aim to maximize their own payoffs via a certain learning algorithm. Notably, we prove the convergence of policy gradient algorithms in concave Cournot games with very limited feedback (i.e., only the price and nothing else). This is the first result (to the best of our knowledge) on the convergence property of algorithms with continuous action spaces that do not fall in the no-regret class.

1.3 Dissertation Outline

The thesis is organized into five parts. Chapter 1 provides the motivation and background. The remaining chapters are as follows:

- Chapter 2: *Decision Making under Environmental Uncertainty: Online Battery Control in Pay-for-Performance Market* focuses on the battery degradation cost modeling and online battery control subject to external uncertainties such as supporting renewable integration and providing frequency regulation services.
- Chapter 3: *Diversity and Multiplexing: Joint Optimization of Battery Storage for Superlinear Gains* extends the online control framework to a multi-application scenario, and demonstrate the superlinear benefits for battery multiplexing.
- Chapter 4: *Decision Making under Model Uncertainty: Input Convex Neural Networks for Building Control and Voltage Regulation* proposes a novel data-driven control framework for complex energy system control with unknown models.
- Chapter 5: *Learning in Multiagent System with Limited Information Exchange: Cournot Competition in Electricity Market* analyzes the interactions of agents in the energy market from a game theory perspective.
- Chapter 6: *Conclusion and Future Works* concludes the thesis and lays out suggestions for future work.

When reading the thesis, each of the chapters can be read independently of one another. Each chapter focuses on a different perspective of uncertainties faced by energy system operation, in accordance with a concrete application scenario faced in the real world. Each chapter is associated with a concise introduction and literature review section to be self-contained.

Chapter 2

DECISION MAKING UNDER ENVIRONMENTAL UNCERTAINTY: ONLINE BATTERY CONTROL IN PAY-FOR-PERFORMANCE MARKET

2.1 Introduction

Plenty of energy storage technologies have been developed to serve different applications, such as pumped hydro-power, batteries, flywheels, and many more [9]. Among these different technologies, *battery energy storage (BES)* (e.g., lithium-ion batteries) features quick response time, high round-trip efficiency, pollution-free operation, and flexible power/energy ratings. These characteristics make it an ideal choice for a wide range of power system applications, including integration of renewable resources [10], grid frequency regulation [11] and behind-the-meter load management of commercial and residential users [12].

The focus of the chapter is on the optimal control of battery energy storage under a general "pay for performance" setup: a battery is incentivized to follow certain instruction signals and is penalized when it cannot. For example, a battery participating in frequency regulation would receive a signal and is paid based on how well it follows the signal. Another important application that falls under this setup is a battery used by customers with onsite renewable generations, where the customers may need to purchase more expensive power from the grid if the battery cannot smooth out the local net demand. The common theme of the problems under the pay for performance setup is that the signal the battery should follow is inherently *random* and the control decisions must be made in *real-time*. Furthermore, battery storage naturally couples the decisions across time because of its finite energy and power capacities.

Therefore, finding the optimal control policy for a battery is essentially a constrained online stochastic control problem [13].

This online problem is challenging for two main reasons: 1) battery degradation and 2) lack of future information. Analogous to cell phone batteries losing their capacity after several years of use, larger batteries used in the grid also losses their capacity with every charge and discharge actions [14]. In fact, overly aggressive use of batteries can often deplete their useful capacities in a matter of months. However, battery degradation is a complex process governed by electrochemical reactions and depends multiple environmental and utilization factors. The second challenge of the lack of information is common to all stochastic control problems. At any given time, a decision must be made without knowing the future signals. This is further complicated by the coupling constraints introduced by the battery. These two challenges are illustrated well in the fast frequency regulation problem. Frequency regulation is a mechanism used by power system operators to correct the short timescale imbalance between generation and demand in the overall grid. In fast regulation (e.g., regD in PJM), a signal representing the imbalance is broadcasted every 2 or 4 seconds. Having enough energy to follow the regulation signals is critical to the function of the power system, especially as renewables increase the uncertainties in both generation and supply. By participating in regulation, a battery receives a fixed payment ahead of the time. However, if it cannot follow the regulation signal, then a penalty is charged based on the mismatch. Therefore, at every time step, a battery must balance its degradation from following the regulation signal with the penalty of doing so, while not knowing the future value of the signal.

2.1.1 Literature Review

The operation of battery energy storage has received much recent research attention because of the importance of batteries to a power system with high penetration of renewables and maturing technologies [10–13, 15–18]. In these works, the degradation cost of the batteries

are modeled in different ways. The authors of [10, 13, 15, 16] assume battery has a fixed lifetime and ignore the degradation cost. This assumption works well when batteries are used sparingly, but tend to lead to overly aggressive actions for finer time resolution applications such as frequency regulation Other energy storage control studies include degradation models either based on battery charging/discharging power [11, 12] or energy throughput [17, 18]. For example, [12] assumes a convex degradation cost model based on battery charging/discharging power for households demand response, and [17, 18] assign a constant price $2\$/MWh$ based on battery energy throughput. These degradation models are convenient to be incorporated in existing optimization problems, at a cost of losing accuracy in quantifying the actual degradation cost. For example, a Lithium Nickel Manganese Cobalt Oxide (NMC) battery has *ten* times more degradation, when operated at near 100% cycle depth of discharge (DoD), compared to operating at 10% DoD for the *same* amount of charged power or energy throughput [19]. However, the impact of cycle depth is difficult to capture using power or energy based degradation functions.

The battery aging process is fundamentally described by a set of partial differential and algebraic equations [20, 21], however, they are in some sense too detailed to be used in power system applications. Even with dedicated state-of-the-art algorithms, these equations take several seconds to solve, making them too slow to be used in applications like frequency regulation where one receives a signal every 2 or 4 seconds. To mitigate these difficulties, we use a semi-empirical degradation model that combines theoretical battery aging mechanism with experimental observations. This model is motivated by viewing battery capacity fading as a material fatigue process, where a deeper charge/discharge cycle stresses battery much more than an equivalent number of shallower cycles.

The online nature of the battery control problem has perhaps received more attention from the control community. Multiple types of approaches have been developed, including model predictive control [15, 22, 23], stochastic and dynamic programming [10, 24–27]. The authors of [22] derive a model predictive control (MPC)-based for battery energy storage and wind

integration, although without any performance guarantees. Recent works [15, 23] do include results that bound the performance gap of online algorithms, but it is difficult to evaluate the quality of these bounds since they are either quite loose or depend on complicated optimization problems themselves that grow with the time of operation. In addition, none of these bounds considers a cycle-based degradation problem. Our results in this paper provide an online algorithm with a constant gap to the offline optimal that is independent of the length of the operation time.

In addition to the MPC type of algorithms, another widely used strategy is dynamic programming (DP). For example, [10] and [24] consider using DP for storage operation with a co-located wind farm, [25] and [26] for operating storage with end-user demands, and [27] for storage with demand response. However, for real-time control problems, the battery state space, action space, and the instruction signal are all continuous. Standard DP discretization approaches tend to cause the dimension of the problem to grow exponentially. Also, implementing these algorithms requires the distributional information of the random instruction signal, which may not be readily available. In contrast, our algorithm does not require any distributional information.

As a summary, most previous studies have attacked the battery control problem by focusing on one of the two aforementioned challenges. On one hand, by assuming the degradation of batteries is a quadratic function of the charge/discharge powers, we recover a type of constrained stochastic quadratic regulation problem where the key challenge is the lack of future information [28, 29]. On the other hand, one can focus on the degradation of the batteries, by employing accurate electrochemical models while assuming full knowledge of the future [20, 30]. Both directions have led to significant advances by still remains unsatisfactory. Even by assuming the signal that a battery faces is Gaussian, a constrained linear quadratic Gaussian problem is still extremely challenging to solve and provide any theoretical performance guarantees. Similarly, solving the optimization problem with accurate electrochemical models is by no means trivial even under full knowledge, and it is usually

difficult to adapt the solutions to an online form. Given these difficulties, batteries still only serve as emergency backup, or used actively in grid services when they are owned by the utilities and are subsidized under renewable portfolio incentives.

2.1.2 Our contributions

Instead of attacking the complexity of the degradation function or the lack of future information one at a time, we address these two challenges together in a joint fashion. Surprisingly, we provide a provably near-optimal online algorithm for battery control. In particular, we show that under a form of so-called cycle based degradations, there is an *online strategy that is within a constant additive gap of the optimal offline strategy under all possible future signals*. We explicitly characterize this gap and relate it to the set of possible future signals. The key insight of this result comes from a better understanding of the degradation of electrochemical batteries and how it relates to the control problem. In past approaches, online battery control was studied in the time domain, leading to complex optimization problems. In contrast, we look at the problem in the *cycle-domain*, where the problem naturally decouples according to each cycle of the charge/discharge profile. This approach has a loose analogy with time/frequency duality, where some problems are much simpler in the frequency domain than in the time domain. Altogether, the approach proposed in this chapter makes three contributions to the state-of-art in battery control:

1. We present an electrochemically accurate and trackable battery degradation model, called the rainflow cycle-based degradation. We prove the cost model is convex, which enables it to be used in various battery optimization problems with optimality guarantee.

2. We provide a subgradient algorithm to solve the optimization problem efficiently and optimally for offline battery planning and dispatch. In addition, we offer an online battery control policy with a simple threshold structure, and achieve near-optimal performance w.r.t. the offline controller that has complete future information.

2.2 Problem Formulation

In this section, we describe the battery operation model, the rainflow cycle-based battery aging cost, and the pay for performance market setup. Then we state the main optimization problem on *how to balance profit from frequency regulation and the degradation cost of battery operation in an online fashion.*

2.2.1 Battery Operations

We consider an operation defined over finite discrete control time steps $t \in \{1, \ldots, T\}$, and each control time interval has a duration of τ.[1] Let x_t be the energy stored in the battery–the state of charge (SoC)–at time t. By convention, x_t is a normalized quantity between 0 (empty battery) and 1 (full battery). At any time interval, the battery can either charge with power c_t (in units of kW) or discharge with power d_t (in units of kW). Then its state of charge evolves according to the following linear difference equation:

$$x_{t+1} - x_t = \frac{\tau \eta_c}{E} c_t - \frac{\tau}{\eta_d E} d_t. \tag{2.1}$$

where η_c and η_d are the charging and discharging efficiency,[2] and E (in units of kWh) is the rated energy capacity of battery. We use bold symbols $\mathbf{x}, \mathbf{c}$, and $\mathbf{d}$ to denote the vector version of SoC, charging powers and discharging powers, respectively.

For a given battery, it has three types of operational constraints. The first is the limits its SoC, where the stored energy in the battery is constrained to be within a particular region. This constraint can arise from either health concerns since batteries like lithium-ion should not be charged completely full or discharged to be completely empty. It can also arise if batteries are used for other applications such as backup. In this paper, we assume that the

[1] In practice, τ is set by the power electronic based battery management system, and is normally in the scale of milliseconds [31].

[2] By convention, η_c is between 0 and 1 and η_d is larger than 1.

SoC limits are given. The other two constraints on battery operation are the rate constraints on the charging and discharging powers. These constraints are written as:

$$\underline{x} \leq x_t \leq \overline{x},\ 0 \leq c_t \leq P, \text{ and } 0 \leq d_t \leq P,$$

where $\underline{x}$ and $\overline{x}$ is the minimum and maximum SoC of the battery, respectively; P is the battery power rating.

We consider an optimization problem where a battery is incentivized to follow an instruction signal $\mathbf{r}$. Suppose the operation revenue is $\mathbf{R}(\mathbf{c}, \mathbf{d}, \mathbf{r})$, a function of battery power output $\mathbf{c}, \mathbf{d}$ and the instruction signal. The operational cost comes from the battery degradation, denoted here by $f(\mathbf{c}, \mathbf{d})$, a function of battery charging/discharging responses. The exact form of $f(\cdot)$, namely *the rainflow cycle-based degradation function*, is introduced in the next section. The optimization objective is to maximize the net utility of the battery, which equals to operational revenue subtracting the cost. The overall problem is:

$$\max_{\mathbf{c},\mathbf{d}} R(\mathbf{c}, \mathbf{d}, \mathbf{r}) - f(\mathbf{c}, \mathbf{d}) \tag{2.2a}$$

$$\text{s.t. } x_{t+1} = x_t + \frac{\tau \eta_{\mathrm{c}}}{E} c_t - \frac{\tau}{\eta_{\mathrm{d}} E} d_t\,, \tag{2.2b}$$

$$\underline{x} \preceq \mathbf{x} \preceq \overline{x}\,, \tag{2.2c}$$

$$0 \preceq \mathbf{c} \preceq P\,, \tag{2.2d}$$

$$0 \preceq \mathbf{d} \preceq P\,, \tag{2.2e}$$

where (2.2b) is the state evolution equation, (2.2c) is the SoC constraint, (2.2d) and (2.2e) are the power constraints. Component-wise inequality between two vectors is denoted by $\preceq$. Note here we may include a constraint that storage cannot charge and discharge at the same time [32], but it turns out that this condition will always be satisfied in our setting.

Solving (2.2) has proven to be difficult for two reasons. The first is that most realistic cycle-based degradation functions are not well understood (e.g., they are not known to be convex), making the deterministic version of (2.2) nontrivial [33]. The second is that in

real-time applications such as frequency response, the signal $\mathbf{r}$ is inherently random and difficult to forecast [34], while the state of the problem x_t is constrained and coupled over time. Therefore, even for relatively simple forms of f (e.g. $f = \sum c_t^2 + d_t^2$), there are no optimal or provably suboptimal online algorithms. The next section describes the rainflow cycle-based degradation model in detail, and the rest of the paper shows that rather surprisingly, this realistic will lead to a simple provable optimal online algorithm.

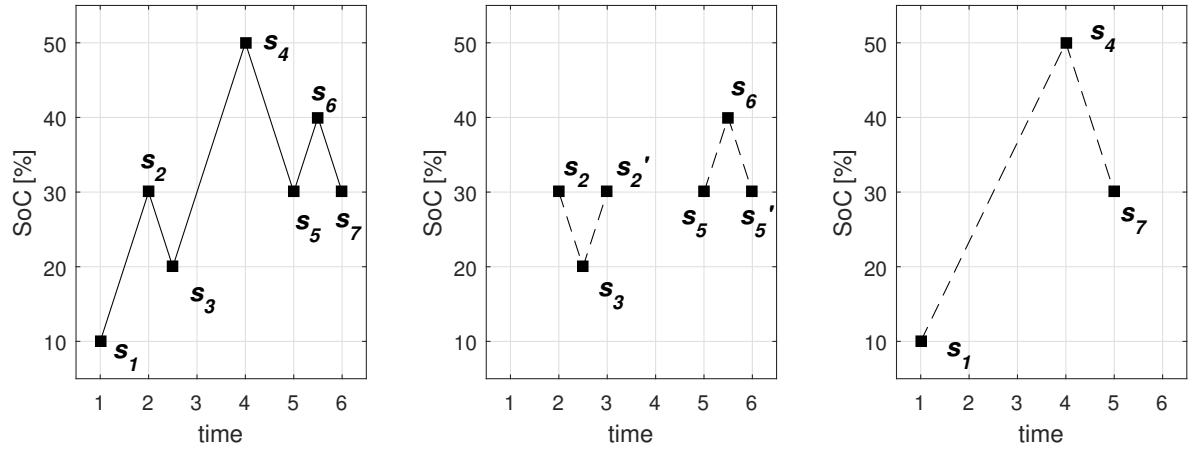
2.2.2 Cycle Counting via Rainflow

To model the battery degradation cost $f(\mathbf{c}, \mathbf{d})$, we take the rainflow cycle-based method, the most widely used semi-empirical method used in practice [33, 35–38]. The cycle aging of electrochemical battery cells is evaluated based on stress cycles, and the rainflow method identify cycles from local extrema in a SoC profile. A local extrema point indicates that the battery switched from charge mode to discharge mode, or vice versa. We use $s_1, s_2, \ldots$ to denote the extrema, including both minima and maxima. Figure 2.2.2 gives an example of a profile with seven extrema.

Given a SoC profile with its local extrema, the goal of the rainflow algorithm is to extract the *cycle depths* between the local maxima and minima of the profile. The key feature of the algorithm is that it does not necessarily calculate the depth between two successive extrema. Rather, it first finds the charging cycle (or discharging cycle) sorted by the depth of the cycles based on the entire profile. For the example in Fig. 2.2.2, there are 3 charging cycles: the deepest one is between s_1 and s_4 (40% SoC), the other two cycles are both of 10% SoC (between s_3 and s_2' in Fig. 2.2.2 and between s_5 and s_6).

For a general SoC profile, the rainflow counting method is given below in Algorithm 1. Note there are multiple equivalent algorithms, and the one we adopt here is based on [21].

This algorithm essentially moves through the extrema to look for charging and discharging



(a) SoC profile with 7 local extrema, including start and end points.

(b) Balanced charging and discharging cycles.

(c) Charging and discharging cycle left after other cycles are extracted.

Figure 2.1: Rainflow cycle counting example

half-cycles [39]. After a cycle is identified, its associated end points are removed, and the process is repeated until no points are left. Figure 2.1 shows the progression of Algorithm 1 through an example profile.

Let Rainflow be the functional form of the rainflow counting algorithm in Algorithm 1, then it takes a SoC profile as an input and outputs the cycle depths:

$$(\mathbf{v}, \mathbf{w}) = \text{Rainflow}(\mathbf{x}) \tag{2.3}$$

where $\mathbf{v}$ is the vector of charging half cycles and $\mathbf{w}$ is the vector of discharging half cycles. Since cycle depths only depend on the relative differences of the turning point of the SoC profile and not on the initial SoC value, they can be calculated from $(\mathbf{c}, \mathbf{d})$

$$(\mathbf{v}, \mathbf{w}) = \text{Rainflow}\Big(\frac{\tau\eta_{\text{c}}}{E}\mathbf{c} - \frac{\tau}{\eta_{\text{d}}E}\mathbf{d}\Big). \tag{2.4}$$

2.2.3 Battery Degradation Cost

After counting the cycles, a cycle depth stress function $\Phi(u) : [0, 1] \to \mathbb{R}^+$ is used to model the life loss from a single cycle of depth u measured in terms of (normalized) changes in the SoC. This function indicates that if a battery cell is repetitively cycled with depth u, then it can operate $1/\Phi(u)$ number of cycles before reaching its end of life. In practice, this function can be estimated through empirical measurements in Fig. 2.2 is normally included by battery manufactures [40]. For most electrochemical batteries, $\Phi(u)$ is a convex function [19, 35–38], popularly parameterized as a power function αu^β or exponential functions $\alpha e^{\beta u}$ [41]. Because cycle aging is a cumulative fatigue process [19, 35], the total life loss ΔL is the sum of the life loss from all half cycles:

$$\Delta L(\mathbf{v}, \mathbf{w}) = \sum_{i=1}^{|\mathbf{v}|} \frac{\Phi(v_i)}{2} + \sum_{i=1}^{|\mathbf{w}|} \frac{\Phi(w_i)}{2}, \tag{2.5}$$

where $|\cdot|$ is the cardinality of a vector. By convention, the factor $1/2$ is included. Here we assumed that a charging half cycle and a discharging half cycle has the same stress function Φ, but our results do not change if different functions are used. For example, to calculate cycle aging for the profile in Fig. 2.2.2, we set $v_1 = 0.1$, $v_2 = 0.1$, $v_3 = 0.4$ and $w_1 = 0.1$, $w_2 = 0.1$, $w_3 = 0.2$. If we substitute the rainflow algorithm as in (2.4) into (2.5), the incremental cycle aging can therefore be written as a function of the control actions $\mathbf{c}$ and $\mathbf{d}$. To convert the loss of life to a cost, let B be the battery cell replacement unit cost in \$/kWh and E be the capacity of the battery in kWh. Then the cycle aging cost function $f(\mathbf{c}, \mathbf{d})$ is

$$f(\mathbf{c}, \mathbf{d}) = \Delta L(\mathbf{c}, \mathbf{d}) \cdot E \cdot B. \tag{2.6}$$

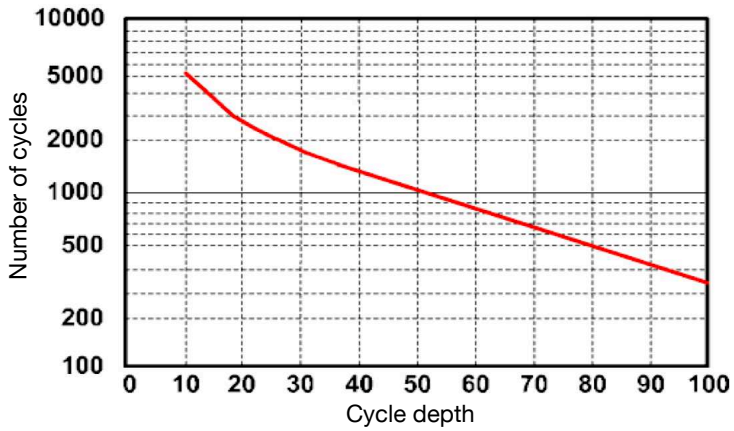


Figure 2.2: Battery cycle depth and operating number curve. The x-axis is the cycle depth in percent, and y-axis is the number of cycles that battery could be operated under certain condition before the end of life.

2.2.4 Revenue Model

Here we describe a version of a pay for performance mechanism widely used by system operators. This mechanism has a two-stage structure. In the first stage, ahead of real-time, a payment C (in units of \$) is provided to the participant. Here, we assume that this payment is known and given and focus on the second stage. The second stage occurs in real-time, where a participant is given a signal $\mathbf{r}$ and faces a penalty if it cannot follow the signal. That is, it by pays a over-response price $\theta \in R^+$ (\$/MWh) for surplus injections or deficient demands during each dispatch interval, and a under-response price $\pi \in R^+$ (\$/MWh) for deficient injections or surplus demands. Then the total revenue is:

$$\begin{aligned} R(\mathbf{c}, \mathbf{d}, \mathbf{r}) =& C - \tau\theta \textstyle\sum_{t=1}^{T} |\eta_c c_t - \frac{d_t}{\eta_d} - r_t|^+ \\ & - \tau\pi \textstyle\sum_{t=1}^{T} |r_t - \eta_c c_t + \frac{d_t}{\eta_d}|^+ , \end{aligned} \tag{2.7}$$

where $\eta_c c_t - \frac{d_t}{\eta_d}$ is the net charging power, $r_t \in [-P, P]$ is the instructed regulation dispatch set-point for the dispatch time step t, with the convention positive values in r_t represents

charging instructions.

This model captures the essence of two important applications of storage in the grid: frequency regulation[3] and the demand shaping. In frequency regulation, C is the capacity payment and r_t is the regulation signal sent every 2 to 4 second by the system operator. The penalty prices θ and π are published values. In demand shaping, a battery would enter into an agreement with an utility to keep demand of a customer at prescribed levels at payment C and r_t can be thought as the net time-varying demand of the user. Here the penalty prices are also determined ahead of time. An important future direction is to extend our results to settings where the penalty prices are random in themselves, such as real-time arbitrage [42, 43].

2.2.5 Optimization Problem

Summarizing the previous sections, we are left with the following optimization problem:

$$\min_{\mathbf{c},\mathbf{d}} \tau \sum_{t=1}^{T} \left[\theta |\eta_c c_t - \frac{d_t}{\eta_d} - r_t|^+ - \pi |r_t - \eta_c c_t + \frac{d_t}{\eta_d}|^+ \right] + \left[\sum_{i=1}^{|\mathbf{v}|} \frac{\Phi(v_i)}{2} + \sum_{i=1}^{|\mathbf{w}|} \frac{\Phi(w_i)}{2} \right] \cdot B \cdot E \tag{2.8a}$$

$$\text{s.t. } x_{t+1} = x_t + \frac{\tau \eta_{\mathrm{c}}}{E} c_t - \frac{\tau}{\eta_{\mathrm{d}} E} d_t \,, \tag{2.8b}$$

$$\underline{x} \preceq \mathbf{x} \preceq \overline{x} \,, \tag{2.8c}$$

$$0 \preceq \mathbf{c} \preceq P \,, \tag{2.8d}$$

$$0 \preceq \mathbf{d} \preceq P \,. \tag{2.8e}$$

$$(\mathbf{v}, \mathbf{w}) = \text{Rainflow}(\mathbf{x}). \tag{2.8f}$$

We are interested in solving (2.8) in two settings:

Offline: In the off-line setting, the entire sequence of the signal $\mathbf{r}$ is given. This is important

[3] In practice, different system operators have slightly different rules for frequency response. Instead of cumbersome accounting for these rules, we focus on the simplified structure which is given in (2.7).

in many planning and validation problems.

Online: Here, we only allow c_t and d_t to depend on the current and past information: $\{r_t, r_{t-1}, \ldots, r_1\}$. This models the real-time decisions that batteries need to make for charging and discharging.

2.3 Convexity and Subgradient Algorithm

2.3.1 Summary of theoretical results

The main contributions of this chapter provide positive results to both the offline and online solutions of (2.8). For the offline setting, we have the following theorem:

Theorem 1. ***Convexity.*** Suppose the battery cycle aging stress function Φ is convex. Then the offline version of the optimization problem in (2.8) is convex in the charge and discharge variables.

This theorem settles an open question about cycle-based degradation cost functions [39, 44] and is used in the proof of the optimality of the online policy. The penalty term in the objective function (2.8a) is clearly convex in $\mathbf{c}$ and $\mathbf{d}$, but the convexity of the term associated with the cycle stress functions is not obvious because of the nonlinear Rainflow$(\cdot)$ function in (2.8f). Previously, problems like the one in (2.8) are solved via generic optimization programs (e.g., fmincon in Matlab), which can be extremely computationally expensive even for small problems. As we illustrate in Section 2.5, frequency regulation problems with time horizon longer than 4 hours take longer than 8 hours to solve using existing approaches, but can be solved in less than 3 minutes using a subgradient algorithm specifically developed for (2.8). Of course, by Theorem 1, the algorithm is optimal.

Next we state the optimality result with respect to the online optimization problem. Let J_g denote the of the value of (2.8) under an online algorithm g, and J^* denote the offline optimal where all information are known at the beginning. Then we have:

Theorem 2. ***Online optimality***. Suppose the battery cycle aging stress function Φ is strictly convex. There exist a threshold online algorithm that has a constant worst-case optimality gap that is independent of the operation time duration T.

The control policy g in Theorem 2 is a fairly straightforward threshold policy and is given as Algorithm 2 Section 2.4. The bound in the theorem is much tighter compared to standard bounds for online optimization problems. Normally, one would compare the averaged regret, namely $\lim_{T\to\infty} \frac{1}{T}(J_g - J^*)$ and a sublinear regret is considered to be "good" [45, 46]. Here, our result essentially shows that one can solve the online version of (2.8) with *zero regret*, since the constant ϵ do not depend on T. In contrast, most existing algorithms cannot even achieve sublinear regret. Again, the key to our result is to explore the particular cyclic structure of the rainflow based cost functions. By a case study on PJM frequency regulation market in Section 2.5, we show that the proposed control algorithm could significantly improve the operational revenue up to 30% and the battery can last as much as 4 times longer. A useful corollary of Theorem 2 showing when the gap is 0:

Corollary 1. ***Zero-optimality Gap*** If $\pi\eta_{\mathrm{d}} = \theta/\eta_{\mathrm{c}}$, then there is no gap between the online algorithm and the optimal offline algorithm with full information.

This corollary holds if the battery has the same charging and discharging efficiency[4] and the penalty prices for over and under injections are the same, then there exists an optimal online algorithm.

2.3.2 Convexity proof

Here we sketch the proof for Theorem 2. Without loss of generality, we only consider the cost of charging cycles given the interchangeable and symmetric nature of charging and discharging variables. A detailed proof is given in Appendix A.

[4]Again, we remind the reader that it is standard convention to write charging and discharging efficiencies differently in the energy storage community. Here, equal efficiency means $\eta_{\mathrm{d}} = \frac{1}{\eta_{\mathrm{c}}}$.

To prove Theorem 2, it suffices to show that the mapping from the SoC profile $\mathbf{x}$ to degradation cost:

$$f(\mathbf{x}) = \left[\sum_{i=1}^{|\mathbf{v}|} \frac{\Phi(v_i)}{2} + \sum_{i=1}^{|\mathbf{w}|} \frac{\Phi(w_i)}{2}\right]$$
$$(\mathbf{v}, \mathbf{w}) = \text{Rainflow}(\mathbf{x})$$

is convex in terms of $\mathbf{x}$ given the cycle stress function $\Phi(\cdot)$ convex. That is, for any two SoC time series $\mathbf{x}, \mathbf{y} \in \mathbb{R}^T$,

$$f(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1-\lambda) f(\mathbf{y}), \forall \lambda \in [0,1]. \tag{2.9}$$

Intuitively, given two SoC series $\mathbf{x}$ and $\mathbf{y}$, if they change in different directions, the two cancel each other out so that the left hand side of (2.9) is less than the right hand side by the convexity of Φ. When $\mathbf{x}$ changes in exactly the same direction as $\mathbf{y}$ for all time steps, the equality holds. The difficulty of proving this result lies in the fact that the rainflow function in Algorithm 1 is a many-to-many function that maps a sequence in $\mathbb{R}^T$ to a set of cycle depth of *indeterminate length*. The proof uses induction as described in the rest of this section.

Unit step decomposition

First, we introduce the step function decomposition of SoC signal. Any SoC series $\mathbf{x}$ could be written out as a finite sum of step functions, where

$$\mathbf{x} = \sum_{t=1}^{T} P_t U_t \,, \tag{2.10}$$

where U_t is a unit step function with a jump at time t defined as:

$$U_t(\tau) = \begin{cases} 1 & \tau \geq t \\ 0 & \text{otherwise.} \end{cases}$$

Fig. 2.3 gives an example of step function decomposition of $\mathbf{x}$. We use this decomposition to

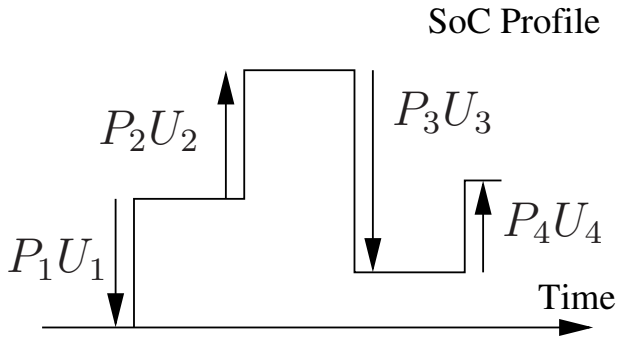


Figure 2.3: Decomposition of an example SoC Profile into 4 step functions.

write out $\mathbf{x}$, $\mathbf{y}$ and $\lambda\mathbf{x} + (1-\lambda)\mathbf{y}$ as finite sum of step functions, where

$$\mathbf{x} = \sum_{t=1}^{T} P_t U_t \,, \mathbf{y} = \sum_{t=1}^{T} Q_t U_t \,, \tag{2.11}$$

$$\lambda\mathbf{x} + (1-\lambda)\mathbf{y} = \sum_{t=1}^{T} Z_t U_t \,. \tag{2.12}$$

Note for $\mathbf{x}$ and $\mathbf{y}$ of different length, we can take T to be the maximum length since 0 can be appended to the shorter profile.

We use induction to prove Theorem 1 on the number of non-zero step changes in $\mathbf{y}$. The base case is given in the next subsection, where y has a single step change.

Initial case

We first show that $f(\mathbf{x})$ is convex when a profile has only one non-zero step change:

Lemma 1. *Under the conditions in Theorem 1, the rainflow cycle-based cost function f satisfies*

$$f(\lambda\mathbf{x} + (1-\lambda)Q_t U_t) \leq \lambda f(\mathbf{x}) + (1-\lambda) f(Q_t U_t) \,, \forall\, \lambda \in [0,1] \,,$$

where $\mathbf{x} \in \mathbb{R}^T$, and $Q_t U_t$ is a step function with a jump happens at time t with amplitude Q_t.

The proof of this initial case requires analyzing the impact on all cycle depths from the single step and is given in Appendix A.

Induction Steps

Assuming Theorem 1 is true if one of the two profiles $\mathbf{x}$ or $\mathbf{y}$ has a single non-zero step. Now, assume f is convex up to the sum of K step changes (arranged by time index):

$$f\Big(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}\Big) \leq \lambda f(\mathbf{x}) + (1-\lambda) f(\mathbf{y}), \lambda \in [0,1]$$

if $\mathbf{y}$ has K non-zero step changes ($K < T$). We need to show f is convex up to the sum of $K+1$ step changes (i.e., $\mathbf{y}$ is of length $K+1$).

The induction step proof relies on a case-by-case analysis. It contains three major conditions, where 1) Z_{K+1} (the amplitude of K+1 step of the combined profile) and Z_K are in the same direction 2) Z_K and Z_{K+1} are in different directions, with $|Z_K| \geq |Z_{K+1}|$ or 3) Z_K and Z_{K+1} are different directions, with $|Z_K| < |Z_{K+1}|$. Each major category may contain some further sub-cases and requires careful accounting. Showing convexity for each sub-case finishes the overall convexity proof and the detailed reasoning is given in Appendix A.

2.3.3 Subgradient Algorithm

The convexity of the offline problem in (2.8) suggestes that it can solved efficiently. However, the degradation cost term $f(\mathbf{c}, \mathbf{d})$ is not continuously differentiable (not differentiable at cycle junction points). This has contributed to the current difficulty in solving (2.8) even for planning and evaluation problems. Here we provide an efficient subgradient algorithm. To begin with, we re-write the constrained optimal battery control problem in (2.8) as an

unconstrained optimization problem using a log-barrier function [47]:

$$\min_{\mathbf{c},\mathbf{d}} J(\cdot) := \tau \sum_{t=1}^{T}\Big[\theta|\eta_c c_t - \frac{d_t}{\eta_d} - r_t|^+ - \pi|r_t - \eta_c c_t + \frac{d_t}{\eta_d}|^+\Big] + \left[\sum_{i=1}^{|\mathbf{v}|}\frac{\Phi(v_i)}{2} + \sum_{i=1}^{|\mathbf{w}|}\frac{\Phi(w_i)}{2}\right]EB$$
$$-\frac{1}{\lambda}\cdot\Big\{\sum_{t=1}^{T}\log[\overline{x} - x(t)] + \sum_{t=1}^{T}\log[x(t) - \underline{x}] + \sum_{t=1}^{T}\log[P - c_t] + \sum_{t=1}^{T}\log[c_t] + \sum_{t=1}^{T}\log[P - d_t] + \sum_{t=1}^{T}\log[d_t]\Big\} \tag{2.13}$$

when $\lambda \to +\infty$, the unconstrained problems (2.13) becomes equivalent to the original constrained problem. With proper step size, the subgradient algorithm is guaranteed to converge to the optimal solution with a user-defined precision level [47].

The major challenge of solving Eq. (2.13) lies in the second term. We need to find the mathematical relationship between charging cycle depth v_i and charging power c_t, as well as the relationship between discharging cycle depth w_j and discharging power d_t. Recall that the rainflow cycle counting algorithm introduced in Section 2.2.2, each time index is mapped to at least one charging half cycle or at least one discharging half cycle. Some time steps sit on the *junction* of two cycles. For example, in Fig. 2.1, $s_2^{'}$ lies on the junction of charge half cycle $s_1 - s_4$ and charge half cycle $s_3 - s_2^{'}$. No time step belongs to more than two cycles.

Let T_{v_i} be all the time indexes that belong to the charging half cycle i and let the time indexes belonging to the discharge half cycle j be set T_{W_j}. Then

$$T_{v_1} \cup \ldots \cup T_{v_{|\mathbf{v}|}} \cup T_{w_1} \cup \ldots \cup T_{W_{|\mathbf{w}|}} = \{1, \ldots, T\}\,, \tag{2.14}$$

$$T_{v_i} \cap T_{w_j} = \emptyset\,,\ \forall i, j\,. \tag{2.15}$$

Eq. (2.15) shows there is no overlapping between a charging and a discharging cycle. That is, each half-cycle is either charging or discharging. The cycle depth therefore equals to the sum of battery charging or discharging within the cycle time frame,

$$v_i = \sum_{t \in T_{v_i}} \frac{\tau\eta_c}{E} c_t\,, \tag{2.16}$$

$$w_j = \sum_{t \in TW_j} \frac{\tau}{\eta_d E} d_t. \tag{2.17}$$

The rainflow cycle cost $f(\mathbf{x})$ is not continuously differentiable. At each cycle junction point, it has more than one subgradient. We use $\partial f(\mathbf{x})|_{c_t}$ to denote a subgradient at c_t. Since the SoC profile $\mathbf{x}$ is a function of $\mathbf{c}$, by the chain rule, we have

$$\partial f(\mathbf{x})|_{c_t} = \Phi^{'}(v_i)\frac{B\tau\eta_c}{2}\,, t \in T_{v_i}\,, \tag{2.18}$$

where v_i is the depth of cycle that c_t belongs to. Note, at junction points, c_t belongs to two cycles so that the subgradient is not unique. We can set v_i to any value between v_{i1} and v_{i2}, where v_{i1} and v_{i2} are the depths of two junction cycles c_t belongs to.

Similarly for discharging cycle, a subgradient at d_t is,

$$\partial f(\mathbf{x})|_{d_t} = \Phi^{'}(w_j)\frac{B\tau}{2\eta_d}\,, t \in TW_j \tag{2.19}$$

where w_j is the depth of the cycle that d_t belongs to. At the junction point, w_j could be set to any value between w_{j1} and w_{j2}, which are the two junction cycles d_t belongs to.

Therefore, we write the subgradient of $J(\cdot)$ with respect to c_t and d_t as $\partial J|_{c_t}$ and $\partial J|_{d_t}$, where

$$\begin{aligned}\partial J|_{c_t} = &-\frac{\partial R(\mathbf{c},\mathbf{d},\mathbf{r})}{\partial c_t} + \Phi^{'}(v_i)\frac{B\tau\eta_c}{2} - \frac{1}{\lambda}\Big\{\sum_{k=t}^{T}\frac{1}{x(k)-\overline{x}}(\frac{\tau\eta_c}{E})\\ &+\sum_{k=t}^{T}\frac{1}{x(k)-\underline{x}}(\frac{\tau\eta_c}{E}) + \frac{1}{c_t - P} + \frac{1}{c_t}\Big\}, t \in T_{v_i}\end{aligned} \tag{2.20}$$

$$\begin{aligned}\partial J|_{d_t} = &-\frac{\partial R(\mathbf{c},\mathbf{d},\mathbf{r})}{\partial d_t} + \Phi^{'}(w_j)\frac{B\tau}{2\eta_d} - \frac{1}{\lambda}\Big\{-\sum_{k=t}^{T}\frac{1}{x(k)-\overline{x}}(\frac{\tau}{\eta_d E})\\ &-\sum_{k=t}^{T}\frac{1}{x(k)-\underline{x}}(\frac{\tau}{\eta_d E}) + \frac{1}{d_t - P} + \frac{1}{d_t}\Big\}, t \in TW_j\end{aligned} \tag{2.21}$$

The update rules for c_t and d_t at the kth iteration are,

$$c_{(k)}(t) = c_{(k-1)}(t) - \alpha_k \cdot \partial J|_{c_{(k-1)}(t)}\,,$$

$$d_{(k)}(t) = d_{(k-1)}(t) - \alpha_k \cdot \partial J|_{d_{(k-1)}(t)}\,,$$

where α_k is the step length at kth iteration. Since the subgradient method is not a decent method [48], it is common to keep track of the best point found so far, i.e., the one with smallest function value. At each step, we set

$$J_{(k)}^{best} = \min\left\{J_{(k-1)}^{best}, J(\mathbf{c}_{(k)}, \mathbf{d}_{(k)})\right\},$$

Since the $J(\cdot)$ is convex, choosing an appropriate step size guarantees convergence.

2.4 Online Control Policy

In this section, we introduce the proposed online battery control policy which balances the cost of deviating from the instruction signal and the cycle aging cost of batteries while satisfying operation constraints. This policy takes a *threshold form* and achieves an optimality gap that is *independent of the total number of time steps*. Therefore in term of regret, this policy achieves the strongest possible result: the regret do not grow with time. Note we assume the regulation capacity has already been fixed in the previous capacity settlement stage

2.4.1 Control Policy Formulation

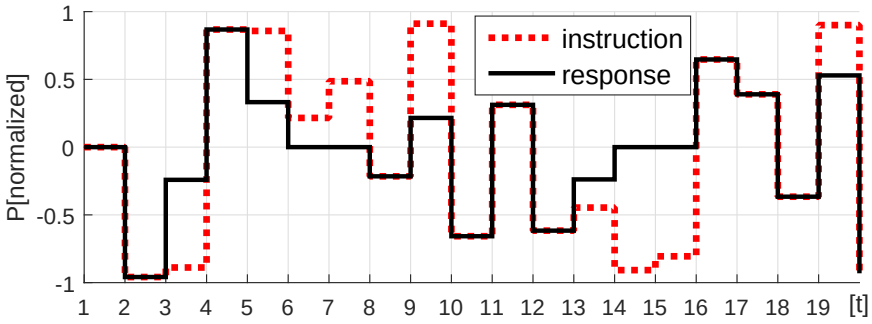
The key part of the control policy is to calculate thresholds that bounds the SoC of the battery as functions of the deviation penalty and degradation cost. Let $\hat{u}$ denote this bound on the SoC and it is given by:

$$\hat{u} = \dot{\Phi}^{-1}\left(\frac{\pi\eta_{\mathrm{d}} + \theta/\eta_{\mathrm{c}}}{B}\right), \tag{2.22}$$

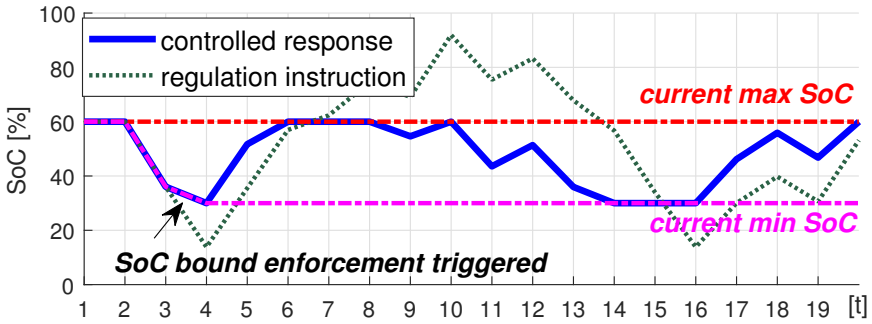
where $\dot{\Phi}^{-1}(\cdot)$ is the inverse function of the derivative of the cycle stress function $\Phi(\cdot)$.

The proposed control policy is summarized in Algorithm 2, and Fig. 2.4 shows a control example of the proposed policy, in which the battery follows the regulation instruction until the distance between its maximum and minimum SoC reaches $\hat{u}$. The detailed formulation

is as follows. We assume at a particular control step t, x_t (battery state of charge) and r_t (frequency regulation signal) are observed, and the proposed regulation policy has the following form: $g_t(x_t, r_t) = \begin{bmatrix} c_t & d_t \end{bmatrix}$. The control policy employs the following strategy



(a) Instruction (dotted line) vs. response (solid line).



(b) Optimal (solid) vs. profile if signal is followed perfectly (dotted)

Figure 2.4: Illustration of the control policy. The policy keeps track the current maximum and minimum SoC level. When the distance in between reaches the calculated threshold $\hat{u}$, the policy starts to constrain the response. Deeper charge and discharge cycles are avoided.

$$\text{If } r_t \geq 0,\ c_t = \min\left\{\frac{E}{\tau\eta_{\mathrm{c}}}(\overline{x}_t - x_t), r_t\right\} \tag{2.23}$$

$$\text{If } r_t < 0,\ d_t = \min\left\{\frac{E\eta_{\mathrm{d}}}{\tau}(x_t - \underline{x}_t), r_t\right\} \tag{2.24}$$

where $\overline{x}_t$ and $\underline{x}_t$ are the upper and lower storage energy level bound determined by the

controller at the control interval t for enforcing the SoC band $\hat{u}$

$$\begin{aligned} \overline{x}_t &= \min\{\overline{x}, x_t^{\min} + \hat{u}\} \\ \underline{x}_t &= \max\{\underline{x}, x_t^{\max} - \hat{u}\} \end{aligned} \tag{2.25}$$

and $x_t^{\max}$, $x_t^{\min}$ is the current maximum and minimum battery storage level since the beginning of the operation, which are updated at each control step as

$$\begin{aligned} x_t^{\max} &= \max\{x_{t-1}^{\max}, x_t\} \\ x_t^{\min} &= \min\{x_{t-1}^{\min}, x_t\}\,. \end{aligned} \tag{2.26}$$

2.4.2 Optimality Gap to Offline Problem

Theorem 2 states that the gap between the online policy in Algorithm 2 and an offline optimal solution is bounded by a constant. This constant can be explicitly characterized. To do this, we define three new functions:

$$J_{\mathrm{u}}(u) = EB\Phi(u) + E(\theta/\eta_{\mathrm{c}} + \pi\eta_{\mathrm{d}})u \tag{2.27a}$$

$$J_{\mathrm{v}}(v) = (1/2)EB\Phi(v) + (E/\eta_{\mathrm{c}})\theta v \tag{2.27b}$$

$$J_{\mathrm{w}}(w) = (1/2)EB\Phi(w) + E\eta_{\mathrm{d}}\pi w \tag{2.27c}$$

where J_{u} is the cost associated with a full cycle (made up of a charging half cycle and a discharging half cycle with equal magnitude), J_{v} for a charge half cycle, and J_{w} for a discharge half cycle. The detailed transforming procedure is discussed in the Appendix II-A.

If the cycle depth stress function $\Phi(\cdot)$ is strictly convex, then it is easy to see that (2.22) is the unconstrained minimizer to (2.27a). Similarly, the unconstrained minimizers of (2.27b) and (2.27c) are:

$$\hat{v} = \dot{\Phi}^{-1}\Big(\frac{\theta/\eta_{\mathrm{c}}}{B}\Big), \quad \hat{w} = \dot{\Phi}^{-1}\Big(\frac{\pi\eta_{\mathrm{d}}}{B}\Big)\,. \tag{2.28}$$

The following theorem offers the analytical expression for ϵ.

Theorem 3. *If function $\Phi(\cdot)$ is strictly convex, then the worst-case optimality gap for the proposed policy $g(\cdot)$ in Theorem 2 is*

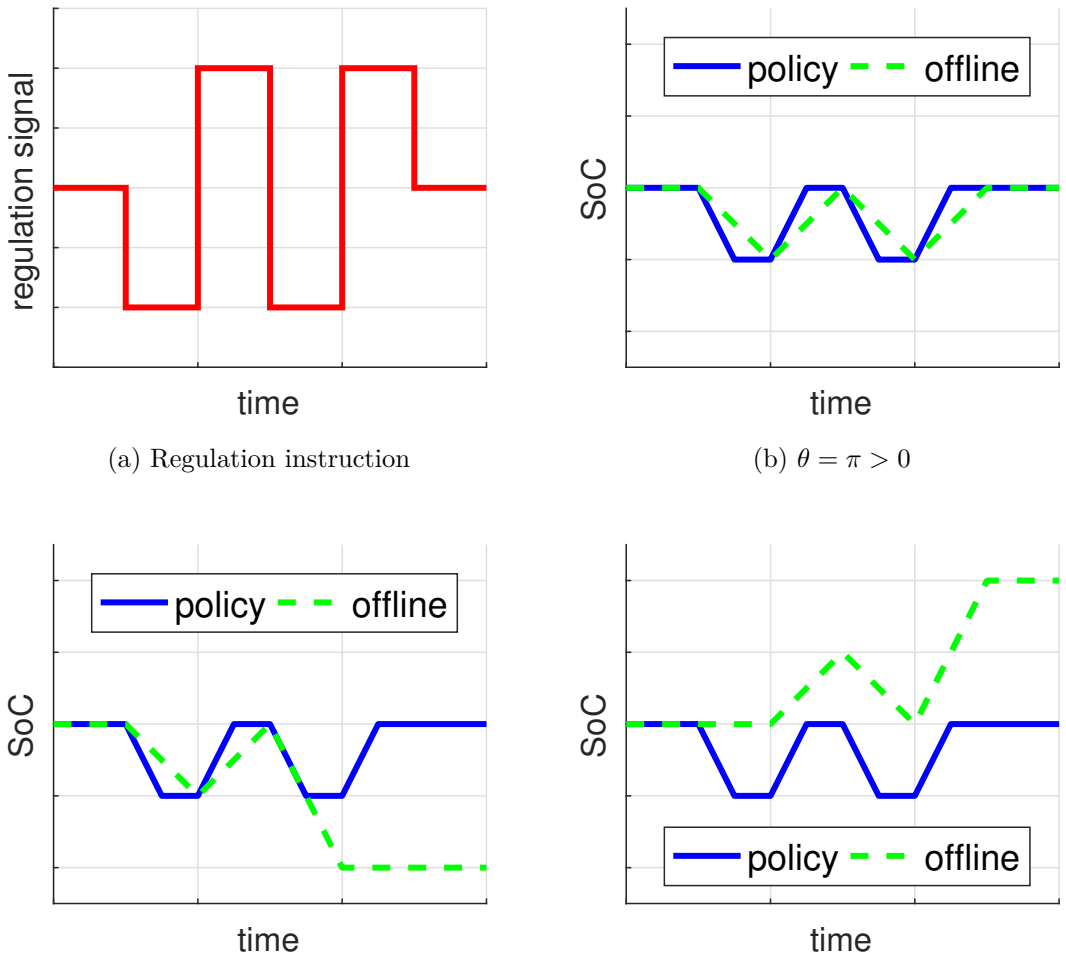
$$\epsilon = \begin{cases} \epsilon_{\mathrm{w}} & \text{if } \pi\eta_{\mathrm{d}} > \theta/\eta_{\mathrm{c}} \\ 0 & \text{if } \pi\eta_{\mathrm{d}} = \theta/\eta_{\mathrm{c}} \\ \epsilon_{\mathrm{v}} & \text{if } \pi\eta_{\mathrm{d}} < \theta/\eta_{\mathrm{c}} \end{cases} \tag{2.29}$$

where

$$\epsilon_{\mathrm{w}} = J_{\mathrm{w}}(\hat{u}) + 2J_{\mathrm{v}}(\hat{u}) - J_{\mathrm{w}}(\hat{w}) - 2J_{\mathrm{v}}(\hat{v}) \tag{2.30}$$

$$\epsilon_{\mathrm{v}} = 2J_{\mathrm{w}}(\hat{u}) + J_{\mathrm{v}}(\hat{u}) - 2J_{\mathrm{w}}(\hat{w}) - J_{\mathrm{v}}(\hat{v})\,. \tag{2.31}$$

Note that Corollary 1 follows from Theorem 3 directly. We defer the proof of the latter to Appendix A. The intuition is that battery operations consist mostly full cycles due to limited storage capacity because the battery has to be charged up before discharged, and vice versa. Enforcing $\hat{u}$–the optimal full cycle depth calculated from penalty prices and battery coefficients–ensures optimal responses in all full cycles. In cases that $\pi\eta_{\mathrm{d}} = \theta/\eta_{\mathrm{c}}$, $\hat{u}$ is also the optimal depth for half cycles, and the proposed policy achieves optimal control. In other cases, the optimality gap is caused by half cycles because they have different optimal depths. However, half cycles have limited occurrences in a battery operation because they are incomplete cycles [49], so that the optimality gap is bounded as stated in Theorem 3. Fig. 2.5 shows some examples of the policy optimality when responding to the regulation instruction (Fig 2.4.2) under different price settings. The proposed policy has the same control action in all three price settings because of the same $\hat{u}$. The policy achieves optimal control in Fig 2.4.2 because $\hat{u}$ is the optimal depth for all cycles. In Fig 2.4.2 and Fig 2.4.2, half cycles have different optimal depths and the policy is only near-optimal. However, the offline result also selectively responses to instructions with a zero penalty price (charge instructions in Fig 2.4.2, discharge instructions in Fig 2.4.2), because it returns the battery to a shallower cycle depth with smaller marginal cost.



(a) Regulation instruction

(b) $\theta = \pi > 0$

(c) $\theta = 0, \pi > 0$

(d) $\theta > 0, \pi = 0$

Figure 2.5: Example illustration of the policy optimality under different price settings. The value of $\theta + \pi$ is the same in all cases and the round-trip efficiency is assumed to be one, so $\hat{u}$ is the same in all cases.

2.5 Simulation Results

2.5.1 Simulation Setting

We compare the proposed control policy with the offline optimal result and a simple control policy proposed in [10]. The maximum state of charge (SoC) level $\overline{x}$ is set to 95% and the minimum SoC level $\underline{x}$ is set to 10%. This assumed battery storage consists of lithium-ion battery cells that can perform 3000 cycles at 80% cycle depth before reaching end of life, and these cells have a polynomial cycle depth stress function concluded from lab tests [41]: $\Phi(u) = (5.24 \times 10^{-4})u^{2.03}$, and the cell replacement price is set to 300 \$/kWh.Suppose the battery power rating is 1MW and energy rating is 1MWh.

2.5.2 Optimality Gap

We simulate regulation using random generated regulation traces to exam the optimality of the proposed policy and to validate Theorem 2 and 3. We generate 100 regulation signal traces assuming a uniform distribution between $[-1, 1]$, and design nine test cases. Each test case has different market prices and battery round-trip efficiency $\eta = \eta_{\mathrm{d}}\eta_{\mathrm{c}}$. In order to demonstrate the time-invariant property of the optimality gap, Case 7 to 9 are designed to have twice the duration of Case 1 to 6 by repeating the generated regulation signal trace.

The 100 generated regulation traces are simulated using the proposed policy, the simple policy [10], and the offline subgradient solver for each test case. Table 2.1 summarizes the test results. The penalty prices, round-trip efficiency, and the number of simulation control intervals used in each test case are listed, as well as the control SoC bound $\hat{u}$ and the worst-case optimality gap ϵ that are calculated using simulation parameters. The simulation results are recorded under the maximum optimality gap and the average objective value.

This test validates Theorem 3 since ϵ is exactly the same as the recorded maximum optimality

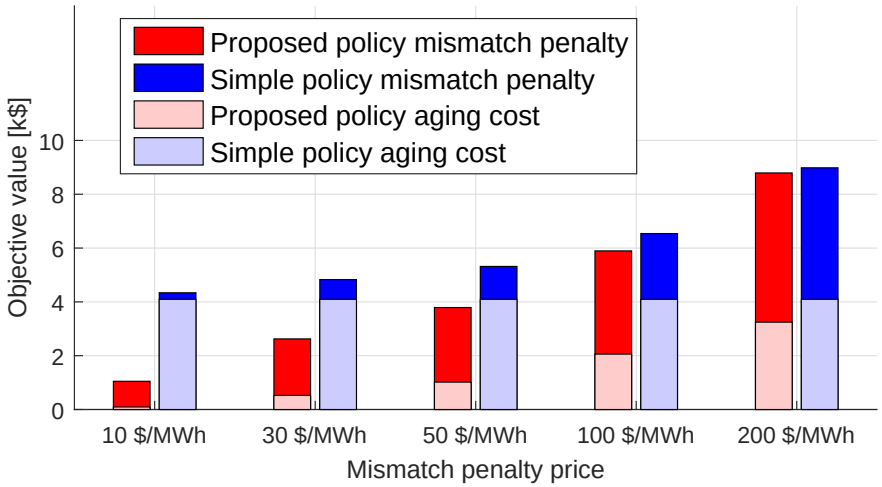


Figure 2.6: Regulation operating cost break-down comparison between the proposed policy and the simple policy. Although the proposed policy has higher penalties, the cost of cycle aging is significantly smaller, so it achieves better trade-offs between degradation and mismatch penalty.

gap for the proposed policy in all cases (both highlighted in pink), while the gap for the simple policy is significantly larger (highlighted in grey). In particular, the proposed policy achieves exact control results in Case 1 to 3 because $\theta/\eta_c = \pi\eta_d$, while Case 4 to 9 have non-zero gaps because the round-trip efficiency is less than ideal ($\eta < 1$). We also see that as penalty prices become higher, the control band $\hat{u}$ becomes wider and the battery follows the regulation instruction more accurately. The simple policy also provides better control results at higher penalty prices. Case 7 to 9 have the same parameter settings as to Case 4 to 6, except that the dimension of regulation signal doubled. The proposed policy achieves the same worst-case optimality gap – which again verifies that the worst-case optimality gap of the proposed online control policy is independent of operation time T.

2.5.3 Simulation using Realistic Regulation Signal

In this simulation we compare the proposed policy with the simple policy using the regulation signal trace published by PJM Interconnection [50]. The control time interval for this signal is 2 seconds and the duration is 4 weeks. Table 2.2 shows the difference of computation time between the subgradient algorithm in Section 2.3.3 and a standard numerical solver [33] implemented using fmincon in Matlab. It turns out that the latter does not converge for problem horizon of longer than 4 hours. All experiments conducted on a Macbook Pro with 2.5 GHz Intel Core i7, 16 GB 1600 MHz DDR3.

Table 2.2: Computation Time

Time horizon (min)	60	120	240	720	1440
Subgradient solving time(s)	23.9	62.5	156.3	673.5	2522
fmincon solving time(s)	264	2006	29800	$\sim$	$\sim$

We repeat the simulation using different penalty prices. We let $\theta = \pi$ in each test case and

set the round-trip efficiency to 85%. Fig. 2.6 summarizes the simulation results in the form of regulation operating cost versus penalty prices, the cycle aging cost and the regulation mismatch penalty are listed for each policy. Because the simple control [10] does not consider market prices, its control actions are the same in all price scenarios, and the penalty increases linearly with the penalty price. The proposed policy causes significantly smaller cycle aging cost and better control results. As the penalty price increases, the gap between the two policies becomes smaller since $\hat{u}$ becomes closer to 100%. According to the historical billing data of PJM regulation market [51], the regulation mismatch penalty price is usually below 50$/MWh. Under such price setting, we save around more than 30% and the battery can last as much as 4 times longer compared to the simple policy case.

2.6 Conclusion

In this chapter, we talk about the optimal control of battery energy storage under a general "pay-for-performance" setup, where batteries need to trade-off between following instruction signals and the impact of degradation from charging and discharging actions. We demonstrate that under electrochemically accurate cycle-based degradation models, the battery control problem can be formulated as a convex online optimization problem. Based on this result, we develop an online control policy that has a bounded time-invariant worst-case optimality gap and is strictly optimal under certain market scenarios. From the case study in the PJM regulation market, we verify the proposed degradation model and online control policy can significantly reduce operational cost and extend battery lifetime. The proposed degradation model, subgradient solver algorithm, and online control policy have a broad application scope in various battery planning and operation optimization problems.

Algorithm 1: Rainflow Counting Algorithm.

Data: A SoC profile with a finite number of local extrema $\mathcal{S} = \{s_1, s_2, \dots\}$.

Result: A set of charging depths $\mathcal{V} = \{v_1, v_2, \dots\}$ and a set of discharging depths $\mathcal{W} = \{w_1, w_2, \dots\}$.

Start from the beginning of the profile;

while $\mathcal{S}$ *is not empty* **do**

  Read from $\mathcal{S}$ to obtain $s_1, s_2, \dots$;

  if *There are more than three points in* $\mathcal{S}$ **then**

    Calculate $\Delta s_1 = |s_1 - s_2|$, $\Delta s_2 = |s_2 - s_3|$, $\Delta s_3 = |s_3 - s_4|$;

    if $\Delta s_2 \leq \Delta s_1$ *and* $\Delta s_2 \leq \Delta s_3$ **then**

      A full cycle of depth Δs_2 associated with s_2 and s_3 has been identified. Add Δs_2 to both $\mathcal{V}$ and $\mathcal{W}$;

      Remove s_2 and s_3 from the profile, set $\mathcal{S} = \{s_1, s_2, s_5, s_6 ...\}$;

    else

      Shift the identification forward and repeat with $\mathcal{S} = \{s_1, s_3, s_4, s_5 ...\}$;

    end

  else

    if $s_1 \leq s_2 \leq s_3$ **then**

      Add $s_3 - s_1$ to $\mathcal{V}$;

    else if $s_1 \geq s_2 \geq s_3$ **then**

      Add $s_1 - s_3$ to $\mathcal{W}$;

    else if $s_1 \geq s_2$, $s_2 \leq s_3$ **then**

      Add $s_3 - s_2$ to $\mathcal{V}$, $s_1 - s_2$ to $\mathcal{W}$;

    else

      Add $s_2 - s_1$ to $\mathcal{V}$, $s_2 - s_3$ to $\mathcal{W}$;

    end

    Set $\mathcal{S}$ to the empty set.

  end

end

Algorithm 2: Proposed Control Policy

Result: Determine battery dispatch point c_t, d_t

`// initialization`

set $\Phi\left(\frac{\pi\eta_{\mathrm{d}}+\theta/\eta_{\mathrm{c}}}{B}\right) \rightarrow \hat{u}$, $x_0 \rightarrow x_0^{\max}$, $x_0 \rightarrow x_0^{\min}$;

while $t \leq T$ **do**

`// read` x_t `and update controller`

set $\max\{x_{t-1}^{\max}, x_t\} \rightarrow x_t^{\max}$, $\min\{x_{t-1}^{\min}, x_t\} \rightarrow x_t^{\min}$;

set $\min\{\overline{x}, x_t^{\min} + \hat{u}\} \rightarrow \overline{x}_t$;

set $\max\{\underline{x}, x_t^{\max} - \hat{u}\} \rightarrow \underline{x}_t$;

`// read` r_t `and enforce soc bound`

if $r_t \geq 0$ **then**

set $\min\left\{\frac{E}{\tau\eta_{\mathrm{c}}}(\overline{x}_t - x_t), r_t\right\} \rightarrow c_t$, $0 \rightarrow d_t$;

else

set $0 \rightarrow c_t$, $\min\left\{\frac{E\eta_{\mathrm{d}}}{\tau}(x_t - \underline{x}_t), r_t\right\} \rightarrow d_t$;

end

`// wait until next control interval`

set $t + 1 \rightarrow t$;

end

Table 2.1: Simulation with Random Generated Regulation Signals.

	θ	π	η	N	$\hat{u}$	ϵ	Maximum optimality gap [$]		Average objective value [$]		
Case	[$/MWh]	[$/MWh]	[%]		[%]	[$]	Theoretical	Simple	Offline	Proposed	Simple
1	50	50	100	100	11.1	0.00	0.00	183.9	117.4	117.4	200.2
2	100	100	100	100	21.9	0.00	0.00	127.5	168.7	168.7	209.0
3	200	200	100	100	42.8	0.00	0.00	47.9	219.4	219.4	226.7
4	50	50	85	100	11.2	0.06	0.06	184.9	117.2	117.3	202.9
5	80	20	85	100	11.7	3.83	3.83	181.4	108.0	110.7	198.9
6	20	80	85	100	10.6	2.19	2.19	192.6	122.4	123.8	206.8
7	50	50	85	200	11.2	0.06	0.06	408.8	235.6	235.7	388.3
8	80	20	85	200	11.7	3.83	3.83	400.6	219.5	222.2	375.4
9	20	80	85	200	10.6	2.19	2.19	421.4	247.6	248.9	401.1

Chapter 3

DIVERSITY AND MULTIPLEXING: JOINT OPTIMIZATION OF BATTERY STORAGE FOR SUPERLINEAR GAINS

3.1 Introduction

In the previous chapter, we consider the real-time battery control problem for a single application, under environmental uncertainty. However, it has been recognized that because of the high capital cost of batteries [52], serving a single application is sometimes difficult to justify their investments [53]. In addition, picking a single application does not consider the possibility of multiple revenue streams and may leave "money on the table". Consequently, a recent line of research has started to analyze the co-optimization of batteries for multiple services [54, 55].

In this chapter, we consider the *joint optimization* of using a battery storage system for both *peak shaving* and *frequency regulation* for a commercial customer. Peak shaving can be used to reduce the demand charge for customers and the (fast) frequency regulation is an ideal service to provide for batteries because of their near-instantaneous response time. The challenge in combining these two applications lies in their *vastly different timescales*: peak demand charge is calculated every month on a smoothed power consumption profile (e.g. 15-minute averages), while fast frequency regulation requires a decision every 2-4 seconds.

The key observation in our work is that serving different applications over different timescales is economically beneficial to the battery: by exploring the diversity in different applications, we can obtain a so-called *superlinear gain*. An example of the superlinear gain is presented in

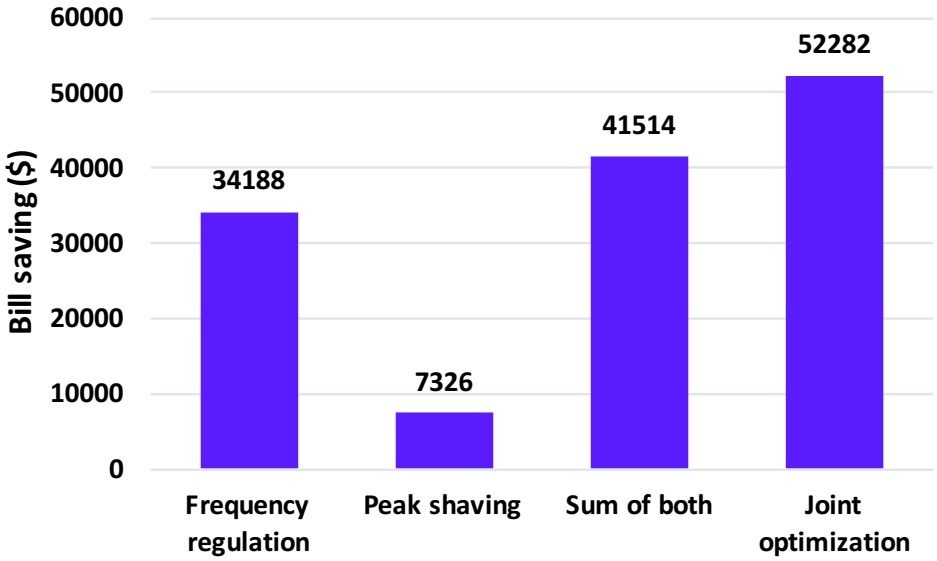


Figure 3.1: Annual electricity bill savings for a 1MW data center (in the PJM control area, total bill of $488,370) under different battery usage scenarios. Savings from joint optimization is larger than the sum of savings from frequency regulation and peak shaving.

Fig. 1. It gives the annual electricity bill savings for a 1MW data center under three scenarios, using batteries for frequency regulation service, peak shaving, and joint optimization. For joint optimization, we use a simple online threshold algorithm given in Section IV. While for peak shaving and regulation service, the solutions are offline optimal. The super-linear gain arises for reasons that would be explored in depth in the rest of the chapter, but briefly speaking, the randomness of frequency regulation signal could contribute to more efficient peak shaving. By exploring the diversity and mutual benefit in different applications, we have this non-linear behavior.

It should be noted that the key function of the existing batteries in data centers is to provide backup capacity. The proposed joint optimization framework is deploying only part of the battery energy capacity while a large portion of the battery energy has been reserved for backup purpose. A more detailed discussion on the division of battery for grid service and

backup is provided in Section V-A.

3.1.1 Literature Review and Our Contributions

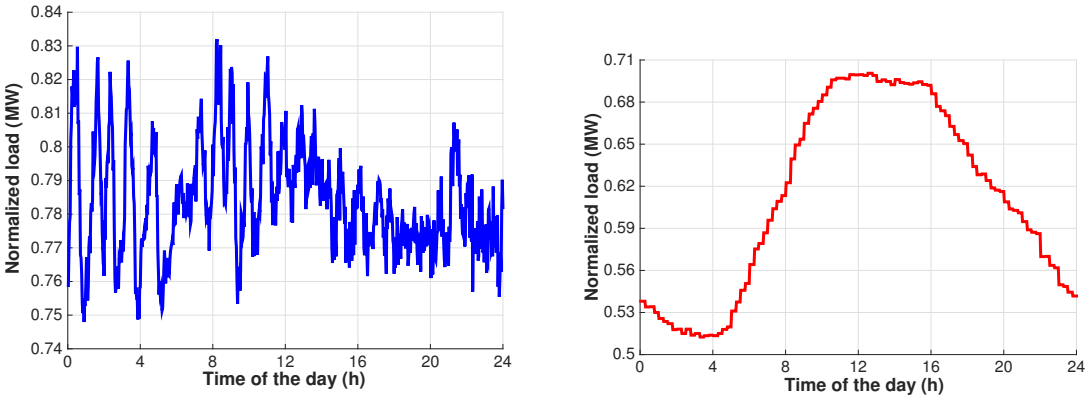
The line of literatures consider co-optimization of storage starts from [55]. In [55], the authors analyze the economics of using storage device for both energy arbitrage and frequency regulation service. The work in [56] extended this "dual-use" idea by considering plug-in electric vehicles as grid storage resource for peak shaving and frequency regulation. Both works showed that dual-use of storage often leads to higher profits than single applications. However, the aforementioned works mainly rely on heuristic analysis under different price and user patterns without directly using optimization models. The work of [57] bridged the methodology gap by proposing a systematic co-optimization framework, which could be applied for evaluating different application combinations at different timescales. This framework assumes that all future information is known, so it cannot be extended directly to deal with potential uncertainties from energy and ancillary service markets (e.g.,. price, frequency regulation signal, etc.). To deal with uncertainties, [53, 54] formulate the battery co-optimization problem as a stochastic program. In [53], stochastic programming was solved to obtain hourly optimal decisions. The work in [54] included applications of different time-scales in its optimization and tackled computational challenges by taking advantage of the problem's nested structure.

Our work is close in spirit to [54], which captures both the future market uncertainties and timescale difference of multiple applications. However, compared with [53, 54], our work contributes in two significant ways:

- We propose a joint optimization framework for batteries to perform peak shaving and provide regulation services. This framework accounts for battery degradation, operational constraints, and the uncertainties in both the customer load and regulation signals. All of the previous works, to our knowledge, do not include the operational

cost of batteries in their optimization models, which can potentially lead to aggressive charging/discharging responses and severely suboptimal operations [53–57]. Since batteries cycle multiple times a day when used for frequency regulation and peak shaving, the degradation effect plays an important role in determining their operations.

- We show that there is a *superlinear gain*: where the revenue from joint optimization is larger than the *sum* of performing the individual applications. We quantify this gain using real world data from two large commercial users: a Microsoft data center and the University of Washington EE & CSE building. Figure 3.2 gives an example daily load profile for both cases. The superlinear gain is fundamentally different from previous observations in [54–56] which only compared the revenue from co-optimization with one single application rather than the sum of the applications. The results in [53] hinted at the relationship between co-optimization revenue and the sum of multiple revenue streams, while mainly focusing on the trade-off between different applications and their "subadditivity". The key observation in our work is that batteries can achieve much larger economic benefits than previously thought if they jointly provide multiple services by exploring the diversity of different applications.



(a) Microsoft data center

(b) UW EE&CSE building

Figure 3.2: Example day load of Microsoft data center and UW EE & CSE building. Both loads are normalized with respect to their rated power.

3.2 Problem Formulation

This section provides some basic definitions and the detailed model setup. We consider a finite time horizon partitioned into T discrete intervals, indexed by $t \in \{1, 2, ..., T\}$. Table 3.1 summarizes the terms and notations used throughout this chapter. The rest of the section sets up the overall optimization problem in three steps. First, we explain how the electricity bill is calculated for a large commercial user. Then we focus on two potential applications of using batteries: *peak shaving* and *frequency regulation*.

3.2.1 Electricity Bill of Commercial Users

We consider commercial consumers whose electricity bill consists of two parts: energy charge and peak demand charge. Let $s(t)$ be the power consumption at at t and t_s be the size of a time step. Then the energy charge is given by:

$$J^{elec} = \lambda_{elec} \sum_{t=1}^{T} s(t) \cdot t_s, \tag{3.1}$$

where λ_{elec} is the price of energy with a unit of $\$/MWh$. The peak demand charge is based on the maximum power power consumption. In practice, this charge is calculated from a running average of power consumption over 15 or 30 minutes. Let $\bar{s}(t)$ denote the smoothed demand and then the peak demand charge can be written as,

$$J^{peak} = \lambda_{peak} \max_{t=1,2,...,T} [\bar{s}(t)]. \tag{3.2}$$

For the rest of the chapter, the time step size t_s is absorbed into the price coefficients for simplicity. Hence, the total electricity bill for a commercial user over time T is,

$$J = \lambda_{elec} \sum_{t=1}^{T} s(t) + \lambda_{peak} \max_{t=1,2,...,T} [\bar{s}(t))] \,, \tag{3.3}$$

This cost function is convex in $s(t)$ since it is a linear combination of linear functions and a piece-wise max function. We will investigate how to reduce the total cost (3.3) by using

Table 3.1: Summary of Notations

Symbols	Definition
λ_{elec}	Energy charge ($/MWh$)
λ_{peak}	Peak demand charge ($/MW$)
λ_c	Frequency regulation capacity revenue ($/MWh$)
λ_b	Battery degradation cost ($/MWh$)
λ_{cell}	Battery cell cost ($/Wh$)
λ_{mis}	Frequency regulation mismatch penalty ($/MWh$)
$r(t)$	Frequency regulation signal
$b(t)$	Battery power: $b(t) = b^{dc}(t) - b^{ch}(t)$ (MW)
$b^{ch}(t), b^{dc}(t)$	Battery charging and discharging power (MW)
η_c, η_d	Battery charging and discharging efficiency
$s(t)$	Commertical building load (MW)
$y(t)$	Energy baseline of commertical building (MW)
C	Frequency regulation capacity bid (MW)
t_s	Time resolution (seconds)
P	Battery power capacity (MW)
E	Battery energy capacity (MWh)
$SoC_{ini}, SoC_{min}, SoC_{max}$	Initial/ minimal/ maximal battery energy percentile
q	superlinear saving ratio
J	Original electricity bill
J^{peak}	Peak demand charge
J^{elec}	Energy charge
J^p	Optimal electricity bill under peak shaving
J^r	Optimal electricity bill under frequency regulation
J^{joint}	Optimal electricity bill under joint optimization
C^*	Day-ahead decision on frequency regulation capacity
U^*	Day-ahead decision on optimal peak shaving threshold

battery energy storage (BES). Specifically, we consider two applications, peak shaving and frequency regulation.

3.2.2 Peak Shaving

The peak demand charge of commercial users could be as large as their energy cost. Therefore smoothing or flattening peak demand represents an important method of reducing their electrical bills. A myriad of methods for peak shaving have been proposed in the literature, e.g., using energy storage [58], load shifting and balancing [59]. Here we focus on using batteries. Batteries can discharge energy when demand is high and charge in other times to smooth user's consumption profiles. Let $b(t)$ denote the power injected by the battery, with the convention that $b(t) > 0$ represents discharging and $b(t) < 0$ represents charging. Then $s(t) - b(t)$ is the actual power draw from the grid. Let $\mathbf{b} = [b(1) \ \dots \ b(T)]$ be the vector of battery actions. The total electricity bill becomes

$$\begin{aligned} J^P = \lambda_{elec} \sum_{t=1}^{T} [s(t) - b(t)] + f(\mathbf{b}) \\ + \lambda_{peak} \max_{t=1\dots T} [(\bar{s}(t) - \bar{b}(t))] \end{aligned} \tag{3.4}$$

where $\bar{s}(t)$ is the averaged power injection of the battery and $f(\mathbf{b})$ models the degradation effect of using the battery. Here we consider a linear battery degradation model where $f(\mathbf{b}) = \lambda_b |b(t)|$ and λ_b is the battery degradation cost coefficient.

3.2.3 Frequency regulation Service

Besides doing peak shaving, commercial users could earn revenue by providing grid services. In this chapter, we consider using batteries owned by these users to participate in the frequency regulation market. In particular, we adopt a simplified version of the PJM regulation market [11]. Fig. 3.3 gives an example of the PJM fast regulation signal for 2 hours.

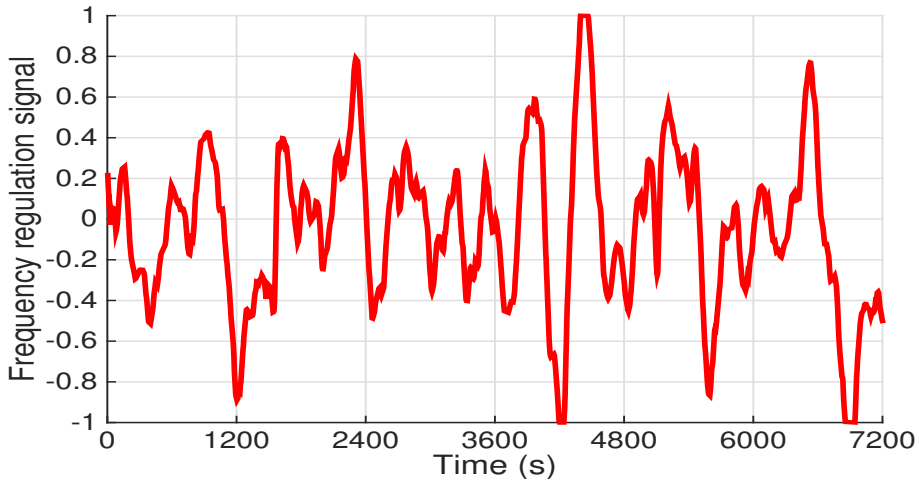


Figure 3.3: PJM fast regulation signal for 2 hours.

Compared with traditional frequency regulation signals, it has a much faster ramping rate and is designed to have a zero-mean within a certain time interval, which is well aligned with the characteristics of batteries. For providing frequency regulation service, the grid operator pays a per-MW option fee λ_c to a resource with stand-by regulation power capacity C. While during the regulation procurement period, the resource is subjected to a per-MWh regulation mismatch penalty (λ_{mis}) for the absolute error between the instructed dispatch and the resource's actual response. Let $r(t)$ be the normalized regulation signal, and the revenue from providing regulation service over time T is:

$$R = \lambda_c C \cdot T - \lambda_{mis} \sum_{t=1}^{T} |b(t) - Cr(t)| - f(\mathbf{b}), \tag{3.5}$$

where $f(\mathbf{b})$ is again the operating cost of the battery.

3.3 Joint Optimization Framework

3.3.1 The joint optimization model

We consider using a battery to provide frequency regulation service and peak shaving simultaneously, thus to boost the economic benefits. The stochastic joint optimization problem is given in (3.6), which captures both the uncertainty of future demand $s(t)$ and the uncertainty of future regulation signals $r(t)$.

$$J^{joint} = \min_{C, b^{ch}(t), b^{dc}(t), y(t)} \lambda_{elec} \sum_{t=1}^{T} E_{\mathbf{s}}[s(t) - b(t)] + \lambda_{peak} \max_{t=1\ldots T} E_{\mathbf{s}}\Big[\bar{s(t)} - \bar{b(t)}\Big] + \sum_{t=1}^{T} f(b(t))$$
$$- E_{\mathbf{rs}}\Bigg[\lambda_c T \cdot C - \lambda_{mis} \sum_{t=1}^{T} | -s(t) + b(t) + y(t) - Cr(t)|\Bigg] \tag{3.6a}$$

$$\text{s.t. } b(t) = b^{dc}(t) - b^{ch}(t)\,, \tag{3.6b}$$

$$C \geq 0\,, \tag{3.6c}$$

$$SoC_{min} \leq \frac{SoC_{ini} + \sum_{\tau=1}^{t}\Big[b^{ch}(\tau)\eta_c - \frac{b^{dc}(\tau)}{\eta_d}\Big]t_s}{E} \leq SoC_{max} \tag{3.6d}$$

$$0 \leq b^{ch}(t) \leq P^{max}\,, \tag{3.6e}$$

$$0 \leq b^{dc}(t) \leq P^{max}\,. \tag{3.6f}$$

The objective function (3.6a) minimizes the total electricity cost of a commercial user for the next day, including the energy cost, peak demand charge, battery degradation cost and regulation service revenue. The optimization variables are frequency regulation capacity C, battery charging/discharging power $b^{ch}(t)$, $b^{dc}(t)$ and frequency regulation baseline load $y(t)$. Participants in regulation market should report a baseline $y(t)$ to the grid operator ahead of their service time [11]. For a commertical user, the baseline $y(t)$ is its load forecasting. Constraint (3.6c) guarantees a non-negative regulation capacity bidding. (3.6d), (3.6e) and (3.6f) represent the battery SoC limit and battery power limits.

3.3.2 Benchmark

To show the gain of joint optimization, we describe two benchmark problems: the offline (deterministic) peak shaving problem and the offline (deterministic) frequency regulation service problem.

The offline peak shaving problem is:

$$J^p = \min_{b(t)} \lambda_{elec} \sum_{t=1}^{T} [s(t) - b(t)] + \lambda_{peak} \max_{t=1\dots T} [\bar{s(t)} - \bar{b(t)}] + \sum_{t=1}^{T} f(b(t)) \tag{3.7a}$$

$$\text{s.t. } b(t) = b^{dc}(t) - b^{ch}(t)\,, \tag{3.7b}$$

$$SoC_{min} \leq \frac{SoC_{ini} + \sum_{\tau=1}^{t} \left[b^{ch}(\tau)\eta_c - \frac{b^{dc}(\tau)}{\eta_d} \right] t_s}{E} \leq SoC_{max} \tag{3.7c}$$

$$0 \ \leq b^{ch}(t) \leq P^{max}\,, \tag{3.7d}$$

$$0 \ \leq b^{dc}(t) \leq P^{max}\,. \tag{3.7e}$$

The above problem is convex in terms of $\mathbf{b}$. We solve it and denote the optimal bill value as J^P. The offline frequency regulation problem is:

$$R^* = \max_{C, b(t)} \lambda_c T \cdot C - \lambda_{mis} \sum_{t=1}^{T} |b(t) - Cr(t)| - \sum_{t=1}^{T} f(b(t)) \tag{3.8a}$$

$$\text{s.t. } b(t) = b^{dc}(t) - b^{ch}(t)\,, \tag{3.8b}$$

$$C \geq 0\,, \tag{3.8c}$$

$$SoC_{min} \leq \frac{SoC_{ini} + \sum_{\tau=1}^{t} \left[b^{ch}(\tau)\eta_c - \frac{b^{dc}(\tau)}{\eta_d} \right] t_s}{E} \leq SoC_{max} \tag{3.8d}$$

$$0 \ \leq b^{ch}(t) \leq P^{max}\,, \tag{3.8e}$$

$$0 \ \leq b^{dc}(t) \leq P^{max}\,. \tag{3.8f}$$

The above regulation revenue maximization problem does not consider the effect of providing frequency regulation service on electricity bills. In these benchmarks, we assume that complete knowledge of the future. In essence, the benchmarks here represent the best possible performance of any algorithms that solve these problems individually.

Recall that frequency regulation is a service managed by grid operators, while as an end consumer, the commercial user's electricity supply contract with the utility is unchanged, thus the user still subjects to the energy and peak demand charge. Therefore, the overall electricity bill J^r is,

$$J^r = \lambda_{elec} \sum_{t=1}^{T} [s(t) - b^r(t)] + \lambda_{peak} \max_{t=1\ldots T} [\bar{s(t)} - \bar{b}^r(t)] - R^* \,, \tag{3.9}$$

where $b^r(t)$ is the optimal battery responce for frequency regulation service and R^* the optimal service revenue.

Both of the benchmark problems are convex because all of the constraints are linear and objectives are an addition of convex functions (pointwise maximum is a convex function). To solve these problems, we use the CVX package for Matlab, a generic package for solving convex problems. We used a 2.5 GHz Intel Core i7 Macbook with 16 GB memory. The problem size can be fairly large, since the time resolution is 4s. But even for an 8-hour horizon, the problem can be solved in about 10 minutes.

3.3.3 The superlinear gain

Our results highlight that a *superlinear* gain can often be obtained: the saving from the *stochastic* joint optimization can be larger than the *sum* of two benchmark optima. In mathematical form, superlinear gain denotes the following phenomenon, which often holds in practice,

$$J - J^{joint} > (J - J^r) + (J - J^p) \,, \tag{3.10}$$

Table 3.2: One day electricity bill for Microsoft data center under four scenarios: original bill, bills after providing frequency regulation service, peak shaving, and joint optimization. Bill savings and saving ratios are listed in the third column.

	Total ($)	Savings ($)	Peak charge ($)	Battery cost ($)	FR gain ($)
Original	1345.7	**0**	461.5	0	0
FR only	1254.6	**91.1 (6.77 %)**	528.7	123.1	301.4
PS shaving	1331.9	**23.8 (1.76%)**	424.8	12.9	0
Joint Optimization	1194.5	**151.2 (11.24%)**	465.8	117.2	272.7

where the left side of Eq. (3.10) is the saving from joint optimization, and the right side represents the sum of savings from two benchmark problems. The key observation in such case is the "super-additivity". The revenue from co-optimization of multiple applications is not only higher than the revenue from any single application (which may be obvious), but also higher than the sum of revenues from all individual applications.

We provide an example to demonstrate the superlinear gain. Table 3.2 gives the daily electricity bill under four scenarios for a 1MW data center: the original bill (batteries are not used), using battery only for frequency regulation, using battery only for peak shaving, and using battery for both services. The bill savings are marked in red, from which we observed that saving from joint optimization is larger than the sum of each individual applications. The load curve and regulation signals for that day are given in Fig. 3.1.1 and Fig. 3.3.

3.4 Online Algorithm

The previous section demonstrates that battery joint optimization could have superlinear gains. In this section, we propose a battery control algorithm for joint optimization. The challenge in combining peak shaving and frequency regulation service together lies in their

vastly different timescales. To deal with the timescale difference, we divided the optimization problem into two stages, 1) day-ahead decision on peak shaving threshold and frequency regulation capacity bidding; 2) real-time control of battery charging/discharging. Fig. 3.4 summarizes the workflow of the overall control algorithm. In this section, we first introduce the load prediction and scenario reduction methods for solving the day-ahead optimization problem, and then a real-time battery operation algorithm is presented.

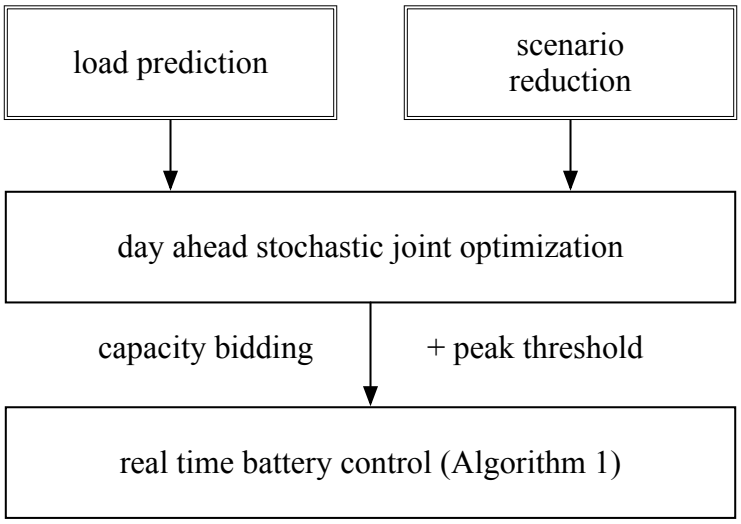


Figure 3.4: Work flow of the proposed battery control method. We use load prediction and scenario reduction to solve the day-ahead stochastic optimization problem. Then we feed in the capacity bidding and peak threshold for real-time control.

3.4.1 Load prediction and scenario generation

We use a multiple linear regression (MLR) model [60] for day-ahead load prediction, which is simple, easy to implement in commercial users' site, yet achieves high prediction accuracy. Details of the load prediction algorithm are given in Appendix A. We used the 10-fold cross validation method to evaluate the MLP load prediction model, and the resulting mean

absolute percentage error (MAPE) is 3.7% for Microsoft data center load and 2.3% for University of Washington EE & CSE building. MAPE is a measure of prediction accuracy, which is calculated by averaging the absolute deviation between real value and prediction divided by the actual value.

To deal with the uncertainty of future regulation signal, a scenario-based method is implemented. Here, we use one-year historical data to empirically model the distributions of regulation signals. Each daily realization of the regulation signal is called a "scenario", and thus we obtain 365 scenarios. We applied the forward scenario reduction algorithm in [61] to select the best subset of scenarios. We set the number of selected scenarios as 10, which strives for a balance between performance and computational complexity. Therefore, we have a set of 10 scenarios for regulation signals denoted as Ω and each scenario associated with a realization probability ω_i, which together compose the uncertainty set.

We solve the stochastic problem in (3.6) using the load prediction $\hat{s}(t)$ and contructed regulation signal uncertainty sets Ω. Define the optimal battery response and regulation capacity as $b^*(t)$ and C^*, then the optimal peak shaving threshold U^* is,

$$U^* = \max_{t=1\ldots T} \left[\bar{\hat{s}}(t) - \bar{b}^*(t) \right], \tag{3.11}$$

3.4.2 Real-time control for battery charging/discharging

Section 3.4.1 describes how to make day-ahead decisions on capacity bidding and peak threshold. Here we introduce a simple real-time battery control algorithm for joint optimization. It is computationally efficient, only requires the measurement of battery's real-time state of charge (SoC) and achieves near-optimal performance compared with the offline optima with perfect foresight. Although more sophisticated methods such as model predictive control [62] or dynamic programming [54] have been proposed, they are not needed in this case.

The intuition for the real-time joint optimization control algorithm comes from the optimal

Table 3.3: Comparison between the simple threshold algorithm and the offline solution with full information

	Data center	UW EE & CSE
Total days simulated	183	365
Average original daily bill ($)	1338.1	985.3
Average joint optimization bill with perfect insight ($)	1184.3	857.2
Average joint optimization bill with simple online control ($)	1194.9	863.6

battery control algorithm for frequency regulation service. Recall the benchmark frequency regulation problem with objective (3.8a): under linear battery cost model, given a fixed capacity bidding C, we have a simple yet optimal real-time battery control method. Theorem 4 describes the optimal control algorithm for batteries providing frequency regulation service.

Theorem 4. *Assume $\lambda_b < \lambda_{mis}$.*[1] *If the marginal battery charging/discharging cost is constant within the operation region, that is, $f(b(t)) = \lambda_b|b(t)|$. For a given capacity C, the optimal battery response $b^*(t)$ for providing regulation service is:*

- $\min\{Cr(t), P, \frac{[SoC(t)-SoC_{min}]E}{\eta_c t_s}\}$, *if* $r(t) \geq 0$
- $\max\{Cr(t), -P, \frac{[SoC(t)-SoC_{max}]E\eta_d}{t_s}\}$, *if* $r(t) < 0$

where $SoC(t)$ of the battery state of charge at the beginning of time step t.

The proof of Theorem 4 is given in the Appendix C. As the theorem shows, when the marginal operation cost for battery charging/discharging is constant, the optimal battery control policy is a simple threshold policy. Following Theorem 4, we propose a real-time control algorithm for joint optimization in Algorithm 3. Table 3.3 gives a comprehensive comparison between the simple online control algorithm and the offline optima with perfect foresight based on

[1] Otherwise the battery would not be used at all.

half year of simulation results of Microsoft data center and one-year data of UW EE & CSE building. Both results show that by implementing the simple threshold algorithm, we can achieve near-optimal performance compared with the perfect foresight case.

3.5 Simulation Results

We provide a case study using half year power consumption data from Microsoft data center and one year data from University of Washington EE & CSE building. The regulation signal is from PJM fast frequency regulation market, where the considered Microsoft data center locates. We implement the simple threshold control algorithm in Section 3.4 for battery joint optimization. Simulation results demonstrate that over 80% of time, we will have the superlinear benefits by joint optimization. We also perform qualitative and quantitative analysis to provide some insights on why and when the superlinear gain happens.

3.5.1 Parameter setup

Assume that the battery optimization horizon is 1 day and the time granularity of t is $4s$, so that $T = 4320$. The electricity price is $47\$/MWh$ and peak demand charge is $12\$/kW$ per month. For regulation service, suppose the regulation capacity payment is $50\$/MWh$ and set mismatch penalty to guarantee at least 80% performance score. The BES for optimization is Lithium Manganese Oxide (LMO) battery, with high power capacity and low energy capacity. Within the SoC operation region $SoC_{min} = 0.2$ and $SoC_{max} = 0.8$, LMO battery has a constant marginal degradation cost w.r.t. how much energy is charged and discharged.

Here we consider using existing batteries in commercial users, e.g., the backup battery in data center, to participate in power market and reduce users' electricity bills. The key function of these batteries for users is to provide backup capabilities and the proposed joint optimization framework is deploying only part of the battery energy capacity. We assume the overall

Algorithm 3: Real-time control for joint optimization

Input : C^*, U^*, P, E

Initialize : $t = 0$, $SoC(1) = SoC_{init}$

for $t = 1 \rightarrow T$ **do**

Receive regulation signal, $r(t)$.

Locate the current time t in its corresponding peak calculated window, $[\tau_o, \tau_e]$.

Calculate peak demand value of the current period,

$$U = \frac{\sum_{\tau=\tau_o}^{t}[s(\tau) - b(\tau)]}{(t - \tau_o)},$$

`/* Simple control */`

if $U \leq U^*$ **then**

$b(t) = C^* \cdot r(t)$;

else

$b(t) = C^* \cdot r(t) + (U - U^*)$;

end

`/* Power and SoC thresholds */`

if $b(t) \geq 0$ **then**

$b(t) = \min\{b(t), P, \frac{[soc(t) - soc_{min}]E}{\eta_c t_s}\}$; `// battery discharge`

else

$b(t) = \max\{b(t), -P, \frac{[soc(t) - soc_{max}]E\eta_d}{t_s}\}$; `// battery charge`

end

Update the battery SoC status,

$$soc(t+1) = soc(t) - \frac{[b^{ch}(\tau)\eta_c - b^{dc}(\tau)/\eta_d]t_s}{E},$$

Proceed to the next control step: $t \leftarrow t + 1$

end

Table 3.4: Comparison between different portions of battery energy capacity used for joint optimization.

Grid service capacity	Annual bill saving ($)	Life expectancy (Year)
3 minutes	52,282	3.58
5 minutes	80,547	1.34
10 minutes	105,140	0.86

battery has a 1MW power capacity and 15 minutes energy capacity, which is a typical size of an industrial-scale grid-tied battery. Different portions of the total energy capacity are considered for grid service, 3 minutes, 5 minutes and 10 minutes respectively. The results are presented in Table 3.4, where the metrics for comparison under different scenarios are the annual bill savings and battery life expectancy. An aggressive user may try to replace their battery in a yearly basis for the largest bill reduction. More likely, for a building or a data center, a 3 year cycle is preferred. In fact, most data centers already replace their batteries every 3 to 4 years for reliability reasons [63], so using 3 minutes of the battery capacity for grid services would lead to considerable gains without any additional burdens. Of course, the remaining portion of battery energy storage is reserved for emergency backup.

Therefore we assume for joint optimization usage, the battery power rating is $1MW$, energy capacity E is 3 minute max power output, and battery cell price is $0.5\$/Wh$. If a LMO battery can be operated for $N = 10,000$ cycles when the average cycle DoD is 60%. Using Eq. (3.12), we calculate the battery degradation cost as $83\$/MWh$.

$$\lambda_b = \frac{\lambda_{cell} \cdot 10^6}{2N \cdot (SoC_{max} - SoC_{min})} \tag{3.12}$$

In order to evaluate the performance of the proposed battery joint optimization algorithm, we compare the savings from joint optimization with the sum of savings from benchmark peak shaving and regulation service problems. A criteria q (joint optimization saving ratio)

is defined as below,

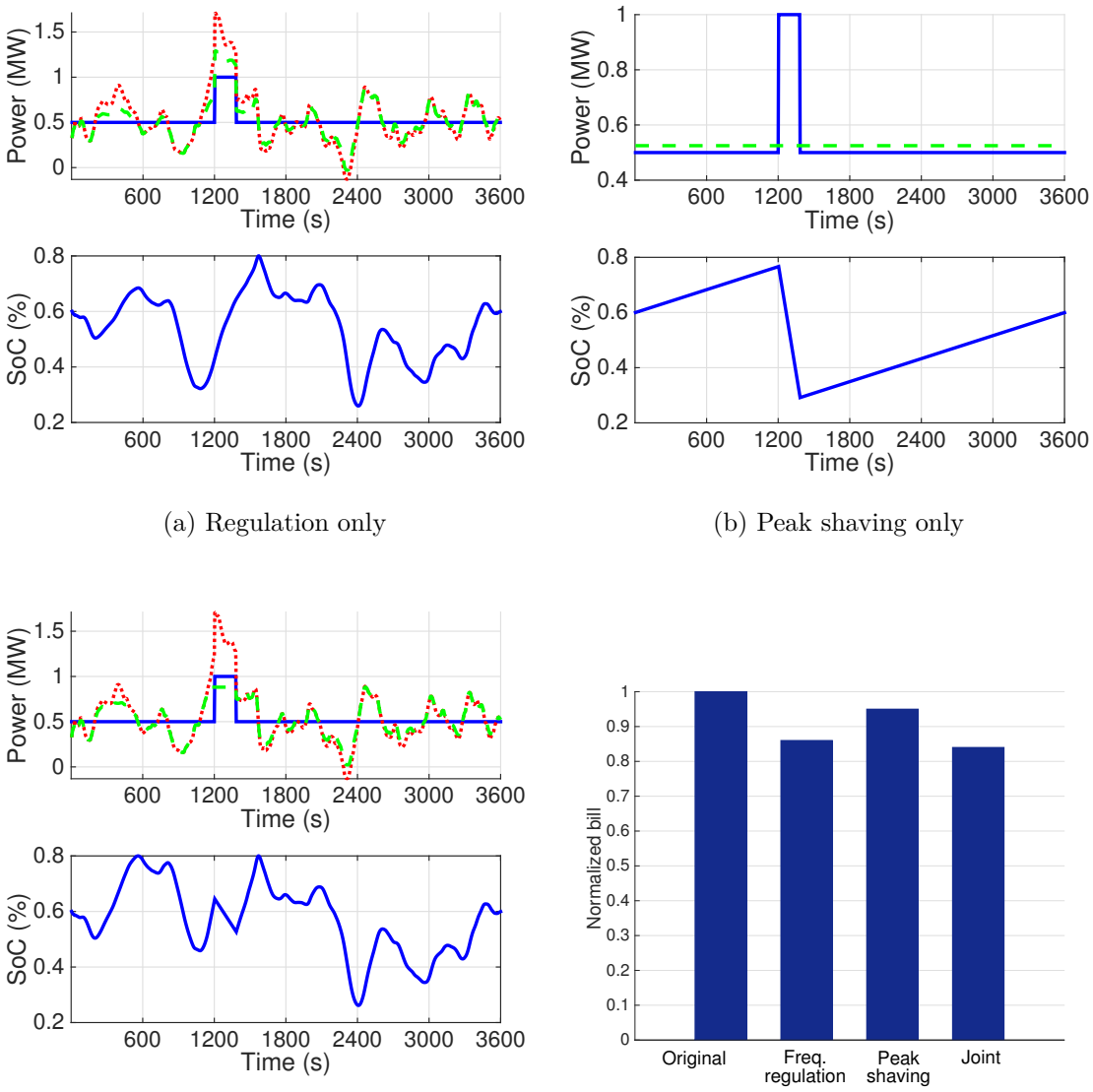
$$q = \frac{(J - J^{joint}) - [(J - J^{r}) + (J - J^{p})]}{J}, \tag{3.13}$$

which describes the percentile of superlinear saving compared to the original bill.

3.5.2 Results for synthetic load: peak shape and superlinear gain

In the previous sections (Figure 1 and Table II), we observed that by doing battery joint optimization, we have such superlinear benefits. One natural question may come up, why we have the superlinear gains? Before we dive into more simulations on real data, we pick a simple rectangle peak case where the base load is 0.5 MW, peak load is 1MW in this section for qualitative analysis, in order to better understand the conditions that lead to superlinear gains. We change the duration of peak from 3 minutes (a sharp peak) to 15 minutes (a flat peak) in order to study the effect of peak shape on the probability of superlinear gain.

Intuitively, the superlinear gain is related to the shape of demand curve. Consider two different peaks, a narrow peak (Fig. 3.5) and a wide peak (Fig. 3.6), we find that the main difference lies in the peak shaving part. For a 3 minute short-time peak, the battery could shave a large portion of the peak before hitting the SoC bound (Fig.3.5 (b)). Thus, we save a lot from only doing peak shaving and the two applications do not interact much , and there is no superlinear saving. However, when the peak duration is long, it takes more battery energy to shave the same height off the peak. As seen from Fig.3.6 (b), the battery doesn't respond much in the peak shaving only case because the cost of using battery gets close (or even exceeds) the saving from reduced peak demand charge. This argument is verified by Fig.3.6 (d), where we find only doing peak shaving does not reduce the bill much. But if we consider joint optimization, the randomness of regulation signal helps break down the one flat peak into several short-time peaks, and we could save more from doing peak shaving on top of providing regulation service. This is where the superlinear saving comes from.



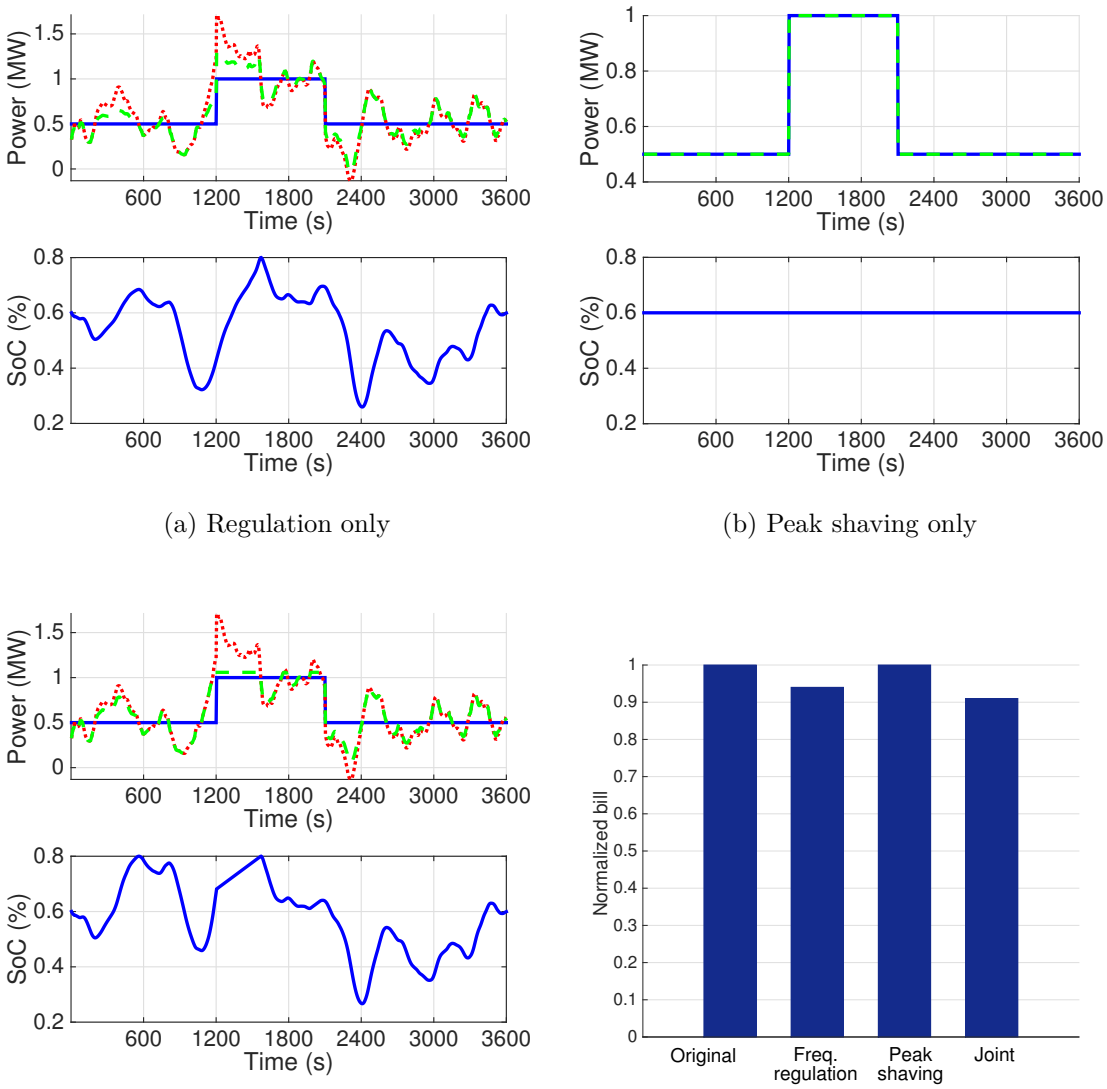
(a) Regulation only

(b) Peak shaving only

(c) Joint optimization

(d) Bills comparison

Figure 3.5: Electricity bills for narrow peak (base load 0.5 MW, peak load 1MW, peak duration 3 minutes). **Labels:** In subfigures (a), (b), (c), the upper plot is power consumption; the lower plot is battery SoC curve. Blue solid line is the original load; red dotted dash line denotes demand+frequency regulation signal; green dashed line is the actual net consumption. Fig. d are normalized bill where the original bill is set to 1.



(a) Regulation only

(b) Peak shaving only

(c) Joint optimization

(d) Bills comparison

Figure 3.6: Electricity bills for narrow peak (base load 0.5 MW, peak load 1MW, peak duration 15 minutes). **Labels:** In subfigures (a), (b), (c), the upper plot is power consumption; the lower plot is battery SoC curve. Blue solid line is the original load; red dotted dash line denotes demand+frequency regulation signal; green dashed line is the actual net consumption. Fig. d are normalized bill where the original bill is set to 1.

Table 3.5: Half year simulation results of Microsoft data center

Microsoft data center	
Total days simulated	183
Days having superlinear gain	151
Average bill saving by joint optimization	10.72%
Super linear saving ratio (q)	2.71%

Table 3.6: One year simulation results of UW EE & CSE building

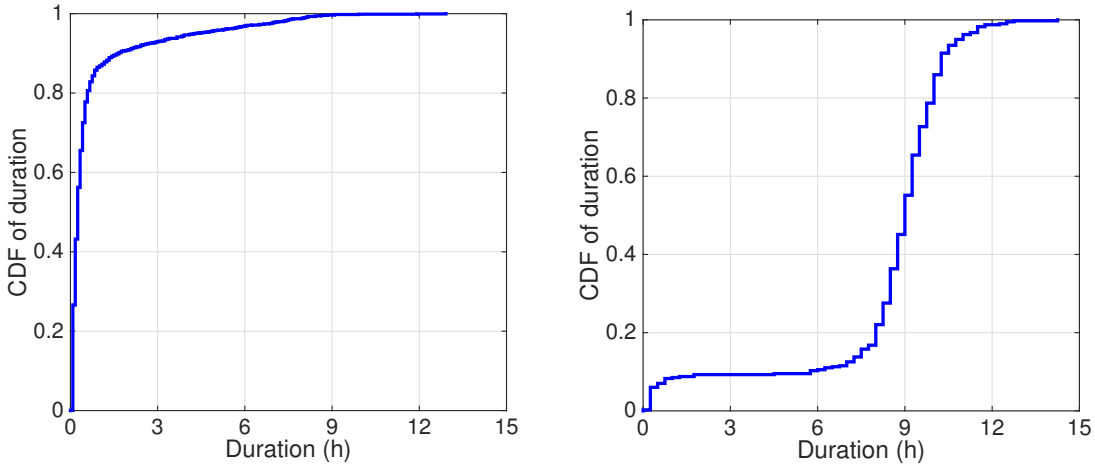
UW EE &CSE building	
Total days simulated	365
Days having superlinear gain	362
Average bill saving by joint optimization	12.35%
Super linear saving ratio (q)	3.91%

3.5.3 Results for real-life data: Microsoft data center and UW EE & CSE building

This section conducts simulations with data from Microsoft data center and UW EE & CSE building. Table 3.5 and 3.6 summarize the simulation results. We consider using a 1MW, 3 minutes battery for grid service, and the reported numerical results are based on the proposed simple online control algorithm. For a 1MW data center with $488,370 annual electricity bill, the cost saving by joint optimization is around $52,282 (10.72%), with $13,234 extra saving compared with the *sum* of benchmark optima. For UW EE & CSE building, 362 out of the 365 days, we have the superlinear gain. The annual electricity bill for UW EE & CSE building is around $359,634, from which we save $44,420 (12.35%) by implementing battery joint optimization. The superlinear gain is $14,061 per year.

To link the quantitative analysis of synthetic load in the previous section to the real-life cases

of data center and UW EE & CSE load, we preform a statistical analysis of load peaks. We plot the Cumulative Distribution Function (CDF) of peak duration for the data center and the EE&CSE building in Fig. 3.7, where the average peak duration for data center is 0.75h (about 45 minutes) and 8.33h for the building.



(a) Microsoft data center

(b) UW EE & CSE building

Figure 3.7: Cumulative distribution of peak duration for two case studies.

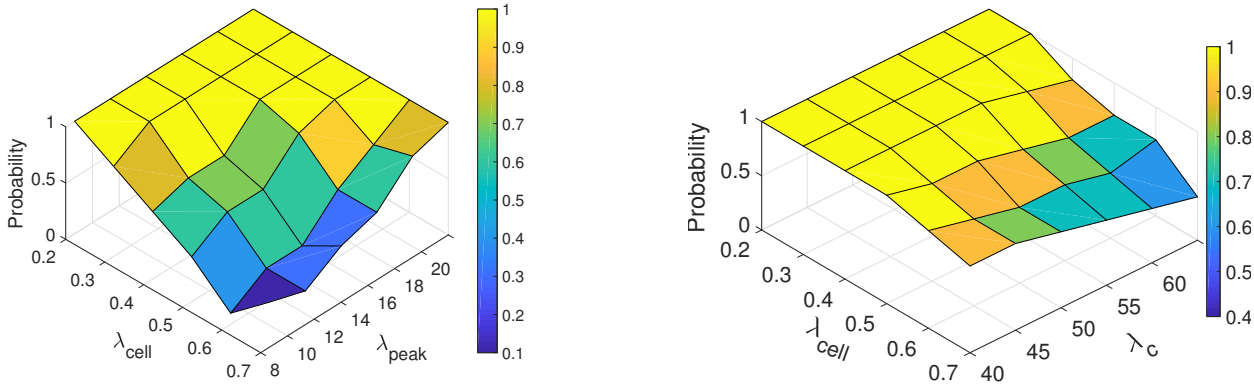
According to the quantitative analysis in Figures 6 and 7, the proposed battery joint optimization has a larger gain for flat peaks compared to sharp peaks. Since the randomness of regulation signal help break down the one flat peak into several short-time peaks, we could save more from doing peak shaving on top of providing regulation service. The exact definition of "long" and "short" peaks depend on the size of the battery. For a 3 minute battery, if the peak is shorter than 3 minutes, then performing joint optimization is not critical and we do not have a superlinear gain. On the other hand, for a peak that is longer than 3 minutes, it is important to use the regulation signal to break it up into smaller peaks. Therefore, both case studies have high superlinear gain probability, which is greater than 80%. The superlinear gain ratio of UW building (99%) is higher than the ratio of the data

center (82.5%) since there are virtually no peaks shorter than the battery capacity in the former and a still a few short peaks in the latter.

3.5.4 Sensitivity analysis

In addition, we preform sensitivity analysis about how different price settings, including different demand charge prices λ_{peak}, battery degradation costs λ_b and regulation payments λ_c, influence the superlinear gain ratio. In order to quantatively evaluate the conditions when superlinear gain will happen and generalize the analysis to all kinds of potential scenarios, we pick a simple load curve with a rectangle peak (base load 0.5 MW, peak load is 1 MW, peak duration 15 minutes) and a truncated Gaussian signal as frequency regulation signal, with $\mu = 0$, $\sigma^2 = 0.12$ and range $[-1, 1]$.

Fig. 10a shows how the chance of having superlinear gain changes with regard to battery cell price and peak demand charge. The probability of superlinear gain increases as the battery cell price goes down, or as the peak demand charge goes up. The "physical origin" of the superlinear gain is the positive interaction between peak shaving and frequency regulation service. Since the randomness in the frequency regulation signal break the flat peak into several smaller peaks, more savings are obtained by performing peak shaving on the top of frequency regulation. Therefore, as the peak demand price goes up, it yields more economic benefits to jointly optimize the two applications. Similarly, the as the battery prices decreases, it can be used more aggressively for both applications. As battery prices continue to decrease in the future, the benefits of joint optimization will increase. Fig. 10b demonstrates how the probability of having superlinear relates to battery cell price and regulation capacity payment. The chance of having superlinear gain is the highest when both the battery cell price and regulation capacity payment are low. When the capacity payment is high enough, it yields much more economic benefits to provide frequency regulation service than peak shaving. In such condition, the probability of having superlinear gain decreases.



(a) Probability of superlinear gain V.S. battery cell price λ_{cell} and peak demand charge λ_{peak}

(b) Probability of superlinear gain V.S. battery cell price λ_{cell} and frequency regulation capacity payment λ_c

Figure 3.8: Superlinear gain probability V.S. price coefficients

3.6 Conclusion

This chapter addresses the multiplexing of battery in large commercial users to reduce the electricity bills. We consider two sources of cost savings: reducing peak demand charge and gaining revenue in the frequency regulation market. We formulate a framework that jointly optimizes battery usage for both of these applications. Surprisingly, we observe that a superlinear gain can often be obtained: the savings from joint optimization can be larger than the sum of the individual savings from devoting the battery to one of the applications. We also develop an online control algorithm which achieves the superlinear gain.

Chapter 4

DECISION MAKING UNDER MODEL UNCERTAINTY: INPUT CONVEX NEURAL NETWORKS FOR BUILDING CONTROL AND VOLTAGE REGULATION

4.1 Introduction

The previous two chapters talk about energy storage control under environmental uncertainty. In the following chapter, we will focus on internal uncertainties and how to optimally control the system subject to them. It turns out the key is to carefully design machine learning architectures that leverage the physics of the system.

One key challenge faced in the control of energy systems is that they tend to have complicated and poorly understood dynamics, sometimes with legacy components are built over a long period of time [64]. Therefore detailed models for these systems may not be available or may be intractable to construct. For instance, since buildings account for 40% of the global energy consumption [65], many approaches have been proposed to operate buildings more efficiently by controlling their heating, ventilation, and air conditioning (HVAC) systems [66]. Most of these methods, however, suffer from two drawbacks. On one hand, a detailed physics model of a building can be used to accurately describe its behavior, but this model can take years to develop. On the other hand, simple control algorithms have been developed by using linear (RC circuit) models [67] to represent buildings, but the performance of these models may be poor since the building dynamics can be far from linear [68].

In recent years, with the growing deployment of sensors in energy systems, a large amount

of operational data have been collected, such as in smart meters [69], user behaviors [70] and PMUs [71]. Using these data, the system dynamics can be learned directly and then automatically updated at periodic intervals. One popular method is to parameterize these complex system dynamics using deep neural networks, yet few research investigated how to integrate deep learning models into real-time closed-loop control of physical systems. A key reason that deep neural networks have not been directly applied in control is that even though they provide good performances in learning system behaviors, optimization on top of these networks is challenging [72]. A deep neural network, for example, may be much better in learning the relationship between temperature set points in a building and its power consumption than a linear model, yet it is not necessarily the case that it would be better to be used to *optimize* the setpoints for energy consumption reduction. Neural networks, because of their structures, are generally not convex from input to output. Therefore, many control applications (e.g., where real-time decisions need to be made) choose to favor the computational tractability offered by linear models despite their poor fitting performances.

4.1.1 Literature Review

The work in [73] was an impetus for this work. The key differences are that the goal in [73] is to show that ICNN can achieve similar classification performances as conventional neural networks and how the former can be used in inference and prediction problems. Our goal is to use these networks for optimization and closed-loop control, and in a sense that we are more interested in the overall system performances and not directly the performance of networks. We also extend the class of networks to include RNNs to capture dynamical systems.

Control and decision-making have used deep learning mainly in model-free end-to-end controller settings [74, 75] (shown in Fig. 4.1 (a)). However, much of the success relies heavily on a reinforcement learning setup where the optimal state-action relationship can be learned via a large number of samples. However, many physical systems do not fit into

the reinforcement learning process, where both the sample collection is limited by real-time operations, and is difficult to explore the whole design space since suboptimal actions would lead to disastrous results (e.g., training a controller of a power system by trying different actions may lead to blackouts). Moreover, there are usually model constraints in physical systems (e.g., peak output for building cooling system) which could neither be directly modeled nor represented efficiently by end-to-end policies.

To address the above sample efficiency, safety and model constraints incompatibility concerns faced by model-free reinforcement learning algorithms, we consider a model-based control approach in this work. Model-based control algorithms often involve two stages – system identification and controller design. For the system identification stage, the goal is to learn a fixed form of system model to minimize some prediction error [76]. Most efficient model-based control algorithms have used a relatively simple function estimator for the system dynamics identification [77], such as linear model [67] and Gaussian processes [78]. These simplified models are sample-efficient to learn, and can be nicely incorporated in the sub-sequent optimal control problems. However, such simple models may not have enough representation capacity in modeling large-scale or high-dimension systems with nonlinear dynamics. Deep neural networks (DNNs) feature powerful representation capability, while the main challenge of using DNNs for system identification is that such models are typically highly non-linear and non-convex [72], which causes great difficulty for following decision making. Our work shows how the proposed ICNN control algorithm achieves the benefits from both sides of the world. By making the neural network convex from input to output, we are able to both obtains good identification accuracies and tractable computational optimization problems.

A recent work from [77] is close in spirit as our proposed method. The authors use a model-based approach for robotics control, where they first fit a neural network for the system dynamics and then use the fitted network in an MPC loop. However, since [77] use conventional NN for system identification, they cannot solve the MPC problem to global optimality. Instead, they use a random shooting method where they choose K random action

sequences and choose the action sequence which gives the lowest cost. Compared with the random shooting approach, our proposed method is guaranteed reach the global optima with significantly less computational time, due to the good representation power of ICNNs (Theorem 1) and nice mathematical properties of convex optimization.

4.1.2 Contribution

We tackle the modeling accuracy and control tractability tradeoff faced by many data-driven control approaches, by building on the input convex neural networks (ICNN) to both represent system dynamics and to find optimal control policies. By making the neural network convex from input to output, we are able to obtain *both good predictive accuracies and tractable computational optimization problems.* The overall methodology is shown in Fig. 4.1. Our

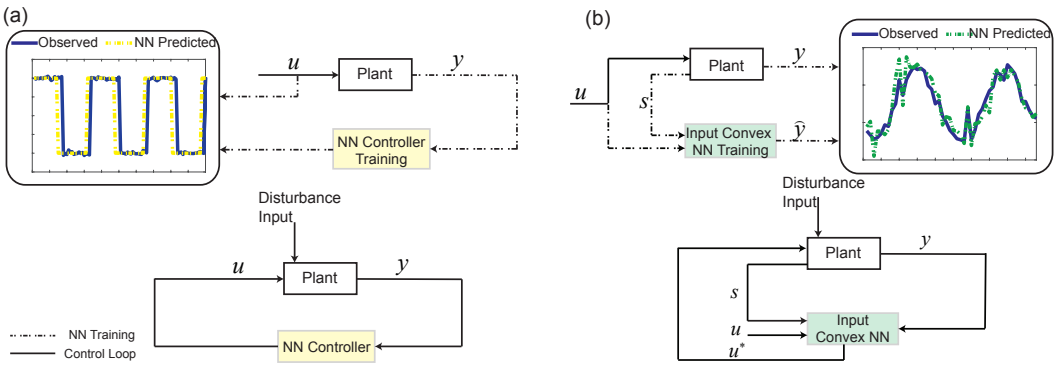


Figure 4.1: (a) Model-free end-to-end controller design, where a model is trained to find the best control actions based on observations. (b) Our proposed model-based method, an input convex neural network is first trained to learn the system dynamics, then we solve a convex predictive control problem to find the best actions.

proposed method (shown in Fig. 4.1 (b)) firstly utilizes an input convex network model to learn the system dynamics and then computes the best control decisions via solving a convex model predictive control (MPC) problem, which is tractable and has optimality guarantees.

This is different from existing methods that uses model-free end-to-end controller which directly maps input to output (shown in Fig. 4.1 (a)). Another major contribution of our work is that we explicitly prove that ICNN can represent all convex functions and is *exponentially* more efficient than widely used convex piecewise linear approximations [79].

4.2 Problem Formuation

In this section, we first setup the general optimization model where a neural network is used in a closed-loop system. The fundamental goal is to optimize system performance which is beyond the learning performance of network on its own. In this section we describe how input convex neural networks (ICNN) can be extremely useful in these systems by considering two related problems. First, we show how ICNN perform in single-shot optimization problems. Then we extend the results to an input convex *recurrent* neural networks (ICRNN), which allows us to both capture systems' complex dynamics and make time-series decisions.

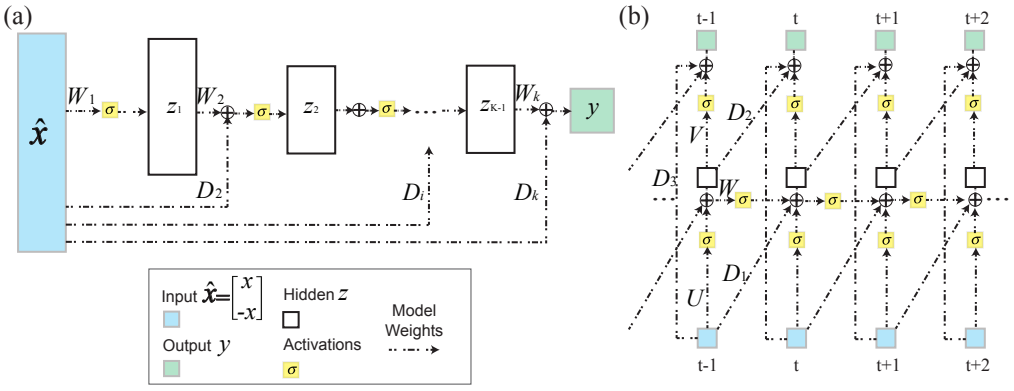


Figure 4.2: Input convex neural networks. (a) Input convex feed-forward neural networks (ICNN). One notable addition is the direct "passthrough" layers $\mathbf{D}_{2:k}$ that connect the inputs to hidden units for better model representation ability. (b) The proposed input convex recurrent neural networks (ICRNN) architectures. In our control settings, we keep all weights in both networks nonnegative, while expanding the inputs with $-\mathbf{u}$.

4.2.1 Single-shot problem

The following proposition states the sufficient condition for a network to be input convex:

Proposition 1. *The feedforward neural network in Fig. 4.2(a) is convex from input to output given that all weights between layers* $\mathbf{W}_{1:k}$ *and weights in the "passthrough" layers* $\mathbf{D}_{2:k}$ *are non-negative, and all of the activation functions are convex and nondecreasing (e.g. ReLU).*

The structure of the input convex neural network (ICNN) structure in Proposition 1 is motivated by the structure in [73] but modified to be more suitable to control of dynamical systems. In [73] it only requires $\mathbf{W}_{2:k}$ to be non-negative while having no restrictions on weights $\mathbf{W}_1$ and $\mathbf{D}_{2:k}$. Our construction achieves the exact representation by expanding the inputs to include both $\mathbf{u}$ ($\in \mathbb{R}^d$) and $-\mathbf{u}$. Then any negative weights in $\mathbf{W}_1$ and $\mathbf{D}_{2:k}$ in [73]'s ICNN structure is set to zero and its negation (which is positive) is added as the weight for corresponding $-\mathbf{u}$. The reason for our construction is to allow the network to be "rolled out in time" when we are dealing with dynamical systems and multiple networks need to be composed together.

An simple example that demonstrates how the proposed ICNN can be used to fit a convex function comes form fitting the $|u|$ function. This function is convex and both decreasing and increasing. Let the activation function be $ReLU(\cdot) = \max(\cdot, 0)$. We can write $|u| = -u + 2ReLU(u)$ [73]. However, in this representation, we need a negative weight, the -1 in front of u, and this would be troublesome if we compose several networks together. In our proposed ICNN structure with all positive weights and input negation duplicates, we can write $|u| = v + 2ReLU(u)$, where we impose a constraint $v = -u$. Such doubline on the number of input variables may potentially make the network harder to train. Yet during control, having all of the weights positive maintains the convexity between inputs and outputs even if multiple steps are considered which will be discussed in Section **??**. The constraint $v = -u$ is linear and can be easily included in any convex optimization.

This proposition follows directly from composition of convex functions [47]. Although it allows for any increasing convex activation functions, in this work we work with the popular ReLU activation function. Two notable additions in ICNN compared with conventional feedforward neural networks are: 1) Addition of the direct *"passthrough" layers* connecting inputs to hidden layers and conventional *feedforward layers* connecting hidden layers for better representation power. 2) the expanded inputs that include both $\mathbf{u}$ and $-\mathbf{u}$. The proposed ICNN structure is shown in Fig. 4.2(a). Note that such construction guarantees that the network is convex and non-decreasing with respect to the expanded inputs $\hat{\mathbf{u}} = \begin{bmatrix} \mathbf{u} \\ -\mathbf{u} \end{bmatrix}$, while the output can achieve either decreasing or non-decreasing functions over $\mathbf{u}$.

Fundamentally, ICNN allows us to use neural networks in decision making processes by guaranteeing the solution is unique and globally optimal. Since many complex input and output relationships can be learned through deep neural networks, it is natural to consider using the learned network in an optimization problem in the form of

$$\min_{\mathbf{u}} \ f(\mathbf{u}; \mathbf{W}) \tag{4.1a}$$

$$\text{s.t.} \quad \mathbf{u} \in \mathcal{U}, \tag{4.1b}$$

where $\mathcal{U}$ is a convex feasible space. Then if f is an ICNN, optimizing over $\mathbf{u}$ is a convex problem, which can be solved efficiently to global optimality. Note that we will always duplicate the variables by introducing $\mathbf{v} = -\mathbf{u}$, but again this does not change the convexity of the problem. Of course, since the weights of the network are restricted to be nonnegative, the performance of the network may be worse. A common thread we observe is that trading off between performance with tractability can be preferable.

4.2.2 Closed-loop control and recurrent neural networks

In addition to the single-shot optimization problem in (4.1), we are interested in optimally controlling a dynamical system. To model the temporal dependency of the system dynamics,

we propose to use recurrent neural networks (instead of feed-forward neural networks). Recurrent networks carry an internal state of the system, which introduces coupling with previous inputs to the system. Fig. 4.2(b) shows the proposed input convex recurrent neural networks (ICRNN) structure. This network maps from input $\hat{\mathbf{u}}$ to output y with memory unit $\mathbf{z}$ according to the following Eq. (4.2),

$$\mathbf{z}_t = \sigma_1(\mathbf{U}\hat{\mathbf{u}}_t + \mathbf{W}\mathbf{z}_{t-1} + \mathbf{D}_2\hat{\mathbf{u}}_{t-1}), \tag{4.2}$$

$$y_t = \sigma_2(\mathbf{V}\mathbf{z}_t + \mathbf{D}_1\mathbf{z}_{t-1} + \mathbf{D}_3\hat{\mathbf{u}}_t), \tag{4.3}$$

where $\hat{\mathbf{u}} = \begin{bmatrix} \mathbf{u} \\ -\mathbf{u} \end{bmatrix}$, and D_1, D_2, D_3 are added direct "passthrough" layers for augmenting representation power. If we unroll the dynamics with respect to time, we have $y_t = f(\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, ..., \hat{\mathbf{u}}_t; \theta)$ where $\theta = [\mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{D_1}, \mathbf{D_2}, \mathbf{D_3}]$ are network parameters, and σ_1, σ_2 denote the nonlinear activation functions. The next proposition states a sufficient condition for the network to be input convex.

Proposition 2. *The network shown in Fig. 4.2(b) is a convex function from inputs to output if all weights U, V, W, D_1, D_2, D_3 are non-negative, and all activation functions are convex and nondecreasing (e.g. ReLU).*

The proof of this proposition again follows directly from the composition rule of convex functions. Similarly to the ICNN case, by expanding the inputs vector to include both $\mathbf{u}$ and $-\mathbf{u}$ and restricting all weights to be non-negative, the resulted ICRNN structure is a convex and non-decreasing mapping from inputs to output.

The proposed ICRNN structure can be leveraged to represent system dynamics for close-loop control. Consider a physical system with discrete-time dynamics, at time step t, let's define $\mathbf{s}_t$ as the *system states*, $\mathbf{u}_t$ as the *control actions*, and y_t as the *system output*. For example, for the real-time control of a building system, $\mathbf{s}_t$ includes the room temperature, humidity, etc; $\mathbf{u}_t$ denotes the building appliance scheduling, room temperature set-points, etc; and output y_t is the building energy consumption. In addition, there maybe exogenous variables that

impact the output of the system, for example, outside temperature will impact the energy consumption of the building. However, since the exogenous variables are not impacted by any of the control actions we take, we suppress them in the formulation below. The time evolution of a system is described by

$$y_t = f(\mathbf{s}_t, \mathbf{u}_t)\,, \tag{4.4a}$$

$$\mathbf{s}_{t+1} = g(\mathbf{s}_t, \mathbf{u}_t) \tag{4.4b}$$

where (4.4b) describes the coupling between the current inputs to the future system states. Physical systems described by (4.4) may have significant inertia in the sense that the outcome of any control actions is delayed in time and there are significant couplings across time periods.

Since we use ICRNNs to represent both the system dynamics $g(\cdot)$ and the output $f(\cdot)$, the control variable $\mathbf{u}$ expands as $\hat{\mathbf{u}}$. The optimal receding horizon control problem at time t can be written as,

$$\underset{\mathbf{u}_t, \mathbf{u}_{t+1}, ..., \mathbf{u}_{t+T}}{\text{minimize}} \quad C(\hat{\mathbf{x}}, \mathbf{y}) = \sum_{\tau=t}^{t+T} J(\hat{\mathbf{x}}_\tau, y_\tau) \tag{4.5a}$$

$$\text{subject to} \quad y_\tau = f(\hat{\mathbf{x}}_{\tau-n_w}, \hat{\mathbf{x}}_{\tau-n_w+1}, ..., \hat{\mathbf{x}}_\tau), \forall \tau \in [t, t+T] \tag{4.5b}$$

$$\mathbf{s}_\tau = g(\hat{\mathbf{x}}_{\tau-n_w}, \hat{\mathbf{x}}_{\tau-n_w+1}, ..., \hat{\mathbf{x}}_{\tau-1}, \hat{\mathbf{u}}_\tau)\,, \forall \tau \in [t, t+T] \tag{4.5c}$$

$$\hat{\mathbf{x}}_\tau = \begin{bmatrix} \mathbf{s}_\tau \\ \hat{\mathbf{u}}_\tau \end{bmatrix},\ \hat{\mathbf{u}}_\tau = \begin{bmatrix} \mathbf{u}_\tau \\ \mathbf{v}_\tau \end{bmatrix}, \forall \tau \in [t, t+T] \tag{4.5d}$$

$$\mathbf{v}_\tau = -\mathbf{u}_\tau, \forall \tau \in [t, t+T] \tag{4.5e}$$

$$\mathbf{s}_\tau \in \mathcal{S}_{feasible}, \forall \tau \in [t, t+T] \tag{4.5f}$$

$$\mathbf{u}_\tau \in \mathcal{U}_{feasible}, \forall \tau \in [t, t+T] \tag{4.5g}$$

where a new variable $\hat{\mathbf{x}} = [\mathbf{s}_t, \hat{\mathbf{u}}_t]$ is introduced for notational simplicity, which called *system inputs*. It is the collection of system states $\mathbf{s}_t$ and duplicated control actions $\mathbf{u}_t$ and $-\mathbf{u}_t$, therefore ensuring the mapping from $\mathbf{u}_t$ to any future states and outputs remains convex.

$J(\hat{\mathbf{x}}_\tau, y_\tau)$ is the control system cost incurs at time τ, that is a function of both the system inputs $\hat{\mathbf{x}}_\tau$ and output y_τ. The functions $f(\cdot)$ and $g(\cdot)$ in Eq. (4.5b)-(4.5c) are parameterized as ICRNNs, which represent the system dynamics from sequence of inputs $(\hat{\mathbf{x}}_{\tau-n_w}, \hat{\mathbf{x}}_{\tau-n_w+1}, ..., \hat{\mathbf{x}}_\tau)$ to the system output y_τ, and the dynamics from control actions to system states, respectively. n_w is the memory window length of the recurrent neural network. The equations (4.5d) and (4.5e) duplicate the input variables $\mathbf{u}$ and enforce the consistency condition between $\mathbf{u}$ and its negation $\mathbf{v}$. Lastly, (4.5f) and (4.5g) are the constraints on feasible system states and control actions respectively. Note that as a general formulation, we do not include the duplication tricks on state variables, so the dynamics fitted by (4.5b) and (4.5c) are non-decreasing over state space, which are not equivalent to those dynamics represented by linear systems. However, since we are not restricting the control space, and we have explicitly included multiple previous states in the system transition dynamics, so the non-decreasing constraint over state space should not restrict the representation capacity by much. In Section.4.3 we theoretically prove the representability of proposed networks.

Optimization problem in (4.5) is a convex optimization with respect to (w.r.t.) inputs $\mathbf{u} = [\mathbf{u}_t, ..., \mathbf{u}_{t+T}]$, provided the cost function $J(\hat{\mathbf{x}}_\tau, y_\tau) = J(\mathbf{s}_\tau, \hat{\mathbf{u}}_\tau, y_\tau)$ is convex w.r.t. $\hat{\mathbf{u}}_\tau$, and convex, nondecreasing w.r.t. $\mathbf{s}_\tau$ and y_τ. A problem is convex if and only if both the objective function and constraints are convex. In the above problem, $J(\mathbf{s}_\tau, \hat{\mathbf{u}}_\tau, y_\tau)$ is convex and nondecreasing w.r.t. $\mathbf{s}_\tau$ and y_τ; $\mathbf{s}_\tau$ and y_τ are parameterized as ICRNNs, i.e., (4.5a) and (4.5b), such that they are convex w.r.t. $\hat{\mathbf{u}}_\tau$. Therefore following the composition rule of convex functions, the objective function is convex w.r.t. inputs $\mathbf{u} = [\mathbf{u}_t, ..., \mathbf{u}_{t+T}]$. Besides, all the equality constraints (4.5d) and (4.5e) are affine. Suppose both the state feasibile set (4.5f) and action feasibile set (4.5g) are convex, the overall optimization is convex.

The convexity of the problem in (4.5) guarantees that it can be solved efficiently and optimally using gradient descend method. Since both the objective function (4.5a) and the constraints (4.5b)-(4.5c) are parameterized as neural networks, and their gradients can be calculated via back-propagation with the modification where cost is propagated to the

input rather than the weights of the network. For implementation, the gradients can be convinently calculated via existing modules such as Tensorflow viaback-propagation. Let $\mathbf{u}^* = \{\mathbf{u}_t^*, \mathbf{u}_{t+1}^*, ..., \mathbf{u}_{t+T}^*\}$ be the optimal solution of the optimization problem at time t. Then the first element of $\mathbf{u}^*$ is implemented to the real-time system control, that is $\mathbf{u}_t^*$. The optimization problem is repeated at time $t+1$, based on the updated state prediction using $\mathbf{u}_t^*$, yielding a model predictive control strategy.

4.3 Input Convex Neural Networks (ICNN)

4.3.1 Representation Power of ICNN

Definition 1. *Given a function $f : \mathbb{R}^d \to \mathbb{R}$, we say that the function $\hat{f}$ approximate f within ϵ if $|f(\mathbf{x}) - \hat{f}(\mathbf{x})| \leq \epsilon$ for all $\mathbf{x}$ in the domain of f.*

Theorem 5. *[Representation power of ICNN] For any Lipschitz convex function over a compact domain, there exists a neural network with nonnegative weights and ReLU activation functions that approximates it within ϵ.*

Lemma 2. *Given a continuous Lipschitz convex function $f : \mathbb{R}^d \to \mathbb{R}$ with compact domain and $\epsilon > 0$, it can be approximated within ϵ by maximum of a finite number of affine functions. That is, there exists $\hat{f}(\mathbf{x}) = \max_{i=1,...,N}\{{\mu_\mathbf{i}}^T\mathbf{x} + b_i\}$ such that $|f(\mathbf{x}) - \hat{f}(\mathbf{x})| \leq \epsilon$ for all $\mathbf{x} \in domf$.*

Sketch of proof for Theorem 5. Supposing Lemma 2 is true, the proof of Theorem 5 boils down to showing that neural network with nonnegative weights and ReLU activation functions can exactly represent a maximum of affine functions. The proof is constructive. We first construct a neural network with ReLU activation functions and both positive and negative weights, then we show that the weights between different layers of the network can be restricted to be nonnegative by a simple duplication trick. Specifically, since the weights in the input layer and passthrough layers in the ICNN can be negative, we simply add a

negation of each input variable (e.g. both $\mathbf{x}$ and $-\mathbf{x}$ are given as inputs) to the network. These variables need satisfy a consistency constraint since one is the negation of the other. Since this constraint is linear, it preserves the convexity of optimization problems. The details of the proofs are given in the Appendix C. □

Similar to Theorem 1, an analogous result about the representation power of ICRNN can be shown for systems with convex dynamics. Given a dynamical system described by rolled out system dynamics $y_t = f(\mathbf{x}_1, \ldots, \mathbf{x}_t)$ is convex, then there exists a recurrent neural network with nonnegative weights and ReLU activation functions that approximates it within ϵ. A broad range of systems can be captured by this model. For example, the linear quadratic (Gaussian) regulator problem can be described using a ICRNN if we identify y as the cost of the regulator [80, 81].[1] An example of a nonlinear system is the control of electrochemical batteries. It can be shown from first principles that the degradation of these types of batteries is convex in their charge and discharge actions [82] and our framework offers a powerful data-driven way to control batteries found in electric vehicles, cell phones, and power systems.

4.3.2 Representation Efficiency of ICNN

In the proof of Theorem 5, we first approximate a convex function by a maximum of affine functions then construct a neural network according to this maximum. Then a natural question is why learn a neural network and not directly the affine functions in the maximum? This approach was taken in [79], where a convex piecewise-linear function (max of affine functions) are directly learned from data through a regression problem.

A key reason that we propose to use ICNN (or ICRNN) to fit a function rather than directly finding a maximum of affine functions is that the former is a much more efficient parameterization than the latter. As stated in Theorem 6, a maximum of K affine functions

[1] It's important to note that y is usually used as the system output of a linear system, but in our context, we are using it to refer to the quadratic cost with respect to the system states and the control input.

can be represented by an ICNN with K layers, where each layer only requires a single ReLU activation function. However, given a single layer ICNN with K ReLU activation functions, it may take a maximum of 2^K affine functions to represent it exactly. Therefore in practice, it would be much easier to train a good ICNN than finding a good set of affine functions.

Theorem 6. *[Efficiency of Representation]*

1. *Let $f_{ICNN} : \mathbb{R}^d \to \mathbb{R}$ be an input convex neural network with K ReLU activation functions. Then $\Omega(2^K)$ functions are required to represent f_{ICNN} using a max of affine functions.*
2. *Let $f_{CPL} : \mathbb{R}^d \to \mathbb{R}$ be a max of K affine functions. Then $O(K)$ activation functions are sufficient to represent f_{CPL} exactly with an ICNN.*

The detailed proof of Theorem 6 is given in Appendix C.

In the following two sections, we demonstrate the effectiveness of ICNN and ICRNN by presenting experimental results on two decision-making problems: energy management of large-scale commercial buildings [83] and the distribution network voltage control, respectively. The proposed method can be used as a flexible building block in decision making problems, where we use we use ICRNN to model the relationship between temperature setpoints and building energy consumption, and ICNN to represent the relationship between the active and reactive power injections to nodal voltage magnitude deviation. Both examples demonstrate that proposed method: 1) discovers the connection between controllable variables and the system dynamics or cost objectives; 2) is lightweight and sample-efficient; 3) achieves generalizable and more stable control performances compared with previous model-based reinforcement learning and simplified linear control approaches.

4.4 Application I: Building Energy Management

We first consider the real-time control problem of building's HVAC (heating, ventilation, and air conditioning) system to reduce its energy consumption. Building energy management remains to be a hard problem in control area. The exact system dynamics are unknown and hard to model due to the complex heating transfer dynamics, time-varying environments and the scale of the system in terms of states and actions [84].

At time t, we assume the building's running profile $\mathbf{x}_t := [\mathbf{s}_t, \mathbf{u}_t]$ is available, where $\mathbf{s}_t$ denotes building system states, including outside temperature, room temperature measurements, zone occupancies and etc. $\mathbf{u}_t$ denotes a collection of control actions such as room temperature set points and appliance schedule. Output is the electricity consumption P_t. This is a model predictive control problem in the sense that we want to find the best control inputs that minimize the overall energy consumption of building by looking ahead several time steps. To achieve this goal, we firstly learn an ICRNN model $f(\cdot)$ of the building dynamics, which is trained to minimize the error between P_t and $f(\mathbf{x}_{t-n_w}, ..., \mathbf{x}_t)$, while n_w denotes the memory window of recurrent neural networks. Then we solve:

$$\begin{aligned}
\underset{\mathbf{u}_t,...,\mathbf{u}_{t+T}}{\text{minimize}} \quad & \sum_{\tau=t}^{t+T} P_t && \text{(4.6a)}\\
\text{subject to} \quad & P_\tau = f_{ICRNN}(\mathbf{x}_{\tau-n_w}, ..., \mathbf{x}_\tau), \forall \tau \in [t, t+T] && \text{(4.6b)}\\
& \mathbf{s}_\tau = g_{ICRNN}(\mathbf{x}_{\tau-n_w}, ..., \mathbf{x}_{\tau-1}, \mathbf{u}_\tau), \forall \tau \in [t, t+T] && \text{(4.6c)}\\
& \underline{\mathbf{u}}_\tau \leq \mathbf{u}_\tau \leq \overline{\mathbf{u}}_\tau, \forall \tau \in [t, t+T] && \text{(4.6d)}\\
& \underline{\mathbf{s}}_\tau \leq \mathbf{s}_\tau \leq \overline{\mathbf{s}}_\tau, \forall \tau \in [t, t+T] && \text{(4.6e)}
\end{aligned}$$

where the objective (4.6a) is minimizing the total energy consumption in future T steps (T is the model predictive control horizon), and (4.6b) is the building energy consumption model, and (4.6c) is used for modeling building states, where both $f_{ICRNN}(\cdot)$ and $g_{ICRNN}(\cdot)$ are

parameterized as ICRNNs. Note that the formulation (4.6) is also flexible with different loss functions. For instance, in practice, we could reuse trained dynamics model (4.6b), and integrate electricity prices into the overall objective so that we could directly learn real-time actions to minimize electricity bills (please refer to Appendix C for more results). The constraints on control actions $\mathbf{u}_t$ and system states $\mathbf{s}_t$ are given in (4.6d) and (4.6e). For instance, the temperature set points as well as real measurements should not exceed user-defined comfort regions.

4.4.1 Experiment setup

To test the performance of the proposed method, we set up a 12-story large office building, which is a reference EnergyPlus commercial building model from US Department of Energy (DoE), with a total floor area of $498,584$ square feet which is divided into 16 separate zones. By using the whole year's weather profile, we simulate the building running through the year and record ($\mathbf{x}_t$, P_t) with a resolution of 10 minutes. We use 10 months' data to train the ICRNN and subsequent 2 months' data for testing. We use 39 building system state variables $\mathbf{s}_t$ (uncontrollable), along with 16 control variables $\mathbf{u}_t$. Output is a single value of building energy consumption at each time step. We set the model predictive control horizon $T = 36$ (six hours). We employ an ICRNN with recurrent layer of dimension 200 to fit the building input-output dynamics $f(\cdot)$. The model is trained to minimize the MSE between its predictions and the actual building energy consumption using stochastic gradient descent. We use the same network structure and training scheme to fit state transition dynamics $g(\cdot)$.

We set the model-based forecasting and optimization benchmark using an linear resistor-circuit (RC) circuit model to represent the heat transfer in building systems, and solve for the optimal control actions via MPC [67]. At each step, MPC algorithm takes into account the forecasted states of the building based on the fitted RC model and implements the current step control actions. We also compare the performance of ICRNN against the conventionally

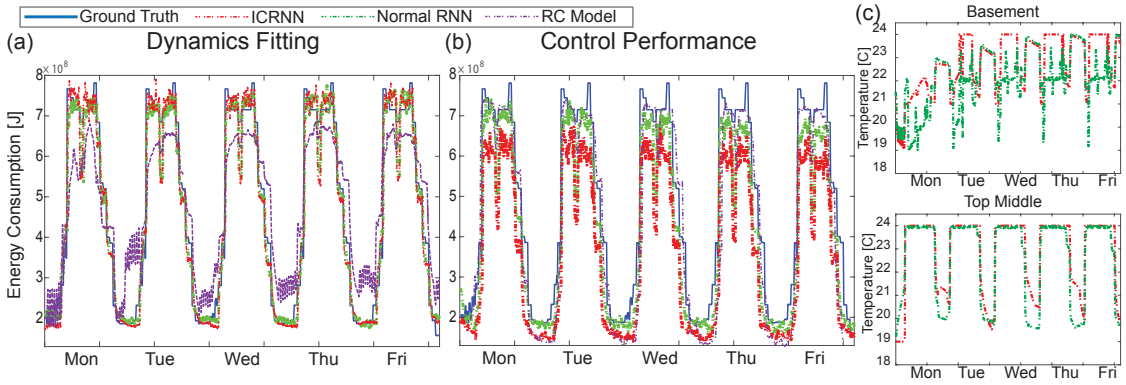


Figure 4.3: Results for constrained optimization of building energy management. (a) ICRNN is able to model the building dynamics as accurately as conventional RNN; (b) Compared to conventional RNN model, ICRNN finds control actions which lead to 11.52% more of energy savings, and (c) ICRNN provides stable control actions while decisions generated by conventional RNN vary dramatically.

trained RNN in terms of building dynamics fitting performance and control performance. To solve the MPC problem with conventional RNN models, we also use gradient-based method with respect to controls. However, since conventional RNN models are generally not convex from input to output, there is no guarantee to reach a global optimum (or even a local one).

In terms of the fitting performance, ICRNN provides a competitive result compared to conventional RNN model. The overall test root mean square error (RMSE) is 0.054 for ICRNN and 0.051 for conventional RNN, both of which are much smaller than the error made by RC model (0.240). Fig. 4.3(a) shows the fitting performance on 5 working days in test data. This illustrates the good performance of ICRNN in modeling building HVAC system dynamics. Then by using the learned ICRNN model of building dynamics, we obtain the suggested room control actions u_t^* by solving the optimal building control problem (4.6). As shown in Fig. 4.3(b), with the same constraints on building temperature interval of $[19°C, 24°C]$, the building energy consumption is reduced by 23.25% after implementing

the new temperature set points calculated by ICRNN. On the contrary, since there is no guarantee for finding optimal control actions by optimizing over conventional RNN's input, the control solutions given by conventional RNN could only reduce 11.73% of electricity. Solutions given by RC model only saves 4.07% of electricity. More importantly, in Fig. 4.3(c) we demonstrate the control actions outputted by our method against MPC with conventional RNN in two randomly selected building zones, the building basement and top floor central area. It shows that our proposed approach is able to find a group of stable control actions for the building system control. While in the conventional RNN case, it generates control set points which have undesirable, drastic variations.

4.5 Application II: Distributed Energy System Voltage Regulation

We now move on to optimally control the voltage level in distribution networks with input convex neural networks. Voltage regulation in distribution networks has played an important role to maintain acceptable voltage magnitudes at all buses. The higher penetration of distributed energy resources (DERs), for example rooftop PV and electric vehicles, could lead to fast voltage fluctuations in distribution networks [85]. To complement slow time-scale control of discrete devices such as tap-changing transformers and switched capacitors, reactive power injections via the inverter-based distributed resources are often proposed for fast time-scale voltage regulations [86].

Research efforts on inverter-based voltage regulations have focused on approaching the control problem through an optimization framework [87]. Consider a power distribution network consisting of a set $\mathcal{N} = \{1, ..., N\}$ of buses and a set $\mathcal{E} \in \mathcal{N} \times \mathcal{N}$ of distribution lines connecting buses. For each bus i, denote V_i as the voltage magnitude and θ_i as the voltage phase angle; let p_i and q_i denote the active and reactive power injections; let $s_i = p_i + jq_i$ be the complex power injection. The corresponding active and reactive power injection vectors are denoted as $\mathbf{p} = \begin{bmatrix} p_1 & p_2 & \cdots & p_N \end{bmatrix}^T, \mathbf{q} = \begin{bmatrix} q_1 & q_2 & \cdots & q_N \end{bmatrix}^T$. For each line $(i, k) \in \mathcal{E}$,

denote line admittance $y_{ik} = g_{ik} - jb_{ik}$ with $b_{ik}, g_{ik} > 0$ as the real and imaginary parts of the admittance matrix element Y_{ik}. For each bus $i \in \mathcal{N}$, its power injection is governed by

$$p_i = \sum_{k=1}^{N} V_i V_k(-g_{ik}\cos(\theta_i - \theta_k) + b_{ik}\sin(\theta_i - \theta_k)) \tag{4.7a}$$

$$q_i = \sum_{k=1}^{N} V_i V_k(g_{ik}\sin(\theta_i - \theta_k) + b_{ik}\cos(\theta_i - \theta_k)). \tag{4.7b}$$

The goal is to address the voltage fluctuations due to higher penetration level of DERs in distribution networks. By making use of the power electronics interfaces of DERs such as PV inverters, the reactive power injections q_i can be controlled within certain limits. By changing reactive power injections, the goal is to maintain voltage magnitude V_i within a small distance from the nominal value $V_{i,0}$ for all buses (e.g., plus/minus 5%). Formally, we can cast voltage regulation as the following optimization problem:

$$\min_{\mathbf{q}} \quad \sum_{i=1}^{N} \alpha_i |V_i - V_{i,0}| \tag{4.8a}$$

$$\text{s.t.} \quad \underline{\mathbf{q}} \leq \mathbf{q} \leq \overline{\mathbf{q}} \tag{4.8b}$$

$$\text{Power Flow Equations (4.7)} \tag{4.8c}$$

where α_i is a weighted parameter which can be adjusted by the system operator. The constraints in (4.8b) capture the hard constraints on available reactive power injections on each bus i. The constraints in (4.8c) capture the power flow models. The active power $\mathbf{p}$ is considered as an exogenous input vector which is not controlled.

Solving (4.8) is not trivial, because the problem is not convex due to the nonlinear relationship between bus voltage magnitudes and powers. There has been a rich body of literatures about reformulation of (4.8) as a convex optimization problem. For each line $(i, k) \in \mathcal{E}$, denote $s_{ik} = p_{ik} + jq_{ik}$ as the complex power flow and $z_{ik} = r_{ik} + jx_{ik}$ as the line impedance. The

DistFlow equations [88] model the distribution network flow as

$$-p_k = p_{ik} - r_{ik} l_{ik} - \sum_{l:(k,l)\in\mathcal{E}} p_{kl} \tag{4.9a}$$

$$-q_k = p_{ik} - x_{ik} l_{ik} - \sum_{l:(k,l)\in\mathcal{E}} q_{kl} \tag{4.9b}$$

$$V_k^2 = V_i^2 - 2(r_{ik}p_{ik} + x_{ik}q_{ik}) + \left(r_{ik}^2 + x_{ik}^2\right) l_{ik} \tag{4.9c}$$

$$l_{ik} = \frac{p_{ik}^2 + q_{ik}^2}{V_i^2}. \tag{4.9d}$$

By further making the following relaxation for (4.9d)

$$l_{ik} \geq \frac{p_{ik}^2 + q_{ik}^2}{V_i^2} \tag{4.10}$$

which can be written as a second-order cone constraint, the relaxed constratins together with the voltage regulation objective is then a Second Order Cone Program (SOCP) [89].

Remark 7. *The relaxed voltage control problem with regulation objective* (4.8a)*, reactive power injection constraint* (4.8b)*, power flow constraints* (4.9a)-(4.9c) *and* (4.10) *is convex. Under many circumstances, this relaxation is tight, see [89, 90].*

In order to further simplify the analysis of the original voltage control problem (4.8), many linearized power flow models are adopted, while Simplified Distflow model is widely used [87, 91], which sets l_{ik} to be zeros, and approximates $V_i^2 - V_k^2$ by $2(V_i - V_k)$:

$$-p_k = p_{ik} - \sum_{l:(k,l)\in\mathcal{E}} p_{kl} \tag{4.11a}$$

$$-q_k = q_{ik} - \sum_{l:(k,l)\in\mathcal{E}} q_{kl} \tag{4.11b}$$

$$V_i - V_k = r_{ik}p_{ik} + x_{ik}q_{ik}. \tag{4.11c}$$

Similarly, by replacing (4.8c) with the linearized version (4.11), we are also able to solve the voltage regulation as a convex optimization problem. However, considering the increasing variability of load and generation in distribution networks, voltage regulation based on

linearized approximation model (4.11) may not be accurate enough to represent the true distribution network models, while the resulting control signals of reactive power injections may not be optimal when applied in the real distribution networks. In summary, to solve voltage regulation as a convex optimization, all these aforementioned approaches require the exact information on line parameters (e.g., line impedances) and network topology. Unfortunately, due to the lack of observability of distribution systems, directly learning the topology is hard without PMU data [92].

Given the practical challenges of voltage regulation, we want to design an optimal controller that satisfies following requirements:

- The controller must learn an accurate representation of the power injections to nodal voltage magnitudes;
- Such representation is easy to be integrated into the optimization framework.

Intuitively, we are trying to design and find functions $|V_i - V_{i,0}| = f_i(\mathbf{p}, \mathbf{q}), i = 1, ..., N$, which could accurately represent the relationship from active and reactive power injections to nodal voltage magnitude deviation. By leveraging historical smart meter data to fit f_i, we want to see if the fitted model could represent the underlying grid. More importantly, if f_i is a convex function from $\mathbf{p}, \mathbf{q}$ to $|V_i - V_{i,0}|$, then the following problem

$$\min_{\mathbf{q}} \quad \sum_{i=1}^{N} \alpha_i |V_i - V_{i,0}| \tag{4.12a}$$

$$\text{s.t.} \quad \underline{\mathbf{q}} \leq \mathbf{q} \leq \overline{\mathbf{q}} \tag{4.12b}$$

$$|V_i - V_{i,0}| = f_i(\mathbf{p}, \mathbf{q}) \tag{4.12c}$$

is still a tractable convex optimization problem. Note that we integrate voltage magnitude deviations constraint (4.12c) into the voltage regulation framework, which is a general formulation to make sure once f_i is convex, (4.12) is a convex optimization problem. Such

formulation is comparable to previous formulations by either treating voltage magnitude deviations as the optimization objective [90] or as box constraints [93].

When the underlying topology and the line parameters are unknown, we propose to first learn a convex mapping from $\{\mathbf{p}, \mathbf{q}\}$ to voltage magnitude deviations using an ICNN. Once fitted using collected observations, we are able to use the same ICNN, and integrate it to (4.12) to find optimal reactive power injections. In order to train ICNN and learn its parameters h_θ, we need to minimize the supervised traning loss. For the kth training instance, it is defined as the mean square error between the ground truth voltage magnitude deviation vector $\mathbf{V}_{target} := \{|V_i^k - V_{i,0}^k|\}, i = 1, ..., N$ and the ICNN output:

$$L(\mathbf{V}_{target}, h_\theta(\mathbf{p}, \mathbf{q})) = \frac{1}{N}||\mathbf{V}_{target} - h_\theta(\mathbf{p}, \mathbf{q})||_2^2, \tag{4.13}$$

and the update of h_θ is based on gradient descent algorithm. In addition, to take the constraints of $W_{2:m} \geq 0$ into account, we need to make sure the gradient descent update always falls into the feasible regions (e.g., nonnegative weights). Hence we use a projected gradient algorithm to guarantee the constraint holds [47].

Definition 2. *The projection of a point y, onto a set X is defined as*

$$\Pi_X(y) = \underset{x \in X}{\operatorname{argmin}} \frac{1}{2}\|x - y\|_2^2 \tag{4.14}$$

Given a starting point $x^{(0)} \in X$ and step-size $\gamma > 0$, projected gradient descent (PGD) extends the standard gradient descent settings with the projection step onto the feasible sets of feasible reactive power. At iteration t, the algorithm takes the following PGD step:

$$x^{(t+1)} = x^{(t)} - \gamma \Pi_X\Big(x^{(t)} - \nabla f\Big(x^{(t)}\Big)\Big), \forall t \geq 1 \tag{4.15}$$

which is implemented iteratively until a certain stoppping criterion (e.g., fixed number of iterations or gradient value is smaller than predefined ϵ) is satisfied. The ICNN weights are then updated as follows,

$$h_\theta = h_\theta - \gamma \Pi_{W_{2:m} \geq 0}(h_\theta - \nabla_{h_\theta}(L(\mathbf{V}_{target}), h_\theta(\mathbf{p}, \mathbf{q}))). \tag{4.16}$$

In practical implementations where there are large groups of measurements $\{\mathbf{p}^k, \mathbf{q}^k, \mathbf{V}^k\}$ with k standing for the index for measurement index, it is possible to use small batch of training data to do PGD steps (4.16). Such practical algorithms, e.g., stochastic gradient descent (SGD), can accelerate training convergence [94]. The training procedure is summarized in Algorithm 4 by using collected training data and SGD training algorithm.

Once the ICNN training process is finished, we fix model parameters h_θ, and use it as a proxy model for the unknown distribution networks model $f_i,\ i = 1, ..., N$ in (4.12c). Since h_θ represents the convex mappings from $\mathbf{q}$ to $|V_i - V_{i,0}|,\ \forall i$, we are now ready to solve (4.12) computationally. In the similar spirit of ICNN training, where we optimize over network weights using gradient descent to minimize training loss, in the voltage regulation setting, we optimize over ICNN inputs $\mathbf{q}$ to minimize the optimization objective $\sum_{i=1}^{N} \alpha_i |V_i - V_{i,0}|$ using gradient descent. Again, to take the constraints of reactive power injection range into account, we need to make sure that the gradient descent update always falls into the feasible reactive power injection regions. By adapting PGD to the trained ICNN, starting from uncontrolled reactive power $\mathbf{q}^{(0)} = \mathbf{q}$, we take iterative PGD steps on the voltage regulation objective (4.12a) until gradient convergence. PGD steps also guarantee the convergence to optimal solution under the convex settings. The overall algorithm for ICNN training and finding optimal reactive power injections are described in Algorithm 4.

4.5.1 Experiment Setup

In this section, we evaluate the proposed data-driven voltage regulation algorithm in two standard IEEE test distribution networks, that is IEEE 13-bus and IEEE 123-bus systems. Linear models, standard neural networks and the optimal SOCP formulations are used for comparison. For both 13-bus and 123-bus system, we use AC power flow model (4.7) to generate $10,000$ instances of simulation data composed of $\{\mathbf{p}, \mathbf{q}, \mathbf{V}\}$. We assume both the distribution network topology and line parameters are not revealed to the optimization

Algorithm 4: ICNN for Voltage Regulation

Data: Learning rate η, Step size γ, Batch size T, Training iterations $n_{training}$, Optimization stopping critertion ϵ. Training dataset $\{\mathbf{p}, \mathbf{q}, \mathbf{V}\}$. Initial model h_θ.

Result: Optimal reactive power injection: $\mathbf{q}^* = \mathbf{q}^{(t)}$

\# ICNN Training;

for *iter* $\leftarrow 0$ **to** n **do**

  # Update parameters for ICNN;

  Sample batch from historical data:;

  $\{\mathbf{p}^k, \mathbf{q}^k, \mathbf{V}^k\}_{k=1}^T \sim \mathbb{P}_x$;

  Update h_θ using stochastic gradient descent:;

  $\mathbf{V}_{target}^k = \{|V_i^k - V_{i,0}^k|\}, i = 1, ..., N$;

  $h_\theta = h_\theta - \eta \Pi_{W_{2:m} \geq 0}(h_\theta - \nabla_{h_\theta}(L(\mathbf{V}_{target}), h_\theta(\mathbf{p}, \mathbf{q})))$;

end

Fix ICNN parameters h_θ;

 # Voltage Regulation via ICNN;

Get measurements $\{\mathbf{p}, \mathbf{q}\}$, $t = 0$, $\mathbf{q}^{(t)} = \mathbf{q}$;

while $\nabla_{\mathbf{q}^{(t)}}(\sum_{i=1}^N h_\theta(\mathbf{p}, \mathbf{q}^{(t)})) > \epsilon$ **do**

  $\mathbf{q}^{(t+1)} = \mathbf{q}^{(t)} - \gamma \Pi_{\mathbf{q}}(\mathbf{q}^{(t)} - \nabla_{\mathbf{q}^{(t)}}(\sum_{i=1}^N h_\theta(\mathbf{p}, \mathbf{q}^{(t)})))$;

  $t \leftarrow t + 1$;

end

Optimal reactive power injection: $\mathbf{q}^* = \mathbf{q}^{(t)}$;

algorithm, except when the optimal SOCP is used as a baseline. We allow plus/minus 20% of reactive power injections at each node as control inputs. We develop three algorithms and compare their performances for two test feeders shown in Fig. 4.4.

- *Linear Model:* We consider using a linear model to fit the unknown dynamics from active and reactive power to the deviations between nodal voltage and the nominal

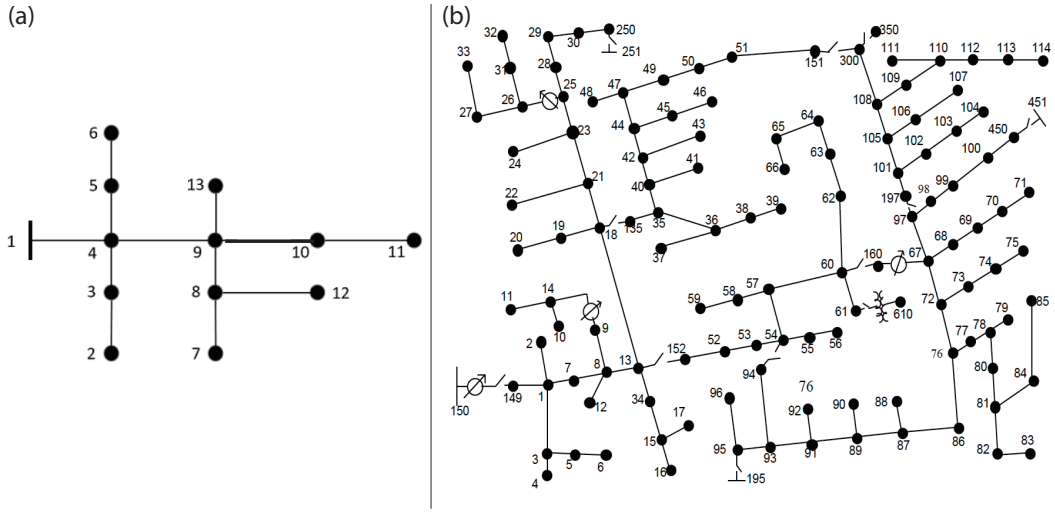


Figure 4.4: Schematic diagram of (a). IEEE 13-bus test feeder and (b). IEEE 123-bus test feeder. Reference buses: 1 and 149.

voltage. Such linearized models have been widely used in power systems literature [87];

- *Neural Networks Model:* We construct standard three-layer and four-layer neural networks for the 13-bus and 123-bus cases, respectively. We tune the parameters of neural networks (e.g., number of neurons, learning rate) and stop the training process once the fitting performance on validation data converges;

- *Input Convex Neural Networks:* We keep the number of layers and matrices the same dimension as those of neural networks models, but add direct layers $D_i, i = 2, ..., k$ correspondingly. We constrain network weights $W_{2:k}$ to be non-negative during training.

To fit the parameters of neural network models, we use mean squared error as the loss function during training. To solve the voltage regualtion problem (4.8), we set $\alpha_i, i = 1, ..., k$ in (4.12) to be 1 in our simulation cases. When α is not equal to 1, we could adapt the optimization problem using weighted sum of voltage deviation correspondingly. Note that we could also flexibly add reactive power costs to the objective function (4.8a), as long as they

Simulation Network	IEEE 13-Bus			
Model	SOCP	Linear	NN	ICNN
Model Fitting MAE	-	9.93%	3.45%	3.86%
Regulated voltage out of 3% tolerance	3.46%	8.65%	7.88%	4.71%
Regulated voltage out of 5% tolerance	0.47%	7.89%	6.86%	1.05%
Computation Time (per instance/s)	0.9684	0.2022	0.3137	0.2512
SImulation Network	IEEE 123-Bus			
Model	SOCP	Linear	NN	ICNN
Model Fitting MAE	-	12.98%	3.56%	4.25%
Regulated voltage out of 3% tolerance		21.46%	14.04%	7.51%
Regulated voltage out of 5% tolerance		19.19%	9.65%	1.64%
Computation Time (per instance/s)		0.2712	0.6297	0.4302

Table 4.1: Comparison between SOCP, Linear model, Neural Networks model, and Input Convex Neural Networks model for IEEE 13-bus and IEEE 123-bus systems.

are convex functions over reactive power injections. All the implementations are conducted on a MacBook Pro with 2.4GHz Intel Quad Core i5.

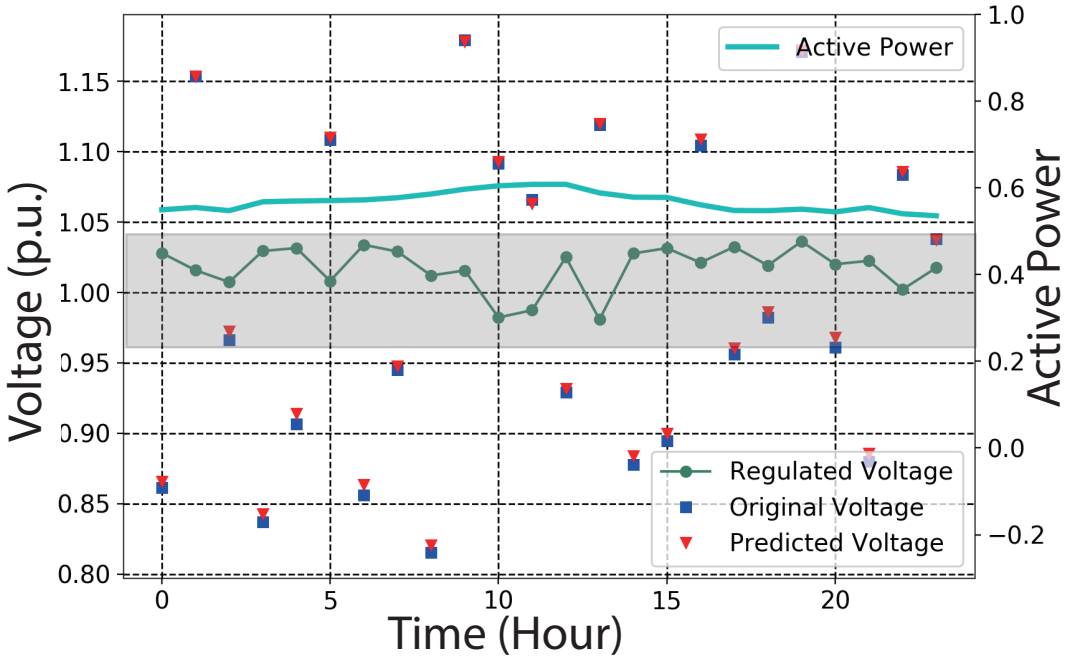


Figure 4.5: Example of voltage regulation over a daily variation for the 13-bus test feeder. The voltage of bus 4 is shown. With ICNN accurately predicting voltages (red triangle), it could regulate voltage within 4% of nominal values (grey box) under varying load level throughout the day.

To benchmark the performance of the proposed algorithms under unknown topology and parameters, we also follow [89] to relax $l_{ij} \geq \frac{P_{ij}^2+Q_{ij}^2}{V_i^2}$ in the Dist-flow equations, and use the same validation datasets to solve the resulting convex SOCP. We calculate the optimal reactive power injections along with the resulting voltage profiles. We use CVX to solve the SOCP and linearized models, and use Tensorflow to set up and solve NN and ICNN models.

We firstly validate that ICNN can be used as a proxy for power flow equations, and predict the nodal voltage magnitude deviations. By using $8,000$ training instances, the ICNN can predict the voltage deviations on the validation instances accurately. As shown in Table 4.1, the mean absolute error (MAE) of ICNN fitting are smaller than 4.3% in both test systems, which are comparable to 3.45% and 3.56% by using neural networks. This is also illustrated in Fig. 4.5, where under different load levels throughout 24 hours, the ICNN can predict all

the nodal voltages accurately. More importantly, linear model's fitting performances are over 2 times worse than the neural networks counterparts. We later show such fitting errors would also impact the controller performances.

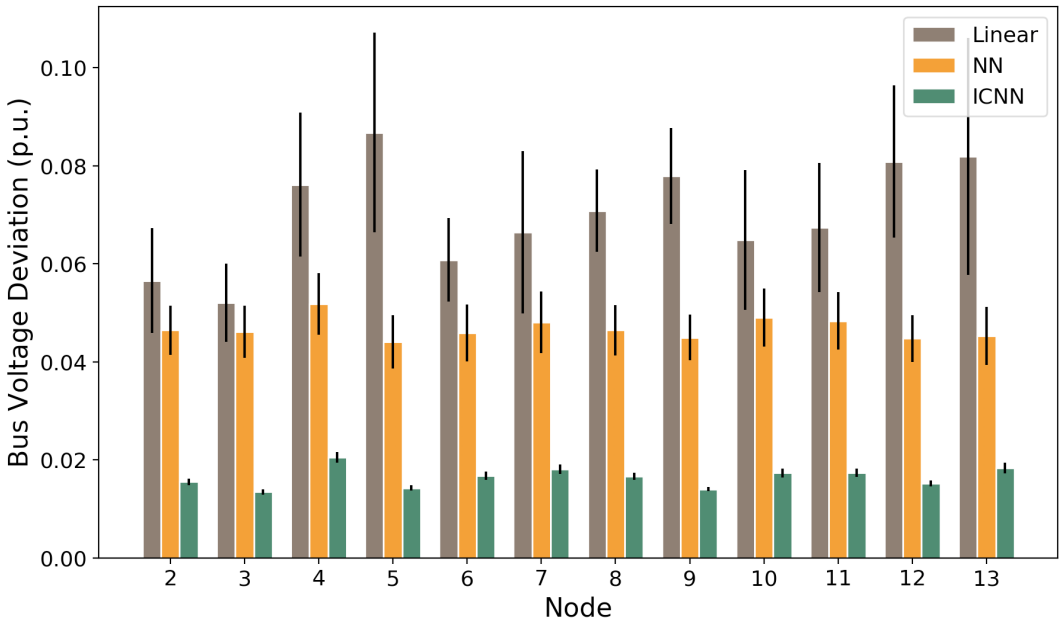


Figure 4.6: Comparisons on nodal voltage deviation of linear-fitted model, neural network model and input convex neuural network on IEEE 13-bus system. On average, the mean voltage deviation for ICNN is 4.3 times better than linear model, and 2.7 times better than standard NN model.

In Figure 4.5, we show the regulated voltage using ICNN in the IEEE 13-bus case. Under this day's load profile, we are able to regulate node 4's voltage magnitude within $\pm 4\%$ per unit with constrained reactive power injections (Equation 4.12b). In Figure 4.6, we show that the mean and variance on each bus's voltage deviations using three models for the 13-bus feeder. On the one hand, with similar fitting performances, ICNN outperforms the standard neural network in regulating nodal voltages. This is due to the fact that neural networks may have many local minima, and the NN-based controller can not find the optimal reactive power injections. On the other hand, even though linear model provides a easier

venue for solving optimization problem, it suffers from inaccurate modeling of the underlying distribution grids, and the regulated bus voltages have greater level of fluctuations. Similar observations also hold in the 123-bus test case, where in Fig. 4.7 we show the nodal voltage comparison using three models, and voltage regulated by ICNN are constrained to be in a much narrower range. More results on voltage regulation performances are summarized in Table 4.1. Under varying load and power generation profiles, ICNN is able to maintain over 98.3% of nodal voltages within 5% deviations from nominal voltages, which are comparable to SOCP solutions. On the contrary, linear fitted models can not scale to larger system, and nearly 20% of voltages are out of 5% tolerance in the 123-bus case. We note that with more injections of reactive power, the proposed control scheme can make use of such injections to achieve better regulation performance.

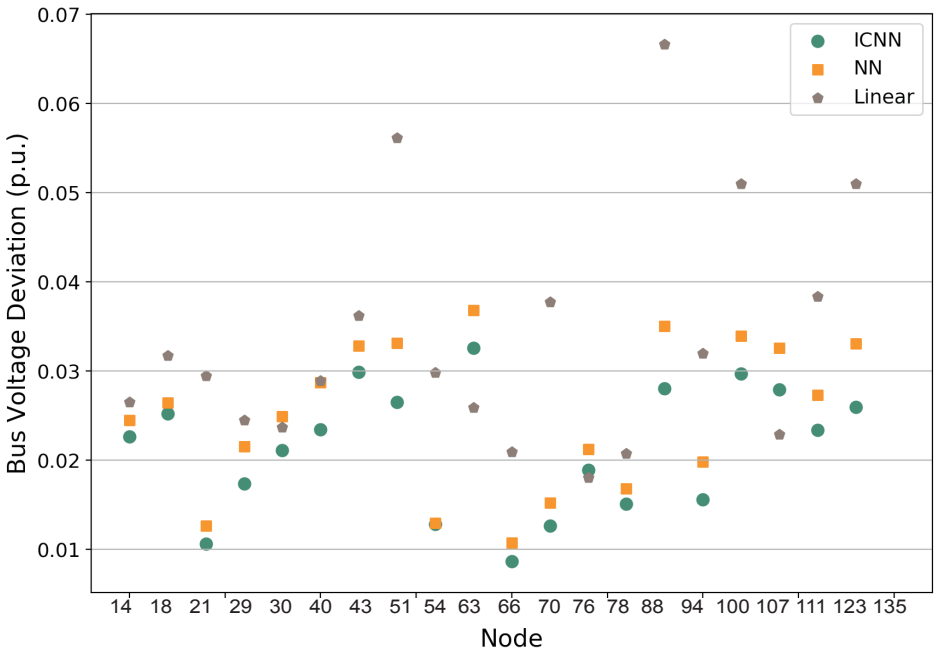


Figure 4.7: Comparisons on 20 randomly selected buses' nodal voltage deviation plots of linear-fitted model, neural network model and input convex neuural network model on IEEE 123-bus system.

We also give an analysis on the computation time for each algorithm. Compared to linear model, optimization based on ICNN generally takes longer time, but it is still able to find the optimal solutions within the acceptable time range. Note that we are solving the ICNN optimization problem using our own solver, while solving SOCP and linear model using the off-the-shelf CVX solver. More importantly, in the 13-bus case, ICNN-based optimization is faster than SOCP solver, and it scales to 123-bus case with moderate computation time increases, while SOCP solver is hard to scale to larger network. An interesting observation is that it takes longer for NN to find solutions compared to ICNN, partly due to the fact that gradient-based optimizer is stuck in some local minima. We discuss the performance of proposed method on distributed case in [8].

4.6 Conclusion

In this chapter, we propose a novel data-driven control framework that uses neural networks designed to be convex from the input to the output. We show that many interesting energy system control problems can be cast as convex optimization problems with the proposed network architecture. Experiments on the building energy management and distribution system voltage control demonstrate the effectiveness of our methodology. This framework bridges machine learning and closed-loop control by using ICNN to learn the unknown system dynamics, which obtain both good predictive accuracy and tractable computational complexity.

Chapter 5

LEARNING IN MULTIAGENT SYSTEM WITH LIMITED INFORMATION EXCHANGE: COURNOT COMPETITION IN ELECTRICITY MARKET

5.1 Introduction

In previous chapters, we consider control of a single energy system subject to uncertainties. But in real-world energy systems, there are millions of individuals with different roles and objectives. What will happen if each individual tries to learn and optimize their behaviors? How would their strategic interactions affect the entire system's stability and efficiency?

In this chapter, we move on to analyze the dynamics of multiagent systems, with limited information exchange. At each step, every agent chooses an action based on their local observations, while they're uncertain about the entire system state and other individuals' actions. We focus on a specific class of games where the agents undergo *Cournot competitions* [95]. Cournot competition is the essential market model for many socio-economic systems such as energy systems [96], transportation networks [97] and healthcare systems [98]. It can be thought as multiple agents competing to satisfy an elastic demand by changing their production levels. For example, most of the US electricity market is built upon a Cournot competition model [96].[1], where energy producers bid into the grid, and a market price is cleared based on the total supply and demand. Each producer's payoff is based on the market price multiplied by its share of the supply. The goal of producers is to maximize their

[1]In this a first-order approximation of the locational marginal pricing used by markets in the United States.

individual payoffs by strategically choosing the production levels.

5.1.1 Literature Review and Our Contributions

When learning is not needed, there is a wealth of results for the Cournot competition. For example, when each agent has full information about the game, including the price function and the cost function of other agents, there are many works characterizing the properties of the Nash equilibrium of the game [99–101]. However, when learning is involved and agents do not have full information, the properties of the game are not well understood. This is even the case in the simplest setting, where the agents only receive the price from the system as the feedback but do not know the price function form nor the actions of other agents.

To answer what happens when agents learn, we must model how they learn - or more precisely, what type of learning algorithms is used. A key technical challenge is that when learning is used, the Cournot game becomes stochastic. Currently, most works focus on no-regret algorithms [102, 103] because they only require a minimal set of assumptions on the game. In addition, the no-regret definition could be directly translated to the coarse correlated equilibrium condition [104] for a wide range of algorithms (e.g., multiplicative-weight [105], online mirror descent [106], Follow-the-Regularized-Leader [107]). However, while the theoretical properties of no-regret algorithms are attractive, they also limit the applicability of these algorithms. In practice, systems and agents are often *not adversarial* to each other, and the competition is often designed to have specific structures. In many games, it is more natural for players to use myopic policies such as reinforcement learning algorithms that directly aim for profit maximization. In addition, the notation of coarse correlated equilibrium can be quite weak, and sharper results are often desired.

Reinforcement learning algorithms can lead to much better performances than no-regret algorithms, but proving their convergence has proven to be challenging [108] since the coupling between the (continuous) actions of the players must be carefully analyzed. Attempts have

been made to discretize the space(e.g., Q-learning [109]) then studying the resulting discrete game, but the dimensionally quickly grows and important features (e.g., convexity) are hard to retain [110, 111].

In this chapter, we directly work with the continuous action and state space by considering agents use *policy gradient* algorithms. In particular, we assume the class of policies where the actions are parameterized by the mean of distributions (e.g., Gaussian policies). The major contribution of this work is in the following: *we prove that when the price function is linear or when there are two agents, there is a unique Nash equilibrium (NE) in the stochastic Cournot game, and the policy gradient dynamics converge exponentially quickly to the NE.* This is the first result (to the best of our knowledge) on the convergence property of algorithms with continuous action spaces that do not fall in the no-regret class.

5.2 Problem Formulation and Preliminaries

Definition 3 (Cournot Game)**.** *Consider N players produce homogeneous products in a limited market, where the action space of player i is its production level $x_i \geq 0$. The utility function of player i is denoted as $\pi_i(\boldsymbol{x}) = p(\sum_{j=1}^{N} x_j)x_i - C_i(x_i)$, where p is the market price (inverse demand) function that maps the total production quantity to a price in $\mathbb{R}$ and $C_i(\cdot)$ is the cost function of player i.*

The goal of each player i in the Cournot game is to choose the best production quantity x_i such that maximizes his utility π_i. An important concept in game theory is the *Nash equilibrium*, at which state no player can increase their payoffs by unilaterally changing their strategies. A Nash equilibrium of the Cournot game defined by $(\pi_1, ..., \pi_N)$ is a vector $\boldsymbol{x}^* \geq 0$ such that for all i:

$$\pi_i(x_i^*, \boldsymbol{x}_{-i}^*) \geq \pi_i(\tilde{x}_i, \boldsymbol{x}_{-i}^*), \quad \text{for all} \quad \tilde{x}_i, \tag{5.1}$$

where $\boldsymbol{x}_{-i}$ denotes the actions of all players except i. In this paper, we restrict our attention to Cournot games satisfying the following assumptions:

Assumption 1. *We assume the price function and cost functions:*

(A1) The price function p is concave, strictly decreasing and twice differentiable on $[0, y_{\max}]$, where $y_{\max}$ is the first point where p becomes 0. For $y > y_{\max}$, $p(y) = 0$. In addition, $p(0) > 0$.

(A2) The cost function $C_i(x_i)$ is convex, strictly increasing, twice differentiable and $p(0) > C_i'(0)$, for all i.

These assumptions are standard in the literature (e.g., see [112] and references within). The assumption $p(0) > C_i'(0)$ is to avoid the triviality of a player never participating in the game. The following proposition shows that Cournot game satisfying the above assumptions has an unique Nash equilibrium.

Proposition 3. *A Cournot game satisfying (A1) and (A2) has exactly one Nash equilibrium.*

Proof. Proof of Proposition 3 refers to Theorem 1 in [113]. □

We consider each agent adopt a *policy-based* method to learn and act. At each time step t, we assume that it has (possibly noisy) information of the system states $\mathbf{s_t}$. This agent maintains a policy $\pi_\theta(\cdot|\mathbf{s_t})$, which is a probability distribution on the action it would take, conditioned on the agent's information $\mathbf{s_t}$. At each time step, after the agent picks actions $\boldsymbol{a}_t \sim \pi_\theta(\cdot|\mathbf{s_t})$, the system releases reward. Subsequently, they update their policy parameters along the gradient direction of their long-term expected reward. Such learning procedure is called *policy gradient* method [114] in the literature. As a key permise for the idea, the policy long-term reward is,

$$J(\theta) = E_{\tau\sim p_\theta(\tau)}[\sum_t r(\mathbf{s}_t, \mathbf{a}_t)] \tag{5.2}$$

and the gradient is given by,

$$\nabla_\theta J(\theta) = E_{\tau\sim p_\theta(\tau)}[(\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(\boldsymbol{a}_t|\boldsymbol{s}_t))(\sum_{t=1}^{T} r(\boldsymbol{s}_t, \boldsymbol{a}_t))]\,, \tag{5.3}$$

where τ and $J(\theta)$ are trajectories and the expected trajectory return under policy π_θ, respectively, and $\nabla_\theta \log \pi_\theta(\boldsymbol{a}_t|\boldsymbol{s}_t))$ is the score function of the policy. Various of policy gradient methods have been proposed by estimating the gradient (5.3) in different ways, including REINFORCE [114] and natural policy gradient [115].

5.3 Stochastic Cournot Game

We discuss the main convergence results in this section. As briefly mentioned, we consider policy-based models of how agents choose and evolve their actions. In particular, as the agents only get the reward as feedback and nothing else, the policy gradient dynamics reduce to the *stateless* version. In particular, we consider a policy that is parameterized by the mean of a distribution. This model includes many popular algorithms, for example, the ubiquitous Gaussian policies and their extensions [116]. Let θ_i denote the *mean* of player i's action, and X_i to be a zero-mean random variable. For convenience, we assume it is continuous and has a bounded density function denoted by $f_i(X_i)$. We say X_i is unimodal at mean if f_i has a global maximum at the mean and no other isolated local maxima [2].

At each time step, player i choose the action to play as $a_i \sim \pi_{\theta_i}(\cdot) = \theta_i + X_i$. Note that in most Cournot games, the action is interpreted as quantity, that cannot be negative. Therefore, player i has to play by drawing a quantify from the rectified distribution $(\theta_i + X_i)^+$, where $a^+ = \max(a, 0)$. Under the Cournot game setup, the expected profit in Eq. (5.2) can be written out as the follows,

$$J_i(\theta_i; \boldsymbol{\theta}_{-i}) = E_{\boldsymbol{X}}\left[p\left(\sum_{j=1}^{N}(\theta_j + X_j)^+\right)(\theta_i + X_i)^+ - C((\theta_i + X_i)^+)\right]. \tag{5.4}$$

[2]For example, Gaussian and uniform distributions are unimodal under this definition.

and the gradient value in Eq. (5.3) equals,

$$\nabla_{\theta_i} J_i = E\left[1(\theta_i + X_i \geq 0)\left\{p'\left(\sum_{j=1}^{N}(\theta_j + X_j)^+\right)(\theta_i + X_i) + p\left(\sum_{j=1}^{N}(\theta_j + X_j)^+\right) - C_i'(\theta_1 + X_1)\right\}\right], \tag{5.5}$$

where $1(\cdot)$ is the indicator function.

We call the game associated with these J_i's the *stochastic Cournot game*, where player i chooses θ_i, observe the profit J_i, and update θ_i according to the payoff gradient. The Nash equilibrium of the stochastic Cournot game is defined as, $(\theta_1^*, ..., \theta_N^*)$ such that $\nabla_{\theta_i^*} J_i = 0, \forall i$. The form of (5.5) has made analyzing the system dynamics difficult compared to standard Cournot games. Firstly, because the actions are rectified, the profit of player i not long just depends on the sum of the other players (as in a Cournot game), but it actually depends on each of the other players' parameters. This rules out many elegant and simple results on the existence and uniqueness of Nash equilibria [117–120]. Secondly, although the realization fo the actions are nonnegative, it is not obvious that θ_i's need to be nonnegative, or even bounded. The main result in the paper is to overcome these challenges and show that under some assumptions, the game is well-behaved and policy gradient updates converge exponentially quickly to the Nash equilibrium in Stochastic Cournot games.

Theorem 8. *Consider a stochastic Cournot game satisfying the assumptions (A1) and (A2). Suppose each player's policy is parameterized as the mean θ_i and a zero-mean random variable X_i that is unimodal at the mean with infinite support, and suppose that all players follow policy gradient in* (5.5) *to update their mean. Then the policies converge to the Nash equilibrium exponentially quickly for all initializations either of the following condition holds:*

1. *The price function is linear.*

2. *The number of players equals two.*

The condition of the theorem includes Gaussian policies, which is a natural choice for

continuous action spaces [114, 121], and such a form also includes popular neural network policies [122] where the mean can be parameterized via a neural network. We also do not restrict the players to be symmetric, and each of the players would adopt different variances or even have completely different classes of distributions. The infinite support requirement of the distribution is a technicality and can be weakened, although it would make the proofs much more cumbersome.

The proof of Theorem 8 proceeds in three lemmas. We defer the full proofs of these lemmas to Appendix D and sketch the proof here. The first step in the proof is to show that we can restrict the actions of the players to a compact region using the following lemma:

Lemma 3. *Under the assumptions of Theorem 8, θ_i can be restricted to $[\underline{\theta}_i, y_{\max}]$, where $\underline{\theta}_i$ is a constant.*

This lemma essentially confines the choices of the players to a compact interval, which sets up the rest of the proof. As a reminder, $y_{\max}$ is the point where the price function becomes 0. The proof of this lemma is based on showing that player i's profit will be suboptimal if it chooses an θ_i outside of the interval, regardless of other players' choices.

Interestingly, to show the parameters of the policy gradients converges to the Nash equilibrium of the stochastic Cournot game for the two cases stated in Theorem 8, we need two different proof techniques. Therefore, we separate them into two lemmas as stated below.

Lemma 4. *Under the assumptions of (A1)-(A2) and suppose the market price is linear, the policy gradient updates converge to the unique Nash equilibrium exponentially fast under all initial conditions.*

Lemma 5. *Under the assumptions of (A1)-(A2) and suppose there are only two players, the policy gradient updates converge to the unique Nash equilibrium exponentially fast under all initial conditions.*

The proof of Lemma 4 leverages Rosen's conditions in [123]. A sufficient condition for the convergence of gradient-based algorithms in concave N-player games is that the game Hessian

is *negative definite*. Therefore, we prove Lemma 4 in two steps. First is to show that the stochastic Cournot games with assumptions of (A1)-(A2) are concave N-player game, and then show the game Hessian is negative definite under linear price functions. However, once the price function is not linear, we cannot directly use Rosen's conditions for the convergence proof, even under the two-player case. The proof of Lemma 5 is based on a dynamical system interpretation. We proved that under the two-player general price case, the game Hessian is strictly diagonally dominant, thus the Nash equilibrium is an exponentially stable fixed point.

We close this section with two remarks. Firstly, our proof provides a sufficient condition for the convergence of policy gradient in Cournot games, that is either the price function is linear, or the player number is no more than two. However, it may not be the necessary condition. In particular, we provide a three-player example with quadratic price function in Section 5.4.2, which also shows convergence behavior. Secondly, it should be noted that in practice, some players may decide to not use policy gradient but some other learning algorithms. We provide some empirical evaluations of the system robustness in Section 5.4.2, by assuming a small portion of players acts randomly. Both directions, 1) generalizing the convergence proof to a broader class of games and 2) dynamics under heterogeneous learning agents are important as future works.

5.4 Numerical experiments

In this section, we exam the performance of policy gradient algorithms in various of Cournot games. We first verify the convergence behavior under linear price and two-player cases. Next, we provide investigative studies on the system behavior under multi-player and players with random actions scenarios is well as intuitions for the robustness behavior.

We perform all the experiments using the natural policy gradient algorithm [115] with a

Gaussian policy. The gradient with respect to the policy parameter θ_i follows,

$$\nabla_{\theta_i} J_i(\theta_i) = E_x[\pi_i(x_i, x_{-i}) \nabla_{\theta_i} \log f_{\theta_i}(x_i)] = \frac{1}{N} \sum_{i=1}^{N} \hat{\pi}_i \nabla_\theta \log f_\theta(x_i)\,, \tag{5.6}$$

where $\pi_i(x_i, x_{-i})$ is the payoff function of player i and $\hat{\pi}_i$ is the observed payoff. In the above formula, $f_{\theta_i}(x_i) = \theta_i + X_i$ is the decision making policy for player i and $X_i \sim N(0, \sigma_i)$. For the action, we have $x_i = (\mu_i + X_i)^+$, where the action is truncated to be non-negative. The update rules for μ_i follows the natural policy gradient in [115]. We choose the standard deviation for each player the same as $\sigma = 0.05$.

5.4.1 Cournot Game Examples

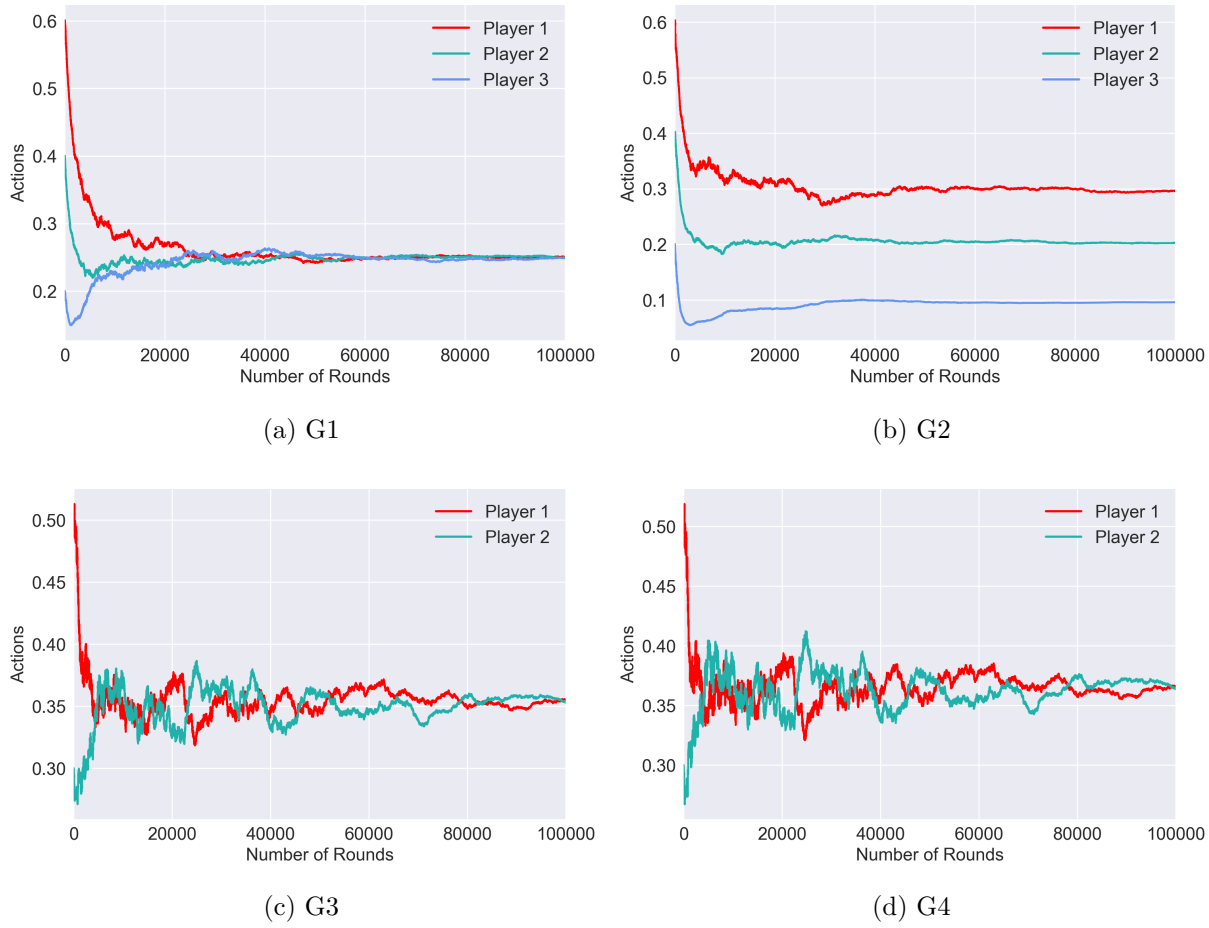
In this section, we verify the convergence behavior of the proposed algorithm in four example Cournot games, with different price and individual cost settings.

G1: three-player with linear price function $p(\boldsymbol{x}) = 1 - (x_1 + x_2 + x_3)$ and no individual cost $C_i(x_i) = 0, \forall i$. The Nash equilibrium is $x_1^* = x_2^* = x_3^* = \frac{1}{4}$.

G2: three-player with linear price function $p(\boldsymbol{x}) = 1 - (x_1 + x_2 + x_3)$ and differnet individual cost $C_i(x_i) = 0.1 \cdot i \cdot x_i$ for player i. The Nash equilibrium is $x_1^* = 0.3, x_2^* = 0.2, x_3^* = 0.1$.

G3: two-player quadratic price function $p(\boldsymbol{x}) = 1 - (x_1 + x_2)^2$ without cost. The Nash equilibrium is $x_1^* = x_2^* = \sqrt{1/8} \approx 0.3536$. **G4:** two-player cubic price function $p(\boldsymbol{x}) = 1 - \frac{1}{2}(x_1 + x_2)^3$ without cost. The Nash equilibrium is $x_1^* = x_2^* = \sqrt[3]{1/20} \approx 0.3684$.

In all of the games, each player simultaneously picks a production level. The price is determined by the sum of productions and broadcasted back to all players. This game is repeated multiple times with all players use policy gradient to learn and act. The dynamics of the policy parameter (i.e., the mean) are plotted in Figure 5.1. In all simulated games with different initializations and settings, the policy parameters converge to the Nash equilibrium, which verifies the theoretical results in Section 5.3.



(a) G1

(b) G2

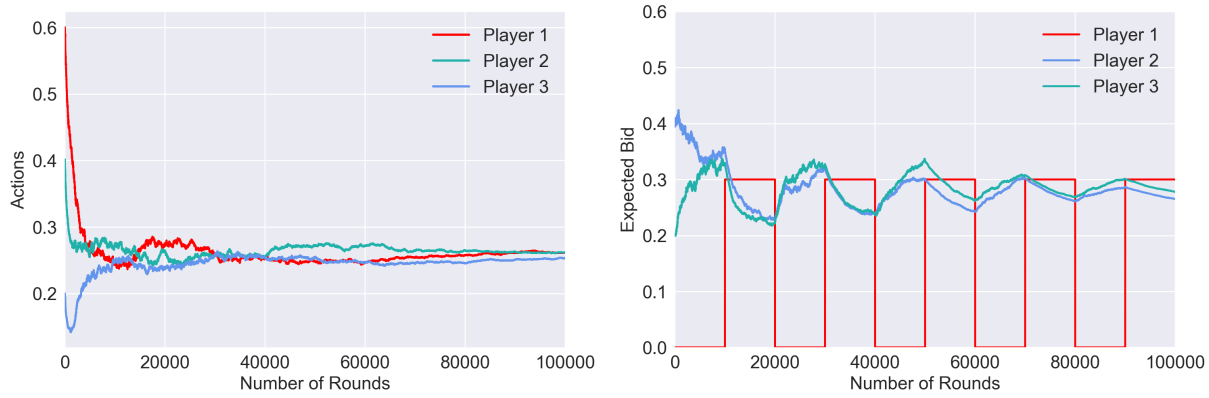
(c) G3

(d) G4

Figure 5.1: Convergence behavior of policy gradient in stochastic Cournot games: (a)-(b) are games with linear price and (c)-(d) are two-player games with general price functions.

5.4.2 Investigative Studies

In this section, we provide two investigative studies relating to system performance under more general setups: 1) multi-agent Cournot game with non-linear price function; 2) hetergenous players that do not follow policy gradient updates. Note that our theoretical result in Section 5.3 does not apply to the following two cases. **G5:** three-player with quadratic price function $p(\boldsymbol{x}) = 1 - (x_1 + x_2 + x_3)^2$ and no cost. **G6:** three-player with linear price function $p(\boldsymbol{x}) = 1 - (x_1 + x_2 + x_3)$ and no individual cost. One player does not follow policy gradient updates. Fig 5.2 shows that both the three-player general price and heterogeneous players



(a) G5 (b) G6

Figure 5.2: Policy gradient dynamics beyond Theorem 1's convergence conditions.

cases also converge to some equilibria, though they do not satisfy the convergence conditions in Theorem 8, These results are promising in the sense that our results might be able to generalize to a broader class of games, and theoretically proving these would be valuable future work. There may also be settings where players are malicious, but designing optimal adversarial tactics and the detection algorithms, by themselves are topics that contain a vast body of literature and is beyond the scope of this work.

5.5 Conclusion

In this chapter, we study the interaction of strategic players in Cournot games with limited feedback. Since each player only has local observation without knowing other players' actions, they face the challenge of decision making under uncertainty. The problem is further complicated as all players are self-interested and are strategically optimizing their own behaviors. Notably, we proved the convergence of policy gradient reinforcement learning to the Nash equilibrium, under two conditions: either the price function is linear or there are two players. An important next step is to exam the convergence performance using real-world electricity market data. In addition, extending the results to more general conditions such as multi-player general price functions is also an important future direction.

Chapter 6

CONCLUSION AND FUTURE WORKS

6.1 Conclusion

This dissertation addressed control of energy systems under various kinds of uncertainties, including environmental uncertainty, model uncertainty, and uncertainties from user interaction and competition. It presents a set of computation algorithms that combines the *physical knowledge* of power systems and *algorithm innovations* of machine learning and optimization, that provide solutions to different energy system control problems including energy storage, smart buildings, and distribution grid voltage control. These approaches obtain both optimality guarantee and tractable computational complexity, which have been demonstrated through mathematical proofs.

In addition to the detailed conclusions given in each chapter, a number of generalizations can be drawn from this dissertation:

- The pathway towards a more sustainable future is built on top of both technology advancement and algorithm innovations (in other words, hardware + computing). To handle the large amount of uncertainties brought by renewables and users, new technology is much-needed. For instance, energy storage is a game-changer for the system since it can shift energy across time and space, thus to relieve the stress of balancing versatile supply and demand in real-time. However, to unlock the full potential of new technology requires tailor-made computing algorithms, such as the battery online control and multiplexing algorithms proposed in Chapter 2 and Chapter 3.

- Data will play a major role in future energy system operation under uncertainties. Ubiquitous measurement and advanced data analytics can help system operators to better understand the "current", and better predict the "future". Instead of making decisions shrouded in the fog, decisions can be optimized with better situational awareness. A good illustration is the ICNN for building energy management and distribution system voltage control in Chapter 4. Both systems are subject to unknown models, but we are able to obtain satisfying performance by reconstructing the model via data and designing efficient control laws.

- Last but not least, it is important to recognize the "social" properties of energy systems beyond the "physical" properties. Energy system interconnects millions of individuals, where each of them having different roles and objectives. With more distributed energy resources (e.g., renewables, storage, controllable load) entering the system, it becomes more challenging to manage the market considering their complex interactions. Yet it creates opportunities if we could leverage the individual capacities and flexibilities to compensate for the uncertainty and balance the system on its own. Chapter 5 provides some promising steps towards a more spontaneous market organization framework via iterative learning and mechanism optimization.

This dissertation concludes that with the technology advancement of distributed energy resources, big data analytics, effective control, and market algorithm design, we are on the right track towards a more flexible, sustainable and intelligent energy system design to handle the ever-increasing amount of operational uncertainties.

6.2 Suggestions for Future Work

This dissertation provides some promising steps towards a more effcient, intelligent and sustainable energy system design, under significant level of uncertainties. Moving forward, some areas of future explorations are exciting.

6.2.1 Learning and Control for Distributed Energy Resources

The current energy system is centrally managed and optimized, where major participants are few large generators. However, with the Smartgrid transformation, the demand-side would play a critical role. Specifically, there will be millions of users, ranging from renewables, electrical vehicles, controllable loads, and smart energy homes. It will lead to a revolution in system control and management scheme - from a centralized, homogeneous, and generator-centric system to a distributed, heterogeneous, and user-centric system.

It is known that if all agents use standard learning and optimization methods such as gradient descent, it can lead to oscillations and divergence even in simple two-player systems. This leads to a set of interesting questions regarding how to design effective distributed learning and control algorithms, as well as incentives, that guarantee the overall system stability and efficiency. In particular,

1. Suppose each agent is running heterogeneous learning algorithms for decision making, collectively do the agents converge to some stable equilibrium? Can we have a unified framework to analyze heterogeneous learning agents?
2. How do agents incorporate forecast information? For example, system operators may provide some predictions about future electricity demand and supply at different time scales. These signals also provide agents with side information, but they are distinct from users' own state observations since they are about the future and may well be wrong. How would agents react to forecasts, and how would users' policies change compared to only the past and current information is used?

Answers to the above questions would enable distributed, and intelligent control for hundreds of thousands of devices across future energy systems, from energy storage, renewable, to smart energy homes. A successful control framework would take into account the individual objectives, guarantee each participant's welfare, and maintain the overall system stability.

6.2.2 Safe and Robust Decision Making

Data-driven control is a promising way to go for future energy system control, such as grid frequency and voltage control under high-level uncertainties and variabilities. However, standard machine learning algorithms ignore the constraints of physical systems and can lead to unsafe decisions. It would require tailor-made applications of learning algorithms that consider the safety and robustness of the decision-making algorithms and have a theoretical guarantee on the performance.

There are different types of safety requirements in energy systems, that require different methods to deal with. For instance, energy storages have power and energy capacity limits, which need to be satisfied at each step of execution, which is referred to as the *safe exploration* problem [124]. There are other types of safety requirements, such as the need for robust decision-making under uncertainties (e.g., frequency control under uncertain renewable generation), which is referred to as the *robust reinforcement learning* problem [125]. In general, the fundamental properties regarding data-driven control systems are not well understood yet, which concerns its translation from simulation to the real world. Future works on both theoretic analyses of data-driven control system safety and applications in real-world energy systems would be highly valuable.

6.2.3 Mechanism Design for Future Energy Markets

Market mechanism design is also a vital component to enable a sustainable future. The introduction of new resources (e.g., renewables, demand-side participation) creates significant opportunities but also brings in challenges, as the current electricity market design is not ready to embrace all the participants. For example, the current electricity pricing mechanism is based on fossil fuel's marginal price. However, solar and wind have zero-marginal cost which calls for a different pricing mechanism. In addition, to hedge the high uncertainties

caused by renewables, the system operator might need to provide more information beyond the current Locational Marginal Price (LMP) feedback. In what follows we lay out a series of important research questions in this future energy market design:

1. How to set price for zero-carbon energy systems? One major challenge for future energy market design is that the utility functions for many participants are hard to quantify, e.g., renewables with zero-marginal-cost but non-negligible fixed investment cost, and demand-side agents with unclear utility curves. An adaptive pricing mechanism might be needed, that iteratively estimate the individual utility functions and optimizing the pricing function based on the estimation.

2. How to release information *spatially* and *temporally* in distributed energy systems? To coordinate a large number of distributed energy resources, information design is also a powerful lever [126]. It might be feasible to share more information within the neighborhood area such that demand and supply can learn to balance locally; and share less information with other areas to maintain adequate competition. Questions of where and when to reveal, and how much information to share are fundamental to the optimal information design in energy markets.

It is important for the market operator to provide the right spectrum of incentives and information, thus to make the most efficient utilization of energy possible.

BIBLIOGRAPHY

[1] "Next era's secret recipe for energy storage: Planning," tech. rep.

[2] Y. Shi, B. Xu, Y. Tan, and B. Zhang, "A convex cycle-based degradation model for battery energy storage planning and operation," in *2018 Annual American Control Conference (ACC)*, pp. 4590–4596, IEEE, 2018.

[3] Y. Shi, B. Xu, Y. Tan, D. Kirschen, and B. Zhang, "Optimal battery control under cycle aging mechanisms in pay for performance settings," *IEEE Transactions on Automatic Control*, vol. 64, no. 6, pp. 2324–2339, 2018.

[4] B. Xu, Y. Shi, D. S. Kirschen, and B. Zhang, "Optimal battery participation in frequency regulation markets," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6715–6725, 2018.

[5] Y. Shi, B. Xu, D. Wang, and B. Zhang, "Using battery storage for peak shaving and frequency regulation: Joint optimization for superlinear gains," *IEEE Transactions on Power Systems*, vol. 33, no. 3, pp. 2882–2894, 2017.

[6] Y. Chen, Y. Shi, and B. Zhang, "Optimal control via neural networks: A convex approach," in *International Conference on Learning Representations (ICLR)*, 2019.

[7] Y. Chen, Y. Shi, and B. Zhang, "Modeling and optimization of complex building energy systems with deep neural networks," in *2017 51st Asilomar Conference on Signals, Systems, and Computers*, pp. 1368–1373, IEEE, 2017.

[8] Y. Chen, Y. Shi, and B. Zhang, "Input convex neural networks for optimal voltage regulation," *arXiv preprint arXiv:2002.08684*, 2020.

[9] B. Dunn, H. Kamath, and J.-M. Tarascon, "Electrical energy storage for the grid: a battery of choices," *Science*, vol. 334, no. 6058, pp. 928–935, 2011.

[10] E. Bitar, R. Rajagopal, P. Khargonekar, and K. Poolla, "The role of co-located storage for wind power producers in conventional electricity markets," in *American Control Conference (ACC), 2011*, pp. 3886–3891, IEEE, 2011.

[11] Y. Shi, B. Xu, B. Zhang, and D. Wang, "Leveraging energy storage to optimize

data center electricity cost in emerging power markets," in *Proceedings of the Seventh International Conference on Future Energy Systems (e-Energy)*, p. 18, ACM, 2016.

[12] N. Li, L. Chen, and S. H. Low, "Optimal demand response based on utility maximization in power networks," in *Power and Energy Society General Meeting, 2011 IEEE*, pp. 1–8, IEEE, 2011.

[13] Y. Xu and L. Tong, "Optimal operation and economic value of energy storage at consumer locations," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 792–807, 2017.

[14] P. Arora, R. E. White, and M. Doyle, "Capacity fade mechanisms and side reactions in lithium-ion batteries," *Journal of the Electrochemical Society*, vol. 145, no. 10, pp. 3647–3667, 1998.

[15] J. Qin, Y. Chow, J. Yang, and R. Rajagopal, "Online modified greedy algorithm for storage control under uncertainty," *IEEE Transactions on Power Systems*, vol. 31, no. 3, pp. 1729–1743, 2016.

[16] H. Pandzic, Y. Wang, T. Qiu, Y. Dvorkin, and D. S. Kirschen, "Near-optimal method for siting and sizing of distributed storage in a transmission network," *Power Systems, IEEE Transactions on*, vol. 30, no. 5, pp. 2288–2300, 2015.

[17] A. A. Akhil, G. Huff, A. B. Currier, B. C. Kaun, D. M. Rastler, S. B. Chen, A. L. Cotter, D. T. Bradshaw, and W. D. Gauntlett, *DOE/EPRI 2013 electricity storage handbook in collaboration with NRECA*. Sandia National Laboratories Albuquerque, NM, 2013.

[18] B. Zakeri and S. Syri, "Electrical energy storage systems: A comparative life cycle cost analysis," *Renewable and Sustainable Energy Reviews*, vol. 42, pp. 569–596, 2015.

[19] M. Ecker, N. Nieto, S. Käbitz, J. Schmalstieg, H. Blanke, A. Warnecke, and D. U. Sauer, "Calendar and cycle life study of li (nimnco) o 2-based 18650 lithium-ion batteries," *Journal of Power Sources*, vol. 248, pp. 839–851, 2014.

[20] P. W. Northrop, V. Ramadesigan, S. De, and V. R. Subramanian, "Coordinate transformation, orthogonal collocation, model reformulation and simulation of electrochemical-thermal behavior of lithium-ion battery stacks," *Journal of The Electrochemical Society*, vol. 158, no. 12, pp. A1461–A1477, 2011.

[21] V. Ramadesigan, P. W. Northrop, S. De, S. Santhanagopalan, R. D. Braatz, and V. R. Subramanian, "Modeling and simulation of lithium-ion batteries from a systems engineering perspective," *Journal of The Electrochemical Society*, vol. 159, no. 3, pp. R31–R45, 2012.

[22] L. Xie, Y. Gu, A. Eskandari, and M. Ehsani, "Fast mpc-based coordination of wind power and battery energy storage systems," *Journal of Energy Engineering*, vol. 138, no. 2, pp. 43–53, 2012.

[23] J. Qin, Y. Chow, J. Yang, and R. Rajagopal, "Distributed online modified greedy algorithm for networked storage operation under uncertainty," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 1106–1118, 2016.

[24] J. H. Kim and W. B. Powell, "Optimal energy commitments with storage and intermittent supply," *Operations research*, vol. 59, no. 6, pp. 1347–1360, 2011.

[25] Y. Xu and L. Tong, "On the value of storage at consumer locations," in *PES General Meeting| Conference & Exposition, 2014 IEEE*, pp. 1–5, IEEE, 2014.

[26] P. M. van de Ven, N. Hegde, L. Massoulié, and T. Salonidis, "Optimal control of end-user energy storage," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 789–797, 2013.

[27] J. L. Melo, T. J. Lim, and S. Sun, "Online demand response strategies for non-deferrable loads with renewable energy," *IEEE Transactions on Smart Grid*, 2017.

[28] E. Chemali, L. McCurlie, B. Howey, T. Stiene, M. M. Rahman, M. Preindl, R. Ahmed, and A. Emadi, "Minimizing battery wear in a hybrid energy storage system using a linear quadratic regulator," in *Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE*, pp. 003265–003270, IEEE, 2015.

[29] B. Bouchard, R. Elie, and C. Imbert, "Optimal control under stochastic target constraints," *SIAM Journal on Control and Optimization*, vol. 48, no. 5, pp. 3501–3531, 2010.

[30] G. He, Q. Chen, C. Kang, P. Pinson, and Q. Xia, "Optimal bidding strategy of battery storage in power markets considering performance-based regulation and battery cycle life," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2359–2367, 2016.

[31] V. Pop, H. J. Bergveld, D. Danilov, P. P. Regtien, and P. H. Notten, *Battery management systems: Accurate state-of-charge indication for battery-powered applications*, vol. 9. Springer Science & Business Media, 2008.

[32] M. R. Almassalkhi and I. A. Hiskens, "Model-predictive cascade mitigation in electric power systems with storage and renewables—part i: Theory and implementation," *IEEE Transactions on Power Systems*, vol. 30, no. 1, pp. 67–77, 2015.

[33] B. Xu, A. Oudalov, A. Ulbig, G. Andersson, and D. Kirschen, "Modeling of lithium-ion

battery degradation for cell life assessment," *IEEE Transactions on Smart Grid*, vol. PP, no. 99, pp. 1–1, 2016.

[34] J. Hughes, A. Dominguez-Garcia, and K. Poolla, "Coordinating heterogeneous distributed energy resources for provision of frequency regulation services," in *Proceedings of the 50th Hawaii International Conference on System Sciences*, 2017.

[35] A. Millner, "Modeling lithium ion battery degradation in electric vehicles," in *Innovative Technologies for an Efficient and Reliable Electricity Supply (CITRES), 2010 IEEE Conference on*, pp. 349–356, IEEE, 2010.

[36] R. H. Byrne and C. A. Silva-Monroy, "Estimating the maximum potential revenue for grid connected electricity storage: Arbitrage and regulation," *Sandia National Laboratories*, 2012.

[37] P. Ruetschi, "Aging mechanisms and service life of lead–acid batteries," *Journal of Power Sources*, vol. 127, no. 1, pp. 33–44, 2004.

[38] J. Wang, J. Purewal, P. Liu, J. Hicks-Garner, S. Soukazian, E. Sherman, A. Sorenson, L. Vu, H. Tataria, and M. W. Verbrugge, "Degradation of lithium ion batteries employing graphite negatives and nickel–cobalt–manganese oxide+ spinel manganese oxide positives: Part 1, aging mechanisms and life estimation," *Journal of Power Sources*, vol. 269, pp. 937–948, 2014.

[39] G. Marsh, C. Wignall, P. R. Thies, N. Barltrop, A. Incecik, V. Venugopal, and L. Johanning, "Review and application of rainflow residue processing techniques for accurate fatigue damage estimation," *International Journal of Fatigue*, vol. 82, pp. 757–765, 2016.

[40] "Northern arizona wind & sun inc. battery cycles v.s. lifespan plot."

[41] I. Laresgoiti, S. Käbitz, M. Ecker, and D. U. Sauer, "Modeling mechanical degradation in lithium ion batteries during cycling: Solid electrolyte interphase fracture," *Journal of Power Sources*, vol. 300, pp. 112–122, 2015.

[42] D. Krishnamurthy, C. Uckun, Z. Zhou, P. Thimmapuram, and A. Botterud, "Energy storage arbitrage under day-ahead and real-time price uncertainty," *IEEE Transactions on Power Systems*, 2017.

[43] J. E. Parsons, C. Colbert, J. Larrieu, T. Martin, and E. Mastrangelo, "Financial arbitrage and efficient dispatch in wholesale electricity markets," Tech. Rep. CEEPR WP 2015-002, MIT Center for Energy and Environmental Policy Research, 2015.

[44] I. Rychlik, "Extremes, rainflow cycles and damage functionals in continuous random processes," *Stochastic processes and their applications*, vol. 63, no. 1, pp. 97–116, 1996.

[45] E. Hazan *et al.*, "Introduction to online convex optimization," *Foundations and Trends® in Optimization*, vol. 2, no. 3-4, pp. 157–325, 2016.

[46] W. B. Powell and S. Meisel, "Tutorial on stochastic optimization in energy—part i: Modeling and policies," *IEEE Transactions on Power Systems*, vol. 31, no. 2, pp. 1459–1467, 2016.

[47] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[48] B. Stephen, L. Xiao, and A. Mutapcic, "Subgradient methods," in *lecture notes of EE392, Stanford University*, Autume 2004.

[49] C. Amzallag, J. Gerey, J. Robert, and J. Bahuaud, "Standardization of the rainflow counting method for fatigue analysis," *International journal of fatigue*, vol. 16, no. 4, pp. 287–293, 1994.

[50] "PJMRegulation Market Issues Senior Task Force."

[51] "PJM Regulation Zone Preliminary Billing Data," Aug 2017.

[52] B. Wasowicz, S. Koopmann, T. Dederichs, A. Schnettler, and U. Spaetling, "Evaluating regulatory and market frameworks for energy storage deployment in electricity grids with high renewable energy penetration," in *In European Energy Market (EEM), 2012 9th International Conference on the , IEEE*, 2012.

[53] X. Xi, R. Sioshansi, and V. Marano, "A stochastic dynamic programming model for co-optimization of distributed energy storage," *Energy Systems*, vol. 5, no. 3, 2014.

[54] B. Cheng and W. Powell, "Co-optimizing battery storage for the frequency regulation and energy arbitrage using multi-scale dynamic programming," *IEEE Transactions on Smart Grid*, 2016.

[55] R. Walawalkar, J. Apt, and R. Mancini, "Economics of electric energy storage for energy arbitrage and regulation in new york," *Energy Policy*, vol. 35, no. 4, pp. 2558–2568, 2007.

[56] C. D. White and K. M. Zhang, "Using vehicle-to-grid technology for frequency regulation and peak-load reduction," *Journal of Power Sources*, vol. 196, no. 8, pp. 3972–3980, 2011.

[57] A. W. Dowling, R. Kumar, and V. M. Zavala, "A multi-scale optimization framework for electricity market participation," *Applied Energy*, vol. 190, pp. 147–164, 2017.

[58] S. Lukas, E. Lobato, and L. Rouco, "Energy storage systems providing primary reserve and peak shaving in small isolated power systems: an economic assessment," *International Journal of Electrical Power & Energy Systems*, vol. 53, pp. 675–683, December 2013.

[59] D. Caprino, M. L. D. Vedova, and T. Facchinetti, "Peak shaving through real-time scheduling of household appliances," *Energy and Buildings*, vol. 75, pp. 133–148, June 2014.

[60] H. Tao, *Short term electric load forecasting.* North Carolina State University, 2010.

[61] H. Wang and J. Huang, "Cooperative planning of renewable generations for interconnected microgrids," *IEEE Transactions on Smart Grid*, vol. 7, pp. 2486–2496, December 2016.

[62] J. Qin, Y. Chow, J. Yang, and R. Rajagopal, "Online modified greedy algorithm for storage control under uncertainty," *IEEE Transactions on Power Systems*, vol. 31, pp. 1729–1743, May 2016.

[63] Cisco Validated Designs, "Data center technolog design guide," tech. rep., Cisco, 2014.

[64] W. Wolf, "Cyber-physical systems," *Computer*, vol. 42, pp. 88–89, 2009.

[65] C.-C. Cheng, S. Pouffary, N. Svenningsen, and J. M. Callaway, "The kyoto protocol, the clean development mechanism and the building and construction sector: A report for the unep sustainable buildings and construction initiative," 2008.

[66] Z. Zhang, R. Deng, T. Yuan, and S. J. Qin, "Distributed optimization of multi-building energy systems with spatially and temporally coupled constraints," in *American Control Conference (ACC), 2017*, pp. 2913–2918, IEEE, 2017.

[67] Y. Ma, A. Kelman, A. Daly, and F. Borrelli, "Predictive control for energy efficient buildings with thermal storage: Modeling, stimulation, and experiments," *IEEE Control Systems*, vol. 32, no. 1, pp. 44–64, 2012.

[68] P. H. Shaikh, N. B. M. Nor, P. Nallagownden, I. Elamvazuthi, and T. Ibrahim, "A review on optimized control systems for building energy and comfort management of smart sustainable buildings," *Renewable and Sustainable Energy Reviews*, vol. 34, pp. 409–429, 2014.

[69] N. K. Suryadevara, S. C. Mukhopadhyay, S. D. T. Kelly, and S. P. S. Gill, "Wsn-based smart sensors and actuator for power management in intelligent buildings," *IEEE/ASME transactions on mechatronics*, vol. 20, no. 2, pp. 564–571, 2015.

[70] J. Kwac, J. Flora, and R. Rajagopal, "Household energy consumption segmentation using hourly data," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 420–430, 2014.

[71] T. Overbye, P. Sauer, C. DeMarco, B. Lesieutre, and M. Venkatasubramanian, "Using pmu data to increase situational awareness," *Power System Engineering Research Center (PSERC) Publication*, vol. 21, pp. 10–16, 2010.

[72] K. Kawaguchi, "Deep learning without poor local minima," in *Advances in Neural Information Processing Systems*, pp. 586–594, 2016.

[73] B. Amos, L. Xu, and J. Z. Kolter, "Input convex neural networks," in *International Conference on Machine Learning*, pp. 146–155, 2017.

[74] T. Wei, Y. Wang, and Q. Zhu, "Deep reinforcement learning for building hvac control," in *The Design and Automation Conference*, 2017.

[75] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1334–1373, 2016.

[76] L. Ljung, "System identification," in *Signal analysis and prediction*, pp. 163–173, Springer, 1998.

[77] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7559–7566, IEEE, 2018.

[78] D. Meger, J. C. G. Higuera, A. Xu, P. Giguere, and G. Dudek, "Learning legged swimming gaits from experience," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 2332–2338, IEEE, 2015.

[79] A. Magnani and S. P. Boyd, "Convex piecewise-linear fitting," *Optimization and Engineering*, vol. 10, no. 1, pp. 1–17, 2009.

[80] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*, vol. 2. Wiley New York, 2007.

[81] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*, vol. 15. Siam, 1994.

[82] Y. Shi, B. Xu, Y. Tan, D. Kirschen, and B. Zhang, "Optimal battery control under cycle aging mechanisms in pay for performance settings," *IEEE Transactions on Automatic Control*, pp. 1–1, 2018.

[83] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, W. F. Buhl, Y. J. Huang, C. O. Pedersen, R. K. Strand, R. J. Liesen, D. E. Fisher, M. J. Witte, *et al.*, "Energyplus: creating a new-generation building energy simulation program," *Energy and buildings*, vol. 33, no. 4, pp. 319–331, 2001.

[84] S. Kouro, P. Cortés, R. Vargas, U. Ammann, and J. Rodríguez, "Model predictive control—a simple and powerful method to control power converters," *IEEE Transactions on industrial electronics*, vol. 56, no. 6, pp. 1826–1838, 2009.

[85] P. M. Carvalho, P. F. Correia, and L. A. Ferreira, "Distributed reactive power generation control for voltage rise mitigation in distribution networks," *IEEE transactions on Power Systems*, vol. 23, no. 2, pp. 766–772, 2008.

[86] P. Jahangiri and D. C. Aliprantis, "Distributed volt/var control by pv inverters," *IEEE Transactions on power systems*, vol. 28, no. 3, pp. 3429–3439, 2013.

[87] H. Zhu and H. J. Liu, "Fast local voltage control under limited reactive power: Optimality and stability analysis," *IEEE Transactions on Power Systems*, vol. 31, no. 5, pp. 3794–3803, 2015.

[88] M. E. Baran and F. F. Wu, "Optimal capacitor placement on radial distribution systems," *IEEE Transactions on power Delivery*, vol. 4, no. 1, pp. 725–734, 1989.

[89] M. Farivar, C. R. Clarke, S. H. Low, and K. M. Chandy, "Inverter var control for distribution systems with renewables," in *2011 IEEE international conference on smart grid communications (SmartGridComm)*, pp. 457–462, IEEE, 2011.

[90] M. Farivar, R. Neal, C. Clarke, and S. Low, "Optimal inverter var control in distribution systems with high pv penetration," in *2012 IEEE Power and Energy Society general meeting*, pp. 1–7, IEEE, 2012.

[91] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Transactions on Power delivery*, vol. 4, no. 2, pp. 1401–1407, 1989.

[92] H. Li, Y. Weng, Y. Liao, B. Keel, and K. E. Brown, "Robust hidden topology identification in distribution systems," *arXiv preprint arXiv:1902.01365*, 2019.

[93] G. Qu and N. Li, "An optimal and distributed feedback voltage control under limited reactive power," in *2018 Power Systems Computation Conference (PSCC)*, pp. 1–7, IEEE, 2018.

[94] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*, pp. 177–186, Springer, 2010.

[95] A. A. Cournot, *Recherches sur les principes mathématiques de la théorie des richesses*. 1838.

[96] D. S. Kirschen and G. Strbac, *Fundamentals of power system economics*. John Wiley & Sons, 2018.

[97] K. Bimpikis, S. Ehsani, and R. İlkılıç, "Cournot competition in networked markets," *Management Science*, vol. 65, no. 6, pp. 2467–2481, 2019.

[98] M. Chletsos and A. Saiti, "Hospitals as suppliers of healthcare services," in *Strategic Management and Economics in Health Care*, pp. 179–205, Springer, 2019.

[99] C. Shapiro, "Theories of oligopoly behavior," *Handbook of Industrial Organization*, vol. 1, pp. 329–414, 1989.

[100] A. Kannan and U. V. Shanbhag, "Distributed computation of equilibria in monotone nash games via iterative regularization techniques," *SIAM Journal on Optimization*, vol. 22, no. 4, pp. 1177–1205, 2012.

[101] A. F. Daughety, *Cournot oligopoly: characterization and applications*. Cambridge university press, 2005.

[102] U. Nadav and G. Piliouras, "No regret learning in oligopolies: cournot vs. bertrand," in *International Symposium on Algorithmic Game Theory (EC)*, pp. 300–311, Springer, 2010.

[103] P. Mertikopoulos and Z. Zhou, "Learning in games with continuous action sets and unknown payoff functions," *Mathematical Programming*, vol. 173, no. 1-2, pp. 465–507, 2019.

[104] T. Roughgarden, "Algorithmic game theory," *Communications of the ACM*, vol. 53, no. 7, pp. 78–86, 2010.

[105] S. Arora, E. Hazan, and S. Kale, "The multiplicative weights update method: a meta-algorithm and applications," *Theory of Computing*, vol. 8, no. 1, pp. 121–164, 2012.

[106] E. Hazan, "Introduction to online convex optimization," *Found. Trends Optim.*, vol. 2, pp. 157–325, Aug. 2016.

[107] A. Kalai and S. Vempala, "Efficient algorithms for online decision problems," *Journal of Computer and System Sciences*, vol. 71, no. 3, pp. 291–307, 2005.

[108] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *arXiv preprint arXiv:1911.10635*, 2019.

[109] D. S. Leslie and E. J. Collins, "Individual q-learning in normal form games," *SIAM Journal on Control and Optimization*, vol. 44, no. 2, pp. 495–514, 2005.

[110] E. Rodrigues Gomes and R. Kowalczyk, "Dynamic analysis of multiagent q-learning with ε-greedy exploration," in *Proceedings of the 26th International Conference on Machine Learning (ICML)*, pp. 369–376, 2009.

[111] G. Arslan and S. Yüksel, "Decentralized q-learning for stochastic teams and games," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1545–1558, 2016.

[112] R. Johari and J. N. Tsitsiklis, "Efficiency loss in cournot games," *Harvard University*, 2005.

[113] F. Szidarovszky and S. Yakowitz, "A new proof of the existence and uniqueness of the cournot equilibrium," *International Economic Review*, pp. 787–789, 1977.

[114] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Proceedings of the 12th International Conference on Neural Information Processing Systems (NeurIPS)*, 1999.

[115] S. M. Kakade, "A natural policy gradient," in *Proceedings of the 15th International Conference on Neural Information Processing Systems (NeurIPS)*, pp. 1531–1538, 2002.

[116] P.-W. Chou, D. Maturana, and S. Scherer, "Improving stochastic policy gradients in continuous control with deep reinforcement learning using the beta distribution," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 834–843, JMLR. org, 2017.

[117] F. Szidarovszky and S. Yakowitz, "Contributions to cournot oligopoly theory," *Journal of Economic Theory*, vol. 28, no. 1, pp. 51–70, 1982.

[118] W. Novshek, "On the existence of cournot equilibrium," *The Review of Economic Studies*, vol. 52, no. 1, pp. 85–98, 1985.

[119] R. Amir, "Cournot oligopoly and the theory of supermodular games," *Games and Economic Behavior*, vol. 15, no. 2, pp. 132–148, 1996.

[120] C. Ewerhart, "Cournot games with biconcave demand," *Games and Economic Behavior*, vol. 85, pp. 37–47, 2014.

[121] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proceedings of the 31st International Conference on International Conference on Machine Learning (ICML)*, 2014.

[122] S. Levine and V. Koltun, "Guided policy search," in *Proceedings of 30th International Conference on International Conference on Machine Learning (ICML)*, 2013.

[123] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," *Econometrica: Journal of the Econometric Society*, pp. 520–534, 1965.

[124] L. Zheng, Y. Shi, L. J. Ratliff, and B. Zhang, "Safe reinforcement learning of control-affine systems with vertex networks," *arXiv preprint arXiv:2003.09488*, 2020.

[125] R. J. A. A. J. S. Y. S. J. K. T. M. T. H. M. R. Daniel Mankowitz, Nir Levine, "Robust reinforcement learning for continuous control with model misspecification," *arXiv preprint arXiv:1906.07516*, 2019.

[126] Y. Shi and B. Zhang, "No-regret learning in cournot games," *arXiv preprint arXiv:1906.06612*, 2019.

[127] H. Royden and P. Fitzpatrick, "Real analysis. 4th," 2010.

[128] M. G. Cox, "An algorithm for approximating convex functions by means by first degree splines," *The Computer Journal*, vol. 14, no. 3, pp. 272–275, 1971.

[129] M. M. Gavrilović, "Optimal approximation of convex curves by functions which are piecewise linear," *Journal of Mathematical Analysis and Applications*, vol. 52, no. 2, pp. 260–282, 1975.

[130] S. Wang, "General constructive representations for continuous piecewise-linear functions," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 9, pp. 1889–1896, 2004.

[131] E. W. Weisstein, "Gershgorin circle theorem," 2003.

[132] L. J. Ratliff, S. A. Burden, and S. S. Sastry, "Characterization and computation of local nash equilibria in continuous games," in *2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 917–924, IEEE, 2013.

Appendix A

PROOF OF THEOREMS FOR CHAPTER 2

A.1 Proof for Theorem 1

Here we provide the detailed proof of Theorem 1 in Chapter 2. Since all SoC profile can be written as the sum of step functions, by induction method, we first need to prove that $f(\mathbf{x})$ is convex up to a step function as base case (Lemma 1).

Proposition 4. *Let $g(\cdot)$ be a convex function where $g(0) = 0$. Let $r_1, r_2, r_3, ..., r_n$ be real numbers, suppose*

- $\sum_{i=1}^{n} r_i = D > 0$
- $|r_i| \leq D, \forall i \in \{1, 2, 3, ..., n\}$

Then,

$$g(\sum_{i=1}^{n} r_i) \geq \sum_{\{i:r_i \geq 0\}} g(r_i) - \sum_{\{i:r_i < 0\}} g(|r_i|). \tag{A.1}$$

Proposition 5. *Consider a step change added to $\mathbf{x}$, where $\mathbf{x}^{'}(t) = \mathbf{x}(t) + Q_t U_t$, $t \in [0, T]$. Suppose Q_t is positive [1], the rainflow cycle decomposition results (only considering charging cycles) for $\mathbf{x}$ and $\mathbf{x}^{'}$ are,*

$$\mathbf{x} : v_1, v_2, ..., v_m, ..., v_M \ ,$$

[1]The proof for negative Q_t is the same, just change Q_t to $|Q_t|$

$$\mathbf{x}^{'} : v_1{}^{'}, v_2{}^{'}, ..., v_n{}^{'}, ..., v_N{}^{'},$$

Define $L = max(M, N)$*, we could re-write the cycles in* $\mathbf{x}$ *and* $\mathbf{x}^{'}$ *as,*

$$\mathbf{x} : \underbrace{v_1, v_2, ..., v_M, 0, 0, ...}_{L},$$
$$\mathbf{x}^{'} : \underbrace{v_1{}^{'}, v_2{}^{'}, ..., v_N{}^{'}, 0, 0, ...}_{L},$$

Define Δv_i *such that,* $v_i{}^{'} = v_i + \Delta v_i \,, \forall i = 1, 2, ..., L$ *The following relations always holds,*

$$\left| \sum_{i=1}^{L} \Delta v_i \right| \leq Q_t \,, \tag{A.2}$$

$$|\Delta v_i| \leq Q_t \,, \tag{A.3}$$

Proof of Lemma 1. Let's consider $\mathbf{x}^{'} = \lambda \mathbf{x} + (1 - \lambda) Q_t U_t$. Then the rainflow cycle decomposition results for $\lambda \mathbf{x}$ and $\mathbf{x}^{'}$ are

$$\lambda \mathbf{x} : \underbrace{\lambda v_1, \lambda v_2, ..., \lambda v_M, 0, 0, ...}_{L}$$
$$\mathbf{x}^{'} : \underbrace{v_1{}^{'}, v_2{}^{'}, ..., v_N{}^{'}, 0, 0, ...}_{L}$$

Define Δv_i such that,

$$v_i{}^{'} = \lambda v_i + (1 - \lambda) \Delta v_i \,, \forall i = 1, 2, ..., L$$

$$
\begin{aligned}
& f(\lambda \mathbf{x} + (1-\lambda) Q_t U_t) \\
=& \sum_{i=1}^{L} \Phi(\lambda v_i + (1-\lambda)\Delta v_i) \\
=& \sum_{i=1}^{L^+} \underbrace{\Phi(\lambda v_i + (1-\lambda)\Delta v_i)}_{\Delta v_i \geq 0} + \sum_{i=1}^{L^-} \underbrace{\Phi(\lambda v_i - (1-\lambda)|\Delta v_i|)}_{\Delta v_i < 0} \\
\leq& \sum_{i=1}^{L^+} [\lambda \Phi(v_i) + (1-\lambda)\Phi(\Delta v_i)] + \sum_{i=1}^{L^-} [\lambda \Phi(v_i) - (1-\lambda)\Phi(|\Delta v_i|)] \\
\leq& \lambda \sum_{i=1}^{L} \Phi(v_i) + (1-\lambda)[\sum_{i=1}^{L^+} \Phi(\Delta v_i) - \sum_{i=1}^{L^-} \Phi(|\Delta v_i|)]
\end{aligned} \tag{A.4}
$$

To continue the proof in (A.4) and derive the final relation, we separate the whole variable space to two cases based on equations (A.2) and (A.3).

(1). Assume $\sum_{i=1}^{L} \Delta v_i = Q_i$, $|\Delta v_i| \leq Q_i$. By Proposition (4), it follows that

$$
\begin{aligned}
f\Big(\lambda \mathbf{x} + (1-\lambda) Q_t U_t\Big) \leq& \lambda \sum_{i=1}^{L} \Phi(v_i) + (1-\lambda)\Big[\sum_{i=1}^{L^+} \Phi(\Delta v_i) - \sum_{i=1}^{L^-} \Phi(|\Delta v_i|)\Big] \\
\leq& \lambda \sum_{i=1}^{L} \Phi(v_i) + (1-\lambda)\Phi(\sum_{i=1}^{L} \Delta v_i) \\
=& \lambda \sum_{i=1}^{L} \Phi(v_i) + (1-\lambda)\Phi(Q_t)
\end{aligned}
$$

(2) Assume $-Q_t \leq \sum_{i=1}^{L} \Delta v_i < Q_t$, $|\Delta v_i| \leq Q_t$.

Add some "virtual cycles" $v_{L+1}^{'}, v_{L+2}^{'}, ..., v_{L+K}^{'}$ at the end of $\mathbf{x}^{'}$, each $v_{L+i}^{'}$ is positive and satisfies that $|v_{L+i}^{'}| \leq Q_t$. So that $\sum_{i=1}^{L+K} \Delta v_i = Q_t$, $|\Delta v_i| \leq Q_t, \forall i \in [1, 2, ..., L+K]$. Write 0

at the end of $\lambda\mathbf{x}$ to achieve the same cycle number.

$$\lambda\mathbf{x} : \underbrace{\lambda v_1, \lambda v_2, ..., \lambda v_M, 0, 0, 0, ..., 0}_{L+K}$$

$$\mathbf{x}^{'} : \underbrace{v_1^{'}, v_2^{'}, ..., v_N^{'}, 0, 0, ..., 0, v_{L+1}^{'}, v_{L+2}^{'}, ..., v_{L+K}^{'}}_{L+K}$$

$$\begin{aligned} f\Big(\lambda\mathbf{x} + (1-\lambda)Q_tU_t\Big) \leq & \lambda\sum_{i=1}^{L}\Phi(v_i) + (1-\lambda)\Big[\sum_{i=1}^{l^+}\Phi(\Delta v_i) - \sum_{i=1}^{l^-}\Phi(|\Delta v_i|)\Big] \\ < & \lambda\sum_{i=1}^{L}\Phi(v_i) + (1-\lambda)\Big[\sum_{i=1}^{l^+}\Phi(\Delta v_i) + \sum_{i=L+1}^{L+K}\Phi(\Delta v_i) - \sum_{i=1}^{l^-}\Phi(|\Delta v_i|)\Big] \\ \leq & \lambda\sum_{i=1}^{L}\Phi(v_i) + (1-\lambda)\Phi(\sum_{i=1}^{L+K}\Delta v_i) = \lambda\sum_{i=1}^{L}\Phi(v_i) + (1-\lambda)\Phi(Q_t) \end{aligned}$$

To sum up,

$$f(\lambda\mathbf{x} + (1-\lambda)Q_tU_t) \leq \lambda\sum_{i=1}^{L}\Phi(v_i) + (1-\lambda)\Phi(Q_t) = \lambda f(\mathbf{x}) + (1-\lambda)f(Q_tU_t)\,, \tag{A.5}$$

where $\lambda \in [0,1]$. □

Lemma 1 shows that $f(\mathbf{x})$ is convex up to every step change in $\mathbf{x}$.

By lemma 1, we already proved the base case convexity. When $K = 1$,

$$f\Big(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}\Big) \leq \lambda f(\mathbf{x}) + (1-\lambda)f(\mathbf{y})\,, \lambda \in [0,1]$$

Next we need to show the induction relation. Suppose that, $f(\mathbf{x})$ is convex up to the sum of K step changes (arranged by time index)

$$f\Big(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}\Big) \leq \lambda f(\mathbf{x}) + (1-\lambda)f(\mathbf{y})\,, \lambda \in [0,1], \mathbf{x}, \mathbf{y} \in \mathbb{R}^K$$

Then we prove $f(\mathbf{x})$ is convex up to the sum of $K+1$ step changes,

The following proposition is needed for the proof.

Proposition 6.

$$f(\sum_{t=1}^{K} P_t U_t) \geq f(\sum_{t=1}^{i-1} P_t U_t + (P_i + P_{i+1})U_i + \sum_{t=i+2}^{K} P_t U_t)\,,$$

In other words, the cycle stress cost will reduce if combining adjacent unit changes.

Proof. The rainflow cycle counting algorithm only considers local extreme points.

I) If P_i and P_{i+1} are the same direction, combining them doesn't affect the value of local extreme points. Therefore the left side cost equals right side cost.

II) If P_i and P_{i+1} are in different directions, suppose P_i is negative and P_{i+1} positive (otherwise the same). Time $t = i$ makes a local minimum point.

- Case a: If $|P_{i+1}| \leq |P_i|$, combining them will raise the value of local minimum point i, thus reducing the depth of cycles which contains i. Therefore, the cost after combining is less than the original cost.

- Case b: If $|P_{i+1}| > |P_i|$, combining them will lead to the removal of local minimum point i.

  In one case, if P_{i-1} and P_i are the same direction, time $t = i - 1$ will make a local minimum point taking the place of time $t = i$. Therefore, the magnitude of the local minimum point decreases, similar to case (a), the total cost after combining is less than the original cost.

  In the other case, if P_{i-1} and P_i are different directions, we lose a full cycle with depth $|P_i|$ after combining. So the cost after combining P_i, P_{i+1} is also less than the original.

□

There are three cases when the profile length go from K to $K + 1$ depending on the values and signs. The arguments are straightforward but tedious, and the interested reader can refer to the online version of [82].

A.2 Proof of Theorem 2 and Theorem 3

A.2.1 Model Reformulation

Both Theorem 2 and Corollary 1 follows directly from Theorem 3. To prove Theorem 3, we rewrite the optimization problem (2.8) as,

$$(\mathbf{c}^*, \mathbf{d}^*) \in \arg\min_{\mathbf{c},\mathbf{d}} f(\mathbf{c}, \mathbf{d}) - \tau \sum_{t=1}^{T} \Big[\theta c_t + \pi d_t\Big] \tag{A.6a}$$

$$\text{subject to (2.8b), (2.8c), and}$$

$$0 \le c_t \le [r_t]^+ \tag{A.6b}$$

$$0 \le d_t \le [-r_t]^+ \tag{A.6c}$$

by observing that a battery's actions would never exceed the regulation signals. $f(\mathbf{c}, \mathbf{d})$ defines the rainflow cycle-based degradation cost.

We utilize the rainflow algorithm to transform the problem into a cycle-based form. The rainflow method maps the entire operation uniquely to cycles, the sum of all charge and discharge power can be represented as the sum of cycle depths as (recall that a full cycle has symmetric depth for charge and discharge)

$$\sum_{i=1}^{|\mathbf{u}|} u_i + \sum_{i=1}^{|\mathbf{v}|} v_i = \frac{\tau \eta_{\mathrm{c}}}{E} \sum_{t=1}^{T} c_t \tag{A.7}$$

$$\sum_{i=1}^{|\mathbf{u}|} u_j + \sum_{i=1}^{|\mathbf{w}|} w_i = \frac{\tau}{\eta_{\mathrm{d}} E} \sum_{t=1}^{T} d_t \,. \tag{A.8}$$

We substitute (A.7) and (A.8) into the reformulated objective function (A.6a) to replace c_t and d_t with cycle depths

$$J_{\mathrm{cyc}}(\mathbf{c}, \mathbf{d}) + J_{\mathrm{reg}}(\mathbf{c}, \mathbf{d}, \mathbf{r}) =$$
$$\sum_{i=1}^{|\mathbf{u}|} J_{\mathrm{u}}(u_i) + \sum_{i=1}^{|\mathbf{v}|} J_{\mathrm{v}}(v_i) + \sum_{i=1}^{|\mathbf{w}|} J_{\mathrm{w}}(w_i) \,. \tag{A.9}$$

A.2.2 Proof for Theorem 3

The following lemmas support the proof for Theorem 3.

Lemma 6. *Suppose an minimizer* $(\mathbf{c}^*, \mathbf{d}^*)$ *of* (2.8) *in the offline setting has the corresponding cycle depths* $(\mathbf{u}^*, \mathbf{v}^*, \mathbf{w}^*)$*. Then the depth of each cycle in this result either reaches the optimal cycle depth or bounded by the operation constraints as*

$$u_i^* = \min(\hat{u}, \overline{u}_i) \tag{A.10a}$$

$$v_i^* = \min(\hat{v}, \overline{v}_i) \tag{A.10b}$$

$$w_i^* = \min(\hat{w}, \overline{w}_i) \tag{A.10c}$$

where $\overline{u}_i$*,* $\overline{v}_i$*,* $\overline{w}_i$ *denote constraint bounds including the regulation instruction signal and battery energy limit.*

Lemma 7. *A cycle depth in the control action of* $g(\cdot)$ *either reaches the depth of* $\hat{u}$ *or is bounded by the operation constraints.*

Lemma 8. *There exists one and only one half cycle with the largest depth in a rainflow residue profile. Other half cycles are in strictly decreasing order either to the left- or to the right-hand side direction of this largest half cycle.*

It is easy to see now from Lemma 6 and Lemma 7 that the proposed control policy achieves optimal control result for all full cycles, and the optimality gap is caused by half cycle results. Consider the following relationship in a rainflow residue profile as in Lemma 8 assuming the largest half cycle is in the discharging direction

$$\ldots < w_{j-1}^* < v_{j-1}^* < w_j^* > v_j^* > w_{j+1}^* > \ldots \tag{A.11}$$

and substitute Lemma 7 into (A.11)

$$\ldots \min\{\hat{v}, \overline{v}_j\} < \min\{\hat{w}, \overline{w}_j\} > \min\{\hat{v}, \overline{v}_{j+1}\} \ldots \tag{A.12}$$

It is easy to see now that if $\hat{w} > \hat{v}$, then the largest possible value for w_j^* is $\hat{w}$, and the largest possible value for v_j^* and v_{j-1}^* is $\hat{v}$, the rest half cycles in (A.11) must have depths smaller

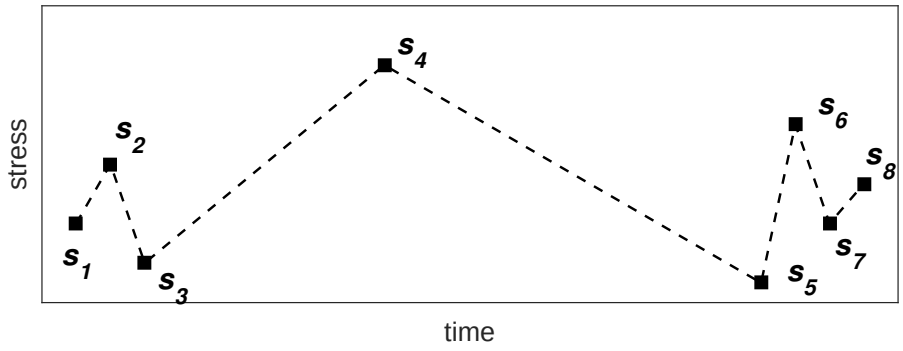


Figure A.1: Illustration for Lemma 8. The largest half cycle is between s_4 and s_5, other half cycles are in strictly decreasing order either to the left- or to the right-hand side direction of this largest half cycle.

than $\hat{v}$, which indicates that their depths are bounded by operation. If $\hat{v} > \hat{w}$, then the largest possible value for w_j^* is $\hat{w}$, and the rest half cycles must have depths smaller than $\hat{w}$. We repeat this analysis for cases that v_j^* is the largest cycle, and summarize the half cycle conditions in Table A.1 Hence, the worst-case optimality gap is caused by that some

Table A.1: Summarizing Half Cycle Depth Conditions

	$\hat{w} > \hat{v}$	$\hat{w} < \hat{v}$
Half cycles of depth $\hat{w}$	At most one	At most two
Half cycles of depth $\hat{v}$	At most two	At most one
Rest half cycles	must be $< \hat{v}$	must be $< \hat{w}$

half cycles have depth $\hat{u}$ or $\hat{w}$, while the control policy enforces $\hat{u}$ as the depth of all cycles unbounded by operation. The gap in Theorem 3 is therefore calculated using half cycle depth conditions in Table A.1.

Proof for Lemma 6: Since cycles are linear combinations of charge and discharge power, and

constraints (A.6b), (A.6c), (2.8c) can be transformed into linear constrains with respect to cycle depths. From Theorem 1, the transformed cycle-based problem is also has a convex objective function with linear constraints. Although exact formulations of the transformed constraints are complicated to express, we use $\overline{u}_i$, $\overline{v}_i$, and $\overline{w}_i$ to denote these binds, which are sufficient for the proof for Theorem 3.

Proof for Lemma 7: The rainflow method always identify the largest cycle as between the

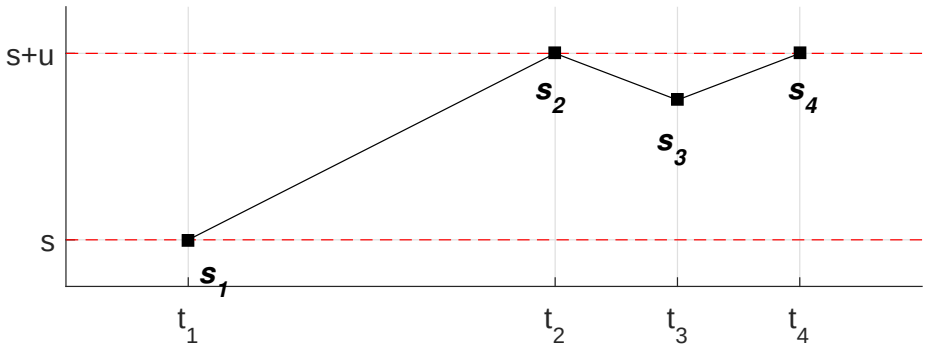


Figure A.2: Illustration for Lemma 7.

minimum and the maximum SoC point. In the proposed policy, any operation that goes outside the defined operation zone will cause the largest cycle depth to change instead of the depth of the cycle it was previous in. For example, in Fig. A.2 the maximum cycle is between SoC s and $s+u$, and the battery is at time t_4. If the battery continue to charge and the SoC goes about $s+u$, then this operation will increase the largest cycle depth instead of the shallower cycles assoicated with extrema s_2, s_3 and s_4.

Proof for Lemma 8: Because the rainflow method identifies a cycle from extrema distances if $\Delta s_{i-1} \geq \Delta s_i \leq \Delta s_{i+1}$, then all extrema in the rainflow residue must satisfy either $\Delta s_{i-1} < \Delta s_i < s_{i+1}$ or $\Delta s_{i-1} < \Delta s_i > s_{i+1}$ or $\Delta s_{i-1} > \Delta s_i > s_{i+1}$, which proofs this lemma.

Appendix B

PROOF OF THEOREMS FOR CHAPTER 3

B.1 Load prediction

Solving the stochastic joint optimization problem in (3.6) requires accurate short-term load forecasting (STLF) for the next 24 hours. A lot of research has been done in the area of STLF. There are two major factors determine the quality of load prediction, *input features* and prediction *model*. On the one hand, selecting features or a group of features which affect the future load most is important. The input features mainly include the effect of nature (eg. temperature) and the effects of human activities (calendar variables, e.g., business hours), and the interaction of above two factors. On the other hand, deciding which kind of models to forecast future load is also crucial. People have been adopting or developing various techniques for day-ahead load forecasting, including regression, time series analysis, neural networks, support vector machine and a combination of the above methods [60].

We used a multiple linear regression (MLR) model that takes $\mathcal{X} = \{$trend, temperature forecasting (TMP), month, Hour $\times$ TMP, month $\times$ TMP, day $\times$ Hour, adjacency day's load, weekend and holiday effect, recent similar days' average$\}$ as input, and use the following MLR model to predict the power demand for next 1 day. Fig B.1 presents the day-ahead load prediction result for a data center.

$$
\begin{aligned}
Y = {} & \beta_0 + \beta_1 \times Trend + \beta_2 \times TMP + \beta_3 \times Month \\
& + \beta_4 \times Hour \times TMP + \beta_5 \times month \times TMP \\
& + \beta_6 \times day \times Hour + \beta_7 \times Load(day-1) \\
& + \beta_8 \times weekend + \beta_9 \times holiday + \beta_{10} \times \bar{Load}(day-1)
\end{aligned}
\tag{B.1}
$$

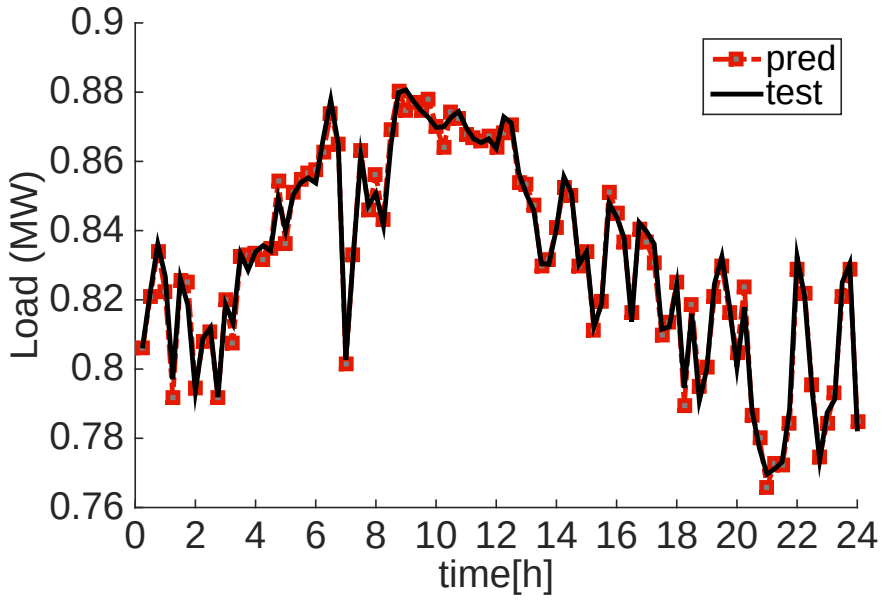


Figure B.1: Data center load prediction. The black curve is the actual demand, and the red line are the day ahead load prediction using MLR. The load is scaled between 0 and 1MW.

B.2 Scenario Selection

In order to solve the stochastic joint optimization problem in (3.6), we also need to model the uncertainty of future regulation signals. We use one-year historical data to empirically

model the distributions of regulation signals. Each daily realization of the regulation signal is called a "scenario", and thus we obtain 365 scenarios.

Because a large number of scenarios will reduce the computational tractability of the joint optimization problem, it is useful to choose a smaller subset of scenarios that can well approximate the original entire scenario set. We applied the forward scenario reduction algorithm in [61] to select the best subset of scenarios, and assign new probabilities to the selected scenarios. The key idea of scenario reduction is to pick a subset of scenarios which preserve as much information as the original set. We set the number of selected scenarios as 10, which strives for a balance between performance and computational complexity by simulation. For visualization clarity, we plot 4 out of the 10 selected scenarios in Fig. B.2.

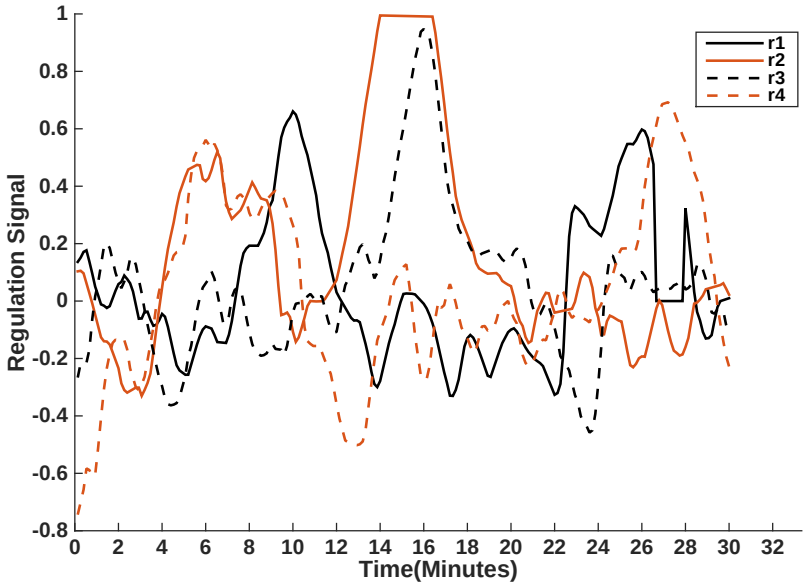


Figure B.2: Selected regulation signal scenarios. r1, r2, r3, r4 are the top four representative frequency regulation signal scenarios.

B.3 Proof of Theorem 1

Here we provide a detailed proof of Theorem 1 in 3.4.2.

Proof. Under linear battery cost model, it is obvious that $b^*(t)$ and $r(t)$ always have the same sign. Or equivalent saying, $b(t)$ and $r(t)$ are always both positive or both negative. Based on the relative sizes of coefficients and sign, there are 5 cases to be considered:

$$\begin{cases} \lambda_b < \lambda_{mis} \begin{cases} r(t) \geq 0 \begin{cases} b(t) \geq Cr(t) \ (i) \\ b(t) < Cr(t) \ (ii) \end{cases} \\ r(t) < 0 \begin{cases} b(t) \geq Cr(t) \ (iii) \\ b(t) < Cr(t) \ (iv) \end{cases} \end{cases} \\ \lambda_b \geq \lambda_{mis} \ (v) \end{cases}$$

(i). $\lambda_b < \lambda_{mis}$, $r(t) \geq 0$ and $b(t) \geq Cr(t)$

In this case, $b(t) \geq 0$, battery is discharging. And the objective function (3.8a) becomes,

$$\underset{C,b(t)}{\text{maximize}} \ \ \lambda_c C + \frac{1}{T} E\{-(\lambda_{mis} + \lambda_b) \sum_{t=1}^{T} b(t) + \lambda_{mis} \sum_{t=1}^{T} Cr(t)\}$$

,

Notice that the coeffiecient in front of $b(t)$ is negative. In order to maximize the objective, we need to minimize $b(t)$ under the following constraints:

$$\begin{cases} b(t) \geq 0 \\ b(t) \geq Cr(t) \\ b(t) \leq P \\ b(t) \leq \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{min}E}{t_s} \end{cases}$$

- If either $Cr(t) > P$ or $Cr(t) > \frac{SoC_{ini}E + \sum_{t=1}^{\tau-1} b(\tau)t_s - SoC_{min}E}{t_s}$, there is no feasible solution.
- If $Cr(t) \leq P$ and $Cr(t) \leq \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{min}E}{t_s}$, optimal solution $b^*(t) = Cr(t)$.

(ii). $\lambda_b < \lambda_{mis}$, $r(t) \geq 0$ and $b(t) < Cr(t)$

In this case, $b(t) \geq 0$, battery is discharging, and the objective function (3.8a) becomes,

$$\underset{C,b(t)}{\text{maximize}} \quad \lambda_c C + \frac{1}{T} E\{(\lambda_{mis} - \lambda_b) \sum_{t=1}^{T} b(t) - \lambda_{mis} \sum_{t=1}^{T} Cr(t)\}$$

,

Notice the coeffiecient in front of $b(t)$ is $(\lambda_{mis} - \lambda_b)$, which is positive in this case. So in order to maximize the objective, we need to maximize $b(t)$ under the following constraints:

$$\begin{cases} b(t) \geq 0 \\ b(t) < Cr(t) \\ b(t) \leq P \\ b(t) \leq \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{min}E}{t_s} \end{cases}$$

So $b^*(t) = min\{Cr(t), P, \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{min}E}{t_s}\}$.

Summarizing case (i) and (ii), we get:

If $\lambda_b < \lambda_{mis}$, $r(t) \geq 0$, $b^*(t) = min\{Cr(t), P, \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{min}E}{t_s}\}$.

(iii). $\lambda_b < \lambda_{mis}$, $r(t) < 0$ and $b(t) \geq Cr(t)$

In this case, $b(t) < 0$, battery is charging, and the objective function (3.8a) becomes,

$$\underset{C,b(t)}{\text{maximize}} \quad \lambda_c C + \frac{1}{T} E\{(-\lambda_{mis} + \lambda_b) \sum_{t=1}^{T} b(t) + \lambda_{mis} \sum_{t=1}^{T} Cr(t)\}$$

,

The coeffiecient in front of $b(t)$ is $(-\lambda_{mis} + \lambda_b)$, which is negative. So in order to maximize the objective, we need to minimize $b(t)$ under the following constraints:

$$\begin{cases} b(t) \leq 0 \\ b(t) \geq Cr(t) \\ b(t) \geq -P \\ b(t) \geq \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{max}E}{t_s} \end{cases}$$

The minimal $b(t)$ is optimal, $b^*(t) = max\{Cr(t), -P, \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{max}E}{t_s}\}$.

(iv). $\lambda_b < \lambda_{mis}$, $r(t) < 0$ and $b(t) < Cr(t)$

In this case, $b(t) < 0$, battery is charging, and the objective function (3.8a) becomes,

$$\underset{C,b(t)}{\text{maximize}} \quad \lambda_c C + \frac{1}{T} E\{(\lambda_{mis} + \lambda_b) \sum_{t=1}^{T} b(t) - \lambda_{mis} \sum_{t=1}^{T} Cr(t)\}$$

,

The coeffiecient in front of $b(t)$ is positive, so in order to maximize the objective, we need to maximize $b(t)$ under the following constraints:

$$\begin{cases} b(t) < 0 \\ b(t) < Cr(t) \\ b(t) \geq -P \\ b(t) \geq \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{max}E}{t_s} \end{cases}$$

- If either $Cr(t) < -P$ or $Cr(t) < \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{max}E}{t_s}$, there is no feasible solution for $b(t)$.
- If $Cr(t) \geq -P$ and $Cr(t) \geq \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{max}E}{t_s}$, we have the optimal $b^*(t) = Cr(t)$.

Summarizing case (iii) and (iv), we get:

If $\lambda_b < \lambda_{mis}$, $r(t) < 0$, $b^*(t) = max\{Cr(t), -P, \frac{SoC_{ini}E + \sum_{\tau=1}^{t-1} b(\tau)t_s - SoC_{max}E}{t_s}\}$.

(v). $\lambda_b \geq \lambda_{mis}$

We know that $b(t)$ and $r(t)$ always have the same sign, so the objective function (3.8a) could be expressed as,

$$\lambda_c \cdot C - \frac{1}{T} E\{\sum_{t=1}^{T} \lambda_{mis}||b(t)| - C|r(t)|| + \sum_{t=1}^{T} \lambda_b |b(t)|\},$$

For each time step t, we take deriviative of the objective function w.r.t. $|b(t)|$,

$$\begin{aligned}
\frac{\delta J}{\delta |b(t)|} &= -(\lambda_{mis} \cdot \vec{1}_{|b(t)>C|r(t)||} - \lambda_{mis} \cdot \vec{1}_{|b(t)<C|r(t)||} + \lambda_b) \\
&= -\lambda_{mis} \cdot \vec{1}_{|b(t)>C|r(t)||} + \lambda_{mis} \cdot \vec{1}_{|b(t)<C|r(t)||} - \lambda_b \\
&\leq \lambda_{mis} \cdot \vec{1}_{|b(t)<C|r(t)||} - \lambda_b \\
&\leq \lambda_{mis} - \lambda_b \leq 0
\end{aligned}$$

Since $\frac{\delta J}{\delta |b(t)|} \leq 0$, so in order to maximize J (the regulation service benefits), $|b| = 0$. Therefore, when $\lambda_b \geq \lambda_{mis}$, $b^*(t) = 0$ is optimal for $\forall C \geq 0$. □

Appendix C

PROOF OF THEOREMS FOR CHAPTER 4

C.1 Toy Example

Consider a synthetic example which contains two circles of noisy input data $\mathbf{u} \in \mathbb{R}^2$, along with discrete data label $y \in \{0, 1\}$ which is based on input coming from inner loop ($y = 0$) or outer loop ($y = 1$). Suppose a decision maker is interested in finding the $\mathbf{u}$ that maximizes the probability of y being 0. This optimization problem can be solved by firstly learning a neural network classifier from $\mathbf{u}$ to y, and then to find the $\mathbf{u}$ point which minimizes the output of the neural network. More specifically, let f_{NN} be a conventional neural network and f_{ICNN} be an ICNN. Then the objective becomes minimizing $f_{NN}(\mathbf{u})$ or $f_{ICNN}(\mathbf{u})$.

Figure C.1 shows the decision boundaries for f_{NN} and f_{ICNN}, respectively. These networks are composed of 2 hidden layers, with 200 neurons in each layer, and are trained using the same random seed, same number of samples (100) until loss convergence. The decision boundaries of a conventional network have many "zigzags", which makes solving (4.1) challenging, especially if $\mathbf{u}$ is constrained. In contrast, the ICNN has convex level sets (by construction) as decision boundaries, which leads to a convex optimization problem.

C.2 Proof of Theorem 1

Proof. Lemma 2 follows from well established facts in function analysis stating that piecewise linear functions are dense in the space of all continuous functions over compact sets [127] and

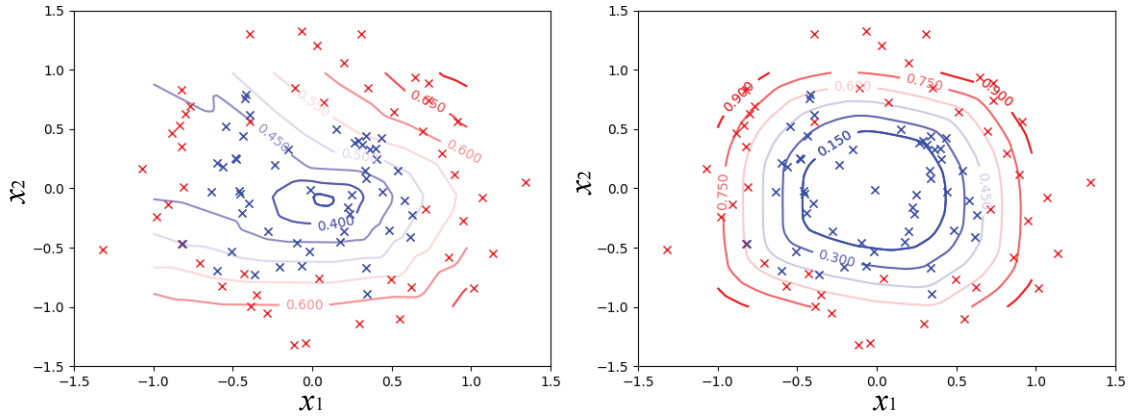


Figure C.1: Toy example on classifying circle data with label 0 (blue cross) and label 1 (red cross) along with conventional neural networks (left) and ICNN (right) decision contour lines. A decision maker is interested in finding a $\mathbf{u}$ that has the highest probability of being labeled 0.

convex piecewise linear functions are dense in the space of all convex continuous functions [128, 129]. Using the fact that convex piecewise linear functions can be represented as a maximum of affine functions [79, 130] gives the desired result in the lemma.

Lemma 1 shows that all continuous Lipschitz convex functions $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ over convex compact sets can be approximated using maximum of affine functions. Then it suffices to show that an ICNN can exactly represent a maximum of affine functions. To do this, we first construct a neural network with ReLU activation function with both positive and negative weights that can represent a maximum of affine functions. Then we show how to restrict all weights to be nonnegative.

As a starting example, consider a maximum of two affine functions

$$f_{CPL}(\mathbf{x}) = \max\{\mathbf{a}_1^T\mathbf{x} + b_1, \mathbf{a}_2^T\mathbf{x} + b_2\}. \tag{C.1}$$

To obtain the exact same function using a neural network, we first rewrite it as

$$f_{CPL}(x) = (\mathbf{a}_2^T\mathbf{x} + b_2) + \max\Big((\mathbf{a}_1 - \mathbf{a}_2)^T\mathbf{x} + (b_1 - b_2), 0\Big). \tag{C.2}$$

Now define a two-layer neural network with layers $\mathbf{z}_1$ and $\mathbf{z}_2$ as shown in Fig. C.2:

$$z_1 = \sigma\Big((\mathbf{a}_1 - \mathbf{a}_2)^T\mathbf{x} + (b_1 - b_2)\Big), \tag{C.3a}$$

$$z_2 = z_1 + \mathbf{a}_2^T\mathbf{x} + b_2 \tag{C.3b}$$

where σ is the ReLU activation function and the second layer is linear. By construction, this neural network is the same function as f_{CPL} given in (C.1).

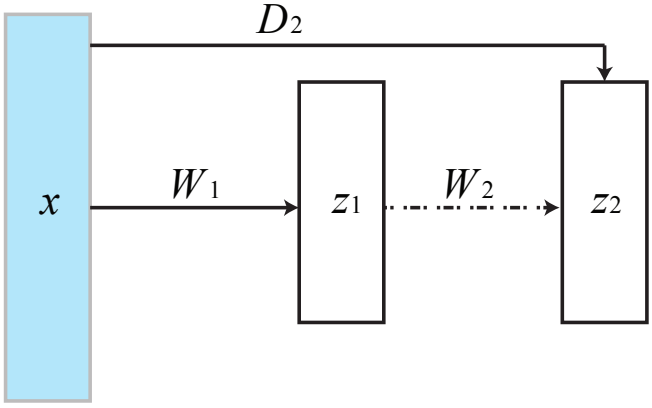


Figure C.2: A simple two-layer neural networks. In alignment with (C.3), W_1 denotes the first-layer weights $\mathbf{a}_1 - \mathbf{a}_2$ and bias $b_1 - b_2$, and W_2 denotes the linear second layer. Direct layer is denoted as D_2 for weights $\mathbf{a}_2$ and bias b_2.

The above argument extends directly to a maximum of K linear functions. Suppose

$$f_{CPL}(\mathbf{x}) = \max\{\mathbf{a}_1^T\mathbf{x} + b_1, ..., \mathbf{a}_K^T\mathbf{x} + b_K\} \tag{C.4}$$

Again the trick is to rewrite $f_{CPL}(\mathbf{x})$ as a nested maximum of affine functions. For notational

convenience, let $L_i = \mathbf{a}_i^T\mathbf{x} + b_i$, $L_i' = L_i - L_{i+1}$. Then

$$\begin{aligned}
f_{CPL} &= \max\{L_1, L_2, ..., L_K\} \\
&= \max\{\max\{L_1, L_2, ..., L_{K-1}\}, L_K\} \\
&= L_K + \sigma(\max\{L_1, L_2, ..., L_{K-1}\} - L_K) \\
&= L_K + \sigma(\max\{\max\{L_1, L_2, ..., L_{K-2}\}, L_{K-1}\} - L_K, 0) \\
&= L_K + \sigma(L_{K-1} - L_K + \sigma(\max\{L_1, L_2, ..., L_{K-2}\} - L_{K-1}, 0), 0) \\
&= ... \\
&= L_K + \sigma\Big(L'_{K-1} + \sigma\Big(L'_{K-2} + \sigma(...\sigma(L'_2 + \sigma(L_1 - L_2, 0), 0), ..., 0)\Big), 0\Big).
\end{aligned}$$

The last equation describes a K layer neural network, where the layers are:

$$\begin{aligned}
z_1 &= \sigma(L_1 - L_2, 0) = \sigma\Big((\mathbf{a}_1 - \mathbf{a}_2)^T\mathbf{x} + (b_1 - b_2)\Big), \\
z_2 &= \sigma(L'_2 + z_1, 0) = \sigma\Big(z_1 + (\mathbf{a}_2 - \mathbf{a}_3)^T\mathbf{x} + (b_2 - b_3)\Big), \\
&..... \\
z_i &= \sigma(L'_i + z_{i-1}, 0) = \sigma\Big(z_{i-1} + (\mathbf{a}_i - \mathbf{a}_{i+1})^T\mathbf{x} + (b_i - b_{i+1})\Big), \\
&..... \\
z_K &= z_{K-1} + L_K = h_K(z_{K-1} + L_K) = \Big(z_{K-1} + \mathbf{a}_K^T\mathbf{x} + b_K\Big).
\end{aligned}$$

Each layer of of this neural network uses only a single activation function.

Although the above neural network exactly represent a maximum of linear functions, it is not convex since the coefficients between layers could be negative. In particular, each layer involves an inner product of the form $(\mathbf{a}_i - \mathbf{a}_{i+1})^T\mathbf{x}$ and the coefficients are not necessarily nonnegative. To overcome this, we simply expand the input to include $\mathbf{x}$ and $-\mathbf{x}$. Namely, define a new input $\hat{\mathbf{x}} \in \mathbb{R}^{2d}$ as

$$\hat{\mathbf{x}} = \begin{bmatrix} \mathbf{x} \\ -\mathbf{x} \end{bmatrix}. \tag{C.5}$$

Then any inner product of the form $\mathbf{h}^T\mathbf{x}$ can be written as

$$\begin{aligned}\mathbf{h}^T\mathbf{x} &= \sum_{j=1}^{d} h_i x_i \\ &= \sum_{i:h_i\geq 0} h_i x_i + \sum_{i:h_i<0} h_i x_i \\ &= \sum_{i:h_i\geq 0} h_i x_i + \sum_{i:h_i<0} (-h_i)(-x_i) \\ &= \sum_{i:h_i\geq 0} h_i \hat{x}_i + \sum_{i:h_i<0} (-h_i)(\hat{x}_{i+d}),\end{aligned}$$

where all coefficients are nonnegative in the above sum.

Therefore any inner product between a coefficient vector and the input $\mathbf{x}$ can be written as an inner product between a nonnegative coefficient vector and the expanded input $\hat{\mathbf{x}}$. Therefore, without loss of generality, we can limit all of the weights between layers to be nonnegative, and thus the neural network to be input convex. Note that in optimization problems, we need to enforce consistency in $\hat{\mathbf{x}}$ be including (C.5) as a constraint. However, this is a linear equality constraint, which maintains the convexity of the optimization problem. □

C.3 Proof of Theorem 2

Proof. The second statement of Theorem 6 directly follows the construction in the proof of Theorem 5, which shows that a maximum of K affine functions can be represent by a K-layer ICNN (with a single ReLU function in each layer). So it remains to show the first statement of Theorem 6.

To show that a maximum of affine functions can require exponential number of pieces to approximate a function specified by an ICNN with K activation functions, consider a network with 1 hidden layer of K nodes and the weights of direct "passthrough" layers are set to 0:

$$f_{ICNN}(\mathbf{x}) = \sum_{i=1}^{K} w_{1i}\sigma(\mathbf{w}_{0i}^T\mathbf{x} + b_i)\,, \tag{C.6}$$

It contains $3K$ parameters: $\mathbf{w}_{0i}$, w_{1i} and b_i, where $\mathbf{w}_{0i} \in \mathbb{R}^d$ and $w_{1i}, b_i \in \mathbb{R}$.

In order to represent the same function by a maximum of affine functions, we need to assess the value of every activation unit $\sigma(\mathbf{w}_{0i}^T\mathbf{x} + b_i)$. If $\mathbf{w}_{0i}^T\mathbf{x} + b_i \geq 0$, $\sigma(\mathbf{w}_{0i}^T\mathbf{x} + b_i) = \mathbf{w}_{0i}^T\mathbf{x} + b_i$; otherwise, $\sigma(\mathbf{w}_{0i}^T\mathbf{x} + b_i) = 0$. In total, we have 2^K potential combinations of piecewise-linear function, including

$$
\begin{aligned}
L_1 &= \left(\sum_{i=1}^{K} w_{1i}\mathbf{w}_{0i}\right)^T \mathbf{x} + \sum_{i=1}^{K} w_{1i}b_i \,, \text{if all } \ \mathbf{w}_{0i}^T\mathbf{x} + b_i \geq 0 \\
L_2 &= \left(\sum_{i=2}^{K} w_{1i}\mathbf{w}_{0i}\right)^T \mathbf{x} + \sum_{i=2}^{K} w_{1i}b_i \,, \text{if } \mathbf{w}_{01}^T\mathbf{x} + b_1 < 0 \text{ and all other } \mathbf{w}_{0i}^T\mathbf{x} + b_i \geq 0 \\
L_3 &= \left(w_{11}\mathbf{w}_{01} + \sum_{i=3}^{K} w_{1i}\mathbf{w}_{0i}\right)^T \mathbf{x} + w_{1i}b_i + \sum_{i=3}^{K} w_{1i}b_i \,, \text{if } \mathbf{w}_{02}^T\mathbf{x} + b_2 < 0 \text{ and other } \mathbf{w}_{0i}^T\mathbf{x} + b_i \geq 0 \\
&\cdots\cdots \,, \\
L_{2^K} &= 0 \,, \text{ if all } \mathbf{w}_{0i}^T\mathbf{x} + b_i < 0.
\end{aligned}
$$

So the following maximum over 2^K pieces is required to represent the single linear ICNN:

$$\max\{L_1, L_2, ..., L_{2^K}\}.$$

□

C.4 Details on Building Energy Management

C.4.1 Minimizing Electricity Costs

To further demonstrate the potential of our proposed control framework in dealing with different real world tasks, we modify the setting of the building control example in Section 4.2 to a more complicated case. Instead of directly minimize the total energy consumption of building, we aim to minimize the total energy cost of building which subject to a varying time-of-use electrical price λ. The optimization problem in (7) should be re-written as,

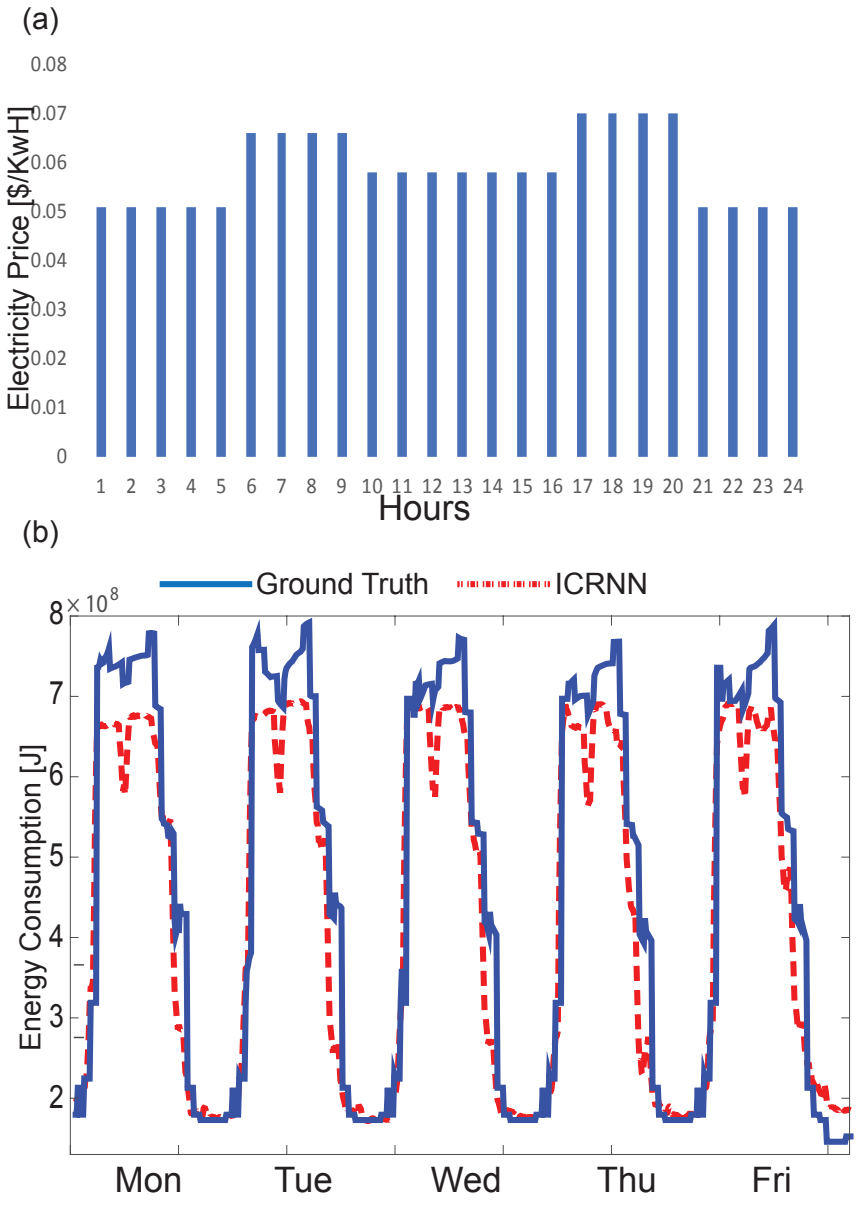


Figure C.3: (a) 24 hour price signal along with (b) optimization results on one-week electricity usage of building using ICRNN.

$$\underset{\mathbf{u}_t,...,\mathbf{u}_{t+T}}{\text{minimize}} \quad \sum_{\tau=0}^{T} \lambda_\tau \cdot f(\mathbf{x}_{t+\tau-n_w}, ..., \mathbf{x}_{t+\tau}) \tag{C.7a}$$

$$\text{subject to} \quad \mathbf{s}_{t+\tau} = g(\mathbf{x}_{t+\tau-n_w}, ..., \mathbf{x}_{t+\tau-1}, \mathbf{u}_{t+\tau}), \forall \tau \tag{C.7b}$$

$$\underline{\mathbf{u}}_{t+\tau} \leq \mathbf{u}_{t+\tau} \leq \overline{\mathbf{u}}_{t+\tau}, \forall \tau \tag{C.7c}$$

$$\underline{\mathbf{s}}_{t+\tau} \leq \mathbf{s}_{t+\tau} \leq \overline{\mathbf{s}}_{t+\tau}, \forall \tau \tag{C.7d}$$

where the objective (C.7a) is minimizing the total energy cost of building in future T steps (T is the model predictive control horizon) subject to time-of-use electricity price λ_τ, and (C.7b) is used for modeling building states, in which $g(\cdot)$ are parameterized as ICRNNs. Same as the previous building control case, we have constraints on both control actions $\mathbf{u}_t$ and system states $\mathbf{s}_t$ are given in (C.7c) and (C.7d). For instance, the temperature set points as well as real measurements should not exceed user-defined comfort regions. In Fig. C.3 we visualize our model flexibility by using Seattle's Time-of-Use (TOU) price from Seattle City Light [1], and minimizing one week's electricity bills. We could see ICRNN capture the long term relationships between control variables and final costs, and raise the energy consumption during off-peak price a little, but reduce the energy consumption during peak hours.

C.4.2 Control Constraints Effects

In Fig. C.4 we add one more comparison on the control constraints effects on the final control performance by using ICRNN. Interestingly, with different set point constraints, the ICRNN finds similar solutions for off-peak electricity usage, which may correspond to necessary energy consumptions, such as lightning and ventilation. Moreover, when we set no constraints on the system, it would cut down more than 80% of total energy during peak hours.

[1] http://www.seattle.gov/light/

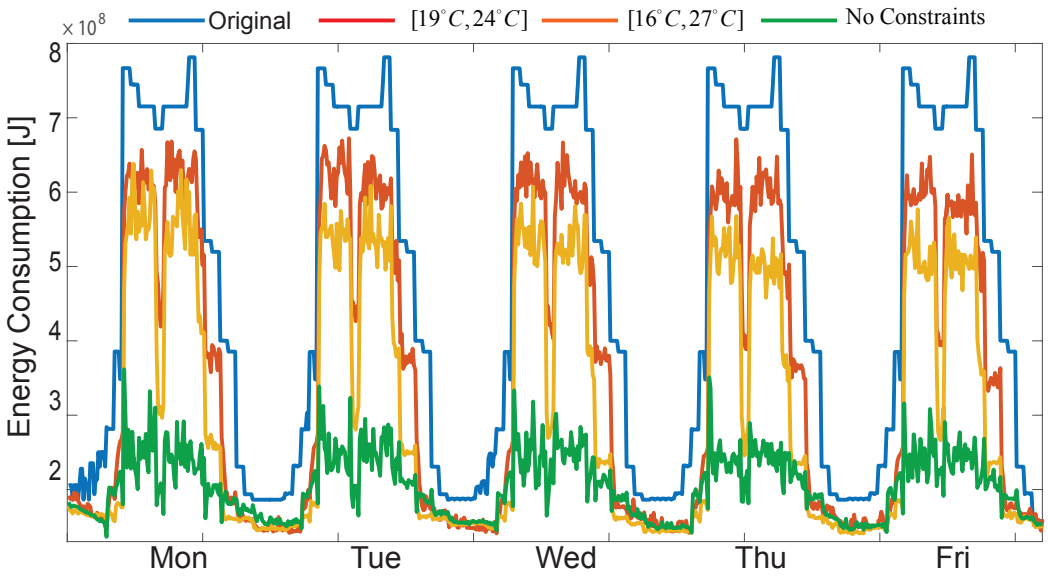


Figure C.4: Results on one-week electricity usage of building using input convex neural network control method based upon different control constrains.

Appendix D

PROOF OF THEOREMS FOR CHAPTER 5

Here we provide the detailed proof for the three major lemmas in Chapter 5.

D.1 Proof of Lemma 3

Without loss of generality, we can consider player 1. Fix the other player's choices of θ's. Define the random variable $Y = \sum_{i=2}^{N}(\theta_i + X_i)^+$. We first prove that it is never beneficial for player 1 to set θ_1 to a value larger than $y_{\max}$. Consider the derivative of J_1 with respect to θ_1

$$\begin{aligned} g_1 &= \frac{\partial}{\partial \theta_1} E\Big[p\Big((\theta_1 + X_1)^+ + Y\Big)(\theta_1 + X_1)^+ - C_1\Big((\theta_1 + X_1)^+\Big)\Big] \\ &= E\big[1(\theta_1 + X_1 \geq 0)\{p'(\theta_1 + X_1 + Y)(\theta_1 + X_1) \\ &\quad + p\Big((\theta_1 + X_1)^+ + Y\Big) - C_i'(\theta_1 + X_1)\Big\}\Big], \end{aligned} \tag{D.1}$$

where $1(\cdot)$ is the indicator function. We want to show that if $\theta_1 \geq y_{\max}$, the derivative is negative. The last term $-E[1(\theta_1 + X_1 \geq 0)C_i'(\theta_1 + X_1)]$ is negative because C_i is strictly increasing. Now consider the first two terms, and let f_Y be the density of Y,

$$\begin{aligned} &E[1(\theta_1 + X_1 \geq 0)\{p'(\theta_1 + X_1 + Y)(\theta_1 + X_1) + p(\theta_1 + X_1 + Y)\}] \\ &= \int_0^\infty \int_0^\infty 1(\theta_1 + x_1 \geq 0)\{p'(\theta_1 + x_1 + y)(\theta_1 + x_1) + p(\theta_1 + x_1 + y)\} f_1(x) f_Y(y) dx dy \\ &\overset{(a)}{=} \int_0^{y_{\max}} \int_{-\theta_1}^{y_{\max} - y - \theta_1} [p'(\theta_1 + x_1 + y)(\theta_1 + x_1) + p(\theta_1 + x_1 + y)] f_1(x) f_Y(y) dx dy \\ &\overset{(b)}{=} \int_0^{y_{\max}} \int_0^{y_{\max} - y} [p'(x_1' + y)(x_1') + p(x_1' + y)] \cdot f_1(x' - \theta_1) f_Y(y) dx dy, \end{aligned} \tag{D.2}$$

where (a) follows from assumption (A1) and (b) from a change of variable from x_1 to $x_1' = \theta_1 + x_1$. Next we show that for any given y, $\int_0^{y_{\max} - y} p'(x_1' + y)(x_1') + p(x_1' + y) dx = 0$.

Using the integration by parts on the first term, denoting $\bar{y} = y_{\max} - y$, we have

$$\int_0^{y_{\max}-y} p'(x_1'+y)(x_1') + p(x_1'+y)dx = p(x_1'+y)x_1'|_{x_1'=0}^{x_1'=\bar{y}} - \int_0^{\bar{y}} p(x_1'+y) + \int_0^{\bar{y}} p(x_1'+y) = 0.$$

By assumption (A1), $p'(x_1'+y)(x_1') + p(x_1'+y)$ is positive at $x_1' = 0$. Therefore, it must undergo a sign change from positive to negative. However, by the unimodality assumption, $\theta_1 \geq y_{\max}$, $f_1(x_1 - \theta_1)$ is an increasing function on the interval $x_1 \in [0, y_{\max} - y]$. Therefore, $\int_0^{y_{\max}-y}[p'(x_1'+y)(x_1') + p(x_1'+y)]f_1(x' - \theta_1)dx < 0$ for all y and this proves that player 1 would never choose θ_1 to be larger or equal to $y_{\max}$.

Now we show that there is a lower bound on θ_1. The partial derivative of g_1 with respect to θ_j is

$$\frac{\partial g_1}{\partial \theta_j} = E\left[1(\theta_1 + X_1 \geq 0, \theta_j + X_j \geq 0) \cdot \left\{p''\left(\sum_{j=1}^{N}(\theta_j + X_j)^+\right) \cdot (\theta_i + X_i) + p'\left(\sum_{j=1}^{N}(\theta_j + X_j)^+\right)\right\}\right] < 0$$

where the inequality follows from p is strictly decreasing and concave. Similar calculations can be used to show that all cross partials are negative. Therefore, player 1 should decrease θ_1 as other players increase their parameters. From the first part of the proof, suppose all other players choose $y_{\max}$ as their play. Even at this choice, g_1 still becomes positive for negative enough θ_1's, therefore implying that the choice of θ_1 is lower bounded by some real number $\underline{\theta}_1$.

D.2 Proof of Lemma 4

Lemma 3 shows that the action space is convex and compact. Let $\boldsymbol{G}$ denote the Hessian of the game, so

$$G_{ij} = \frac{\partial^2 J_i}{\partial \theta_i \partial \theta_j} = \frac{\partial g_i}{\partial \theta_j}.$$

Focusing on the diagonal terms, we have

$$G_{ii} = \frac{\partial^2 J_i}{\partial \theta_i^2} = E\left[1(\theta_1 + X_1 \geq 0) \cdot \left\{ p''\left(\sum_{j=1}^{N} (\theta_j + X_j)^+\right) \right.\right.$$
$$\left.\left. \cdot(\theta_i + X_i) + 2p'\left(\sum_{j=1}^{N} (\theta_j + X_j)^+\right) - C_i''(\theta_i + X_i) \right\}\right] < 0, \tag{D.3}$$

which is negative by assumptions (A1) and (A2). A game is said to be a *concave N-player game* [123] if $G_{ii} < 0, \forall i$ and the action space is convex and compact. Therefore, it is a concave N-player game. The following proposition is given in [123] as a sufficient condition to show when gradient-based algorithms converge to Nash equilibriums:

Proposition 7. *Let $\boldsymbol{G}$ denote the Hessian of a concave N-player game. If $\boldsymbol{G}^T + \boldsymbol{G}$ is negative definite over the space of actions, there is a unique Nash equilibrium and the policy gradient dynamics approach it exponentially quickly for all initializations.*

Using Proposition 7, it suffices for us to show that the negative definiteness of $\boldsymbol{G}^T + \boldsymbol{G}$ under the stochastic Cournot game. The main challenge of proving the negative definiteness lies in the *expectation* term, and we need to relate the properties of the Hessian of a function to its expectation. The following proposition tackles the aforementioned challenge and relates the Hessian property to its expectation.

Proposition 8. *Let $f : \mathbb{R}^N \to \mathbb{R}$ be a continuous function and suppose that the first and second order partial derivatives exist for all points except possibly for a set of measure 0. Let $\boldsymbol{G}$ be the Hessian of f, whenever it exists. Now consider the function $\hat{f} : \mathbb{R}^N \to \mathbb{R}$, where $\hat{f}(y) = E_{\boldsymbol{X}} f(y + \boldsymbol{X})$ and $\boldsymbol{X}$ is random vector in $\mathbb{R}^N$, with continuous and bounded density function and infinite support. Let $\hat{\boldsymbol{G}}$ be the Hessian of $\hat{f}$. Then: i) If $\boldsymbol{G}^T + \boldsymbol{G}$ is negative semidefinite at all points where $\boldsymbol{G}$ exists, then $\hat{\boldsymbol{G}}^T + \hat{\boldsymbol{G}}$ is negative semidefinite. ii) If $\boldsymbol{G}^T + \boldsymbol{G}$ is negative definite for a set of measure larger than 0, then $\hat{\boldsymbol{G}}^T + \hat{\boldsymbol{G}}$ is negative definite.*

Proof. The proof of this proposition is straightforward. By the assumption on the random vector $\boldsymbol{X}$, we can switch the order of differentiation and the expectation. In addition, the

density being continuous allows us to ignore the points where $\boldsymbol{G}$ does not exist. Then

$$\begin{aligned}&\boldsymbol{v}^T(\boldsymbol{G}^T(\boldsymbol{y})+\boldsymbol{G}(\boldsymbol{y}))\boldsymbol{v}\\&=E_{\boldsymbol{X}}\Big[\boldsymbol{v}^T(\boldsymbol{G}^T(\boldsymbol{y}+\boldsymbol{X})+\boldsymbol{H}(\boldsymbol{y}+\boldsymbol{X}))\boldsymbol{v}\Big]\leq 0,\end{aligned}$$

for any $\boldsymbol{v}$. Now suppose $\boldsymbol{G}^T(\boldsymbol{y}+\boldsymbol{x})+\boldsymbol{G}(\boldsymbol{y}+\boldsymbol{x}))$ is negative definite for set of positive measure, then by the continuity of the density function, $\boldsymbol{v}^T(\boldsymbol{G}^T(\boldsymbol{y})+\boldsymbol{G}(\boldsymbol{y}))\boldsymbol{v}<0$ for all nonzero $\boldsymbol{v}$ and $\hat{\boldsymbol{G}}^T+\hat{\boldsymbol{G}}$ is negative definite. □

Let $f_i(\boldsymbol{x})=p(\sum_i x_i^+)x_i^+$, which is continuous and twice differentiable except for a measure zero set on $\mathbb{R}^N$. Since $J_i=E[f_i(\boldsymbol{\theta}+\boldsymbol{X})]$, we need to show $f=\Big[f_1\dots f_N\Big]$ satisfies the condition of Proposition 8. Given a vector $\boldsymbol{x}$, without loss of generality, assume that $x_1,\dots,x_k\geq 0$ and $x_{k+1},\dots,x_N<0$. The second order derivatives of f are,

$$\begin{aligned}\frac{\partial^2 f_i}{\partial x_i\partial x_j}=&1(x_i\geq 0)\cdot\\&\begin{cases}p''(\sum_l x_l^+)x_i+2p'(\sum_l x_l^+)-C''(x_i), i=j\\1(x_j>0)\Big(p''(\sum_l x_l^+)x_i+p'(\sum_l x_l^+)\Big), i\neq j\end{cases}.\end{aligned}$$

Because of the indicator on both $x_i\geq 0$ and $x_j\geq 0$, the Hessian is only nonzero for the *upper left block*. In this block, we have $\forall i,j\leq k$,

$$\frac{\partial^2 f_i}{\partial x_i\partial x_j}=\begin{cases}p''(\sum_{l=1}^k x_l)x_i+2p'(\sum_{l=1}^k x_l)-C''(x_i), i=j\\p''(\sum_{l=1}^k x_l)x_i+p'(\sum_{l=1}^k x_l), i\neq j,\end{cases}.$$

When the price function is linear, the second order derivative term vanishes, i.e. $p''(\sum_{l=1}^k x_l)=0$. Therefore, we have,

$$\frac{\partial^2 f_i}{\partial x_i\partial x_j}=\begin{cases}2p'(\sum_{l=1}^k x_l)-C''(x_i)\text{ if } i=j, i,j\leq k\\p'(\sum_{l=1}^k x_l)\text{ if } i\neq j, i,j\leq k\end{cases}$$

Now we can write $\boldsymbol{G}$ as $\boldsymbol{G}_1 + \boldsymbol{G}_2 + \boldsymbol{G}_3$ where

$$\boldsymbol{G}_1 = \left[\begin{array}{ccc|ccc} \begin{bmatrix} p'(\sum_{l=1}^k x_l) & \cdots & p'(\sum_{l=1}^k x_l) \\ \vdots & \ddots & \vdots \\ p'(\sum_{l=1}^k x_l) & \cdots & p'(\sum_{l=1}^k x_l) \end{bmatrix} & & & \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \\ \hline & \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} & & \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \end{array}\right] \tag{D.4}$$

Both $\boldsymbol{G}_2, \boldsymbol{G}_3$ are diagonal matrices. For $\boldsymbol{G}_2$, it has the i'th component being $p'(\sum_{l=1}^k x_l)$ for $i \leq k$ and 0 for $i > k$. Since X_i have infinite support, there exists cases where $k = N$ (all player sample non-negative actions) for a *measure larger than 0 set*, in which $\boldsymbol{G}_2$ is *negative definite*. For $\boldsymbol{G}_3$, it has the i'th component $-C''(x_i) \leq 0$ for $i \leq k$ and 0 for $i > k$, thus it is negative semi-definite. Therefore, it suffices for us to show the negative semi-definess of $\boldsymbol{G}_1$.

The eigenvalues of $\boldsymbol{G}_1$ in Eq. (D.4) are the combination of eigenvalues of the upper left and lower-right matrices. Eigenvalues of the upper left matrix are $kp'(\sum_{l=1}^k x_l) < 0$ and 0 ($k-1$ repeats), and eigenvalues of the lower right are all zeros. Thus $\boldsymbol{G}_1$ is negative semi-definite.

Therefore, there exists an measure nonzero set of actions, such that $\boldsymbol{G} = \boldsymbol{G}_1 + \boldsymbol{G}_2 + \boldsymbol{G}_3$ is negative definite. By Proposition 8, we have the Hessian of $(J_1, ..., J_N)$, that is $\hat{\boldsymbol{G}}$ is negative definite.

D.3 Proof of Lemma 5

Now, consider the two-player Cournot games with general price function $p(\cdot)$ under assumption (A1) and (A2). There are four cases considering the positiveness of x_1 and x_2.

a) $x_1, x_2 \geq 0$:

$$\boldsymbol{G}_a = \begin{cases} p''(x_1+x_2)x_i + 2p'(x_1+x_2) - C''(x_i), i = j, \\ p''(x_1+x_2)x_i + p'(x_1+x_2), i \neq j, \end{cases}$$

b) $x_1 < 0, x_2 \geq 0$:

$$\boldsymbol{G}_b = \begin{bmatrix} 0 & 0 \\ 0 & p''(x_1+x_2)x_2 + 2p'(x_1+x_2) \end{bmatrix}$$

c) $x_1 \geq 0, x_2 < 0$:

$$\boldsymbol{G}_c = \begin{bmatrix} p''(x_1+x_2)x_1 + 2p'(x_1+x_2) & 0 \\ 0 & 0 \end{bmatrix}$$

d) $x_1 < 0, x_2 < 0$:

$$\boldsymbol{G}_d = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The game Hessian matrix thus follows,

$$\begin{aligned} \hat{\boldsymbol{G}} = E[&1(x_1, x_2 \geq 0)\boldsymbol{G}_a + 1(x_1 < 0, x_2 \geq 0)\boldsymbol{G}_b \\ &+1(x_1 \geq 0, x_2 < 0)\boldsymbol{G}_c + 1(x_1 < 0, x_2 < 0)\boldsymbol{G}_d], \end{aligned} \tag{D.5}$$

$\boldsymbol{G}_a$ is a strictly diagonally dominant matrix since the magnitude of the diagonal entry is strictly larger than the *sum* of the magnitudes of all the other (non-diagonal) entries in each row, i.e., $|p''(x_1+x_2)x_i + 2p'(x_1+x_2) - C''(x_i)| > |p''(x_1+x_2)x_i + p'(x_1+x_2)|, \forall i$. Given that $\boldsymbol{G}_b, \boldsymbol{G}_c, \boldsymbol{G}_d$ are all diagonally dominant matrices, $\hat{\boldsymbol{G}}$ in Eq. (D.5) is a strictly diagonally dominant matrix . Therefore, the eigenvalues of matrix $\hat{\boldsymbol{G}}$ are all in the left–half plane (i.e., the real parts of eigenvalues are negative) by the Gershgorin circle theorem [131]. Proposition 4 in [132] showed that when all eigenvalues of the Hessian are in the open left–half plane, then the Nash equilibrium is an exponentially stable fixed point of the dynamical system generated by the gradient descend algorithm.