

**LEARNING-BASED SITUATIONAL AWARENESS, DECISION MAKING, AND
FLEXIBILITY AGGREGATION FOR POWER DISTRIBUTION SYSTEMS
WITH UNCERTAINTY**

A Dissertation
Submitted to the
Temple University Graduate Board

In Partial Fulfillment
of the Requirements for the Degree
DOCTOR OF PHILOSOPHY

by

Shengyi Wang

May 2022

Examining Committee Members:

Dr. Liang Du, Advisory Chair, Electrical and Computer Engineering, Temple University
Dr. Fauzia Ahmad, Electrical and Computer Engineering, Temple University
Dr. Chang-Hee Won, Electrical and Computer Engineering, Temple University
Dr. Yan Li, Electrical Engineering and Computer Science, Pennsylvania State University

ABSTRACT

The ever-growing penetration of distributed energy resources in both the generation and demand-side brings environmental benefits and technical challenges to electric power distribution systems. Specifically, due to the inherently intermittent nature of renewable energy resources and the invisible behaviors of customers in electricity use, there exists a high level of uncertainty, which has significantly threatened the stable, secure, and dedicated operation of distribution systems. The conventional operation strategies tend to be model-driven based on offline studies or historical experiences, leading to an over-conservative or risky operation solution especially when the system encounters considerable uncertainty. That is, such a deterministic solution is highly difficult to adapt to the various unknown system operating conditions. Therefore, it is imperative to find a proper operation strategy for distribution systems under uncertainty. With the high volume of the real-time measurement data available to the distribution system operator and the huge success of Machine Learning (ML) technologies in the data-intensive industry, it is promising to marriage the knowledge representation and reasoning power of ML to analyze, understand and reveal the potential effects of uncertainty from data itself, finally solving optimal operation problems under uncertainty more efficiently and accurately. This dissertation aims at developing learning-based approaches for three representative and challenging operation problems to have accurate situational awareness, optimal decision making, and efficient flexibility aggregation under uncertainty. The problems are focused on identifying the behind-the-meter Electric Vehicle charging load, scheduling energy storage systems for voltage regulation, and estimating a feasible active-reactive power flexibility region for capacity support.

DEDICATION

To my beloved family.

ACKNOWLEDGMENTS

Time goes so fast, and I will realize one of my dreams to become a Doctorate. Recall this meaningful and challenging journey, and it was bittersweet. Since I started my Ph.D. program, I have met numerous people, all of whom I am incredibly fortunate to have in my life.

First and foremost I would like to express profound gratitude to my supervisor and mentor, Dr. Liang Du. He is a friendly and amiable professor who treats me as his good friends than students. During my Ph.D. study, he always encourages and supports me. Whatever difficulties I encounter, I can talk with him, and get instant and useful help. He not only taught me the required knowledge, skills, and attitudes for being a qualified researcher but also let me know more about myself, which drives me to become a successful man in the future. Without him, it would be impossible for me to have a promising future.

Secondly, I would like to thank my dissertation committee members: Dr. Fauzia Ahmad, Dr. Chang-Hee Won, and Dr. Yan Li for their valuable suggestions to my dissertation. I also would like to thank my group mates: Dongsun Sun, Kaitlyn Sitch, Abtahi Reza, and Zhenyu Zhao for a cherished time spent together in the lab.

Thirdly, I would like to thank my collaborators, especially Dr. Di Shi, Dr. JiaJun Duan, Dr. HongBo Sun, and Dr. Song Zhang for their encouragement and supports during my internship. It was a great privilege to work with them.

Fourthly, I would like to thank my friends: Xiangyu Xu, Zeyu Mao, Yuanqi Gao, Jianqiao Huang, and others who accompanied me in the past few years. It is unforgettable for me to discuss research problems, play basketball in the Pearson Hall, work out in the gym, travel together, etc. They make my Ph.D. study life more colorful and energetic.

Finally, I cannot begin to express my gratitude to my parents. Without their tremendous understanding and encouragement in the past few years, it would be impossible for me to complete my study.

TABLE OF CONTENTS

ABSTRACT	ii
DEDICATION	iii
ACKNOWLEDGMENTS	iv
LIST OF TABLES	ix
LIST OF FIGURES	x
LIST OF ACRONYMS	xiii
CHAPTER 1:INTRODUCTION	1
1.1 Distribution System With Uncertainties	1
1.2 Modern Artificial Intelligence Technologies	2
1.3 Motivations and Objectives	3
1.4 Review of Residential EV Charging Load Profile Identification	5
1.5 Review of Voltage Regulation with High PV Penetration	8
1.6 Review of Power Flexibility Aggregation for TSO-DSO Interaction	10
1.7 Dissertation Organization	13
CHAPTER 2:LITERATURE REVIEW	15

2.1	Deep Neural Networks	15
2.2	Deep Reinforcement Learning	17
2.2.1	Principles of RL Problems	17
2.2.2	DRL	24

**CHAPTER 3:LEARNING-BASED SITUATION AWARENESS UNDER UN-
CERTAINTY 27**

3.1	Problem Formulation	28
3.1.1	Definitions	28
3.1.2	EV Charging Profile Identification as NILM	29
3.2	Benchmark Algorithm: HMMs	31
3.2.1	Applicability	31
3.2.2	Implementation Details	31
3.3	The Proposed DGM: Representation and Parameter Learning	32
3.3.1	Representation Layer Embedded Markov Chain	33
3.3.2	Probability Distribution Approximation in DGM	35
3.3.3	Supervised Learning in DGM	40
3.4	The Proposed DGM: Exact Inference	42
3.5	Numeric Results	46
3.5.1	Experiment Setup and Evaluation Metrics	46
3.5.2	EV Charging Status Classification	47
3.5.3	EV Charging Profile Elements Analyses	50
3.6	Conclusion	51

CHAPTER 4:LEARNING-BASED DECISION MAKING UNDER UNCERTAINTY 54

4.1	Modeling of Multi-phase Unbalanced LV Distribution Network	55
4.2	ESS Scheduling for Voltage Regulation	58
4.2.1	Total Daily System Voltage Regulation Cost	58
4.2.2	ESS-based Voltage Regulation Problem Formulation	61
4.3	Real-time Voltage Regulation as MDP	63
4.4	Learn-to-schedule Feedback Control Framework Via SAC	66
4.5	Simulation Results	72
4.5.1	Customized 6-Bus Test System	74
4.5.2	Modified IEEE 34-Bus Test System	75
4.6	Conclusion	79

CHAPTER 5:LEARNING-BASED FLEXIBILITY AGGREGATION UNDER UNCERTAINTY 80

5.1	System Model	81
5.1.1	Device Model	81
5.1.2	Network Model	86
5.1.3	Uncertainty Propagation Model	89
5.2	Flexibility Region Construction	89
5.2.1	Non-Feeder-Level Power Flexibility Region	90
5.2.2	Feeder-Level Power Flexibility Region	92
5.3	Inner Approximation Framework	96
5.3.1	Problem Formulation	96
5.3.2	Data-driven Solution	98

5.4	Numerical Simulation	101
5.4.1	Deterministic Region Approximation	104
5.4.2	Deterministic Region Approximation Under Uncertainty	107
5.5	Conclusion	108
CHAPTER 6: CONCLUSION AND FUTURE WORK		112
6.1	Conclusion	112
6.2	Future Work	114
REFERENCES		115
PUBLICATIONS		123

LIST OF TABLES

3.1	The proposed DGM architecture and parameter settings	40
3.2	Performance comparison using five-fold cross-validation and non-transfer learning setting	49
3.3	Performance Comparison Using Transfer Learning Setting	51
5.1	Parameters of the device model	102

LIST OF FIGURES

1.1	Illustration of the distribution system with various distributed energy resources	1
1.2	Illustration of the proposed learning-in-the-loop distribution system operation framework responding to a high level of uncertainty	4
3.1	Overview of EV charging profile identification. (a): a sample aggregated power consumption profile; (b): its corresponding EV charging status (charging started at t_s and ended at t_e); and (c) its corresponding EV charging profile.	28
3.2	The general procedure of the proposed identification algorithm under the architecture of NILM can be divided into four steps.	30
3.3	Graphical illustration of how aggregated profiles and EV charging profiles are represented by an HMM (black) vs. the proposed representation-layer-embedded Markov model (blue), which introduces an additional layer of representations.	32
3.4	Illustration of a sample standardized aggregated power consumption profile (left) and its corresponding representation layer (right).	33
3.5	An example likelihood function $p(x_t y_t)$ in HMM, where the learned likelihood distributions for the two distinctive scenarios $p(x_t y_t = 0)$ and $p(x_t y_t = 1)$ are represented by the blue and red lines, respectively.	34
3.6	Demonstration of a CNN layer, which consists of three sequential operations: Conv1D, UpSampling1D, and MaxPooling1D.	38
3.7	Visualization of feature maps extracted from convolutional layers with sixteen filters.	39
3.8	Implementation flowchart of the minibatch training in DGM, including both forward and backward propagation.	43

3.9	Example demonstration of how to calculate $F_c(t = 3, y_t = 0)$ where the red sign highlights the possible minimum cost paths (the directed edges) from the last time step ($t = 2$) to the current time step ($t = 3$).	44
3.10	Selected typical examples of inference results including (a) once-charging, (b) twice-charging in day and night, (c) twice-charging in two nights, and (d) multiple-charging, respectively. Each row shows the aggregated power consumption profile, the probability of an EV at ON, the optimal cost over time steps t and $t - 1$ given y_t , the measured EV charging status profile, and the inferred EV charging status profile, respectively.	45
3.11	Illustration of convergence of the training (black line) and validation (blue line) losses as the epochs increase during training.	46
3.12	Performance comparison using five-fold cross-validation and non-transfer learning settings.	49
3.13	The analysis of some example EV charging profile elements including start charging time, end charging time, duration, and charging times for four representative houses in terms of their F1 values, where the blue and red lines are the measured and classified distributions, respectively.	52
4.1	Illustration of an MDP for ESSs based voltage regulation	64
4.2	Overview of the proposed learn-to-schedule feedback controller	70
4.3	Schematic diagram of (a) the customized 6-bus and (b) modified IEEE 34-bus systems where the black, red and blue line correspond the line with three phases, only a phase and only b phase, respectively.	73
4.4	Illustration of neural network architecture of actor and critic network where blue layer is a concatenation layer	74
4.5	Illustration of learning performance (a) during training and voltage regulation performance (b) during execution where the dash line is a baseline	75
4.6	Illustration of (a) a sample daily profile with load, PV, and ESSs; (b) active power injection profile at bus 5; (c) ESS SOC profiles; and (d) voltage profile at bus 5 where dash lines represent upper/lower bounds.	76
4.7	Voltage dynamics of phase a bus 5 (a) w/o and (b) w control	77
4.8	Illustration of learning performance (a) during training and voltage regulation performance (b) during execution where the dash line is a baseline	77

4.9	Illustration of (a) a sample daily profile with load, PV, and ESS; (b) active power injection profile at bus 11; (c) ESS SOC profiles; and (d) voltage profiles at bus 11, with ESSs at bus 7, 9, 10, 11, 13, 23, 32, 33 are relabeled as ESS 1, 2, 3, 4, 5, 6, 7, 8, respectively	78
4.10	Voltage dynamics of phase a bus 11 (a) w/o and (b) w control.	79
5.1	Illustration of the power region at time t for (a): a PV inverter and (b): an ESS where the area depicted by a blue line is the original nonlinear power region, and the area colored by yellow is the polygonal power region.	83
5.2	Illustration of the proposed inner approximation framework.	97
5.3	Illustration of the artificially-designed system.	102
5.4	Illustration of PV and uncontrollable load data.	102
5.5	Illustration of the approximated deterministic region. The points in the red rectangle region can provide the capacity support to the transmission system while the ones in the blue rectangle region require the capacity support from the transmission system.	105
5.6	Illustration of the flexibility metrics: γ_p^t , γ_q^t and γ_s^t in the entire aggregation period.	106
5.7	Illustration of the effect of PV penetration level on the optimization results.	107
5.8	Illustration of the effect of PV penetration level on the approximated region.	109
5.9	Illustration of the approximated deterministic region without considering forecast error uncertainty.	110
5.10	Illustration of the robust deterministic approximated region.	111
5.11	Illustration of the effect of ϵ on the optimality and robustness.	111

LIST OF ACRONYMS

AI	Artificial Intelligence
CNN	Convolutional Neural Network
DDPG	Deep Deterministic Policy Gradient
DER	Distributed Energy Resource
DGM	Deep Generative Model
DL	Deep Learning
DNN	Deep Neural Network
DP	Dynamic Programming
DQN	Deep Q-network
DRL	Deep Reinforcement Learning
DSO	Distribution System Operator
ELBO	Evidence Lower Bound
ESS	Energy Storage System
EV	Electric Vehicle
FHMM	Factorial Hidden Markov Model
HMM	Hidden Markov Model
HVAC	Heating, Ventilation and Air-Conditioning
LV	Low Voltage
MAP	Maximizing <i>a Posterior</i>
MDP	Markov Decision Process
ML	Machine Learning
MPC	Model Predictive Controller

NILM Non-Intrusive Load Monitoring
OLTC On-Load Tap Changer
PV Photovoltaic
QP Quadratic Programming
ReLU Rectified Linear Unit
RL Reinforcement Learning
SAC Soft Actor Critic
SGD Stochastic Gradient Decent
SOC State of Charge
SVR Step Voltage Regulator
TSO Transmission System Operator
UDS Unbalanced Distribution Network
V2G Vehicle-to-Grid
VAE Variational Autoencoder
WT Wind Turbine

CHAPTER 1

INTRODUCTION

In this chapter, an overview of the research background is first provided, and then the motivations and objectives are presented. After that, some literature reviews for three representative and challenging research problems are conducted. Finally, the dissertation organization is outlined.

1.1 Distribution System With Uncertainties

The main objective of electric power distribution systems is to deliver energy from the transmission system to end-users at medium voltage and low voltage. In the last two decades, some tremendous changes have been witnessed in the distribution system. On one hand, unlike the conventional distribution system, there are various Distributed Energy Resources (DERs), such as Wind Turbines (WTs), Photovoltaics (PVs), Energy Storage Systems (ESSs), microgrids, Electric Vehicle (EV) charging stations, and residential/commercial buildings, as shown in Figure 1.1.

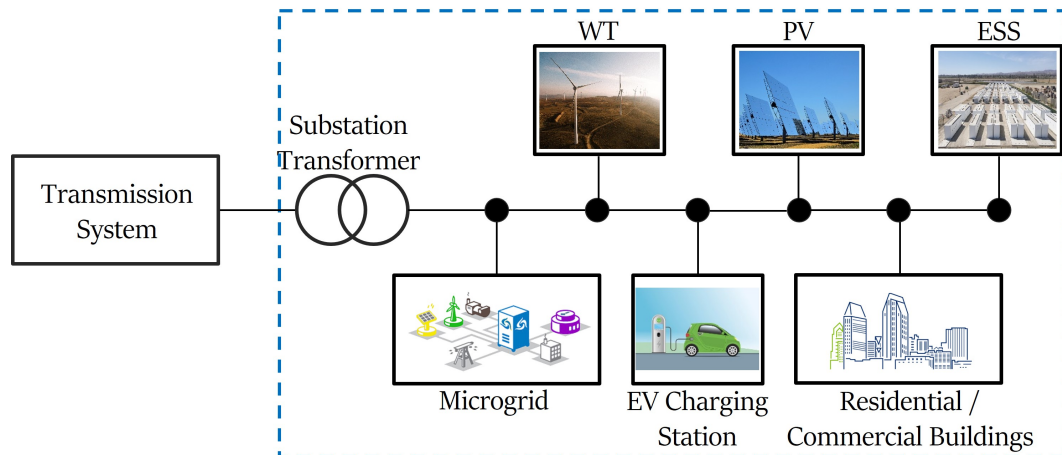


Figure 1.1: Illustration of the distribution system with various distributed energy resources

The future trend is that DERs will become the main part of energy producers and consumers. For example, according to Annual Energy Outlook 2021 [1] released from U.S. Energy Information Administration, renewable electric generating technologies will account for almost 60 of the approximately 1,000 gigawatts of cumulative capacity additions from 2020 to 2050, and the solar generation will be responsible for almost three-quarters of the increase in renewable generation. On the other hand, a massive number of smart meters are installed in the distribution system, which provides detailed data required for accurate information and automated decision support.

With the changes in current distribution systems, there are many uncertainty sources. On the side of generation, the uncertainty mainly comes from the renewable energy sources, which in nature are stochastic and intermittent depending on the weather conditions. On the side of demand, the uncertainty is the behaviours of customers in electricity use, especially with more and more prosumers or active stakeholders, e.g., EVs and ESSs, emerging in the distribution system. These two types of uncertainty result in great fluctuation and unpredictability, and impose challenges to the operation of distribution systems. Moreover, the insufficient real meter measurements and the measurement error may also create uncertainty, which has a further negative impact on the grid operation. The uncertainty has to be addressed in order to prevent the power quality from degrading and maintain the reliability and stability of distribution systems.

1.2 Modern Artificial Intelligence Technologies

Artificial Intelligence (AI) is defined as intelligence demonstrated by machine, as opposed to natural intelligence displayed by animals including humans. Modern AI technologies tend to refer to learning based computational intelligence, such as ML, Deep Learning (DL), and Reinforcement Learning (RL), which has gained the huge success in multiple research fields such as engineering, medicine and economics. ML is a branch of AI, which refers to the automated detection of meaningful patterns in datasets. DL can be

considered as a sub-class of ML, which is focused on neural network based methods with many nonlinear layers for pattern recognition. Unlike ML and DL, RL is a sub-field of ML for decision making by trial and error.

1.3 Motivations and Objectives

As mentioned before, the high penetration of DERs as well as the inaccurate and insufficient data measurement can introduce much uncertainty, which has an adverse impact on the operation of distribution systems. The traditional operation strategies tend to be model-driven based on offline studies or historical experiences, leading to a over-conservative or risky operation solution especially when a high level of uncertainty exists in the system. That is, they have the limited capabilities to adapt to various unknown system operating conditions. Therefore, it is imperative to find a proper operation strategy for distribution system with uncertainty.

Nowadays, with the high volume of the real-time measurement data available to the Distribution System Operator (DSO) and the significant success of modern AI technologies in the data-intensive industry, it could be promising to utilize ML or other advanced learning technologies to analyze, understand and reveal the potential effects of unknown operating conditions from the data, finally addressing optimal operation problems under uncertainty in a more efficient and accurate way. As shown in Figure 1.2, a learning-in-the-loop distribution system operation framework is proposed to respond to a high level of uncertainty. At each operation interval (depending on the specific task), a closed-loop procedure is implemented. First, the grid operation data is collected from the smart meters. Second, the AI agent learns from the data and provides useful information and insights to the DSO. Third, with the assistance or guidance of the AI agent, the DSO can properly schedule the distribution assets (i.e., controllable DERs or other devices) to maintain the reliability and efficiency of the distribution system while mitigating the adverse effect of uncertainty sources.

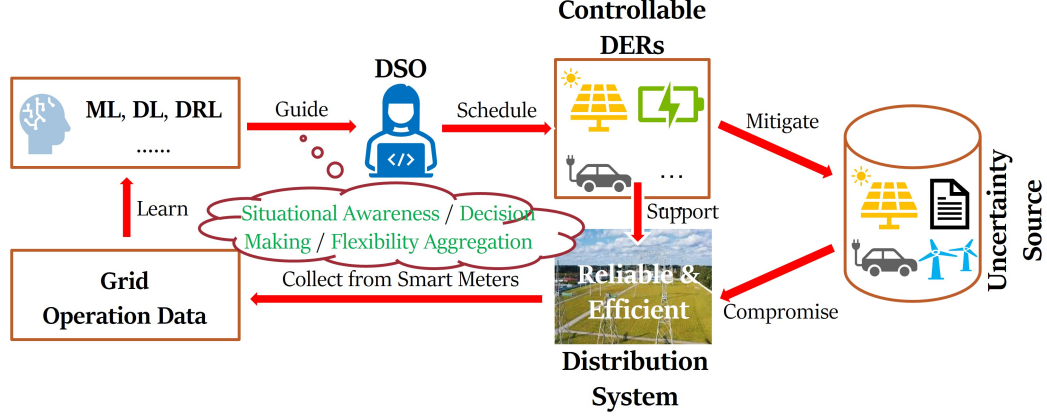


Figure 1.2: Illustration of the proposed learning-in-the-loop distribution system operation framework responding to a high level of uncertainty

In this dissertation, centered around the above-mentioned framework, the goal is to develop learning-based methods for three representative and challenging operation problems: situational awareness, decision making, and flexibility aggregation, specified as follows.

- Insufficient meter installment renders the distribution system less visible, leading to inaccurate grid models and inadequate situational awareness. This dissertation is concentrated on residential EV charging load profile identification. For the consideration of installment cost, normally each household has only one meter to measure total energy consumption. The residential EV charging load as an important uncertain source attracts more and more attention of the DSO for modeling the accurate EV charging load profile. Therefore, the first task is to develop a learning-based method to estimate the behind-the-meter EV charging load and enhance the situational awareness under uncertainty. The relative literature reviews can be found in Section 1.4.
- High PV penetration can degrade the power quality especially on voltage profiles due to its intermittent nature. Moreover, with its large-scale integration, it is difficult to model uncertainty or capture the system dynamics and stochastic nature. Although

there exist many deterministic methods to schedule voltage regulation devices, it seems challenging to make an optimal schedule decision in a similar way in an environment with a high level of uncertainty. Therefore, the second task is to develop a learning-based method to schedule ESSs for voltage regulation in Low Voltage (LV) networks with high PV penetration and improve the computational performance of decision making under uncertainty. The relative literature reviews can be found in Section 1.5.

- Apart from the engagement in the problems of situational awareness and decision making, it is essential for the DSO to interact with the Transmission System Operator (TSO). The distribution system with controllable DERs can provide great power flexibility to support transmission system operation, i.e., power flexibility aggregation. This dissertation is focused on estimating an approximate and feasible power flexibility region. However, the forecast error uncertainty of PV generation and load has an adverse impact on the ahead-of-time flexibility region estimation, which hinders the efficient and safe implementation of power flexibility aggregation. Therefore, the third task is to develop a learning-based method to find a region that can hedge against the forecast error uncertainty as much as possible. The relative literature reviews can be found in Section 1.6.

1.4 Review of Residential EV Charging Load Profile Identification

The worldwide electricity demand profile is experiencing a paradigm shift with increasing penetration of electrified transportation. In the U.S., it is expected that transportation electrification will drive domestic electricity demand rise through 2050 [2], by when over 2.3 million new light-duty EVs will be sold annually [3]. Across the globe, many major economies have announced their intentions to end the sale of internal combustion engine vehicles [4] within several decades. The impact of high volume of

EVs on power grids has been extensively studied in literature [5]. In general, EVs have been considered as active loads which could provide flexibility in terms of grid services [6] through Vehicle-to-Grid (V2G) modes [7] or transactive controls [8].

In the literature, aggregated EV charging demands are modeled as a stochastic part of the overall load model. However, the uncertainty in individual EV charging profiles (i.e., start charging time, initial State of Charge (SOC), charging power, and charging duration) [9] and traffic conditions [10] makes it difficult to accurately derive real-time EV charging demand models under various scenarios. Therefore, probabilistic distributions are typically assumed. In [11] and [12], the charging start time is represented by the normal distribution. Similarly, a truncated normal distribution is suggested to represent the arriving time and parking time at commercial buildings [13] for EV charging duration. Furthermore, in [14], EV charging duration is assumed to be exponentially distributed. Moreover, the initial SOC is modeled as a random variable under log-normal distribution [9]. However, it is questionable whether these assumptions from locational models can be used in other regions. For example, charging start times in rural residential, urban residential, and commercial districts at different seasons are unlikely to be the same. Therefore, in recent years, pilot projects have been carried out globally to collect and analyze EV charging profiles in the Netherlands [15], U.K. [16], Australia [17], and California [18].

However, most of the historical data is only small-scale and sampled at commercial charging stations. For residential applications, it is costly to (intrusively) install additional sampling devices into existing residential EV chargers and (more importantly) unrealistic to sample and communicate EV charging information to system operators, another recent research effort [19–23] focuses on utilizing widely-available smart meter data to non-intrusively, locally, and reliably estimate EV charging profiles in real-time to preserve privacy and avoid unnecessary investment in additional infrastructure.

To conclude above discussions, it is of great interests for system operators and planners to extract EV charging profiles from smart meter data in a non-intrusive manner such that

1) unrealistic and uncertain assumptions (as pointed out in the above discussions) can be alleviated; and 2) EV charging profiles can be accurately extracted in real-time to support both short-term system operations and long-term planning.

To the best of our knowledge, reference [20] is probably the first to adopt Non-Intrusive Load Monitoring (NILM) and apply benchmark algorithms such as the Hidden Markov Models (HMMs) [24] to detect events and disaggregate EV charging profiles from low-frequency smart meter readings. HMM is a general modeling technique suited to represent a sequence of hidden features in time or space, in which each hidden feature causes or emits an observation. In the context of EV charging profile identification, the aggregated power consumption profile and EV charging profile can be treated as observation and hidden feature, respectively. Obviously, the former is dependent on the latter, i.e., the latter may take different proportions of the former. In addition, to more accurately model the aggregated power consumption profile, profiles of other loads can also be considered by utilizing a variant of HMM called Factorial Hidden Markov Models (FHMMs) [25–28]. Reference [25] proposed a Conditional Factorial Hidden Semi-Markov Model which works well for appliances with simple or modestly complex power signatures, but suffers from complex signatures. Reference [26] developed a Difference Additive Factorial Approximate MAP, which can perform exact inference and is computationally efficient. A convex formulation of approximate inference avoids susceptibility to local optima. Reference [27] presented a solution by designing a hierarchical probabilistic model, which has efficient and effective estimation of latent states. Reference [28] proposed a ensemble methods named hierarchical FHMM, which handles the correlations between devices in order to strengthen independence assumption of devices and preserve the one-at-a-time condition.

Although the FHMM is a favorable model, which is relatively easy to train, it has some drawbacks. Firstly, FHMMs work well for discrete-state (e.g., ON/OFF or High/Medium/Low) loads but have difficulty in determining optimal number of quantized

states for continuous loads. Secondly, FHMMs require detailed *a priori* information such as the number of loads in the aggregated signal and thus are unable to tackle unknown loads. Finally, the computational complexity of an exact inference grows exponentially with the number of sequence lengths and the number of appliances. Meanwhile, an approximate inference may compromise the accuracy. Therefore, it is desired to design an algorithm to mitigate the aforementioned issues. Apart from FHMM-based non-intrusive approaches, reference [21] presents an unsupervised algorithm to extract EV charging loads from the smart meter data using independent component analysis, and reference [23] proposes a training-free algorithm based on bounding-box fitting and load signatures.

1.5 Review of Voltage Regulation with High PV Penetration

The global electricity industry is undertaking a significant paradigm shift from traditionally being centralized, passive, and rigid to potentially being distributed, active, and autonomous. The accelerating proliferation of DERs, such as distributed generations, ESSs, and electrified transportation, has been making positive technical, economic, and environmental impacts. The inherently intermittent nature of DERs has brought many challenges to distribution networks, especially on voltage profiles [29–31]. For instance, it has been shown that in the daytime, high solar generation can cause over-voltage issues [32], while at night, excessive EV charging demands could also result in under-voltage issues [33], especially for rural and remote areas. Conventional voltage regulation devices, such as On-Load Tap Changers (OLTCs), Step Voltage Regulators (SVRs), and shunt capacitors, are too slow or inflexible to accurately regulate voltage in distribution networks with high DER penetrations [34, 35].

In general, voltage regulation methods are categorized into two groups: active power-based and reactive power-based [36]. Contrast to high-voltage transmission lines, the R/X ratio of LV distribution networks is inherently much higher, which implies that voltage magnitudes in LV distribution networks are more sensitive to active power injections [29,

35]. Therefore, this work will focus on active power control for autonomous voltage regulation in LV Unbalanced Distribution Network (UDS).

Besides conventional strategies such as generation curtailment [37], many recent works utilize distributed ESSs for voltage regulation. Reference [34] coordinates PV inverters, ESSs, and SVRs every five seconds to follow volatile PV fluctuations. ESS charging/discharging is adaptively tuned to avoid SOC exhaustion and implemented with no real-time communication requirement. A similar problem is considered to dynamically estimate SOC, adjust participating factors, and regulate ESS charging/discharging by consensus algorithms [38], in which real-time information exchange among neighboring ESS is required. Moreover, a Model Predictive Controller (MPC) is proposed to autonomously coordinate ESSs and grid devices and alleviate voltage variation caused by PV fluctuations [39]. A similar MPC-based coordination of ESSs and OLTCs for voltage regulation is proposed in [40]. However, how the model accuracy might impact those model-driven methods is not discussed. To address a common constraint of single time-scale, a multi-time-step strategy is designed using back-and-forth communication to optimally coordinate ESS by calculating global voltage violation sensitivities [41].

Compared to model-based methods, model-free control is more preferable to avoid modeling challenges [42], such as RL [43]. In RL, autonomous agents can actively learn (in a model-free manner) to perform a given task by trial and error without any guidance from human operators. A batch RL algorithm is proposed in [44] to determine optimal tap settings, in which a Markov Decision Process (MDP)-based policy is formulated to map voltage magnitude measurements to tap ratio changes. Each OLTC is assigned an action-value function, which is sequentially learned by the least-square policy iteration with the distribution network modeled by the *LinDistFlow* [45, 46].

Moreover, RL is typically combined with deep neural networks, i.e., Deep Reinforcement Learning (DRL), for enhanced representational capability in high-dimensional spaces [47, 48]. A safe off-policy DRL algorithm is proposed in [49] to model

Volt-VAR control as a constrained MDP with discrete action space. A popular value-based DRL formulation, called the Deep Q-network (DQN), is constructed with designed rewards in [50] to handle time-varying operating conditions, in which statuses/ratios of capacitors, SVRs, and inverters are modeled as agents' actions. A two-time-scale DQN is proposed in [51] to incorporate both on-off capacitor actions (hourly or daily) and smart inverter injections (within milliseconds). It is shown that, for autonomous voltage control, the Deep Deterministic Policy Gradient (DDPG) method requires less action corrections than DQN [52].

To the authors' best knowledge, model-free voltage regulation with multiple ESSs autonomously coordinated has not been well addressed. To fill this gap, this work aims at designing a DRL-based framework to autonomously schedule day-ahead ESS operations for real-time voltage regulation in UDS. Note that although in literature UDS could refer to that each bus can have one, two, or three nodes, most recent voltage regulators consider three-phase distribution systems with significant power or voltage imbalance across phases due to unbalanced DER installation and fluctuations [35, 53–57]. Furthermore, although some recent works have used three-phase balanced distribution system to reduce the modeling complexity, three-phase UDS is more consistent with practice and is thus adopted in this dissertation proposal. Specifically, the UDS considered in this dissertation proposal models the placement of loads and DERs at different phases and the coupling effect between different phases, which cannot be reflected in the simplified balanced distribution systems.

1.6 Review of Power Flexibility Aggregation for TSO-DSO Interaction

The penetration of DERs such as solar generations, ESSs, and demand response has considerably increased over the last years, which promotes a transformation of distribution systems from passive to active state. Specifically, unlike the conventional distribution system treated as a passive load on transmission side, the modern distribution system

with large-scale integration of DERs can not only meet energy demand of local loads, but also actively release significant power flexibility to support transmission system operation. The power flexibility typically refers to the ability of a distribution system to modulate its exchanged power with the transmission system at the substation interface. Through coordinated transmission and distribution (T&D) dispatch (i.e., flexibility aggregation), the potential power flexibility can be fully exploited to make power grids more efficient and resilient [58, 59].

In practice, aggregating a large population of DERs and exploiting power flexibility for system-wide operation and control are challenging due to computational complexity and cybersecurity risks. Some existing works [58–60] are focused on applying disaggregation-based methods (e.g., generalized Benders decomposition) to jointly dispatch all generators in both the transmission and distribution networks in a formulation of economic dispatch. However, these methods suffer from slow convergence issues especially when there exists a large number of DERs. Meantime, it requires frequent boundary information exchange, which aggravate communication burden issues. Other alternatives for flexibility aggregation are to characterize a feasible time-varying power flexibility region on distribution side, which can be integrated into mature transmission optimization programs to mitigate the complexity from coupled transmission-distribution networks.

With massive DER devices, multiple time periods and a complex network environment with multiple coupled factors (i.e., time-coupled, power-coupled, and phase-coupled constraints), it is intractable to procure an exact and feasible power flexibility region. Therefore, more research efforts are devoted to approximation methods. Reference [61] models the power flexibility of an individual flexible load as a polytope, and then their aggregate flexibility is represented by the Minkowski sum of all these polytopes. Reference [62] models the aggregate P-Q feasible domain over time as a series of time-moving ellipsoids, and their parameters are obtained by a data-driven system identification procedure. In [63], a box-shape active-power flexibility region is modeled, and a distributed

model predictive control framework is developed to optimize the parameters of such an approximated region. Reference [64] proposes a two-stage adaptive robust optimization framework for power flexibility aggregation with a hyperbox approximation in [64] and an ellipsoid approximation in [65].

From the perspective of control system, the power flexibility aggregation can be regarded as a process of controlling the power output of DERs to track the power flexibility command. The majority of existing works are focused on designing control rules of DERs. However, the power output of DERs as control variables is redundant. That is, a power flexibility command can be realized by multiple dispatch alternatives, which may hinder efficient design of control rules and flexibility evaluation schemes. To our best knowledge, [65] is probably the first to reconstruct the DER power region and design control rules of converted control variables, which can regulate power flexibility as the DERs do while providing a new insight into control law design of DERs. Through this reconstruction, the redundancy issue can be avoided, and the feasibility of power flexibility can be easily guaranteed in some case.

Apart from the above-mentioned redundancy issue, approximation results in the ahead-of-time flexibility region evaluation may be compromised by the forecast error uncertainty. That is, if the forecast model has low accuracy, there may exist some infeasible points in the approximated region leading to the termination of the aggregation process. To our best knowledge, few works address this issue. In [66], the forecast error uncertainty is explicitly characterized by Gaussian Mixture Model and incorporated into a chance-constrained optimization problem. The results show its higher aggregation efficiency than the scenario-based methods such as the Monte-Carlo simulation. The [67] proposes a data-driven probabilistic-based flexibility region estimation method by constructing a distributionally robust optimization problem to handle unknown forecast error uncertainties and find a conservative solution.

1.7 Dissertation Organization

The main content of this dissertation consists of six chapters including this introduction. The remainder of this dissertation is organized as follows.

Chapter Two provides a brief introduction to several background about neural network and reinforcement learning. Additional background will be introduced in later chapters if necessary.

Chapter Three proposes a Deep Generative Model (DGM)-driven inference framework for non-intrusive, real-time identification of EV charging profiles. Firstly, the joint probability distribution for available smart meter data (which can actually be considered as time series) is modeled by DGMs. The posterior distributions are approximated by neural networks whose parameters are obtained by variational inference and supervised learning. Secondly, the EV charging status is inferred from DGMs via dynamic programming. Finally, the target EV charging profile can be reconstructed according to the rated power of EV models and inferred status.

Chapter Four proposes a DRL-based voltage regulation method for high PV penetrated distribution systems. Firstly, an ESS-based voltage regulation problem is proposed, with the objective of minimizing the expected total daily system-wide voltage regulation cost subject to operational constraints. Secondly, this problem is converted to an MDP with a partial objective (i.e., negative voltage regulation compensation at single time step) as rewards, with constraints treated as an action space flexibly. Finally, the preceding problem is further formulated as an entropy-constrained MDP problem to make agents explore their action spaces more widely and wisely, which can be approximately solved by a state-of-the-art DRL method called Soft Actor Critic (SAC), without the need of modeling uncertainties.

Chapter Five proposes an inner approximation framework for an ahead-of-time flexibility region estimation considering the forecast error uncertainty. Firstly, the distribution system including device, network, and uncertainty propagation is modeled

as a polyhedral power region. After that, through a coordinate transformation, two types of flexibility regions called non-feeder-level and feeder-level flexibility regions are defined. The latter can be viewed as an aggregation of the former. Finally, a chance-constrained Chebyshev centering optimization model is developed to approximate multiple maximum inscribed balls, which can be solved by simultaneously and unsupervisedly learning multiple hyperplanes.

Finally, Chapter Six summarizes the main contributions and lists outcomes of this dissertation.

CHAPTER 2

LITERATURE REVIEW

In this chapter, a brief introduction to Deep Neural Networks (DNNs) and DRL is provided. The DNNs are used to parameterize the probability distributions in Chapter Three, and the control law and optimal value function in Chapter Four. DRL is involved in Chapter Four. The additional background will be introduced if necessary in later chapters.

2.1 Deep Neural Networks

The first artificial neural network component “perceptron” developed in [68] can be dated back to 1958, which is designed to mimic the way the human brain processes visual data and identifies recognizable objects. After that, it was extended to multilayer perceptrons, also called deep feedforward networks, which are the basic deep neural network architectures.

Given an input \mathbf{x} and an output \mathbf{y} , the goal of a deep feedforward network is to approximate some function $\mathbf{y} = f^*(\mathbf{x})$. For example, for a classifier, \mathbf{x} is a vector of features while \mathbf{y} is a probability distribution of categories. A feedforward network defines a mapping $\mathbf{y} = f(\mathbf{x}, \boldsymbol{\theta})$ where parameters $\boldsymbol{\theta}$ can be learned by training data samples so that f can become the best function approximation to f^* . With the following representation

$$\underbrace{\mathbf{x}}_{\text{input layer}} \rightarrow \underbrace{f}_{\text{hidden layers}} \rightarrow \underbrace{\mathbf{y}}_{\text{output layer}},$$

the models are called feedforward. The first layer of the models is called the input layer and the final layer of the models is called the output layer. When the models have more than two layers, the intermediate layers are called hidden layers.

Take a three-layer feedforward network as an example

$$f(\mathbf{x}, \boldsymbol{\theta}) = g(\mathbf{W}^\top \mathbf{x} + \mathbf{b})$$

where

- $\boldsymbol{\theta} = \{\mathbf{W}, \mathbf{b}\}$ denotes neural network parameters,
- \mathbf{W} is the weights of a linear transformation,
- \mathbf{b} is the biases, and
- g is a nonlinear function called an activation function.

Note that in modern neural networks, the default recommendation [69] for g is to use the rectifier linear unit, i.e.,

$$g(\mathbf{z}) = \max(0, \mathbf{z})$$

.

Given a training set $\{\mathbf{x}^{(i)}, \mathbf{y}^{(i)}\}_{i=1}^n$, the parameters $\boldsymbol{\theta}$ can be learned by solving the following minimization problem

$$\min_{\boldsymbol{\theta}} \frac{1}{n} \sum_{i=1}^n L(f(\mathbf{x}^{(i)}, \boldsymbol{\theta}), \mathbf{y}^{(i)})$$

where $L(f(\mathbf{x}, \boldsymbol{\theta}), \mathbf{y})$ is a loss function that can be used to evaluate how good the approximated function is. This problem can be solved by using back-propagation and variants of stochastic gradient decent algorithm such as the Adam optimizer [70].

Based on this feedforward network, with some tricks casted on loss functions and intermediate layer representation, some other types of neural network architecture such as convolutional neural networks, recurrent neural networks etc. can be built. Refer to the book [69] for more relative details.

2.2 Deep Reinforcement Learning

RL is a representative bio-inspired learning approach. Through the trial-and-error interaction in an unknown environment, RL can learn a decision or control policy. This section starts with principles of RL problems and then moves to DRL.

2.2.1 Principles of RL Problems

Agent

The agent is a learner, or a decision maker, or a controller.

Environment

Most stochastic and discrete-time environments can be viewed as MDPs, or MDP problems. For a MDP, the agent and environment interact at each of a sequence of discrete time steps, $t = 0, 1, 2, 3, \dots$. At each time step t , the agent receives state $s_t \in \mathcal{S}$, and on that basis selects an action $a_t \in \mathcal{A}$. One time step later, the environment responds to the received action, and sends the next state $s_{t+1} \in \mathcal{S}$ as well as a numerical reward $r_t \in \mathcal{R} \subset \mathbb{R}$ to the agent.

The state can be seen as some representation of the environment which contains sufficient information used for decision making. The state evolves based on environment dynamics:

$$P_{ss'}^a := \Pr\{s_{t+1} = s' | s_t = s, a_t = a\} \quad (2.1)$$

where s' , s and a are the values of random variables s_{t+1} , s_t and a_t , respectively. The Equation 2.1 follows the Markov property, i.e., the state at current time step depends only on the state and action of precedent steps.

State-Action Samples

In an unknown environment, it is difficult to obtain an exact representation of environment dynamics like Equation 2.1. Therefore, the state-action samples (s_t, a_t, s_{t+1}) are considered as another representation of environmental dynamics, which can be collected by interacting between agent and environment, either physically in the real world or virtually in a computer.

Policy

The policy π is a function mapping state to action, which describes how the agent responds to the environment. In a MDP problem, the policy is Markovian, i.e., the action is only determined by the current state. This kind of policies can be categorized into deterministic policy and stochastic policy. The deterministic policy $\pi(s)$ selects the same action at the same state, i.e.,

$$a = \pi(s), \quad s \in \mathcal{S} \quad (2.2)$$

while the stochastic policy $\pi(a|s)$ selects action according to a learned probability distribution where

$$\sum_{a \in \mathcal{A}} \pi(a|s) = 1, \quad s \in \mathcal{S} \quad (2.3)$$

In general, stochastic policies are more efficient than their deterministic counterparts because they allow the agent to explore better policies.

When both state space and action space are discrete, a policy can be represented as the look-up table, i.e., an array that holds a set of pre-computed actions in all state positions. For large-scale problems especially in an environment with continuous or high-dimensional state and action space, such a tabular policy is not always computationally

efficient. Alternatively, a parameterized function can be used to represent the policy to reduce the computational burden:

$$\begin{aligned}\pi_\theta(s) &:= \pi(s, \theta) : \text{deterministic policy} \\ \pi_\theta(a|s) &:= \pi(a|s, \theta) : \text{stochastic policy}\end{aligned}\tag{2.4}$$

where θ is the parameter of the policy. The choice of parameterization is a good way of injecting prior knowledge about the desired form of the policy.

Reward and Return

In RL, a reward signal r_t is determined by a function r with reference to the current state s_t , current action a_t , and next state s_{t+1} , defined as:

$$r_{ss'}^a := r(s_t = s, a_t = a, s_{t+1} = s')\tag{2.5}$$

The return G_t is defined as some specific function of the reward sequence. The most popular definition is the discounted return:

$$G_t = \sum_{i=0}^{+\infty} \gamma^i r_{t+i}\tag{2.6}$$

where $\gamma \in [0, 1]$ is the discounting factor. If $\gamma = 0$, the agent is myopic and concerns only immediate reward. As γ approaches 1, the agent becomes more farsighted. The purpose of discounting returns is to provide a tractable measure, i.e., G_t is finite.

Value Function

To evaluate how good a policy is, there are two kinds of value functions: state-value function and action-value function. The former is a function of the state only, defined as

the expected return under a policy π starting from s :

$$v_\pi(s) := \mathbb{E}_\pi[G_t|s] \quad (2.7)$$

The latter is a function of both state and action, defined as the expected return under a policy π taking action a in state s :

$$q_\pi(s, a) := \mathbb{E}_\pi[G_t|s, a] \quad (2.8)$$

When both state and action spaces are small, the tabular function can be used to represent value functions. Otherwise, parameterized functions are better choices with less computation and storage burden.

A fundamental property inherited in value functions is that it naturally holds a certain recursive relationship, i.e., self-consistency condition. Looking ahead from a state to its successor states, the recursive relationship between $v_\pi(s)$ and $q_\pi(s, a)$ can be easily found:

$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) q_\pi(s, a) \quad (2.9)$$

and

$$q_\pi(s, a) = \sum_{s' \in \mathcal{S}} P_{ss'}^a (r_{ss'}^a + \gamma v_\pi(s')) \quad (2.10)$$

Combining Equation 2.9 and Equation 2.10, the self-consistency condition of state-value function can be found:

$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) \left[\sum_{s' \in \mathcal{S}} P_{ss'}^a (r_{ss'}^a + \gamma v_\pi(s')) \right] \quad (2.11)$$

Reusing Equation 2.9 and Equation 2.10 with next state s' and next action a' , the self-consistency condition of action-value function can be found:

$$q_\pi(s, a) = \sum_{s' \in \mathcal{S}} P_{ss'}^a [r_{ss'}^a + \gamma \sum_{a' \in \mathcal{A}} \pi(a'|s') q_\pi(s', a')] \quad (2.12)$$

The self-consistency condition builds the connection between value function and policy. Given a policy and environment dynamics, the so-called model-based evaluation method is performed by solving the self-consistency condition, like dynamic programming, approximate dynamic programming, etc.

Goal

The goal of MDP problems is to find an optimal solution that maximizes the expected return

$$\max_{\pi} J(\pi) = \max_{\pi} \mathbb{E}_{s_0} [v_\pi(s_0)] \quad (2.13)$$

where $J \in \mathbb{R}$ is called objective function, $s_0 \in \mathcal{S}$ is initial state, and π is the policy to be optimized. For problem Equation 2.13, its solution can be classified as model-based and model-free, depending on whether the environment dynamics is known or not.

Optimality

For any finite MDPs, there at least exists an optimal policy, i.e., π_* , that is better than or equal to any other policies. An optimal policy naturally results in corresponding optimal state-value function,

$$v_*(s) = \max_{\pi} v_\pi(s), \quad \forall s \in \mathcal{S} \quad (2.14)$$

and optimal action-value function

$$q_*(s, a) = \max_{\pi} q_\pi(s, a), \quad \forall s \in \mathcal{S} \quad \text{and} \quad \forall a \in \mathcal{A} \quad (2.15)$$

The relation of $v_*(s)$ and $q_*(s, a)$ is

$$v_*(s) = \max_{a \in \mathcal{A}} q_*(s, a) \quad (2.16)$$

$$q_*(s, a) = \sum_{s' \in \mathcal{S}} P_{ss'}^a [r_{ss'}^a + \gamma v_*(s')] \quad (2.17)$$

An optimal policy can be found by maximizing optimal action-value function:

$$\pi_* = \pi_*(a|s) = \begin{cases} 1 & \text{if } a = a^* \\ 0 & \text{if } a \neq a^* \end{cases} \quad (2.18)$$

where

$$a^* = \arg \max_a q_*(s, a)$$

By combining Equation 2.16 and Equation 2.17, the self-consistency condition of $v_*(s)$ can be found:

$$v_*(s) = \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} P_{ss'}^a [r_{ss'}^a + \gamma v_*(s')] \quad (2.19)$$

Reusing Equation 2.16 and Equation 2.17 with next state s' and next action a' , the self-consistency condition of $q_*(s, a)$ can be found:

$$q_*(s, a) = \sum_{s' \in \mathcal{S}} P_{ss'}^a [r_{ss'}^a + \gamma \max_{a' \in \mathcal{A}} q_*(s', a')] \quad (2.20)$$

The recursive relationship derived in Equation 2.19 and Equation 2.20 is also called Bellman equation, which can be used to find the optimal action. Compared with Equation 2.19 and Equation 2.20, using the latter is more efficient without need to use environmental dynamics and know successor states and their values beforehand. There are two methods to solve Bellman equation: policy iteration and value iteration. The former involves alternating steps, i.e., policy evaluation and policy improvement. The policy is

evaluated by its corresponding value function in the step of policy evaluation, and then its new policy is searched in the step of policy improvement. Even though neither of them is optimal, they together can gradually converge to the solution of Bellman equation as long as the policy will become better and better in the phase of policy improvement. The latter solves Bellman equation by using the fixed-point iteration technique. Specifically, the value function is treated as the variable to be iterated. The optimal policy can be greedily searched after an optimal value function is found. The convergence of value iteration depends on whether the Bellman operator is contractive.

Solution

In this part, the connection of optimal policy π_* and optimal solution π_* of the problem defined by Equation 2.13 is explained.

According to the Jensen's inequality,

$$\max_{\pi} J(\pi) = J(\pi_*) \leq \mathbb{E}_{s_0}[\max_{\pi} v_{\pi}(s_0)] = \mathbb{E}_{s_0}[v_*(s_0)] = J(\pi_*) \quad (2.21)$$

Denote $\pi \in \Pi$ as the policy set. When π_* is inside the policy set, i.e., $\pi_* \in \Pi$,

$$J(\pi_*) \leq J(\pi_*) \quad (2.22)$$

By combining Equation 2.21 and Equation 2.22, $J(\pi_*) = J(\pi_*)$, and π_* is one of optimal solutions. Therefore, the solution of Bellman equation is equivalent to that of the problem defined by Equation 2.13. At this time, the optimal solution has nothing to do with the initial state distribution. When π_* is not inside the policy set, i.e., $\pi_* \notin \Pi$, only Equation 2.21 holds, and at this time the initial state distribution can affect the optimality of the problem defined by Equation 2.13. Therefore, if Π can be equal to full policy space, the problem defined by Equation 2.13 degenerates into finding the optimal state-value function $v_*(s)$ and its corresponding optimal policy π_* . For the methods solving for Bellman

equation, there is no constraint on the policy set, and thus $\pi_* \in \Pi$. Apart from using Bellman equation to find an optimal policy, the other type of methods is to directly optimize an policy (often parameterized like $\pi_\theta(s)$) with respect to the scalar performance index $J(\theta)$. When π_θ is neural networks or other functions with almost-universal approximate ability, the problem defined by Equation 2.13 can be rewritten as

$$\max_{\theta} J(\theta) = \max_{\theta} \mathbb{E}_s[v_{\pi_\theta}(s_0)] = \mathbb{E}_s[\max_{\theta} v_{\pi_\theta}(s_0)] \quad (2.23)$$

It can be seen from the problem defined by Equation 2.23 that the initial state distribution has no impact on the optimality, and the objective is still to find the optimal state-value function $v_*(s)$. Normally, the gradient decent algorithm can be used to solve problem defined by Equation 2.23

$$\theta \leftarrow \theta + \eta \nabla_{\theta} J(\theta) \quad (2.24)$$

where η is the learning rate, and $\nabla_{\theta} J(\theta)$ is called policy gradient. The key idea underlying policy gradients is to push up the probabilities of actions that lead to higher return, and push down the probabilities of actions that lead to a lower return, until the optimal policy is found. This direct method requires effective estimation of policy gradient, which is a costly procedure. And it always finds a local maximum. All in all, the essence of RL is to find the optimal state-value or action-value function.

2.2.2 DRL

Traditional RL [43] algorithms exist several limitations. On one hand, they suffer from the issue of “curse of dimensionality” when the state and action spaces are large or continuous. As a result, it is intractable to compute or store a gigantic look-up table for value and policy functions. On the other hand, hand-crafted state representations are typically required. To overcome the above limitations, function approximation methods

are developed to approximate value and policy functions with some parameterized function classes. Driven by the advances of DL, DRL that utilizes DNNs to approximate value and policy functions is becoming increasingly popular, which features the following technical points:

Replay Buffer

The DNNs can equip RL with the ability to automatically extract features and generalize to a large state space. But the training data sampled from the interaction are highly correlated, violating the independence assumption. Therefore, in DRL, a replay buffer denoted by \mathcal{D} is adopted to store a large number of transitions (s_t, a_t, r_t, s_{t+1}) , which can break the correlation among training data by randomly sampling a mini-batch data from the replay buffer when updating neural networks.

Value Function Approximation

Firstly, the action-value function $q_\pi(s, a)$, sometimes called q-function, can be approximated by neural network $q_w(s, a)$ with weights w . Through Temporal-Difference learning, i.e, one-step bootstrapping, the w can be updated by

$$w \leftarrow w + \alpha[r_t + \gamma q_w(s_{t+1}, a_{t+1}) - q_w(s_t, a_t)] \nabla_w q_w(s_t, a_t) \quad (2.25)$$

where α is a learning rate, and the gradient $\nabla_w q_w(s_t, a_t)$ can be calculated efficiently using the back-propagation method, and this updating process is called q-learning.

Secondly, to more conveniently find an optimal policy, some DRL algorithms directly use neural networks to approximate optimal action-value functions $q_*(s, a)$. Similar to one-step bootstrapping, the w can be updated by

$$w \leftarrow \arg \min_w \mathbb{E}_{(s_t, a_t, r_t, s_{t+1}) \sim \mathcal{D}}[r_t + \gamma \max_{a_{t+1}} q_{\hat{w}}(s_{t+1}, a_{t+1}) - q_w(s_t, a_t)] \quad (2.26)$$

where $q_{\hat{w}}(s, a)$ is a target optimal action-value function. In modern DRL algorithms such as DQN[71], DDPG[72] etc., the $q_{\hat{w}}(s, a)$ can be approximated by another neural network with weights \hat{w} , called target network, which is a clone of $q_w(s, a)$. By introducing target networks, the non-stationary distribution of training data, i.e., the training data may not be identically distributed, can be alleviated, which significantly improves the stability of training process. After a fixed number of updates of $q_w(s, a)$, the $q_{\hat{w}}(s, a)$ is renewed by replacing \hat{w} with the latest learned w . Note that it is not easy to solve a maximum problem in Equation 2.26 due to the nonlinear and complex formulation of $q_{\hat{w}}(s, a)$, which hinders the use of value-based RL methods. In actor-critic RL methods, by replacing a_{t+1} with the optimal policy evaluated at state s_{t+1} , the maximum operator can be removed.

Thirdly, the state-based function $v_{\pi}(s)$ can also be approximated by neural network, which can be further combined with the state-dependent action advantage function to determine the action-value function. The main benefit of this factoring is to generalize learning across actions without changing the underlying RL algorithm.

Policy Function Approximation

In policy-based and actor-critic RL methods, neural networks are widely used to parameterize control policies π_{θ} , which can be solved by Equation 2.24. In order to reduce the variance of the gradient, an extra baseline term is added to calculate the gradient. Compared the stochastic policy with the deterministic policy, the latter is more sample-efficient because its policy gradient only integrates over the state space while the former integrates over both state and action spaces. One major issue regarding a deterministic policy is the lack of exploration, which may lead to a poor policy. A common way to encourage exploration is to perturb a deterministic policy with exploratory noises, e.g., adding a Gaussian noise ξ to the policy with $a = \pi_{\theta}(s) + \xi$.

CHAPTER 3

LEARNING-BASED SITUATION AWARENESS UNDER UNCERTAINTY

The proliferation of EVs brings environmental benefits and technical challenges to power grids. An identification algorithm which can accurately extract individual EV charging profiles out of widely available smart meter measurements has attracted great interests. This chapter proposes a non-intrusive identification framework for EV charging profile extraction, which is driven by DGM. First, the proposed DGM is designed as a representation layer embedded into the Markov process and used to model the joint probability distribution of available time-series data. A novel contribution is to approximate posterior distributions by neural networks whose parameters are obtained by variational inference and supervised learning. Second, the EV charging status is inferred from the DGM via dynamic programming. Lastly, the desired EV charging profile can be reconstructed by the rated power of EV models and inferred status.

The remainder of this chapter is organized as follows. Section 2.1 defines the EV charging profile identification problem considered in this dissertation and then formulates it with in the architecture of NILM. Next, Section 2.2 reviews the framework of HMM, which will be used as a benchmark algorithm. Furthermore, Section 2.3 proposes a DGM to model the joint probability distribution of the available aggregated consumption data, of which parameters are obtained by variational inference and supervised learning. Section 2.4 utilizes dynamic programming to perform exact inference of the DGM for the EV charging status. Moreover, Section 2.5 discusses numerical validation setup and results. Finally, Section 2.6 concludes this chapter.

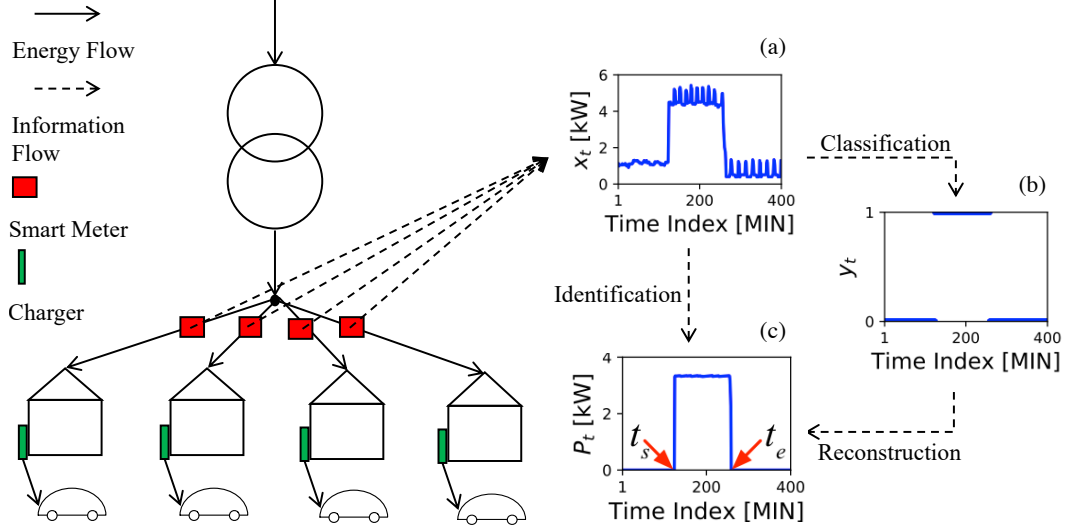


Figure 3.1: Overview of EV charging profile identification. (a): a sample aggregated power consumption profile; (b): its corresponding EV charging status (charging started at t_s and ended at t_e); and (c) its corresponding EV charging profile.

3.1 Problem Formulation

The EV charging profile identification problem considered in this dissertation proposal is presented in Figure 3.1, in which a sample aggregated power consumption profile is shown in Figure 3.1(a), with its corresponding EV charging profile shown in Figure 3.1(c). Furthermore, Figure 3.1(b) shows the corresponding EV charging status (charging started at time t_s and ended at time t_e).

3.1.1 Definitions

Given an aggregated power consumption profile $x = (x_1, \dots, x_T)$, i.e., a timed sequence of a total of T power consumption data points, determine its corresponding EV charging profile P (or $P(x)$ if the source power consumption profile x is relevant). Note that the power consumption profile is called aggregated as most smart meters measure the power consumption of the whole household and thus include all loads (i.e., aggregated). An EV charging profile is thus a timed sequence (of the same length T) of EV charging power consumption data points. In other words, the value P_t of P at time step t denotes the

amount of power by EV charging.

Moreover, at time step t , the charging status y_t of an EV is binary, i.e., either ON (i.e., $y_t = 1$ if P_t is greater than a pre-defined threshold P_{th}) or OFF ($y_t = 0$ otherwise). Furthermore, the probability of an EV at its y_t is denoted by $p(y_t)$. When $y_t = 1$, $p(y_t = 1) = 1$ and $p(y_t = 0) = 0$. When $y_t = 0$, $p(y_t = 1) = 0$ and $p(y_t = 0) = 1$.

3.1.2 EV Charging Profile Identification as NILM

The objective of the EV charging profile identification problem considered in this dissertation proposal is to determine EV charging profile P given aggregated power consumption profile x . Therefore, the scope of this work falls within the framework of an NILM problem. Most techniques used for NILM problems in the literature consist of two sub-tasks: 1) classification and 2) reconstruction. The former task aims at classifying the load operation status into known categories, and the latter task is to reconstruct load consumption profiles using classification results. For example, if the first task returns that the charging status of a certain model of EV at time step t is classified to be ON (i.e., $y_t = 1$) with rated power consumption around 6.7 kW (i.e., $P_t = 6.7$), then the latter task would focus on reconstructing its corresponding EV charging profile. Therefore, this dissertation proposal follows [22, 73] to assume that EV charging power level and corresponding models can be identified separately and mainly focuses on EV charging status classification and converts the EV charging profile identification problem into a binary EV charging status classification task.

Therefore, the EV charging profile identification problem this dissertation proposal aims to solve can be formulated as follows: given an aggregated power consumption profile $x = (x_1, \dots, x_T)$, determine y_t of an EV at each time step $t = 1, \dots, T$. The general procedure of how the proposed EV charging profile identification problem is studied in this dissertation proposal is presented in Figure 3.2. First of all, each generative process for time-series data is modeled by a joint probability distribution in Step 1. Secondly,

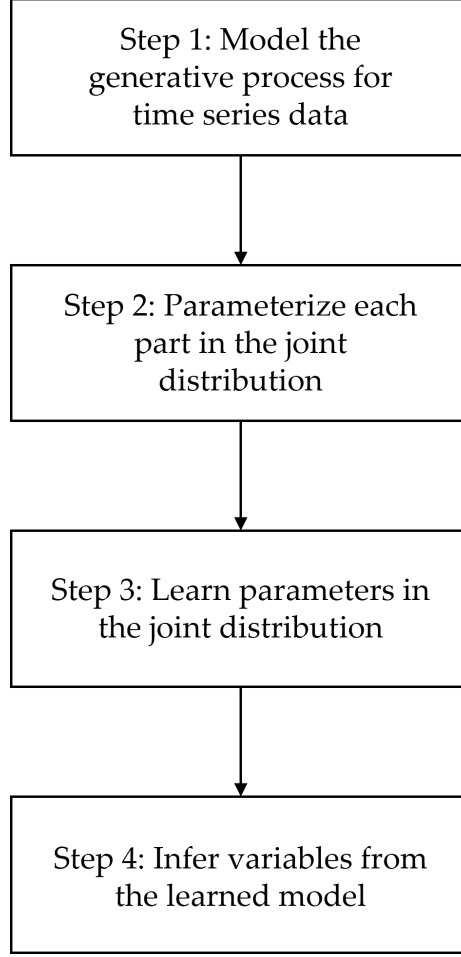


Figure 3.2: The general procedure of the proposed identification algorithm under the architecture of NILM can be divided into four steps.

each component (e.g., evidence and transition probabilities) in the joint distribution from Step 1 is approximated by a common parametric density function in Step 2. Thirdly, parameters of the density function (in the joint distribution) from Step 2 are learned by maximum likelihood estimation in Step 3. Finally, the above-defined identification problem is converted to a Bayesian inference process by the proposed DGM in Step 4. The above steps are further specified in Sections 3.3 and 3.4.

3.2 Benchmark Algorithm: HMMs

3.2.1 Applicability

HMMs have been extensively utilized for EV charging profile and other load profile identification in the field of NILM. The reasons that HMMs are used as benchmark algorithms for EV Charging profile identification are summarized as next three points. First, HMMs are a widely acknowledged tool for modeling time series like the collected smart meter data (i.e., aggregated power consumption profiles and the corresponding EV charging profiles). Second, HMMs are relatively easy to train, and can be quickly deployed in practical scenarios. Third, mainly due to its low identification accuracy with a single Markov chain, it is desired to improve it by enhancing the representational ability of the Markov chain in this proposal.

3.2.2 Implementation Details

Given the aggregated power consumption profiles and their corresponding EV charging profiles, their joint probability distribution can be modeled by a standard HMM [74]. The graphical illustration of such an HMM is given by Figure 3.3, in which an aggregated power consumption profile $x = (x_1, \dots, x_T)$ and its corresponding EV charging status profile $y = (y_1, \dots, y_T)$ (depicted by nodes) are modeled as random variables. Under this model, the aggregated power consumption at each step x_t only depends on some unobserved or latent EV charging status y_t , which are depicted by direct edges. Therefore, the former can be generated/emitted by the latter. This conditional generative process can be modeled by $p(x_t|y_t, \theta)$, which is typically known as emission probability [74]. The Markovianity is defined such that y_t only depends on y_{t-1} . Note that the bottom representation layer (shown in blue) in Figure 3.3 is the proposed model, which will be discussed in the next section.

Such an HMM can model the following joint probability distribution reflecting the

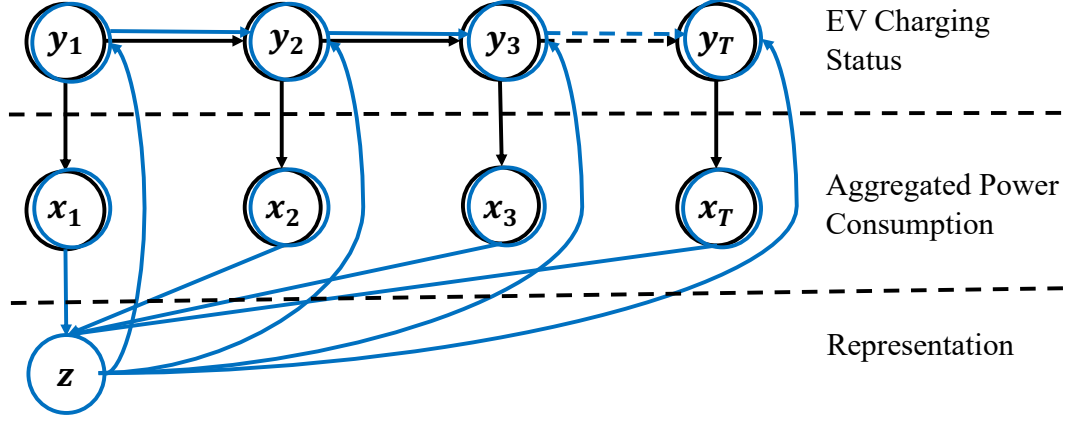


Figure 3.3: Graphical illustration of how aggregated profiles and EV charging profiles are represented by an HMM (black) vs. the proposed representation-layer-embedded Markov model (blue), which introduces an additional layer of representations.

generative process of the time series x and y :

$$p(x, y | \phi) = \underbrace{p(y_1 | \pi)}_{\text{Prior}} \underbrace{\left[\prod_{t=2}^T p(y_t | y_{t-1}, A) \right]}_{\text{Transition Probability}} \underbrace{\left[\prod_{t=1}^T p(x_t | y_t, \theta) \right]}_{\text{Emission Probability}} \quad (3.1)$$

where $\phi = \{\pi, A, \theta\}$ denotes parameters of HMMs, π is the categorical distribution of the initial status, A is the status transition matrix, and θ is the mean and deviation if the emission probability is Gaussian, which can be obtained by counting frequencies through supervised learning. With ϕ learned, the EV charging status y can be inferred by Maximizing *a Posterior* (MAP) $p(y|x, \phi)$ which can be solved by the Viterbi Algorithm [75, p.629], which is a well-known method for exact inference.

3.3 The Proposed DGM: Representation and Parameter Learning

This section presents in details the proposed DGM-driven framework, which simplifies the classification complexity using only a single (but enhanced) Markov chain and utilizes deep neural networks to approximate posterior probability distributions with weights trained via supervised learning.

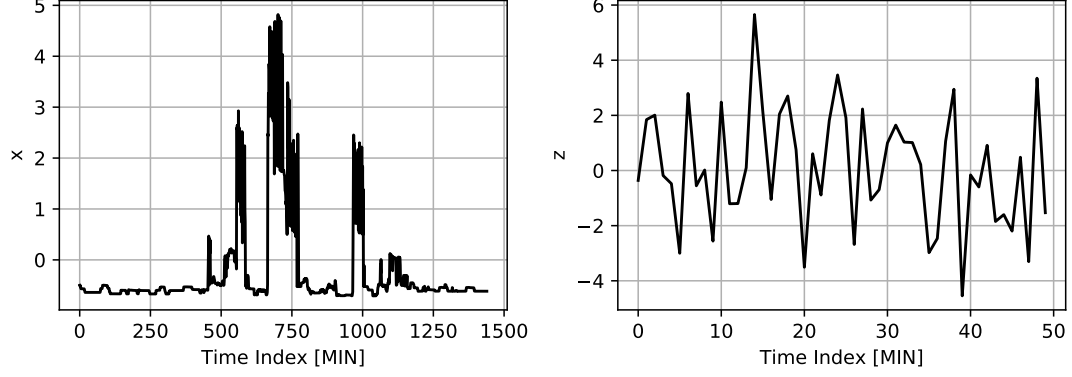


Figure 3.4: Illustration of a sample standardized aggregated power consumption profile (left) and its corresponding representation layer (right).

3.3.1 Representation Layer Embedded Markov Chain

A major innovation of the proposed framework for EV charging status classification is to embed a representation layer (denoted by z) into the Markov chain. The z can be seen as abstract but meaningful features from any aggregated power consumption profile x . As shown in Figure 3.4, the left figure shows a standardized aggregated power consumption profile x , and the right figure shows the value of the corresponding representation layer z . The motivation behind the proposed representation layer can be summarized as follows.

- Firstly, although a single Markov chain can classify EV charging status, the representation capability by raw, aggregated power consumption profiles (i.e., inputs) is relatively weak due to similar and ambiguous features;
- Secondly, nonlinear dynamics in raw time-series inputs might present higher non-stationary variances and thus fail to provide useful features;
- Finally, the learned likelihood distributions $p(x_t|y_t = 0)$ and $p(x_t|y_t = 1)$ from raw inputs may have significant overlaps, which could cause mis-classification issues in later steps. As illustrated in Figure 3.5, the learned (Gaussian mixture) likelihood distributions for $p(x_t|y_t = 0)$ and $p(x_t|y_t = 1)$ are represented by the blue and red lines respectively and overlap in the dash rectangle area.

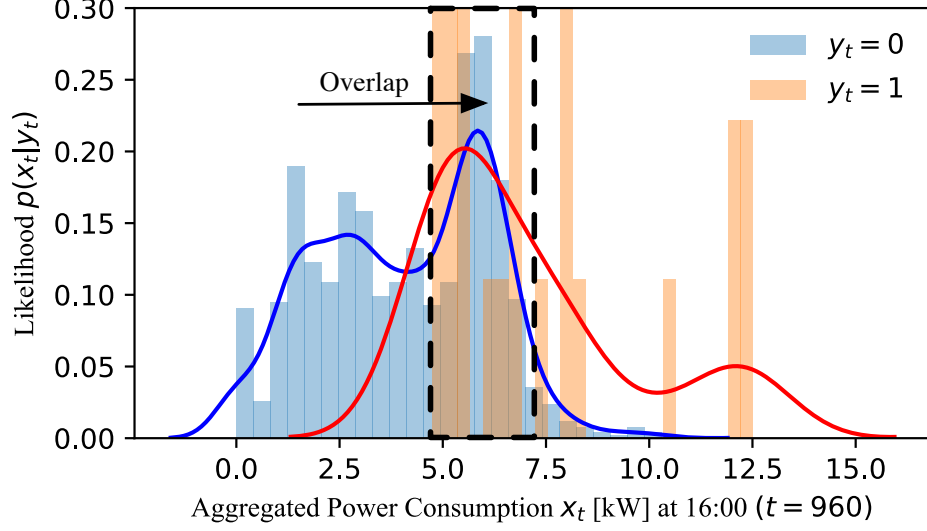


Figure 3.5: An example likelihood function $p(x_t|y_t)$ in HMM, where the learned likelihood distributions for the two distinctive scenarios $p(x_t|y_t = 0)$ and $p(x_t|y_t = 1)$ are represented by the blue and red lines, respectively.

The proposed representation-layer-embedded Markov model is illustrated by Figure 3.3, which depicts the generative process of x and y with z (depicted by nodes) also modeled as a random variable. Considered as measurements, x can always be observed, while y can only be partially observed and z always cannot be observed. Compared with the conventional HMM which only contains the upper and middle layers, the proposed representation-layer-embedded Markov model adds a representation layer with the same nodes as the HMM but without directed paths from y to x , which models likelihood distributions $p(x_t|y_t)$. Both methods generate the same x and y from different perspectives. The HMM assumes that x can be generated by or correlated with y . Moreover, the proposed model assumes that the feature z can generate y (depicted by directed edges). Generating such a z needs to have a powerful feature extractor, which is one of reasons to use deep neural networks.

The two directed edges connecting y_{t-1} to y_t and z to y_t meet the Markov property, i.e., current charging status only relies on the previous charging status and representation layer. The advantage of adding a representation layer is to alleviate the above-discussed likelihood distributions overlapping issue (as shown in Figure 3.5), i.e., the proposed model learns the

posterior distributions $p(z|x)$ and $p(y|z) = \prod_{t=1}^T p(y_t|z)$ instead of all $p(x_t|y_t)$, which will be studied in details in later sections.

3.3.2 Probability Distribution Approximation in DGM

In this work, a DGM is developed based on the proposed representation-layer-embedded Markov model with the following characteristics:

1. **Deep**: using deep neural networks to approximate distributions;
2. **Generative**: taking x as input to generate z and then using z to generate a sequence of y_t , which can be abstracted into the following joint probability distribution

$$p(x, y, z|\phi) = \underbrace{p(x)}_{\text{Evidence}} \underbrace{p(y_1|\pi)}_{\text{Prior}} \underbrace{\left[\prod_{t=2}^T p(y_t|y_{t-1}, A) \right]}_{\text{Transition Probability}} \underbrace{p(z|x, \theta)}_{\text{Posterior of } z} \underbrace{\left[\prod_{t=1}^T p(y_t|z, \theta) \right]}_{\text{Posterior of } y_t} \quad (3.2)$$

where $\phi = \{\pi, A, \theta\}$ denotes parameters of the proposed representation-layer-embedded Markov model.

The probability distributions in the right-hand side of Equation 3.2 are discussed in details as follows.

- $p(y_1|\pi)$: $\pi \in [0, 1]^2$ is prior probability distribution of the initial EV charging status, in which $\pi_i := p(y_1 = i)$, $i \in \{0, 1\}$, represents the probability that initially an EV is at charging status i . With N labelled initial $y_1^{(n)}$, $n = 1, \dots, N$, π_i can be estimated by

$$\pi_i^* = \begin{cases} \frac{\sum_{n=1}^N y_1^{(n)}}{N} & \text{if } i = 1; \\ 1 - \frac{\sum_{n=1}^N y_1^{(n)}}{N} & \text{if } i = 0. \end{cases} \quad (3.3)$$

- $p(y_t|y_{t-1}, A)$: $A \in \mathbb{R}^{2 \times 2}$ is the probability transition matrix, in which A_{ij} represents

the probability that the EV charging status transits from i at $t - 1$ to j at t :

$$A_{ij} = p(y_t = j | y_{t-1} = i) \quad \text{where} \quad i, j \in \{0, 1\}. \quad (3.4)$$

With N labelled $y_t^{(n)}, n = 1, \dots, N$, A_{ij} can be estimated by

$$A_{ij}^* = \frac{\sum_{n=1}^N \sum_{t=2}^T \mathbf{1}_{ij}(y_t^{(n)} = j | y_{t-1}^{(n)} = i)}{N(T-1)} \quad (3.5)$$

where $\mathbf{1}_{ij}(y_t = \cdot | y_{t-1} = \cdot)$ is an indicator function whose output is 1 if and only if $y_t^{(n)} = j$ and $y_{t-1}^{(n)} = i$.

- $p(z|x, \theta)$ and $p(y_t|z, \theta)$: Recall that an abstract but meaningful feature, z is always unobservable. Therefore, the true posterior distribution of z given x is unknown. This dissertation proposal follows [76, 77] to assume that $p(z|x, \theta)$ takes on an approximate Gaussian form, i.e., a multivariate Gaussian with a diagonal covariance, given as

$$\log p(z|x, \theta) = \log \mathcal{N}(z | \mu_z(x), \sigma_z^2(x) \mathbf{I}) \quad (3.6)$$

where $\mu_z(x)$ and $\sigma_z(x)$ are the mean and standard deviation of z , respectively. Because y_t is a binary variable, it is assumed that y_t follows a Bernoulli distribution

$$\log p(y_t|z, \theta) = \log(\mu_{y_t}(z)^{y_t} (1 - \mu_{y_t}(z))^{1-y_t}) \quad (3.7)$$

where $\mu_{y_t}(z)$ is the mean of y_t , which can also be interpreted as the probability that an EV is at ON charging status given z at t , i.e., $\mu_{y_t}(z) = p(y_t = 1|z, \theta)$. Thus the $p(y|z, \theta)$ is a multivariate Bernoulli distribution that is a product of Bernoulli distribution of each y_t .

In this dissertation, DNNs [69, Ch. 6 and 9] are used to approximate $p(z|x, \theta)$ and $p(y_t|z, \theta)$. Compared to linear regression models, DNNs can better extract features in a

robust manner (against noise and error in data) and generalize to new data. Moreover, recurrent layers are often used in DNNs to process time-series data. However, recurrent layers have difficulties in handling long sequences, such as sequences more than 100 time steps [69, Ch. 10]. Therefore, instead of recurrent layers, this dissertation proposal utilizes fully connected (FC) and Convolutional Neural Network (CNN) layers. The former layer is the weighted sum of inputs transformed by nonlinear activations into outputs. The latter layer employs convolution operations to extract nonlinear features [78]. The CNN layer is demonstrated in Figure 3.6, where

- **Conv1D** is a one-dimensional (1D) convolutional layer applied to time series. Its parameters consist of a set of learnable filters, which can capture features such as the spatial structure (e.g., change points) and local information (e.g., magnitudes) of input time series. During the forward pass, each filter is convolved from the beginning of a time series towards its end, computing the dot product between the entries of the filter and the input and producing a feature map of that filter. For instance, an example time series $1, 1, 2, -1, 1, -2, 1$ with a filter $1, 0, -1$ is given in Figure 3.6. The convolution operation can be considered as the dot product, e.g., the first output -1 after the convolution operation can be calculated as $1 \times 1 + 1 \times 0 + 2 \times (-1) = -1$. Note that the output $-1, 2, 1, 1, 0$ after the convolution operation is called a feature map.
- **UpSampling1D** is an operation to upsample (i.e., repeat and resize) the feature map. For instance, given a feature map $-1, 2, 1, 1, 0$ as shown in Figure 3.6, the output after the upsampling operation with factor 2 is $-1, -1, 2, 2, 1, 1, 1, 1, 0, 0$.
- **MaxPooling1D** is an operation to reduce the input size by taking the maximum value of sliding windows in the original input. For instance, considering the feature map $-1, 2, 1, 1, 0$, the output after **MaxPooling1D** with size 3 can be calculated as $2, 2, 1$, where the corresponding sliding windows are $[-1, 2, 1]$, $[2, 1, 1]$, and $[1, 1, 0]$.

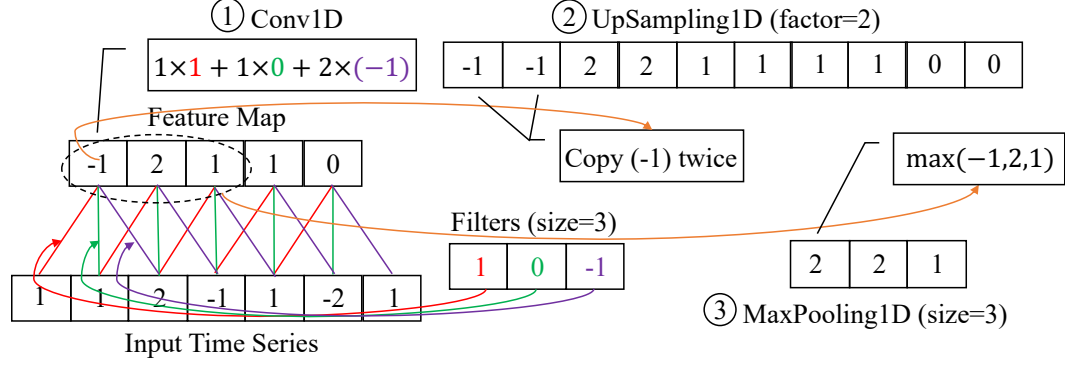


Figure 3.6: Demonstration of a CNN layer, which consists of three sequential operations: Conv1D, UpSampling1D, and MaxPooling1D.

The above three operations can be performed with or without overlaps and paddings, which would imply different final output sizes. The complete network architecture and parameter settings of the proposed DGM are shown in Table 3.1, which presents approximation actions of $p(z|x, \theta)$ and $p(y|z, \theta)$ in each layer. The approximation of $p(z|x, \theta)$ is carried out by two convolutional layers with the same padding, two pooling layers with stride size of 2, and a FC layer for $\mu_z(x)$ and $\log \sigma_z(x)^2$. Moreover, the approximation of $p(y|z, \theta)$ is conducted by a FC layer, two upsampling layers, and two convolutional layers with the same padding. Since $\mu_{y_t}(z)$ is a probability, the sigmoid function is selected to be the activation of the last layer.

Compared with FC layers, the proposed DGM can benefit from convolutional layers. On one hand, convolutional layers can automatically capture useful features from local patterns. As shown in Figure 3.7, an one-dimensional convolutional layer with 16 filters is applied to a sample aggregated power consumption profile to illustrate this advantage. Each subplot shows the identical power consumption profile (lines in black), EV charging operation status (lines in red), and feature map outcomes of one filter (lines in blue). Each trained filter aims at extracting both the spatial structure (e.g., change points) and local information from the input and providing feature maps for its downstream layer. In this illustration, filters #3, #4, #6, #7, #8, #12, #13, and #16 can effectively extract feature maps that match the EV charging status change points (step changes in red lines). The other

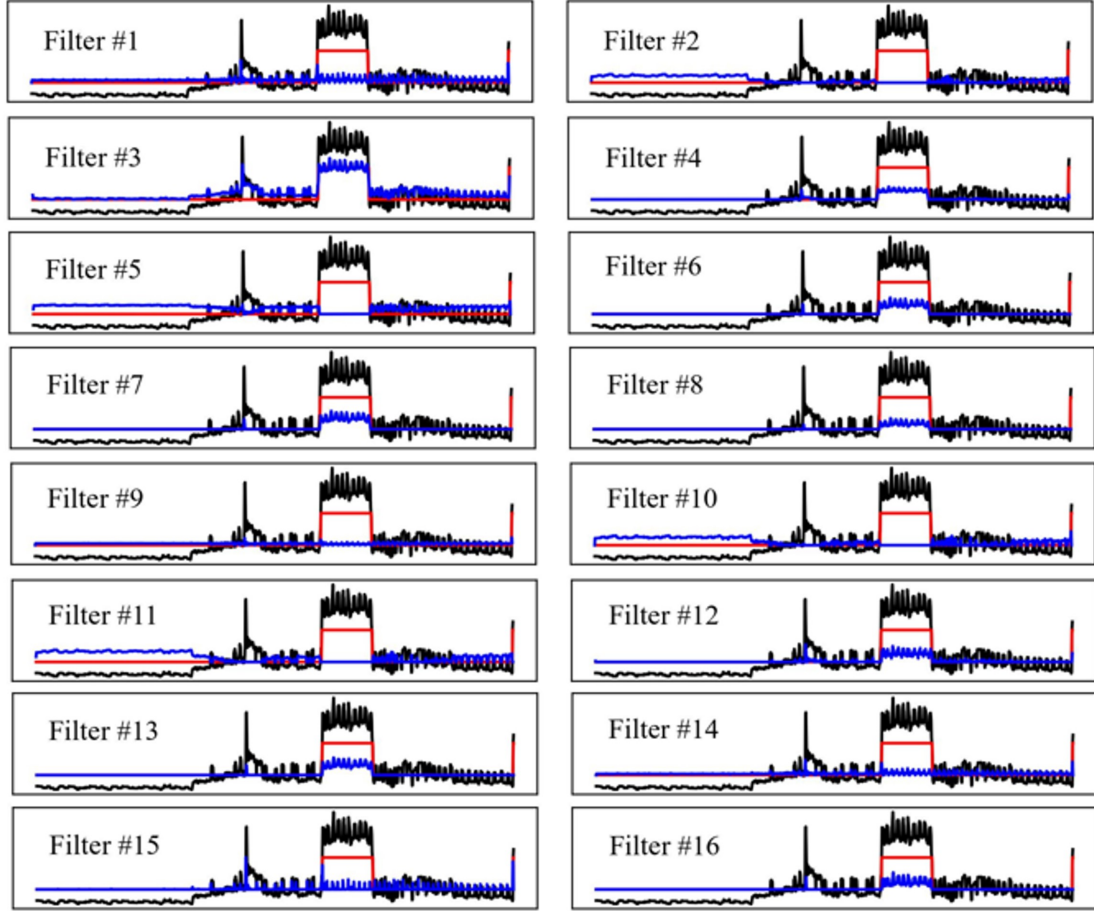


Figure 3.7: Visualization of feature maps extracted from convolutional layers with sixteen filters.

filters also introduce positive impacts, but not as strong as these ones. For different inputs, in general there would be a different subset of filters that perform dominating roles. On the other hand, convolutional layers with weight sharing can reduce the number of neural network weights and indirectly prevent overfitting.

Remark 1: The number of weights in convolutional layers depends on the product of the number and the size of convolution filters (e.g., 16×3), while the number of weights in FC layers relies on the product of the input and output size (e.g., 360×50). In general, the latter is much larger than the former. Therefore, given the same number of layers, a DNN with more convolution layers requires less weights.

Remark 2: Though it is generally true that recurrent layers perform better in capturing

Table 3.1: The proposed DGM architecture and parameter settings

Layer Name	$p(z x, \theta)$	$p(y z, \theta)$
Input	1440	50
Layer1	Conv1D,16,3,ReLU	FC,360,ReLU
Layer2	MaxPooling1D,2	UpSampling1D,2
Layer3	Conv1D,1,3,ReLU	Conv1D,16,3,ReLU
Layer4	MaxPooling1D,2	UpSampling1D,2
Layer5	FC,50/50,-	Conv1D,1,3,sigmoid

*Conv1D denotes 1D convolution layer followed by number and size of filters and an activation layer; MaxPooling1D denotes 1D max pooling layer followed by size of the max pooling windows; UpSampling1D denotes 1D upsampling layer followed by upsampling factors; FC denotes a fully connected layer followed by number of neurons and an activation layer

temporal information than convolutional layers, they tend to have difficulties in handling very long time series. Some recent results [78, 79] have shown that convolutional layers can achieve better performances than recurrent layers in some applications. Moreover, this dissertation proposal utilizes convolutional layers for different purposes, such as extracting spatial structures. As a result, the temporal information is not processed in the neural network part of the proposed model. However, since the proposed DGM retains the Markov property of HMMs, the corresponding temporal information is addressed by transition probabilities. In other words, convolutional layers with Markov property is proposed here as an alternative of recurrent layers for time series.

3.3.3 Supervised Learning in DGM

With labelled dataset $(\mathcal{X}, \mathcal{Y})$, θ in Equation 3.6 and Equation 3.7 can be determined by maximum likelihood [75]. The marginal distribution of each $(x, y) \in (\mathcal{X}, \mathcal{Y})$ is obtained from the joint distribution Equation 3.2 by marginalizing over the latent variable z

$$\log p(x, y|\theta) = \log \int_z p(x, y, z|\theta) dz. \quad (3.8)$$

Maximizing Equation 3.8 could lead to complicated expressions with no closed-form solutions since 1) the integral of the marginal distribution is intractable when $p(z|x, \theta)$ and $p(y|z, \theta)$ are approximated by DNNs with nonlinear hidden layers and 2) batch optimization is costly for large amount of data. Following recent advances in variational inference [80], the proposed DGM can be trained by maximizing the Evidence Lower Bound (ELBO) under data distribution. A lower bound $\mathcal{L}(\theta|x, y)$ on the marginal distribution of (x, y) is given by

$$\begin{aligned} \mathcal{L}(\theta|x, y) = & \log p(x) + \log p(y_1) + \sum_{t=2}^T \log p(y_t|y_{t-1}) \\ & + \sum_{t=1}^T E_{z \sim p(z|x, \theta)} [\log p(y_t|z, \theta)]. \end{aligned} \quad (3.9)$$

Therefore, the ELBO under data distribution $x, y \sim p_{data}$ can be written as

$$\mathcal{L}(\theta|\mathcal{X}, \mathcal{Y}) = E_{x, y \sim p_{data}} [\mathcal{L}(\theta|x, y)]. \quad (3.10)$$

Then Equation 3.6 and Equation 3.7 can be trained by

$$\theta^* = \arg \min_{\theta} -\mathcal{L}(\theta|\mathcal{X}, \mathcal{Y}). \quad (3.11)$$

Since z is stochastic and thus gradients cannot be backpropagated, reparameterization is used to sample z , i.e., given x and a unit Gaussian noise $\epsilon \sim \mathbf{N}(\mathbf{0}, \mathbf{I})$, $z = \mu_z(x) + \epsilon \sigma_z(x)$. Note that the noises injected into the representation layer enables the proposed DGM to learn continuous feature representations. Note that such a sampling process of z is similar to the Variational Autoencoder (VAE) [76]. Therefore, in this dissertation proposal the number of samples z is set to be 1 with a large minibatch size N_b in accordance with the experimental setting in the VAE. Based on Equation 3.6, Equation 3.7, and Equation 3.11, it can be concluded that

$$\begin{aligned}
\theta^* = \arg \min_{\theta} & - \sum_{n_b=1}^{N_b} \sum_{t=1}^T y_t^{(n_b)} \log \mu_{y_t}(z^{(n_b)}) \\
& + (1 - y_t^{(n_b)}) \log(1 - \mu_{y_t}(z^{(n_b)})) \leftarrow \mathbf{Loss}
\end{aligned} \tag{3.12}$$

where $z^{(n_b)} = \mu_z(x^{(n_b)}) + \epsilon^{(n_b)} \sigma_z(x^{(n_b)})$ and $\epsilon^{(n_b)} \sim \mathbf{N}(\mathbf{0}, \mathbf{I})$, $x^{(n_b)}$ and $y^{(n_b)}$ are the n_b -th instance from minibatch, $z^{(n_b)}$ is generated from $x^{(n_b)}$, and $y_t^{(n_b)}$ is the t -th element of $y^{(n_b)}$. From Equation 3.12, it can be seen that the training objective is to minimize the binary multi-label classification loss. In this dissertation proposal, $p(z|x, \theta)$ and $p(y_t|z, \theta)$ are both differentiable functions containing different neural layers composed of multilayer perceptrons, convolution, max-pooling, upsampling, Rectified Linear Unit (ReLU), and sigmoid. Therefore, the gradient-descent based training methods Adam [81] is applied, which is fairly insensitive to the choice of hyperparameters.

Furthermore, this dissertation proposal utilizes the minibatch training, also known as the minibatch gradient descent, which is a variation of the gradient descent algorithm that splits the training dataset into small batches. The implementation flowchart of the minibatch training in DGM is shown in Figure 3.8. For each batch, the forward propagation first generates z and outputs $\mu_{y_t}(z)$, and then the back propagation calculates model loss and update model weights.

3.4 The Proposed DGM: Exact Inference

Once the proposed model is trained with ϕ^* , the next step is to infer EV charging status y^* given aggregated consumption profile x via MAP, i.e.,

$$y^* = \arg \max_y p(y|x, \phi^*), \tag{3.13}$$

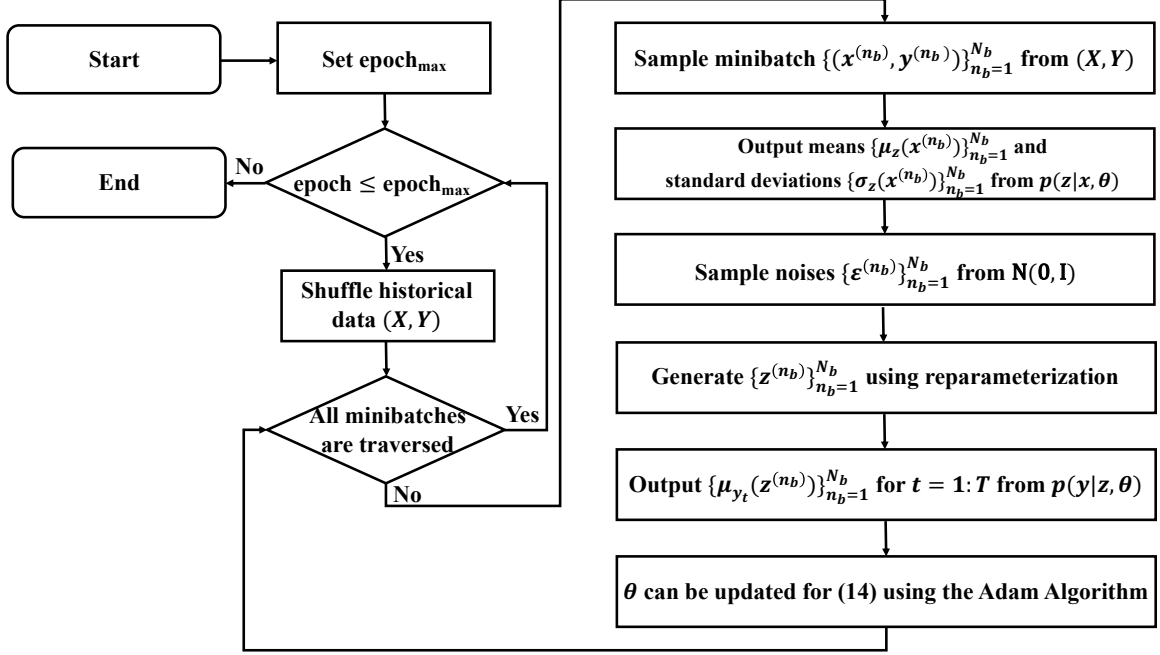


Figure 3.8: Implementation flowchart of the minibatch training in DGM, including both forward and backward propagation.

which is approximated (with z sampled from $p(z|x, \theta^*)$) by

$$y^* = \arg \max_y \log p(y_1) + \sum_{t=2}^T \log p(y_t|y_{t-1}) + \sum_{t=1}^T \log p(y_t|z). \quad (3.14)$$

Note that y^* in (Equation 3.14) can be further inferred via Dynamic Programming (DP) in two stages. For forward induction, at each time step t , the first step is to solve the following

$$F_c(t, y_t) = \min_{y_{t-1}} \{F_c(t-1, y_{t-1}) - \log p(y_t|z) - \log p(y_t|y_{t-1})\} \quad (3.15)$$

where $F_c(t, y_t)$ is the optimal cost function over time steps t and $t-1$ given y_t , and $F(1, y_1) = -\log p(y_1) - \log p(y_1|z)$. An example is shown in Figure 3.9 to demonstrate the calculation of $F_c(t, y_t)$, where the values on the nodes are the cost (or values of the negative logarithm of posterior) and the values on the directed edges are the cost of transporting a unit from one node to the other (i.e., the negative logarithm of transition probability).

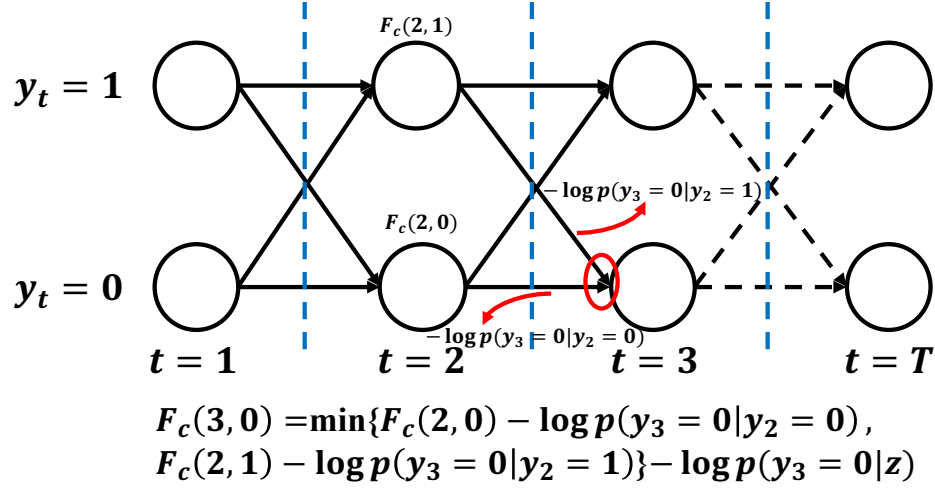


Figure 3.9: Example demonstration of how to calculate $F_c(t = 3, y_t = 0)$ where the red sign highlights the possible minimum cost paths (the directed edges) from the last time step ($t = 2$) to the current time step ($t = 3$).

Therefore, the overall process of forward induction is to find the minimum cost. Note that the complexity of this implementation is $O(4T)$.

For backward induction, the second step is solve the following and find the minimum cost route,

$$y_{t-1}^* = \arg \min_{y_{t-1}} \{F_c(t-1, y_{t-1}) - \log p(y_t^*|z) - \log p(y_t^*|y_{t-1})\} \quad (3.16)$$

where $y_T^* = \arg \min_{y_T} F(T, y_T)$. The complexity of this implementation is $O(2T)$. Figure 3.10 shows four typical inference results corresponding to (a) once-charging , (b) twice-charging in day and night, (c) twice-charging in two nights, and (d) multiple-charging, respectively. It can be observed that the measured and inferred EV charging status are almost identical, which validates the effectiveness of the proposed framework. The incorporation of $p(y_1)$ and $p(y_t|y_{t-1})$ into the graph enables the model to consider the past events at the expense of increased computational complexity of inference.

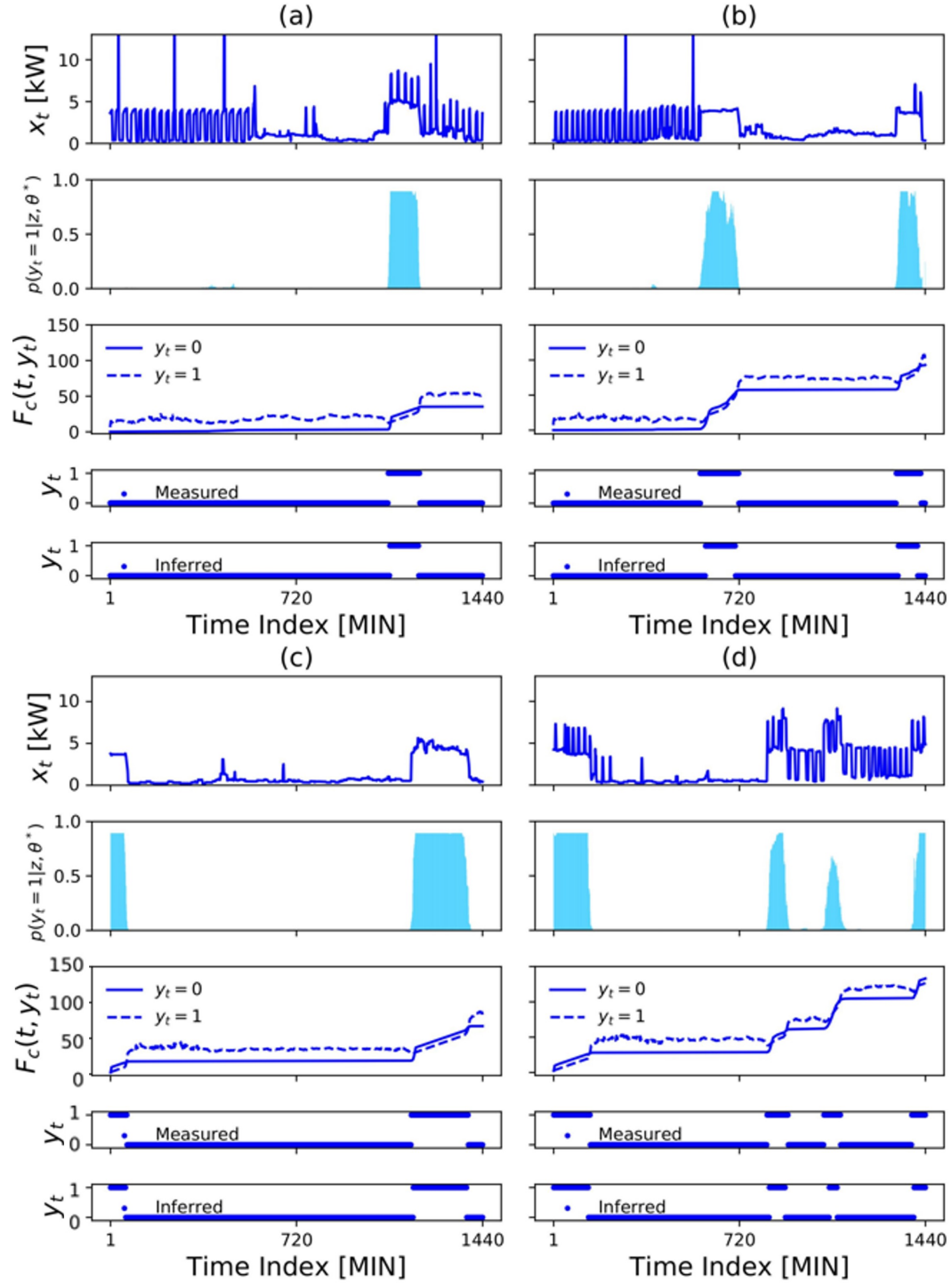


Figure 3.10: Selected typical examples of inference results including (a) once-charging, (b) twice-charging in day and night, (c) twice-charging in two nights, and (d) multiple-charging, respectively. Each row shows the aggregated power consumption profile, the probability of an EV at ON, the optimal cost over time steps t and $t - 1$ given y_t , the measured EV charging status profile, and the inferred EV charging status profile, respectively.

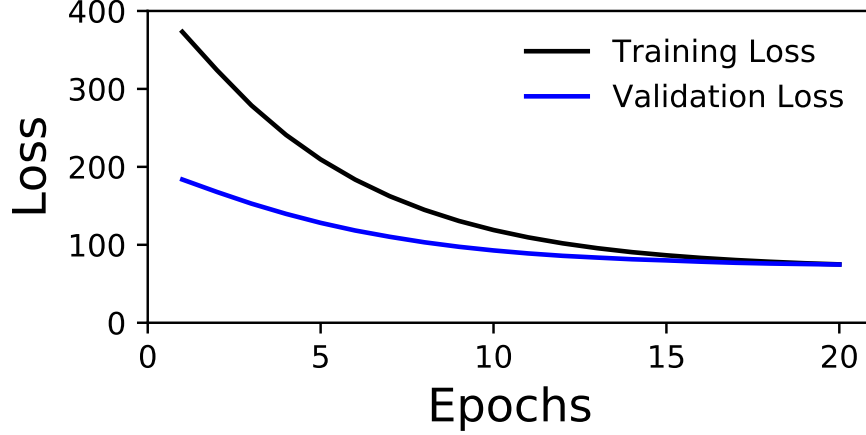


Figure 3.11: Illustration of convergence of the training (black line) and validation (blue line) losses as the epochs increase during training.

3.5 Numeric Results

In this section, the proposed algorithm is validated on the Pecan Street dataset [82], which consists of measurement of circuit-level household electricity consumption data from nearly 1,000 homes across the U.S. Each such home have eight extra channels to record power consumption by major appliances such as Heating, Ventilation and Air-Conditioning (HVAC), refrigerators, and EVs.

3.5.1 Experiment Setup and Evaluation Metrics

The DGM in this dissertation proposal is trained using Adam with an epoch of 20, a mini-batch size of 100, and a learning rate of 0.001. All neuron weights are initialized using Glorot initialization [83]. After data pre-cleaning with removal of bad data points, the aggregated power consumption profiles and EV charging profiles are then standardized and binarized, respectively. The main program is executed on an Intel i7-7820X 8-Core CPU while the training of the proposed DGM including the forward and backward propagation is implemented on a TITAN Xp GPU using TensorFlow as the computational framework. It is observed that the loss of the model goes to convergence as the epoch increases, as shown in Figure 3.11. To evaluate performance of the proposed algorithm, the following

classification metrics are employed,

$$\begin{aligned}\text{Accuracy} &= \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \\ \text{Recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{Precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ \text{F1} &= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}\end{aligned}$$

where

- TP is the *true positive* indicator, i.e., is the number of cases where the DGM classifies the EV charging status as ON and the actual status is indeed ON;
- TN is the *true negative* indicator, i.e., the number of cases where the DGM classifies the EV charging status as OFF and the actual status is indeed OFF;
- FP is the *false positive* indicator, i.e., the number of cases where the DGM classifies the EV charging status as ON but the actual status is OFF; and
- FN is the *false negative* indicator, i.e., the number of cases where the DGM classifies the EV charging status as OFF but the actual status is ON.

3.5.2 EV Charging Status Classification

There are 93 houses with EV charging activities in the Pecan Street dataset, with one data point per minute per house. With each aggregated power consumption profile defined to be of 24 hours, i.e., daily profiles. In this work, both transfer-learning-based and non-transfer-learning-based settings are utilized for the purpose of comparison on performances. The difference between these two settings is that for non-transfer learning, the set of houses used in training is typically the same as the set of houses used in testing. On the contrast, for transfer learning, the set of houses used in testing is typically different

from those used in training, which is a powerful method to check whether a certain model can “transfer” knowledge from one dataset to another.

For the non-transfer learning setting, to reduce bias and variance caused by the source data and better evaluate the effectiveness of the proposed DGM, the five-fold cross-validation (i.e., all available data is first shuffled and divided into five subsets, and each trial takes one subset for testing and the other four subsets for training) [75] is performed. As shown in Figure 3.12, the box plot and the green triangle are used to visualize the variance and mean of evaluation results for the following three scenarios, respectively.

- The first scenario is the proposed DGM without any noise injected into the test sets, denoted as “DGM w/o noise”;
- The second scenario is the proposed DGM with a Gaussian noise (zero mean and half standard deviation, i.e., $\pm 0.5kW$), denoted as “DGM w/ noise”;
- The third scenario is the HMM without any noise, denoted as “HMM w/o noise”.

It can be observed that the variance of evaluation results is small for all five data partitions, and thus the proposed DGM is reasonably stable. On average, the proposed DGM increases accuracy, precision, and F1 by 8.20%, 134.39%, 56.43%, respectively, at the cost of reducing recall by 19.31% compared with the HMM on the five different data partitions. In terms of F1, the proposed DGM is better than the HMM with better average performance of accuracy.

Furthermore, another comparative experiment is performed to demonstrate the robustness of the proposed DGM against noise in the data. From Figure 3.12, the performance of the proposed DGM could be slightly affected by noise and error in the data (the accuracy, precision, recall, and F1 drop by 0.82%, 2.83%, 19.02%, and 13.61%). However, at a reasonable noise level, the proposed DGM still greatly outperforms HMM. Detailed evaluation results are provided in Table 3.2

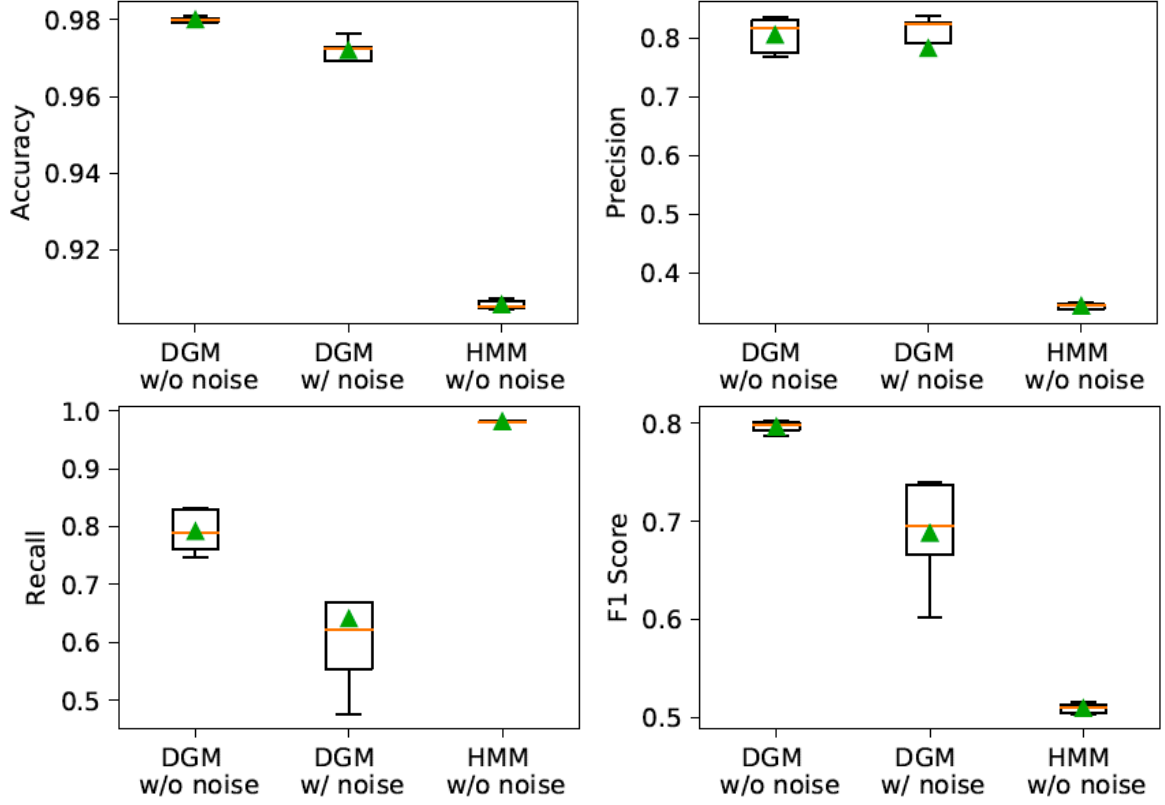


Figure 3.12: Performance comparison using five-fold cross-validation and non-transfer learning settings.

Table 3.2: Performance comparison using five-fold cross-validation and non-transfer learning setting

	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Overall
Accuracy	0.9794 / 0.9692 / 0.9055	0.9801 / 0.9724 / 0.9077	0.9802 / 0.9763 / 0.9084	0.9794 / 0.9728 / 0.9061	0.9809 / 0.9692 / 0.9058	0.9800±0.0006 / 0.9720±0.0027 / 0.9057±0.0011
Precision	0.7683 / 0.6348 / 0.3409	0.8369 / 0.8373 / 0.3494	0.8306 / 0.8275 / 0.3527	0.7751 / 0.7913 / 0.3466	0.8169 / 0.8233 / 0.3417	0.8056±0.0285 / 0.7828±0.0756 / 0.3437±0.0045
Recall	0.8318 / 0.8818 / 0.9822	0.7462 / 0.5548 / 0.9827	0.7613 / 0.6700 / 0.9820	0.8301 / 0.6224 / 0.9826	0.7894 / 0.4765 / 0.9819	0.7917±0.0035 / 0.6411±0.1369 / 0.9812±0.0004
F1 Score	0.7978 / 0.7370 / 0.5061	0.7879 / 0.6661 / 0.5155	0.7935 / 0.7394 / 0.5190	0.8006 / 0.6953 / 0.5124	0.8021 / 0.6021 / 0.5070	0.7964±0.0052 / 0.6880±0.0509 / 0.5091±0.0049

Note that top/middle/bottom numbers in each cell represent metrics of DGM w/o noise, DGM w/ noise, and HMM w/o noise, respectively.

For transfer learning setting, all data are split into the training dataset of 73 households and testing dataset shown in the first column of Table III. For household with dataid 6871, it is an extreme case where there is no any EV charging events. From Table 3.3, it is shown that the proposed DGM increases accuracy, precision, and F1 by 10.8%, 89% and 42.8% at the cost of reducing recall by 17.2% compared with the HMM. In terms of F1, DGM is better than HMM with high average performance of accuracy.

Compared to HMMs, it can be observed that the proposed DGM achieved better performance in *accuracy*, *precision*, and *F1* but only received a lower score in *recall* under both settings. That is because that HMMs cannot accurately classify the EV charging status as “OFF”, i.e., HMMs classifies most “OFF” statuses as “ON” wrongly while the proposed DGM method can mitigate this issue of the HMM method at the cost of classifying a small number of “ON” statuses as “OFF”.

3.5.3 EV Charging Profile Elements Analyses

Four houses (dataid 3036 (a), 370 (b), 1782 (c), and 2018 (d)) are selected according to their different F1 values from high to low. Four elements of EV charging profiles are extracted from measured and corresponding classified EV charging status. Once the extracted results are collected, the distribution of EV charging profile elements can be visualized by Gaussian mixture. As shown in Figure 3.13, the distribution of EV charging profile elements from measured and classified results are represented by the blue and red lines respectively. For each household, it can be seen that the red line is almost identical to the blue line. That is, the distribution of the measured elements can be well approximated by the classified elements. Therefore, the proposed framework is accurate and effective. According to the distribution of the classified elements, it can be summarized that most households charge their EVs after work around 6 p.m and also tend to charge their EVs for one hour and once per day. So these information can be further analyzed to achieve more accurate EV charging profiles.

Table 3.3: Performance Comparison Using Transfer Learning Setting

Dataid	Accuracy [HMM]	Accuracy [DGM]	Precision [HMM]	Precision [DGM]	Recall [HMM]	Recall [DGM]	F1 Score [HMM]	F1 Score [DGM]
370	97.49	98.54	74.50	95.49	98.04	83.28	84.67	88.97
545	94.63	98.68	50.86	95.64	96.40	79.93	66.59	87.08
1185	87.96	97.17	31.64	71.84	97.21	82.34	47.74	76.73
1782	85.85	97.70	28.40	86.04	98.23	70.94	44.06	77.77
2018	77.79	96.11	21.69	67.25	99.30	72.33	35.61	69.70
2335	88.90	96.30	20.48	43.48	96.52	86.88	33.79	57.96
2769	90.58	98.34	31.16	79.26	96.69	83.55	47.13	81.35
3036	98.09	99.28	79.74	96.55	97.11	92.98	87.57	94.73
3367	88.59	98.41	35.54	87.67	98.05	87.28	52.17	87.48
4373	93.40	95.00	67.21	95.96	94.86	63.71	78.68	76.58
4641	97.13	96.18	86.22	92.49	93.27	77.56	89.61	84.37
4957	89.28	98.38	30.25	86.72	99.77	77.05	46.42	81.60
5357	53.17	91.67	6.53	22.62	99.85	63.80	12.25	33.40
5749	92.53	98.34	49.72	84.18	99.87	95.40	66.39	89.44
5786	81.68	96.20	2.15	8.29	99.23	83.30	4.20	15.08
6139	93.16	98.56	45.36	86.13	96.48	89.22	61.70	87.65
6871	91.29	92.66	-	-	-	-	-	-
7863	94.61	98.86	50.10	94.04	98.35	84.20	66.38	88.85
8197	67.49	98.11	12.94	79.00	99.62	83.06	22.90	80.98
8669	88.25	96.20	42.71	77.35	98.64	80.32	59.61	78.81
Overall	87.59 ± 10.58	97.03 ± 2.05	40.38 ± 24.03	76.32 ± 25.07	97.76 ± 1.79	80.90 ± 8.58	53.02 ± 24.34	75.71 ± 20.15

3.6 Conclusion

This chapter proposes a DGM driven non-intrusive identification framework for EV charging profile. With the capability of complex density estimation by DGMs, the EV charging status can be modeled and inferred from DGMs via DP. Then EV charging profiles can be reconstructed according to the rated power of EV models and inferred status. Experiments on Pecan Street datasets were conducted to validate the feasibility and effectiveness of the proposed framework. The numerical results show that the proposed method can improve the overall performance compared with the state-of-art HMMs, though

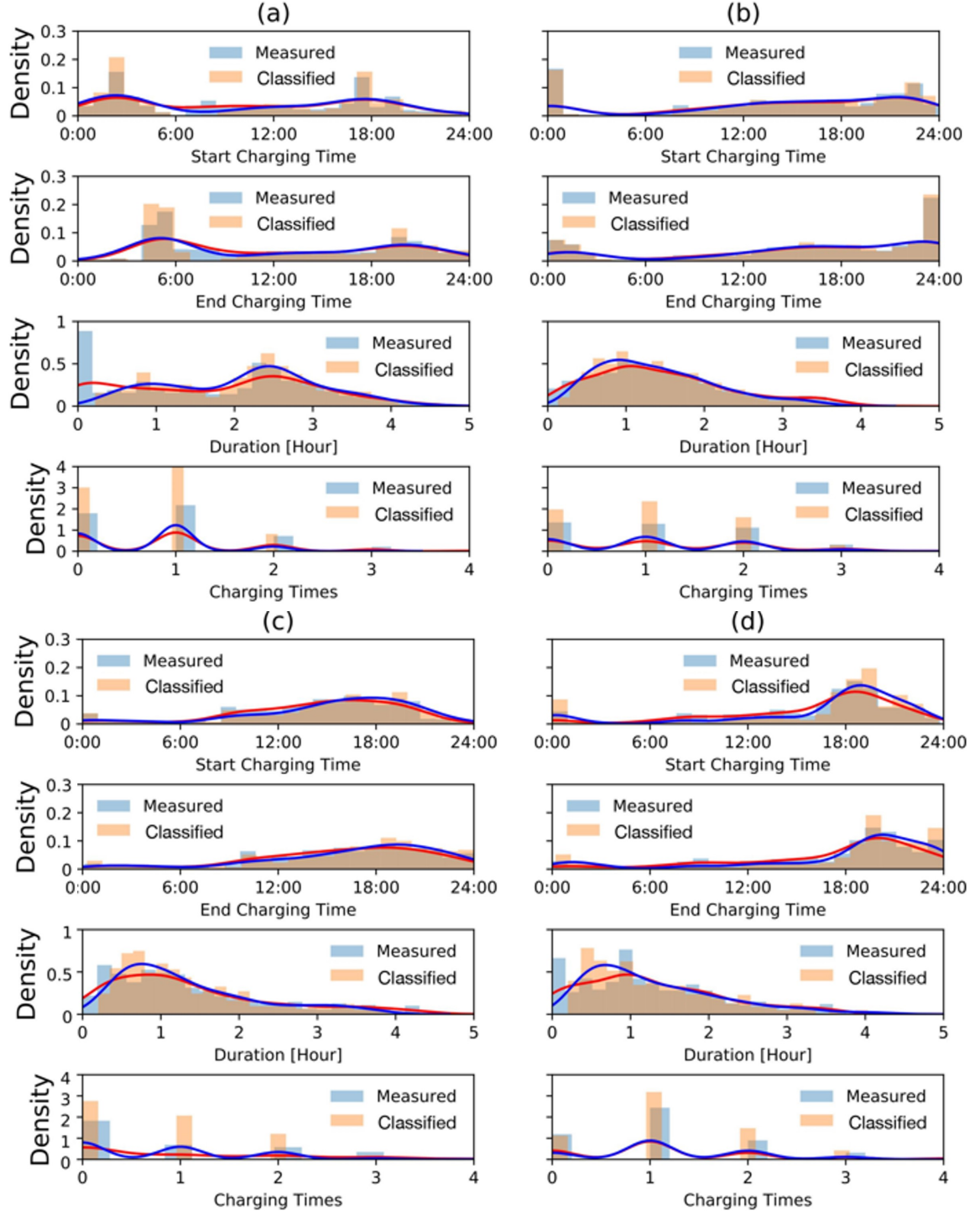


Figure 3.13: The analysis of some example EV charging profile elements including start charging time, end charging time, duration, and charging times for four representative houses in terms of their F1 values, where the blue and red lines are the measured and classified distributions, respectively.

a decrease in the recall was observed. In addition, the proposed framework can well handle noisy and unseen data and thus possesses improved robustness and generalization capabilities. For future research, the proposed framework can be extended to more general multi-class multi-label classification tasks.

CHAPTER 4

LEARNING-BASED DECISION MAKING UNDER UNCERTAINTY

The ever-growing higher penetration of DERs in LV distribution systems brings both opportunities and challenges to voltage support and regulation. This chapter proposes a DRL-based scheduling scheme of ESSs to mitigate system voltage deviations in unbalanced LV distribution networks. The ESS-based voltage regulation problem is formulated as a multi-stage quadratic stochastic program, with the objective of minimizing the expected total daily voltage regulation cost while satisfying operational constraints. While existing voltage regulation methods are mostly focused on one-time-step control, this dissertation proposal explores a day-horizon system-wide voltage regulation problem. In other words, the size of action and state spaces are extremely high-dimensional and need to be delicately handled. Furthermore, in order to overcome the difficulty of modeling uncertainties and develop a real-time solution, a learn-to-schedule feedback control framework is proposed by adapting the problem to a model-free DRL setting.

The remainder of this chapter is organized as follows. Section 3.1 presents a LinDistFlow model for multi-phase UDS. Section 3.2 proposes a quadratic system voltage regulation cost and defines a day-horizon ESS scheduling problem for voltage regulation, which will be formulated as an MDP in Section 3.3. A learn-to-schedule feedback control framework via SAC is presented in Section 3.4. The proposed algorithm is first tested on a customized 6-bus system and then validated on a modified IEEE 34-bus system in Section 3.5. Note that a deterministic Quadratic Programming (QP) based control strategy is used as a benchmark method to evaluate the near-optimality of the proposed strategy. Similar practice is taken in [59] for method comparison. Finally, Section 3.6 concludes this chapter.

4.1 Modeling of Multi-phase Unbalanced LV Distribution Network

Consider a three-phase distributed network represented by a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} := \{0, 1, \dots, N\}$ denotes the set of all nodes (i.e., buses) and \mathcal{E} denotes the set of edges (i.e., distribution lines). Note that this work follows the literature [53] to denote the substation bus as bus 0, and thus the set of the remaining buses is denoted as $\mathcal{N}^+ = \mathcal{N} \setminus \{0\}$. Moreover, each bus $i \in \mathcal{N}^+$ is connected to a unique parent bus π_i and a set of child buses denoted by \mathcal{C}_i . Without loss of generality and for notational simplicity, the line from bus π_i to bus i is unique to bus i and thus can be denoted by the same index as line $i \in \mathcal{E} = \mathcal{N}^+$. Consequently, all buses are labeled in a manner that the index of each bus is always greater than that of its parent bus. In other words, $i > \pi_i$, i.e., monotonically increasing from the substation bus 0 to rural buses.

Furthermore, let $\mathbf{A}^0 = [\mathbf{a}_0 \ \mathbf{A}] \in \mathbb{R}^{N \times (N+1)}$ denote the line-bus incidence matrix, where the first column \mathbf{a}_0 denotes line-bus incidences associated with bus 0 except itself and \mathbf{A} denotes line-bus incidences of the remaining network. Therefore, its entry $A_{ij}^0 = 1$ if $j = \pi_i$ or $A_{ij}^0 = -1$ if $j = i$, and $A_{ij}^0 = 0$ otherwise. Since \mathcal{G} is a connected tree, \mathbf{A} is invertible with the following property

$$\mathbf{A}^{-1} \mathbf{a}_0 = \mathbf{1}_N \quad (4.1)$$

where $\mathbf{1}_N$ is a vector of ones. Consequently, (complex) variables associated with each bus (or line) i at t are represented by 3×1 vectors $\dot{\mathbf{V}}_{\pi_i, t}$ and $\dot{\mathbf{I}}_{i, t}$, respectively, which are coupled by

$$\dot{\mathbf{V}}_{i, t} = \dot{\mathbf{V}}_{\pi_i, t} - \dot{\mathbf{z}}_i \dot{\mathbf{I}}_{i, t}, \quad (4.2)$$

where $\dot{\mathbf{V}}_{i, t} := [\dot{V}_{i, t}^a \ \dot{V}_{i, t}^b \ \dot{V}_{i, t}^c]^\top$ denotes the three-phase voltages of bus i , $\dot{\mathbf{I}}_{i, t} := [\dot{I}_{i, t}^a \ \dot{I}_{i, t}^b \ \dot{I}_{i, t}^c]^\top$ denotes the three-phase currents of bus i , and the complex matrix $\dot{\mathbf{z}}_i = \mathbf{r}_i + j\mathbf{x}_i \in \mathbb{C}^{3 \times 3}$

is the impedance matrix of line i ¹. Finally, squared voltage magnitudes in distribution networks, denoted by $\mathbf{v}_{i,t} := [v_{i,t}^a \ v_{i,t}^b \ v_{i,t}^c]^\top$, can be obtained as follows

$$\begin{aligned} \mathbf{v}_{i,t} &= \dot{\mathbf{V}}_{i,t} \odot \dot{\mathbf{V}}_{i,t}^* = (\dot{\mathbf{V}}_{\pi_i,t} - \dot{\mathbf{z}}_i \dot{\mathbf{I}}_{i,t}) \odot (\dot{\mathbf{V}}_{\pi_i,t} - \dot{\mathbf{z}}_i \dot{\mathbf{I}}_{i,t})^* \\ \Rightarrow \mathbf{v}_{i,t} &= \mathbf{v}_{\pi_i,t} - \dot{\mathbf{V}}_{\pi_i,t} \odot \dot{\mathbf{z}}_i^* \dot{\mathbf{I}}_{i,t}^* - (\dot{\mathbf{V}}_{\pi_i,t} \odot \dot{\mathbf{z}}_i^* \dot{\mathbf{I}}_{i,t}^*)^* + \dot{\mathbf{z}}_i \dot{\mathbf{I}}_{i,t} \odot \dot{\mathbf{z}}_i^* \dot{\mathbf{I}}_{i,t}^* \\ &= \mathbf{v}_{\pi_i,t} - 2\text{Re}[\dot{\mathbf{V}}_{\pi_i,t} \odot \dot{\mathbf{z}}_i^* \dot{\mathbf{I}}_{i,t}^*] + \dot{\mathbf{z}}_i \dot{\mathbf{I}}_{i,t} \odot \dot{\mathbf{z}}_i^* \dot{\mathbf{I}}_{i,t}^* \end{aligned} \quad (4.3)$$

where \odot is the element-wise product operator, i.e., Hadamard product. Note that the last term $\dot{\mathbf{z}}_i \dot{\mathbf{I}}_{i,t} \odot \dot{\mathbf{z}}_i^* \dot{\mathbf{I}}_{i,t}^*$ in Equation 4.3 is relatively small due to the small entries of $\dot{\mathbf{z}}_i$ and thus can be neglected.

The power balance equation at each bus i is given by

$$\underbrace{\dot{\mathbf{S}}_{i,t}}_{\text{power flow from } \pi_i} - \underbrace{(\dot{\mathbf{z}}_{i,t} \dot{\mathbf{I}}_{i,t}) \odot \dot{\mathbf{I}}_{i,t}^*}_{\text{loss}} - \underbrace{\sum_{j \in \mathcal{C}_i} \dot{\mathbf{S}}_{j,t}}_{\text{power flow from } i} = - \underbrace{\dot{\mathbf{S}}_{i,t}}_{\text{power injection}} \quad (4.4)$$

where $\dot{\mathbf{S}}_{i,t} := [\dot{S}_{i,t}^a \ \dot{S}_{i,t}^b \ \dot{S}_{i,t}^c]^\top$ denotes the power flow on line i and $\dot{\mathbf{s}}_{i,t} := [\dot{s}_{i,t}^a \ \dot{s}_{i,t}^b \ \dot{s}_{i,t}^c]^\top$ is the power injection at bus i . Since distribution line losses are relatively small compared to line flows and often neglected, Equation 4.4 can be written as

$$\dot{\mathbf{S}}_{i,t} - \sum_{j \in \mathcal{C}_i} \dot{\mathbf{S}}_{j,t} = -\dot{\mathbf{s}}_{i,t}. \quad (4.5)$$

The power flow formulation in distribution networks is typically approximated linearly, with the assumption that voltages are nearly balanced, i.e., $|\dot{V}_{i,t}^a| \approx |\dot{V}_{i,t}^b| \approx |\dot{V}_{i,t}^c| \approx \tilde{V}_{i,t}$ and $\frac{\dot{V}_{i,t}^a}{\tilde{V}_{i,t}} \approx \frac{\dot{V}_{i,t}^b}{\tilde{V}_{i,t}} \approx \frac{\dot{V}_{i,t}^c}{\tilde{V}_{i,t}} \approx e^{j\frac{2\pi}{3}}$. Surrogating $\dot{\mathbf{V}}_{i,t}$ by $\tilde{V}_{i,t} \dot{\alpha}$, the current vector $\dot{\mathbf{I}}_{i,t}^*$ can be approximately expressed as

$$\dot{\mathbf{I}}_{i,t}^* = \dot{\mathbf{S}}_{i,t} \oslash \dot{\mathbf{V}}_{\pi_i,t} \approx \frac{\dot{\alpha}^* \odot \dot{\mathbf{S}}_{i,t}}{\tilde{V}_{\pi_i,t}} \quad (4.6)$$

¹The dot over an entity (i.e., vector or matrix) means that it is complex.

where \odot is the element-wise division product, $\dot{\alpha} := [1 \ \alpha \ \alpha^2]^\top$ and $\alpha = e^{-j\frac{2\pi}{3}}$. Consequently,

$$\dot{\mathbf{V}}_{\pi_i,t} \odot \dot{\mathbf{z}}_i^* \dot{\mathbf{I}}_{i,t}^* \approx \dot{\alpha} \odot [\dot{\mathbf{z}}_i^* (\dot{\alpha}^* \odot \dot{\mathbf{S}}_{i,t})] \approx \mathbf{diag}(\dot{\alpha}) \dot{\mathbf{z}}_i^* \mathbf{diag}(\dot{\alpha}^*) \dot{\mathbf{S}}_{i,t} \approx \dot{\mathbf{z}}_i^* \dot{\mathbf{S}}_{i,t} \quad (4.7)$$

where $\mathbf{diag}(\mathbf{x})$ is the diagonal matrix with \mathbf{x} on its diagonal, and $\dot{\mathbf{z}}_i := \mathbf{diag}(\dot{\alpha}^*) \dot{\mathbf{z}}_i \mathbf{diag}(\dot{\alpha})$ is a constant matrix. Thus, Equation 4.3 can be reformulated as follows by plugging Equation 4.7 and ignoring the last term

$$\mathbf{v}_{\pi_i,t} - \mathbf{v}_{i,t} = 2\text{Re}[\dot{\mathbf{z}}_i^* \dot{\mathbf{S}}_{i,t}] \quad (4.8)$$

Equation 4.8 and Equation 4.5 then form the (linear) multi-phase LinDistFlow model. For mathematical conciseness, all variables associated with non-substation buses are organized as $\dot{\mathbf{s}}_t := [\dot{\mathbf{s}}_{1,t}^\top \dots \dot{\mathbf{s}}_{N,t}^\top]^\top$ and $\mathbf{v}_t := [\mathbf{v}_{1,t}^\top \dots \mathbf{v}_{N,t}^\top]^\top$, and all variables with lines are organized as $\dot{\mathbf{S}}_t := [\dot{\mathbf{S}}_{1,t}^\top \dots \dot{\mathbf{S}}_{N,t}^\top]^\top$. Therefore, Equation 4.8 and Equation 4.5 can be written in a compact form as

$$\mathbf{M}^{-\top} \mathbf{v}_t = 2\text{Re}[\mathbf{bdiag}(\dot{\mathbf{z}}_i^*) \dot{\mathbf{S}}_t] - v_0 \cdot (\mathbf{a}_0 \otimes \mathbf{1}_3) \quad (4.9)$$

$$\dot{\mathbf{S}}_t = \mathbf{M} \dot{\mathbf{s}}_t \quad (4.10)$$

where \otimes is the Kronecker product, $\mathbf{bdiag}(\mathbf{x}_i)$ is the block diagonal matrix with \mathbf{x}_i as the i -th block, $\mathbf{M} := \mathbf{A}^{-\top} \otimes \mathbf{I}_3$ and v_0 is the squared voltage magnitude of the substation bus. Plugging Equation 4.10 into Equation 4.9 and multiplying both sides of Equation 4.9 by \mathbf{M}^\top , the multi-phase LinDistFlow model can be equivalently converted to the following

equation

$$\begin{aligned}
\mathbf{v}_t &= 2\mathbf{M}^\top \mathbf{Re}[\mathbf{bdiag}(\dot{\mathbf{z}}_i^*) \mathbf{M} \dot{\mathbf{s}}_t] - (-v_0 \mathbf{1}_{3N}) \\
&= 2\mathbf{M}^\top \mathbf{Re}[\mathbf{bdiag}(\dot{\mathbf{z}}_i^*) \mathbf{M} (\mathbf{p}_t + j\mathbf{q}_t)] + v_0 \mathbf{1}_{3N} \\
&= 2\mathbf{M}^\top \mathbf{bdiag}(\mathbf{Re}[\dot{\mathbf{z}}_i^*]) \mathbf{M} \mathbf{p}_t - 2\mathbf{M}^\top \mathbf{bdiag}(\mathbf{Im}[\dot{\mathbf{z}}_i^*]) \mathbf{M} \mathbf{q}_t + v_0 \mathbf{1}_{3N} \\
&= \mathbf{R} \mathbf{p}_t + \mathbf{X} \mathbf{q}_t + v_0 \mathbf{1}_{3N}
\end{aligned} \tag{4.11}$$

where

- $\dot{\mathbf{s}}_t = \mathbf{p}_t + j\mathbf{q}_t$, $\mathbf{p}_t := [\mathbf{p}_{1,t}^\top \dots \mathbf{p}_{N,t}^\top]^\top$, $\mathbf{p}_{i,t} := [p_{i,t}^a \ p_{i,t}^b \ p_{i,t}^c]^\top$ is the active power injection of bus i ,
- $\mathbf{q}_t := [\mathbf{q}_{1,t}^\top \dots \mathbf{q}_{N,t}^\top]^\top$, $\mathbf{q}_{i,t} := [q_{i,t}^a \ q_{i,t}^b \ q_{i,t}^c]^\top$ is the reactive power injection at bus i , and
- Real matrices defined as

$$\mathbf{R} := 2\mathbf{M}^\top \mathbf{bdiag}(\mathbf{Re}[\dot{\mathbf{z}}_i]) \mathbf{M}$$

$$\mathbf{X} := 2\mathbf{M}^\top \mathbf{bdiag}(\mathbf{Im}[\dot{\mathbf{z}}_i]) \mathbf{M}.$$

4.2 ESS Scheduling for Voltage Regulation

4.2.1 Total Daily System Voltage Regulation Cost

The system voltage regulation cost at time t is defined as a quadratic function in terms of the deviations of voltage profiles \mathbf{v}_t from their reference values

$$f_t^{\Delta V} = c_t \cdot \|\mathbf{v}_t - v_0 \mathbf{1}_{3N}\|_2^2 \tag{4.12}$$

where $c_t \geq 0$ is the regulation price², and can be estimated by the utility according to the historical voltage profiles. This cost increases with increasing system voltage deviations

²Similar to [84], the regulation price is a parameter to manifest economic value, which can be determined from historical voltage deviation and PMU data. Estimating the regulation price is beyond the scope of this dissertation proposal, and thus it is randomly set from historical data in numerical simulations.

from the reference value.

Plugging Equation 4.11 into Equation 4.12, $f_t^{\Delta V}$ can then be represented in the form of the active and reactive power injections as

$$f_t^{\Delta V} = c_t \cdot (\mathbf{R}\mathbf{p}_t + \mathbf{X}\mathbf{q}_t)^\top (\mathbf{R}\mathbf{p}_t + \mathbf{X}\mathbf{q}_t) \quad (4.13)$$

According to Equation 4.13, both active and reactive power injections can be utilized to reduce voltage deviations. As discussed above, the R/X ratio is high in LV distribution networks and thus $\|\mathbf{X}\mathbf{q}_t\|_2^2$ is relatively small in comparison with $\|\mathbf{R}\mathbf{p}_t\|_2^2$. Therefore, $f_t^{\Delta V}$ can be approximated by neglecting terms containing the reactive power \mathbf{q}_t

$$f_t^{\Delta V} = c_t \cdot \mathbf{p}_t^\top \mathbf{R}^\top \mathbf{R} \mathbf{p}_t \quad (4.14)$$

which indicates that in LV distribution networks active power injection is more effective to mitigate voltage deviations than the reactive power.

Moreover, the 3×3 complex matrix $\dot{\mathbf{z}}_i$ can be expanded into

$$\dot{\mathbf{z}}_i = \begin{bmatrix} \dot{z}_i^{aa} & \alpha \dot{z}_i^{ab} & \alpha^* \dot{z}_i^{ac} \\ \alpha^* \dot{z}_i^{ba} & \dot{z}_i^{bb} & \alpha \dot{z}_i^{bc} \\ \alpha \dot{z}_i^{ca} & \alpha^* \dot{z}_i^{cb} & \dot{z}_i^{cc} \end{bmatrix}, \quad (4.15)$$

where

$$\begin{aligned} \dot{\mathbf{z}}_i &= \begin{bmatrix} \dot{z}_i^{aa} & \dot{z}_i^{ab} & \dot{z}_i^{ac} \\ \dot{z}_i^{ba} & \dot{z}_i^{bb} & \dot{z}_i^{bc} \\ \dot{z}_i^{ca} & \dot{z}_i^{cb} & \dot{z}_i^{cc} \end{bmatrix} \\ &= \begin{bmatrix} r_i^{aa} + jx_i^{aa} & r_i^{ab} + jx_i^{ab} & r_i^{ac} + jx_i^{ac} \\ r_i^{ba} + jx_i^{ba} & r_i^{bb} + jx_i^{bb} & r_i^{bc} + jx_i^{bc} \\ r_i^{ca} + jx_i^{ca} & r_i^{cb} + jx_i^{cb} & r_i^{cc} + jx_i^{cc} \end{bmatrix}. \end{aligned}$$

Then 3×3 real matrix $\mathbf{Re}[\dot{\mathbf{z}}_i]$ can be decomposed as

$$\begin{aligned} \mathbf{Re}[\dot{\mathbf{z}}_i] &= \tilde{\mathbf{r}}_i + \tilde{\mathbf{x}}_i \\ &= \begin{bmatrix} r_i^{aa} & -\frac{1}{2}r_i^{ab} & -\frac{1}{2}r_i^{ac} \\ -\frac{1}{2}r_i^{ba} & r_i^{bb} & -\frac{1}{2}r_i^{bc} \\ -\frac{1}{2}r_i^{ca} & -\frac{1}{2}r_i^{cb} & r_i^{cc} \end{bmatrix} + \begin{bmatrix} 0 & \frac{\sqrt{3}}{2}x_i^{ab} & -\frac{\sqrt{3}}{2}x_i^{ac} \\ -\frac{\sqrt{3}}{2}x_i^{ba} & 0 & \frac{\sqrt{3}}{2}x_i^{bc} \\ \frac{\sqrt{3}}{2}x_i^{ca} & -\frac{\sqrt{3}}{2}x_i^{cb} & 0 \end{bmatrix}. \end{aligned} \quad (4.16)$$

Since $\|\tilde{\mathbf{x}}_i\|_2^2$ is relatively small in comparison with $\|\tilde{\mathbf{r}}_i\|_2^2$, $\mathbf{Re}[\dot{\mathbf{z}}_i] \approx \tilde{\mathbf{r}}_i$ and the $f_t^{\Delta V}$ can be further approximated as

$$f_t^{\Delta V} = c_t \cdot \mathbf{p}_t^\top \tilde{\mathbf{R}}^\top \tilde{\mathbf{R}} \mathbf{p}_t \quad (4.17)$$

where $\tilde{\mathbf{R}} := 2\mathbf{M}^\top \mathbf{b} \text{diag}(\tilde{\mathbf{r}}_i) \mathbf{M}$. This approximation fully decouples the impact of the line reactance on the voltage magnitudes due to high R/X ratios.

Assuming installation of PV generations, ESSs, and (controllable) loads throughout the LV network, the active power injection profile is defined as

$$\mathbf{p}_t := \mathbf{p}_t^{\text{PV}} + \mathbf{p}_t^{\text{E}} - \mathbf{p}_t^{\text{D}} \quad (4.18)$$

where

- $\mathbf{p}_t^{\text{PV}} := [\mathbf{p}_{1,t}^{\text{PV}\top} \cdots \mathbf{p}_{N,t}^{\text{PV}\top}]^\top$ with $\mathbf{p}_{i,t}^{\text{PV}} := [p_{i,t}^{\text{PV},a} \ p_{i,t}^{\text{PV},b} \ p_{i,t}^{\text{PV},c}]^\top$ denotes PV power generation at bus i at t ,
- $\mathbf{p}_t^{\text{E}} := [\mathbf{p}_{1,t}^{\text{E}\top} \cdots \mathbf{p}_{N,t}^{\text{E}\top}]^\top$ with $\mathbf{p}_{i,t}^{\text{E}} := [p_{i,t}^{\text{E},a} \ p_{i,t}^{\text{E},b} \ p_{i,t}^{\text{E},c}]^\top$ denotes ESS power injection at bus i at t .

If $p_{i,t}^{\text{E},\varphi} \geq 0$, the ESS is discharging; otherwise the ESS is charging. $\mathbf{p}_t^{\text{D}} := [\mathbf{p}_{1,t}^{\text{D}\top} \cdots \mathbf{p}_{N,t}^{\text{D}\top}]^\top$, $\mathbf{p}_{i,t}^{\text{D}} := [p_{i,t}^{\text{D},a} \ p_{i,t}^{\text{D},b} \ p_{i,t}^{\text{D},c}]^\top$ is the power demand of the load at bus i at time t . Insert it into the Equation 4.17 and the $f_t^{\Delta V}$ can be expanded into

$$f_t^{\Delta V} = \underbrace{c_t \cdot \mathbf{p}_t^E \tilde{\mathbf{R}}^\top \tilde{\mathbf{R}} \mathbf{p}_t^E + 2c_t \cdot (\mathbf{p}_t^{\text{PV}} - \mathbf{p}_t^{\text{D}})^\top \tilde{\mathbf{R}}^\top \tilde{\mathbf{R}} \mathbf{p}_t^E}_{\text{negative voltage regulation compensation}} + \underbrace{c_t \cdot (\mathbf{p}_t^{\text{PV}} - \mathbf{p}_t^{\text{D}})^\top \tilde{\mathbf{R}}^\top \tilde{\mathbf{R}} (\mathbf{p}_t^{\text{PV}} - \mathbf{p}_t^{\text{D}})}_{\text{voltage regulation cost before control}} \quad (4.19)$$

In this dissertation, the ESS is the only controllable source for voltage regulation. Therefore, $f_t^{\Delta V}$ can be considered as a quadratic function in terms of ESS power injection. It can be interpreted as the voltage regulation cost after control, i.e., the summation of voltage regulation cost before control and negative voltage regulation compensation by ESSs.

The total daily system voltage regulation cost is defined as the summation of the voltage regulation cost over the total number T of control intervals within a day

$$f_{\text{daily}}^{\Delta V} = \sum_{t=0}^{T-1} f_t^{\Delta V}. \quad (4.20)$$

4.2.2 ESS-based Voltage Regulation Problem Formulation

Finally, the ESS-based voltage regulation problem is formulated as an expected total daily system voltage regulation cost minimization problem subject to ESS operation constraints.

$$\min_{\mathbf{p}^E} \quad \left\{ \mathbb{E}_{\text{soc}_0, \mathbf{p}^{\text{PV}}, \mathbf{p}^{\text{D}}} (f_{\text{daily}}^{\Delta V}) \right\} \quad (4.21)$$

$$\text{s.t.} \quad -\mathbf{1}_T \otimes \bar{\mathbf{p}}^E \leq \underbrace{\mathbf{bdiag}(\mathbf{I}_{3N})}_{3NT \times 3NT} \mathbf{p}^E \leq \mathbf{1}_T \otimes \bar{\mathbf{p}}^E \quad (4.22)$$

$$\underline{\text{soc}} \cdot \mathbf{1}_{3NT} \leq \text{soc} \leq \overline{\text{soc}} \cdot \mathbf{1}_{3NT} \quad (4.23)$$

$$\mathbf{soc} = \mathbf{1}_T \otimes \mathbf{soc}_0 - \mathbf{bdiag}(\mathbf{I}_{3N})(\mathbf{L} \otimes \mathbf{I}_{3N}) \odot (\mathbf{1}_{T \times T} \otimes (\mathbf{1}_{3N} \otimes \Delta \mathbf{e}^\top)) \mathbf{p}^E \quad (4.24)$$

$$p_{i,t}^{E,\varphi} = 0, \quad \forall (i, \varphi) \notin \mathcal{X} \quad (4.25)$$

where

- $\mathbf{p}^E := [\mathbf{p}_1^{E^\top} \dots \mathbf{p}_T^{E^\top}]^\top$ is a vector of power supply of all ESSs ,
- $\mathbf{p}^{PV} := [\mathbf{p}_1^{PV^\top} \dots \mathbf{p}_T^{PV^\top}]^\top$ is a vector of power supply of all PV inverters ,
- $\mathbf{p}^D := [\mathbf{p}_1^{D^\top} \dots \mathbf{p}_T^{D^\top}]^\top$ is a vector of power demand of all loads ,
- $\mathbf{soc} := [\mathbf{soc}_1^\top \dots \mathbf{soc}_T^\top]^\top$ is a vector of state of charge of all ESSs ,
- $\mathbf{soc}_t := [\mathbf{soc}_{1,t}^\top \dots \mathbf{soc}_{N,t}^\top]^\top$ is a vector of state of charge of all ESSs at time t ,
- $\mathbf{soc}_{i,t} := [soc_{i,t}^a \ soc_{i,t}^b \ soc_{i,t}^c]^\top$ is the ESS SOC at bus i ,
- $\bar{\mathbf{p}}^E := [\bar{\mathbf{p}}_1^{E^\top} \dots \bar{\mathbf{p}}_N^{E^\top}]^\top$ is a vector of the maximum power supply of all ESSs ,
- $\bar{\mathbf{p}}_i^E := [\bar{p}_i^{E,a} \ \bar{p}_i^{E,b} \ \bar{p}_i^{E,c}]^\top$ is the maximum ESS power injection at bus i ,
- \underline{soc} and \overline{soc} denote the minimum and maximum allowable SOC, respectively,
- \mathbf{L} is a $T \times T$ lower triangular matrix, i.e., $L_{ij} = 1$ if $i \geq j$ and $= 0$ otherwise,
- $\Delta \mathbf{e} := [\Delta \mathbf{e}_1^\top \dots \Delta \mathbf{e}_N^\top]^\top$, $\Delta \mathbf{e}_i := [\Delta e_i^a \ \Delta e_i^b \ \Delta e_i^c]^\top$. $\Delta e_i^\varphi = \frac{\Delta t}{e_i^\varphi}$ if $(i, \varphi) \in \mathcal{X}$; otherwise $\Delta e_i^\varphi = 0$. Δt is the length of the control interval, e_i^φ is the capacity of the ESS at bus i and \mathcal{X} is a 2-tuple (i, φ) set of positions where bus has ESS installation.

Note that the objective defined by Equation 4.21 minimizes the expected total daily system voltage regulation cost over random variables \mathbf{soc}_0 , \mathbf{p}^{PV} , and \mathbf{p}^D by scheduling the power

supply of ESS. Moreover, constraints defined by Equation 4.22 and Equation 4.23 are to impose limits on the power rate and SOC, respectively, while constraint Equation 4.24 represent dynamics of the ESS without considering energy losses. Finally, constraint Equation 4.25 is to ensure that $p_{i,t}^{\text{E},\varphi} = 0$ at all times if bus has no ESS installation. Therefore, this problem can be considered as a multi-stage quadratic stochastic program. However, optimally solving such a problem is challenging and even intractable. The major difficulties are uncertainties encountered in modern distribution networks. In order to overcome this challenge, a model-free method is proposed in the following section to approximately solve this problem without needs of knowing the uncertainty model.

4.3 Real-time Voltage Regulation as MDP

In this section, the above-formulated ESS-based voltage regulation problem is first converted to an MDP problem where the central controller for scheduling is designed as an agent. To make the agent explore action space more widely and more wisely, the preceding MDP problem is then reformulated as an entropy-constrained MDP problem. An advanced model-free off-policy DRL algorithm called SAC [85] is introduced later to approximately solve such a constrained optimization problem. Finally, a learn-to-schedule feedback control framework is proposed and integrated with a practical deep reinforcement schedule scheme as an optimizer. The learned agent can use real-time operation data to near-optimally schedule ESSs so that the total daily system voltage deviation can be mitigated.

Consider an MDP where state and action are continuous. It is assumed that the state is time dependent and there is no overlap in the state space at different time steps. As shown in Figure 4.1, the agent starts at initial state s_0 . In each control interval $t + 1$, the agent samples an action \mathbf{a}_t from the policy π_t , receives a bounded reward r_t , and transitions to a new state s_{t+1} according to environmental dynamics $p(s_{t+1}|s_t, \mathbf{a}_t)$. After T control intervals, the agent stops in an artificial terminal state s_T . The above-mentioned process

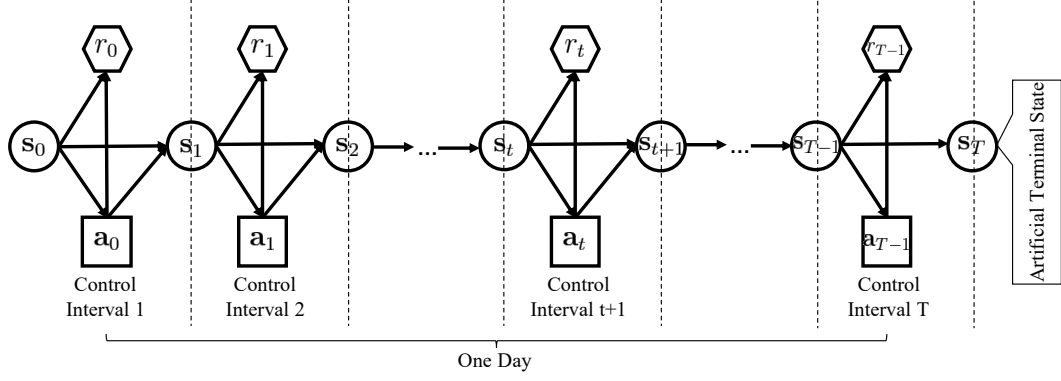


Figure 4.1: Illustration of an MDP for ESSs based voltage regulation

defines a sequential decision-making problem specified as follows

- **AGENT:** An agent is defined as a central controller for scheduling the power supply of the ESS;
- **ACTION:** An action $\mathbf{a}_t \in \mathcal{A} := [-1, 1]^{|X|}$ is defined as a vector of the normalized power supply commands of the ESSs. For ESS installed at bus i and phase φ , given the normalized power supply command $\tilde{p}_{i,t}^{\text{E},\varphi} \in [-1, 1]$, the actual power supply command $p_{i,t}^{\text{E},\varphi} \in [\underline{p}_{i,t}^{\text{E},\varphi}, \bar{p}_{i,t}^{\text{E},\varphi}]$ can be reconstructed by denormalization

$$p_{i,t}^{\text{E},\varphi} = \underline{p}_{i,t}^{\text{E},\varphi} + \frac{1}{2}(\bar{p}_{i,t}^{\text{E},\varphi} - \underline{p}_{i,t}^{\text{E},\varphi})(1 + \tilde{p}_{i,t}^{\text{E},\varphi}) \quad (4.26)$$

where

- $\underline{p}_{i,t}^{\text{E},\varphi} := -\min\{\bar{p}_i^{\text{E},\varphi}, \frac{\overline{\text{soc}} - \text{soc}_{i,t}^{\varphi}}{\Delta e_i^{\varphi}}\}$ for charging, and
- $\underline{p}_{i,t}^{\text{E},\varphi} := \min\{\bar{p}_i^{\text{E},\varphi}, \frac{\text{soc}_{i,t}^{\varphi} - \text{soc}}{\Delta e_i^{\varphi}}\}$ for discharging.

The above normalization forcefully restricts the action to the feasible region of the proposed optimization problem;

- **STATE:** A state $\mathbf{s}_t \in \mathcal{S}_t$ is defined as a vector of the active power injections before control, the states of charge and the scaled time stamp $\frac{t}{T-1}$. We claim that this design is sufficient. On one hand, the solution of the proposed problem is dependent on

the first two factors. On the other hand, the time stamp can make the state time dependent. It should be noted that the artificial terminal state can be anything irrespective of the previous action;

- **REWARD:** The reward $r(\mathbf{s}_t, \mathbf{a}_t)$ (abbreviated as r_t sometimes) is highly related to the objective defined by Equation 4.21 and designed as the voltage regulation compensation

$$r(\mathbf{s}_t, \mathbf{a}_t) = -c_t \cdot [\mathbf{p}_t^E - 2(\mathbf{p}_t^{PV} - \mathbf{p}_t^D)]^\top \tilde{\mathbf{R}}^\top \tilde{\mathbf{R}} \mathbf{p}_t^E; \quad (4.27)$$

- **POLICY:** Each agent selects an action according to a policy that is a family of conditional probability distribution functions $\pi(\mathbf{a}_t|\mathbf{s}_t)$ (abbreviated as π_t sometimes) reflecting the probability that action \mathbf{a}_t being taken in \mathbf{s}_t ;
- **ENVIRONMENT:** The environment responses to actions based on its internal dynamics $p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$ that can be considered as a state transition probability distribution;
- **MDP PROBLEM:** The goal of the agent is designed as finding an optimal policy $\pi_{0:T-1}^*$ so that the expected return from $t = 0$ can be maximized

$$\max_{\pi_{0:T-1}} \mathbb{E}_{\substack{\mathbf{s}_0 \sim p(\mathbf{s}_0), \mathbf{a}_t \sim \pi_t \\ \mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)}} [G_0] \quad (4.28)$$

where

- G_0 is a special case of the return $G_k := \sum_{t=k}^{T-1} r(\mathbf{s}_t, \mathbf{a}_t)$ at $k = 0$;
- $p(\mathbf{s}_0)$ is a probability distribution of the initial state.

The MDP problem defined by Equation 4.28 can be considered as an approximating version of the proposed problem, i.e., the solution space is assumed as a specific form

through feedback policies. This assumption relaxes the dependence of actions on global states over all control intervals and sequentially solves a multi-stage quadratic stochastic program. Once the optimal policy is determined, voltage regulation can be implemented in real-time. However, explicitly solving Equation 4.28 and finding the optimal policy is impractical. Firstly, with many DERs installed, dynamics of the states are difficult to capture. Secondly, high-dimensional state and action spaces lead to a known issue called the curse of dimensionality. Therefore, a model-free DRL is resorted to overcome these two limits and approximately solve Equation 4.28.

4.4 Learn-to-schedule Feedback Control Framework Via SAC

A state-of-art DRL algorithm called SAC is selected to approximately solve Equation 4.28 with the following advantages. Firstly, it can handle continuous action spaces and unknown environmental dynamics. Secondly, by integrating the policy entropy into the MDP, the policy is incentivized to explore more widely while avoiding unpromising avenues, which is beneficial to finding near-optimal policies. Finally, as an off-policy algorithm, SAC is sample-efficient by reusing samples during training and takes less effort to tune hyper-parameters.

In SAC, the problem defined by Equation 4.28 can be reformulated as the entropy constrained MDP problem

$$\begin{aligned} \max_{\pi_{0:T-1}} \quad & \mathbb{E}_{\substack{\mathbf{s}_0 \sim p(\mathbf{s}_0), \mathbf{a}_t \sim \pi_t \\ \mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)}} [G_0] \\ \text{s.t.} \quad & H(\pi_t) \geq H_0 \quad (\forall t) \end{aligned} \tag{4.29}$$

where $H(\pi_t) = - \mathbb{E}_{\mathbf{a}_t \sim \pi_t} [\log \pi(\mathbf{a}_t|\mathbf{s}_t)]$ is the entropy of the policy π_t and H_0 is a predefined minimum policy entropy threshold. Then the goal of the agent is modified as finding an optimal policy $\pi_{0:T-1}^*$ so that the expected return from $t = 0$ can be maximized while satisfying a minimum expected entropy constraint. Although this modification constrains

the original solution space, the current agent is endowed with more rational exploration. To solve a constrained maximization problem, the Lagrangian function is constructed as

$$\mathcal{L}(\pi_{0:T-1}, \gamma_{0:T-1}) = \underbrace{\mathbb{E}_{\substack{\mathbf{s}_0 \sim p(\mathbf{s}_0), \mathbf{a}_t \sim \pi_t \\ \mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)}}}_{\text{maximum entropy objective}} \left[G_0 - \sum_{t=0}^{T-1} \gamma_t \log \pi_t \right] - \gamma_t H_0 \quad (4.30)$$

where $\gamma_t \geq 0$ ($\forall t$) is the dual variable of the corresponding inequality or called “temperature”. For a fixed dual variable, the non-constant term is called “maximum entropy objective”. The dual variable performs a role to control the stochasticity of the optimal policy. If the constraint at time t is satisfied, $\gamma_t = 0$ and the agent is inclined to exploit the current action. If the constraint at time t is violated, $\gamma_t > 0$ and the agent is inclined to explore other actions.

Define an optimal soft q -function $q^*(\mathbf{s}_t, \mathbf{a}_t)$ (abbreviated as q_t^* sometimes)

$$q^*(\mathbf{s}_t, \mathbf{a}_t) = \begin{cases} r_t + \mathbb{E}[q^*(\mathbf{s}_{t+1}, \mathbf{a}_{t+1}) - \gamma_t \log \pi_{t+1}^*], & t \neq T-1 \\ r(\mathbf{s}_{T-1}, \mathbf{a}_{T-1}), & t = T-1 \end{cases} \quad (4.31)$$

where $\mathbf{a}_{t+1} \sim \pi_{t+1}^*$ and $\mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$. It should be noted that the optimal policy at time t is independent of the policy at the previous time steps. Assuming the q_t^* has been evaluated for some t and substituted in the Lagrangian, the optimal policy π_t^* at t for $\forall \mathbf{s}_t \in \mathcal{S}_t$ can be achieved by

$$\max_{\mathbf{a}_t \sim \pi_t} \mathbb{E} [q_t^* - \gamma_t \log \pi_t] \quad (4.32)$$

over π_t , which is an exact step to solve a primal problem for a fixed dual variable. By traversing backwards in time, the Lagrangian with respect to the policy can be optimized.

After solving for the primal problem using Equation 4.32, the dual variable γ_t can be

updated by minimizing

$$\mathbb{E}_{\substack{\mathbf{s}_0 \sim p(\mathbf{s}_0), \mathbf{a}_t \sim \pi_t^* \\ \mathbf{s}_{t+1} \sim p(\mathbf{s}_{t+1} | \mathbf{s}_t, \mathbf{a}_t)}} [-\gamma_t \log \pi_t^* - \gamma_t H_0] \quad (4.33)$$

This is exactly a step to solve a dual problem given an optimal policy. Although alternatively optimizing π_t and α_t provide a possible solution, finding exact solutions to Equation 4.31, Equation 4.32, and Equation 4.33 is in general not practical for continuous states and actions and for unknown environmental dynamics. In other words, the expectation cannot be computed. Therefore, the SAC uses function approximators and Stochastic Gradient Decent (SGD) as well as other standard techniques to approximately solve Equation 4.29, as discussed in the following.

The policy and optimal soft q-function are approximated by a neural network (called actor) $\pi_\phi(\mathbf{a}_t | \mathbf{s}_t)$ with weight ϕ and a neural network (called critic) $q_\theta(\mathbf{s}_t, \mathbf{a}_t)$ with weight θ , respectively. To reduce overestimation of $q_\theta(\mathbf{s}_t, \mathbf{a}_t)$, two parameterized q-functions $q_{\theta_1}(\mathbf{s}_t, \mathbf{a}_t)$ and $q_{\theta_2}(\mathbf{s}_t, \mathbf{a}_t)$ are used instead of $q_\theta(\mathbf{s}_t, \mathbf{a}_t)$ where θ_1 and θ_2 are weights. To improve the stability of learning, two target critic networks $q_{\theta'_1}(\mathbf{s}_t, \mathbf{a}_t)$ and $q_{\theta'_2}(\mathbf{s}_t, \mathbf{a}_t)$ are introduced, which is a copy of the original critic networks $q_{\theta_1}(\mathbf{s}_t, \mathbf{a}_t)$ and $q_{\theta_2}(\mathbf{s}_t, \mathbf{a}_t)$ with an earlier snapshot of weights where θ'_1 and θ'_2 are weights.

Assume that the samples $(\mathbf{s}_t, \mathbf{a}_t, r_t, \mathbf{s}_{t+1}, d)$ including state, action, reward, next state and done have been collected and stored in the replay buffer \mathcal{D} where $d = 1$ if \mathbf{s}_{t+1} is a terminal state; otherwise $d = 0$. The critic network weight θ_i can be learned by minimizing the soft Bellman residual

$$J_q(\theta_i) := \mathbb{E}_{\substack{\mathbf{s}_t, \mathbf{s}_{t+1} \sim \mathcal{D} \\ \mathbf{a}_t \sim \pi_\phi, \mathbf{a}_{t+1} \sim \pi_\phi \\ r_t, d \sim \mathcal{D}}} \left[\frac{1}{2} (q_{\theta_i}(\mathbf{s}_t, \mathbf{a}_t) - y(r_t, \mathbf{s}_{t+1}, \mathbf{a}_{t+1}, d))^2 \right] \quad (4.34)$$

where $\gamma \geq 0$ is a dynamic temperature. A global temperature is used instead of preceding

γ_t because each γ_t has the same objective defined by Equation 4.33. $y(r_t, \mathbf{s}_{t+1}, \mathbf{a}_{t+1}, d) := r_t + (1 - d)(\min_{i \in \{1,2\}} q_{\theta'_i} - \gamma \log \pi_\phi(\mathbf{a}_{t+1}|\mathbf{s}_{t+1}))$ is the target value of $q_{\theta_i}(\mathbf{s}_t, \mathbf{a}_t)$.

According to Equation 4.32 and Equation 4.33, the actor network weight ϕ and the dynamic temperature γ can be learned by minimizing

$$J_\pi(\phi) := \mathbb{E}_{\substack{\mathbf{s}_t \sim \mathcal{D} \\ \mathbf{a}_t \sim \pi_\phi}} [\gamma \log \pi_\phi(\mathbf{a}_t|\mathbf{s}_t) - \min_{i \in \{1,2\}} q_{\theta_i}(\mathbf{s}_t, \mathbf{a}_t)] \quad (4.35)$$

and

$$J(\gamma) := \mathbb{E}_{\substack{\mathbf{s}_t \sim \mathcal{D} \\ \mathbf{a}_t \sim \pi_\phi}} [-\gamma \log \pi_\phi(\mathbf{a}_t|\mathbf{s}_t) - \gamma H_0], \quad (4.36)$$

respectively. The target critic network weight θ'_i can be updated by having them slowly track the learned critic networks

$$\theta'_i \leftarrow \tau \theta_i + (1 - \tau) \theta'_i \quad (4.37)$$

where $\tau \ll 1$ is a parameter for updating the target networks. Assume that the continuous action \mathbf{a}_t is Gaussian distributed with mean $\mu_\phi(\mathbf{s}_t)$ and covariance $\sigma_\phi^2(\mathbf{s}_t)$, where $\mu_\phi(\mathbf{s}_t)$ and $\sigma_\phi^2(\mathbf{s}_t)$ are parameterized by the preceding actor network. The policy function can then be represented as a multivariate Gaussian distribution function

$$\pi_\phi(\mathbf{a}_t|\mathbf{s}_t) = \mathbb{N}(\mu_\phi(\mathbf{s}_t), \sigma_\phi^2(\mathbf{s}_t)) \quad (4.38)$$

To make the action \mathbf{a}_t differentiable, re-parameterization is adopted to sample the actions and an invertible squashing function (\tanh , (i.e., hyperbolic tangent)) is applied after sampling

$$\mathbf{a}_t = \tanh(\mu_\phi(\mathbf{s}_t) + \epsilon_t \sigma_\phi(\mathbf{s}_t)) \quad (4.39)$$

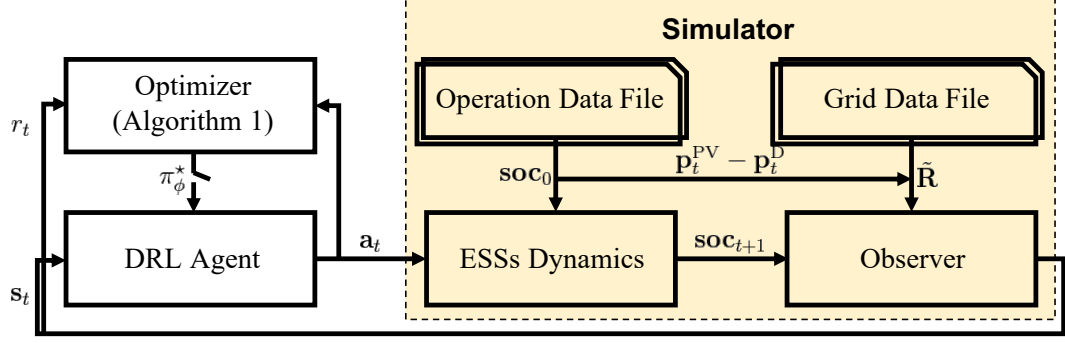


Figure 4.2: Overview of the proposed learn-to-schedule feedback controller

where $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a unit Gaussian noise. Equation 4.34 - Equation 4.39 proposes a deep reinforcement ESS scheduling scheme with a learn-to-schedule feedback control framework. In Algorithm 1, n_{warmup} is a threshold size to start to update network weights and n_{batch} is the batch size. The superscript j denotes the j -th sample in a minibatch. The symbol $\hat{\nabla}$ denotes stochastic gradients. Moreover, λ_{θ_i} , λ_ϕ , and λ_γ are learning rates. The episode means a sample for training, and the action is sampled during training while the mean $\mu_\phi(s_t)$ is output as the action instead of sampling from the policy during execution. From lines 23-25, SGD is used to update weights.

The proposed framework in Figure 5.2 is constituted by three main components: the DRL agent, the optimizer, and the simulator. The agent uses the optimized policy to generate actions. During offline training, the optimizer is in charge of optimizing the policy based on Algorithm 1. After running out the episodes in the training dataset while finding a near-optimal policy, the optimizer will stop. The simulator is an artificial environment to mimic probability distribution $p(s_{t+1}, r_t | s_t, a_t)$. Its main role is to respond to actions taken by the agent. Specifically, at each time t the SOC, as the sole part of the state, dependent on the action transits to the next states of charge using the following dynamics

$$\text{soc}_{t+1} = \text{soc}_t + \Delta e \odot \mathbf{p}_t^E. \quad (4.40)$$

Other parts) of the state, such as active power injections before control, can transit

Algorithm 1 Deep Reinforcement Schedule Scheme

```
1: // offline training
2: Initialize  $\phi, \theta_1, \theta'_1, \theta_2$  and  $\theta'_2$  and  $\alpha$ 
3: Initialize  $\mathcal{D} \leftarrow \emptyset$ 
4: for each episode do
5:   reset the initial state  $s_0$  ;
6:   for  $t=0:T-1$  do
7:     sample action  $\mathbf{a}_t$  from policy  $\pi_\phi(\mathbf{a}_t|s_t)$  ;
8:     denormalize action  $\mathbf{a}_t$  using Equation 4.26 ;
9:     calculate reward  $r_t$  using Equation 4.27 ;
10:    if  $t == T-1$  then
11:      set next state  $s_{t+1}$  arbitrarily ;
12:       $d \leftarrow 1$  ;
13:    else
14:      collect next state  $s_{t+1}$  from the simulator ;
15:       $d \leftarrow 0$  ;
16:    end if
17:     $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, \mathbf{a}_t, r_t, s_{t+1}, d)\}$  ;
18:     $s_t \leftarrow s_{t+1}$  ;
19:  end for
20:  if  $|\mathcal{D}| > n_{\text{warmup}}$  then
21:    sample a minibatch  $\{(s_t^j, \mathbf{a}_t^j, r_t^j, s_{t+1}^j, d^j)\}_{j=1:n_{\text{batch}}}$  from  $\mathcal{D}$  ;
22:    normalize rewards  $\{r_t^j\}_{j=1:n_{\text{batch}}}$  ;
23:    calculate  $\hat{\nabla}_{\theta_i} J_q(\theta_i)$ ,  $\hat{\nabla}_\phi J_\pi(\phi)$  and  $\hat{\nabla}_\gamma J(\gamma)$  ;
24:    update  $\theta_i \leftarrow \theta_i - \lambda_{\theta_i} \hat{\nabla}_{\theta_i} J_q(\theta_i)$  for  $i \in \{1, 2\}$  ;
25:    update  $\phi \leftarrow \phi - \lambda_\phi \hat{\nabla}_\phi J_\pi(\phi)$  ;
26:    update  $\gamma \leftarrow \gamma - \lambda_\gamma \hat{\nabla}_\gamma J(\gamma)$  ;
27:    update  $\theta'_i$  for  $i \in \{1, 2\}$  using Equation 4.37 ;
28:  end if
29: end for
30: output near-optimal policy  $\pi_\phi^*$  ;
31: // online execution
32: for each day do
33:   for  $t = 0:T-1$  do
34:     collect state  $s_t$  from the simulator ;
35:     output action  $\mathbf{a}_t$  using  $\pi_\phi^*$  ;
36:     denormalize action  $\mathbf{a}_t$  using Equation 4.26 ;
37:     execute  $\mathbf{a}_t$  in the simulator ;
38:   end for
39: end for
```

by collecting it directly from operation data. The observer further converts the received information into a desired form of the next state while calculating the reward. During online execution, the simulator is still used except that real-time operation data is used instead of historical data. It can be noted that this framework is non-intrusive, i.e., without requiring direct interaction with a real distribution network for policy search. Before a real-time action is sent to local ESSs, it can be first confirmed by the system operator to avoid risks. In addition, every time period the policy needs to be retrained.

4.5 Simulation Results

The proposed algorithm is tested on the customized 6-bus and modified IEEE 34-bus systems³ where the former and latter have five and fifteen single-phase customers (i.e., DERs), respectively, as shown in Figure 4.3. All DERs are installed at phase a . The base voltage magnitude and power rating are set as 0.416 kV and 0.01 MVA, respectively. The feeder-to-bus and bus-to-bus distances are set as 0.5 km and 0.1 km, respectively. The self and mutual impedances on each line are set as $0.840+0.080j$ and $0.371+0.005j$ ohm/km for the 6-bus system as well as $0.249+0.072j$ and $0.071+0.005j$ ohm/km for the 34-bus system, respectively. The ESSs are the sole voltage controllers in both systems with the capacity and maximum charging/discharging rate set as 5 p.u. \times h and 0.5 p.u., respectively. The minimum and maximum allowable SOC are set as 0.1 and 0.9, respectively. The regulation price is set as 1.5 \$/p.u.².

The daily hour operational data, including PV generation and load profiles, is scaled from one-month's real-world PJM Interconnections data, which is further augmented by perturbation. PV generations and loads are perturbed with a random range from 80% \sim 120% of their original base values. The initial SOC is randomly sampled from a uniform distribution $s_0^j \sim \text{unif}(0.1, 0.9)$. The power factor is set as 0.92.

The architecture of actor and critic networks is presented in Figure 4.4. For the actor

³Line parameters are selected from IEEE Europe LV Test Feeder [86].

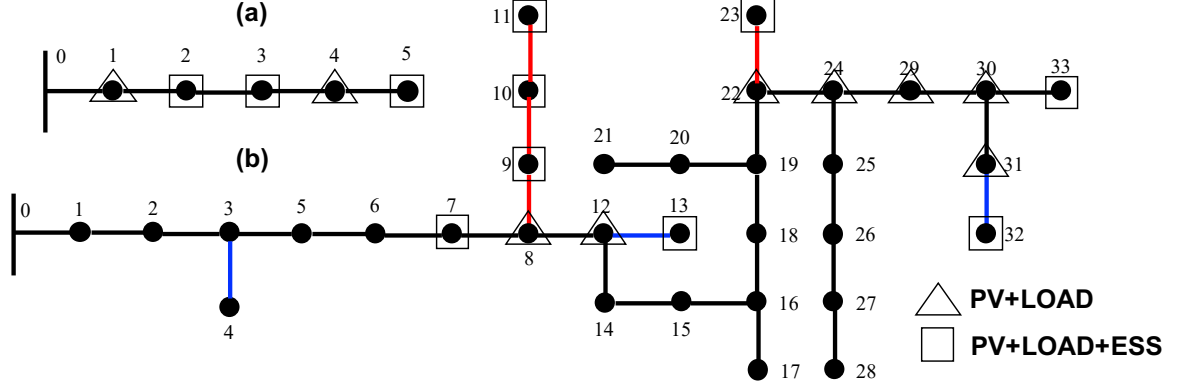


Figure 4.3: Schematic diagram of (a) the customized 6-bus and (b) modified IEEE 34-bus systems where the black, red and blue line correspond the line with three phases, only a phase and only b phase, respectively.

network, the number of neurons in the input, hidden, and output layers are designed as 5/3/1 (active power injection/SOC/time stamp), 2/2, 3/3 (mean/logarithmic standard deviation) for the 6-bus system as well as 15/8/1, 8/4, 8/8 for the 34-bus system, respectively. Similarly, for the critic network, the number of neurons of in the input, hidden, and output layers are designed as 5/3/3/1 (active power injection/SOC/time stamp/action), 2/2/2, and 1 for the 6-bus system as well as 15/8/8/1, 8/4/4, and 1 for the 34-bus system, respectively. Moreover, the fully connected (FC) layer is used to build these two networks. Except the output layer, after each layer the ReLU activation function is applied. The Adam optimizer is adopted for training, with a learning rate of 0.001, 0.001, and 0.001 for actor λ_i , critic λ_ϕ , and temperature λ_γ , respectively. The target critic network is updated with τ set as 0.01. The entropy threshold H_0 , size of replay buffer $|\mathcal{D}|$, batch size n_{batch} , and number of control intervals T are set as -0.05, 1e6, 128 and 24, respectively. Finally, the threshold size n_{warmup} is set as 1280.

In this dissertation proposal, the baseline, i.e., the performance before control actions are applied, is used to evaluate the effectiveness of the proposed algorithms. Moreover, a deterministic QP is constructed with the objective function $f_{\text{daily}}^{\Delta V}$ and the same constraints as the proposed problem Equation 4.22 - Equation 4.25. Since its solution is a strictly lower bound of the original problem, it is used to evaluate the near-optimality. Moreover, since

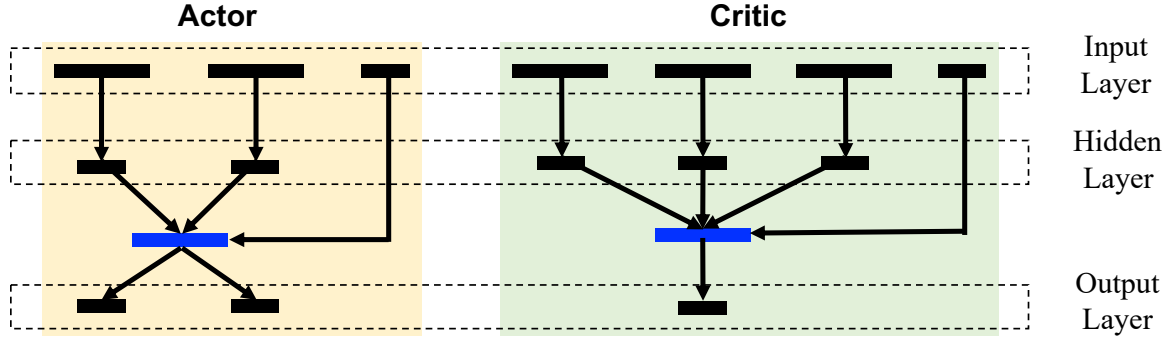


Figure 4.4: Illustration of neural network architecture of actor and critic network where blue layer is a concatenation layer

the matrix $\tilde{\mathbf{R}}^\top \tilde{\mathbf{R}}$ is positive-definite, this quadratic program is solvable, and there always exists a unique global optimizer with the optimum $f_{\text{daily}}^{\Delta V^*}$.

4.5.1 Customized 6-Bus Test System

As shown in Figure 4.5(a), the average return in the first 250 episodes is below the baseline, i.e., the voltage regulation performance after control is worse than the one before control. As the training process continues, the average return has an upward tendency and finally converges to some value. As shown in Figure 4.5(b), it can be seen that the regulation cost optimized by DRL is close to the lower bound achieved by QP solver, and significantly improved in comparison with the cost without control where the average regulation cost is \$1.367 for baseline, \$0.467 for DRL and \$0.334 for QP. In addition, the CPU running time for action generation is 0.0003s in average, which is far less than the length of control interval.

A sample daily profile (including PV generation and load) is provided in Figure 4.6(a). Three ESSs are installed on buses 2, 3, and 5 and labeled as ESS 1, 2, and 3, respectively. Take 9:00-13:00 as an example, it can be observed in Figure 4.6(a) that in this time period over-voltage issues are caused by high solar generation. Consequently, the ESSs are scheduled to charge from the grid and mitigate the voltage deviation (as shown in Figure 4.6(b)) until their maximum SOC limits are reached (as shown in Figure 4.6(c)).

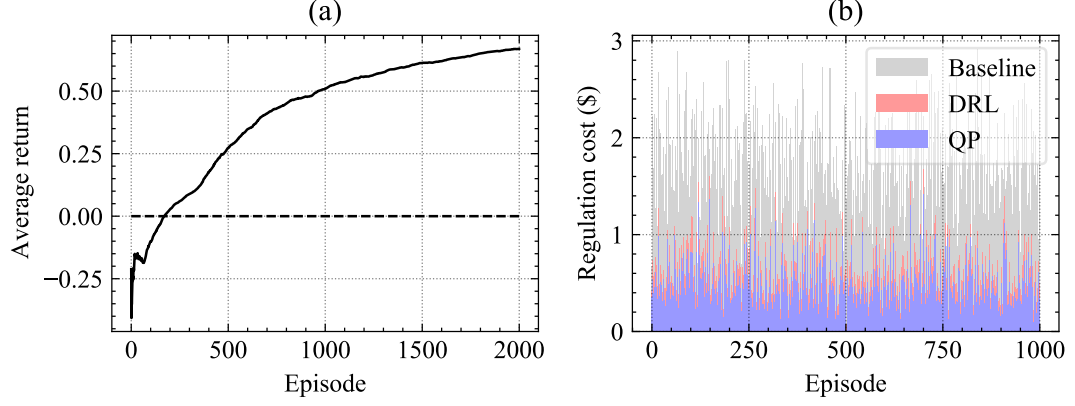


Figure 4.5: Illustration of learning performance (a) during training and voltage regulation performance (b) during execution where the dash line is a baseline

Finally, Figure 4.6(d) shows the comparison of voltage profile at bus 5 with and without ESSs installed. It can be observed that with ESSs installed and scheduled by the proposed method, all three-phase voltage profiles are controlled within the upper and lower bounds, which are far more volatile (i.e., not within the bounds) without ESSs. Finally, Figure 4.7 presents voltage dynamics of phase a at bus 5, which shows that the regulated voltage is more flattened by DRL-scheduled ESSs. Similarly, under-voltage issues between 17:00-23:00 caused by high load demand can be observed and alleviated by scheduling of ESSs as well, as shown in Figure 4.6.

4.5.2 Modified IEEE 34-Bus Test System

As shown in Figure 4.8(a), since the DRL needs to explore possible optimal states at early training stages, the average return in the first 600 episodes is below the baseline. However, with the training process continues, the average return has an upward tendency and finally converges. It can be observed in Figure 4.8(b) that the regulation cost optimized by DRL is close to the lower bound achieved by QP solver, and significantly improved in comparison with the cost without control, where the average regulation cost is \$25.435 for baseline, \$12.542 for DRL and \$8.734 for QP. In addition, the CPU running time for action generation is 0.0004 s in average, which is far less than the length of control interval.

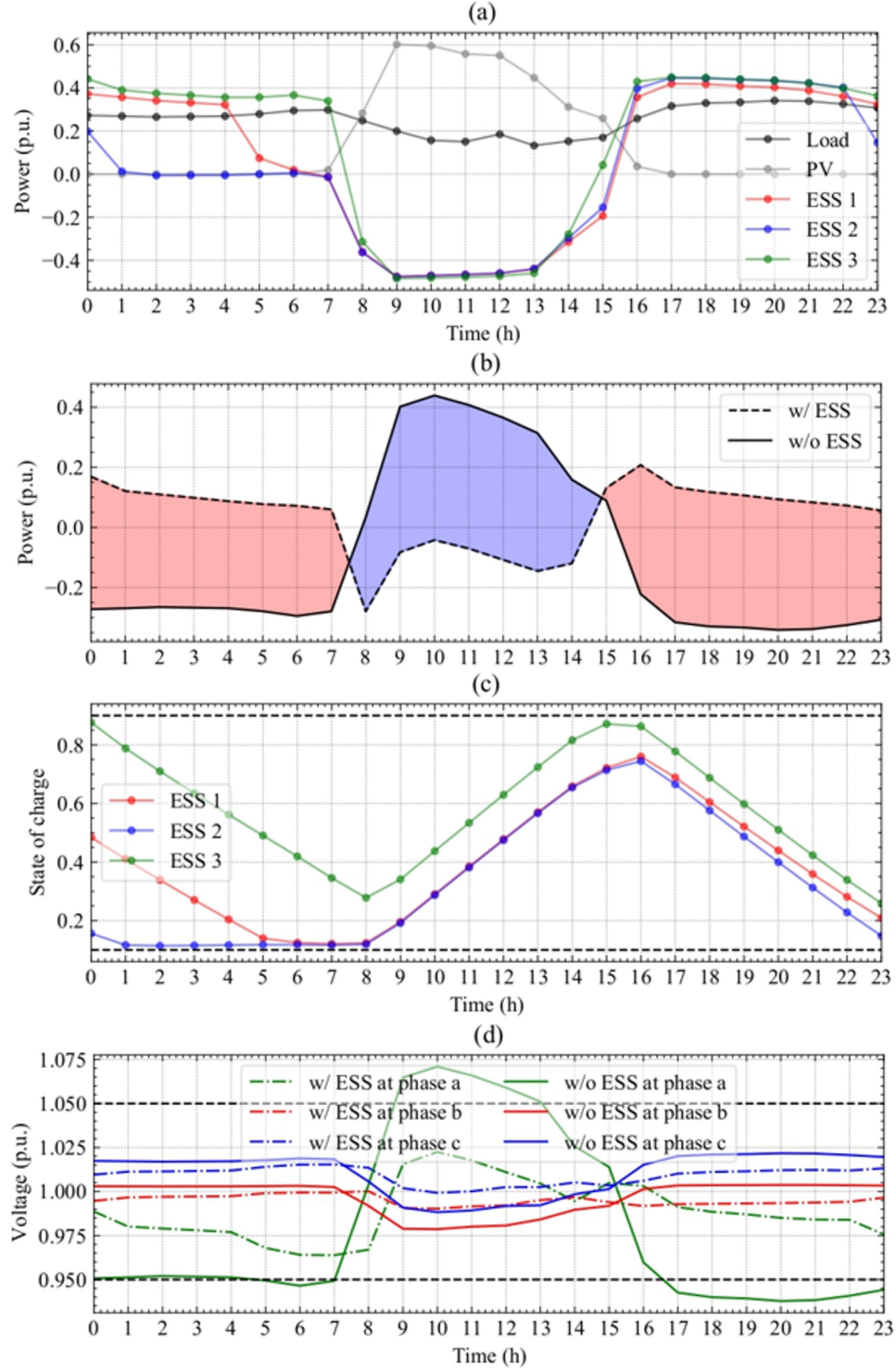


Figure 4.6: Illustration of (a) a sample daily profile with load, PV, and ESSs; (b) active power injection profile at bus 5; (c) ESS SOC profiles; and (d) voltage profile at bus 5 where dash lines represent upper/lower bounds.

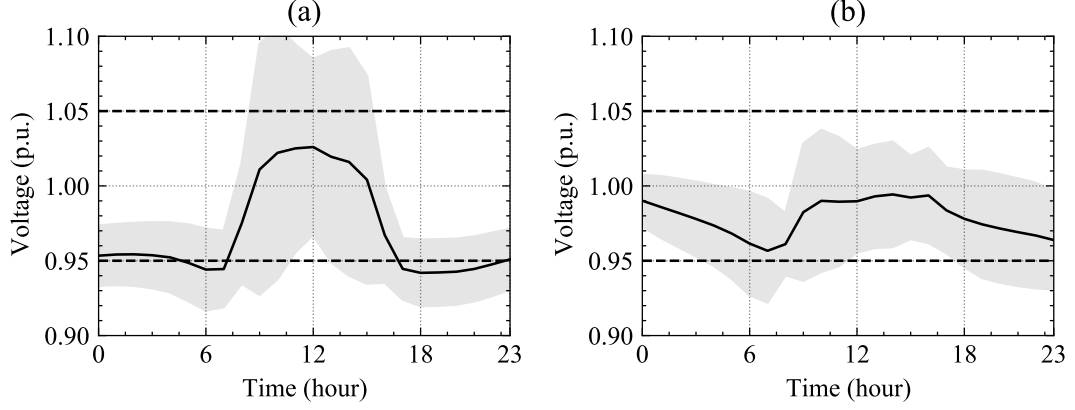


Figure 4.7: Voltage dynamics of phase a bus 5 (a) w/o and (b) w control

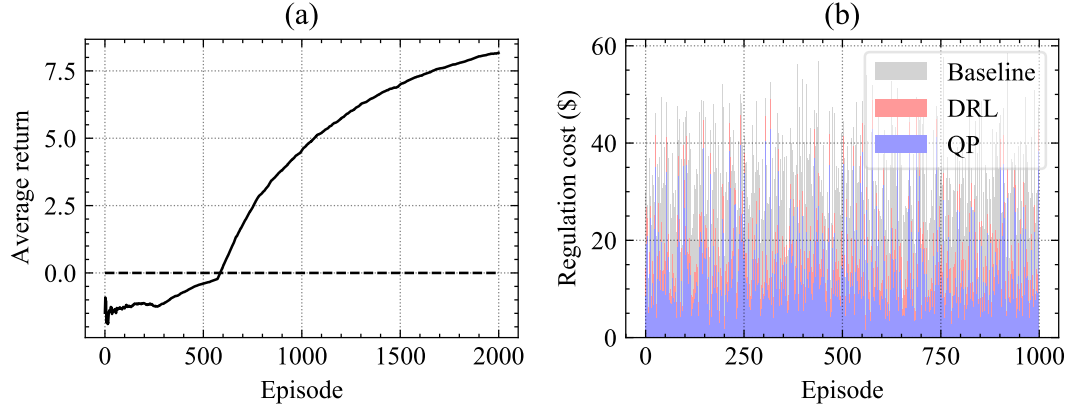


Figure 4.8: Illustration of learning performance (a) during training and voltage regulation performance (b) during execution where the dash line is a baseline

A sample daily profile is shown Figure 4.9, which is different from the profile used in the 6-bus case. The ESSs of phase a are scheduled to charge power from the grid to mitigate the over-voltage issue and discharge power to the grid to mitigate the under-voltage issue. It can be observed that phase b ESSs do not follow previous similar control effect. One possible reason is that the positive reward contribution of phase b ESSs is relatively small. The daily voltage profiles for all three phases are significantly improved with no voltage violations. It can be observed from Figure 4.10 that the regulated voltage dynamics is more flat with the schedule of ESSs by DRL.

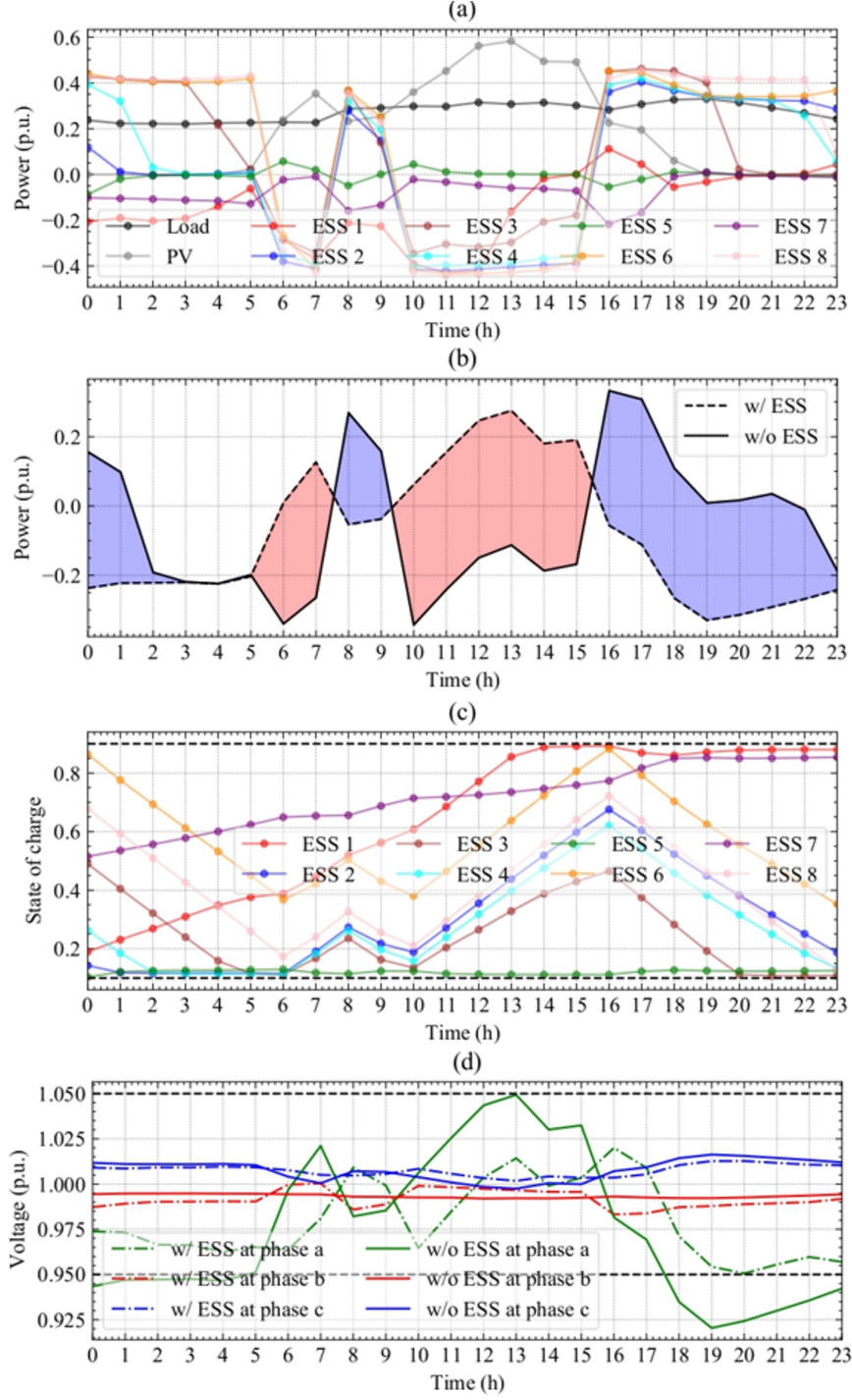


Figure 4.9: Illustration of (a) a sample daily profile with load, PV, and ESS; (b) active power injection profile at bus 11; (c) ESS SOC profiles; and (d) voltage profiles at bus 11, with ESSs at bus 7, 9, 10, 11, 13, 23, 32, 33 are relabeled as ESS 1, 2, 3, 4, 5, 6, 7, 8, respectively

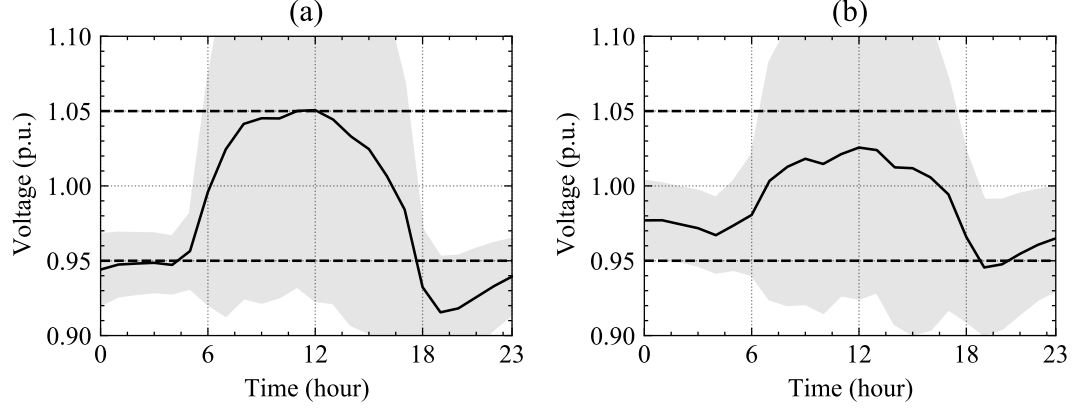


Figure 4.10: Voltage dynamics of phase a bus 11 (a) w/o and (b) w control.

4.6 Conclusion

This chapter proposes a DRL-based ESS-scheduling algorithm for real-time voltage regulation in LV UDS. The voltage regulation problem was formulated as a multi-stage quadratic stochastic program with the objective of minimizing the expected total daily voltage regulation cost, which could explore a day-horizon, network-wide and continuous voltage control effect by the ESSs. To make the problem tractable, the problem was further accommodated to a DRL setting by developing a learn-to-schedule feedback control framework, which can adaptively find an effective, near-optimal, and computational-efficient policy in a simulator environment instead of modeling complicated uncertainties.

CHAPTER 5

LEARNING-BASED FLEXIBILITY AGGREGATION UNDER UNCERTAINTY

The large-scale integration of DERs converts the role of a distribution system from a customer to a prosumer. In this context, the controllable DERs are expected to provide capacity support for the transmission system, which can be considered as power flexibility aggregation. Specifically, it is a process of controlling the power output of DERs to fulfill the desired capacity or flexibility, which can be arbitrarily selected from a feasible power flexibility region. To estimate a feasible power flexibility region, a Chebyshev centering optimization model is developed to approximate the active-reactive power flexibility region as time-decoupled two-dimensional Euclidean balls. For an ahead-of-time power flexibility region estimation problem, the inherent forecast error uncertainty tends to compromise the estimation results and arouse the aggregation safety issue. That is, some infeasible power flexibility points may exist in the estimated region. Once those points are selected, the aggregation process will probably be forcedly terminated. Therefore, this chapter develops a chance-constrained Chebyshev centering optimization model to handle this uncertainty by controlling the risk of violation of operation constraints. By transforming probabilistic constraints into deterministic constraints, solving such a chance-constrained problem can be viewed as unsupervisedly learning multiple conditional hyperplanes from the uncertainty data.

The remainder of this chapter is organized as follows. Section 4.1 presents a system model including the device, network and uncertainty propagation model. Section 4.2 first constructs three types of non-feeder-level power flexibility regions, and then utilizes a coordinate transformation to convert non-feeder-level power flexibility regions into a feeder-level region. Section 4.3 proposes an inner approximation framework. Section 4.4 validates the proposed method. Finally, Section 4.5 concludes this chapter.

5.1 System Model

In this dissertation, a multiphase unbalanced distribution network with buses collected in the index set $\mathcal{N} = \{0, 1, \dots, N\}$ is considered where the index 0 represents the substation bus. For simplicity, assume that all the buses have three phases: a, b, and c. Each electric device can be multi-phase wye-connected or delta-connected to the network. Denote the set $\Phi_Y := \{a, b, c\}$ and $\Phi_\Delta := \{ab, bc, ca\}$. Let the aggregation time horizon be discretized into T periods, and let the duration of each period be τ .

5.1.1 Device Model

In this subsection, three types of typical DERs are considered: PV (with front-end inverters), ESSs, and directly controlled loads (DCLs). Note that the proposed method can be naturally extended to account for other types of DERs so long as individual power region over a whole aggregation period is a polyhedron.

PV Inverters

The index set of buses connected with PV inverters is denoted as \mathcal{N}_{PV} . For a PV inverter installed at phase $\phi \in \Phi_Y \cup \Phi_\Delta$ and bus $k \in \mathcal{N}_{\text{PV}}$, its active and reactive power injections at time t are constrained by

$$0 \leq p_{k,\text{PV}}^{\phi,t} \leq \hat{p}_{k,\text{PV}}^{\phi,t}, \quad \forall t \in \{1, \dots, T\} \quad (5.1)$$

$$p_{k,\text{PV}}^{\phi,t^2} + q_{k,\text{PV}}^{\phi,t^2} \leq s_{k,\text{PV}}^{\phi^2}, \quad \forall t \in \{1, \dots, T\} \quad (5.2)$$

$$|q_{k,\text{PV}}^{\phi,t}| \leq p_{k,\text{PV}}^{\phi,t} \cdot \tan \theta_{k,\text{PV}}, \quad \forall t \in \{1, \dots, T\} \quad (5.3)$$

where $\hat{p}_{k,\text{PV}}^{\phi,t}$ is the expected available maximum active power of this PV inverter at time t , $s_{k,\text{PV}}^{\phi}$ is the nameplate capacity of this PV inverter, and $\theta_{k,\text{PV}}$ is the maximum power factor angle of this PV inverter. Constraints defined by Equation 5.1, Equation 5.2, and Equation 5.3 are active power limit, apparent power capacity limit, and power factor limit, respectively. Assume that over-sized PV inverters are installed, i.e.,

$$s_{k,\text{PV}}^{\phi} = \eta \bar{p}_{k,\text{PV}}^{\phi} (\eta > 1)$$

where $\bar{p}_{k,\text{PV}}^{\phi}$ is the rated active power of this PV inverter, and its maximum power factor angle is set as $\theta_{k,\text{PV}} \leq \theta_{k,\text{PV}}^c = \arccos \frac{1}{\eta}$. Then nonlinear constraint Equation 5.2 can be removed, the power region of this PV inverter at time t becomes a triangle, as shown in Figure 5.1 (a). Define the vector $\mathbf{x}_{k,\text{PV}} := [(p_{k,\text{PV}}^{\phi,t})_{t \in \{1, \dots, T\}}, (q_{k,\text{PV}}^{\phi,t})_{t \in \{1, \dots, T\}}]^{\top}$. According to Equation 5.1 and Equation 5.3, the power region of this PV inverter over a whole aggregation period can be written in a compact way as

$$\mathbf{A}_{k,\text{PV}} \mathbf{x}_{k,\text{PV}} \leq \mathbf{b}_{k,\text{PV}} \quad (5.4)$$

where

$$\begin{aligned} \mathbf{A}_{k,\text{PV}} &= \begin{bmatrix} [-\mathbf{I}_T, \mathbf{0}_{T \times T}] \\ [\mathbf{I}_T, \mathbf{0}_{T \times T}] \\ [-\tan \theta_{k,\text{PV}} \cdot \mathbf{I}_T, \mathbf{I}_T] \\ [-\tan \theta_{k,\text{PV}} \cdot \mathbf{I}_T, -\mathbf{I}_T] \end{bmatrix}, \\ \mathbf{b}_{k,\text{PV}} &= \begin{bmatrix} \mathbf{0}_T \\ \hat{\mathbf{p}}_{k,\text{PV}}^{\phi} \\ \mathbf{0}_T \\ \mathbf{0}_T \end{bmatrix}, \\ \hat{\mathbf{p}}_{k,\text{PV}}^{\phi} &= [\hat{p}_{k,\text{PV}}^{\phi,1}, \dots, \hat{p}_{k,\text{PV}}^{\phi,T}]^{\top}. \end{aligned}$$

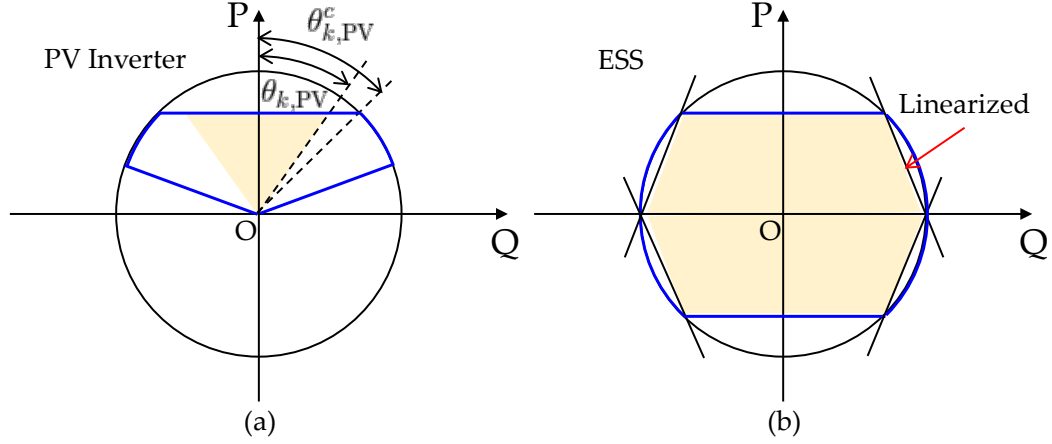


Figure 5.1: Illustration of the power region at time t for (a): a PV inverter and (b): an ESS where the area depicted by a blue line is the original nonlinear power region, and the area colored by yellow is the polygonal power region.

Without loss of generality, three-phase electric devices can be modeled as equivalent single-phase devices, because they have equal power at each phase. With a slight abuse of notation, the superscript ϕ for a three-phase device can be denoted as any phase in Φ_Y or Φ_Δ depending on its connection type. When calculating the system power flow, three-phase power injections can be treated as an equal power injection at each phase provided by the same single-phase device. Note that in this dissertation PV inverters are required to maintain a certain level of power factor at the expense of losing some reactive power flexibility. For example, similar to [87], if the η is chosen as 1.08, the $\theta_{k,PV}^c$ is 22.19° . If the $\theta_{k,PV}$ is designed as 18.19° , i.e., the corresponding minimum power factor is 0.95, the power region of this PV inverter at time t becomes a triangle.

ESSs

The index set of buses connected with ESSs is denoted as \mathcal{N}_{ESS} . For a ESS installed at phase $\phi \in \Phi_Y \cup \Phi_\Delta$ and bus $k \in \mathcal{N}_{\text{ESS}}$, its active and reactive power injections at time t are constrained by

$$-\bar{p}_{k,\text{ESS}}^\phi \leq p_{k,\text{ESS}}^{\phi,t} \leq \bar{p}_{k,\text{ESS}}^\phi, \quad \forall t \in \{1, \dots, T\} \quad (5.5)$$

$$p_{k,\text{ESS}}^{\phi,t^2} + q_{k,\text{ESS}}^{\phi,t^2} \leq s_{k,\text{ESS}}^{\phi^2}, \quad \forall t \in \{1, \dots, T\} \quad (5.6)$$

where $\bar{p}_{k,\text{ESS}}^\phi$ is the maximum discharging rate of this ESS, and $s_{k,\text{ESS}}^\phi$ is the nameplate capacity of this ESS. Constraints defined by Equation 5.5 and Equation 5.6 are active power limit, and apparent power capacity constraint, respectively. To construct a polygonal power region for ESSs, the boundary of constraint Equation 5.6 is linearized, as shown in Figure 5.1 (b). Then Equation 5.6 can be approximated by

$$\begin{cases} p_{k,\text{ESS}}^{\phi,t} - m_{k,\text{ESS}}^\phi q_{k,\text{ESS}}^{\phi,t} \leq n_{k,\text{ESS}}^\phi, & -p_{k,\text{ESS}}^{\phi,t} - m_{k,\text{ESS}}^\phi q_{k,\text{ESS}}^{\phi,t} \leq n_{k,\text{ESS}}^\phi, \\ p_{k,\text{ESS}}^{\phi,t} + m_{k,\text{ESS}}^\phi q_{k,\text{ESS}}^{\phi,t} \leq n_{k,\text{ESS}}^\phi, & -p_{k,\text{ESS}}^{\phi,t} + m_{k,\text{ESS}}^\phi q_{k,\text{ESS}}^{\phi,t} \leq n_{k,\text{ESS}}^\phi \end{cases}, \quad \forall t \in \{1, \dots, T\} \quad (5.7)$$

where $m_{k,\text{ESS}}^\phi = \frac{\bar{p}_{k,\text{ESS}}^\phi}{\sqrt{s_{k,\text{ESS}}^{\phi^2} - \bar{p}_{k,\text{ESS}}^{\phi^2} - s_{k,\text{ESS}}^\phi}}$, and $n_{k,\text{ESS}}^\phi = \frac{\bar{p}_{k,\text{ESS}}^\phi \cdot s_{k,\text{ESS}}^\phi}{s_{k,\text{ESS}}^\phi - \sqrt{s_{k,\text{ESS}}^{\phi^2} - \bar{p}_{k,\text{ESS}}^{\phi^2} - s_{k,\text{ESS}}^\phi}}$. In addition, its SOC at time t is constrained by

$$\underline{e}_{k,\text{ESS}} \leq e_{k,\text{ESS}}^t \leq \bar{e}_{k,\text{ESS}}, \quad \forall t \in \{1, \dots, T\} \quad (5.8)$$

where

- $e_{k,\text{ESS}}^t = e_{k,\text{ESS}}^{t-1} - \frac{\tau \cdot \sigma_{k,\text{ESS}}}{\bar{E}_{k,\text{ESS}}} p_{k,\text{ESS}}^{\phi,t}$, $e_{k,\text{ESS}}^0$ is initial state of charge of this ESS,
- $\sigma_{k,\text{ESS}}$ is an indicator function, i.e., $\sigma_{k,\text{ESS}} = 3$ if this ESS is a three-phase device; otherwise, $\sigma_{k,\text{ESS}} = 1$,
- $\bar{E}_{k,\text{ESS}}$ is the capacity of this ESS, and
- $\underline{e}_{k,\text{ESS}}(\bar{e}_{k,\text{ESS}})$ is the minimum (maximum) allowable SOC of this ESS.

Define the vector $\mathbf{x}_{\text{ESS}} := [(p_{k,\text{ESS}}^{\phi,t})_{t \in \{1, \dots, T\}}, (q_{k,\text{ESS}}^{\phi,t})_{t \in \{1, \dots, T\}}]^\top$. According to Equation 5.5, Equation 5.7 and Equation 5.8, the power region of this ESS over a whole

aggregation period can be written in a compact way as

$$\mathbf{A}_{k,\text{ESS}} \mathbf{x}_{k,\text{ESS}} \leq \mathbf{b}_{k,\text{ESS}} \quad (5.9)$$

where

$$\mathbf{A}_{k,\text{ESS}} = \begin{bmatrix} [-\mathbf{I}_T, \mathbf{0}_{T \times T}] \\ [\mathbf{I}_T, \mathbf{0}_{T \times T}] \\ [\mathbf{I}_T, -m_{k,\text{ESS}}^\phi \cdot \mathbf{I}_T] \\ [-\mathbf{I}_T, -m_{k,\text{ESS}}^\phi \cdot \mathbf{I}_T] \\ [\mathbf{I}_T, m_{k,\text{ESS}}^\phi \cdot \mathbf{I}_T] \\ [-\mathbf{I}_T, m_{k,\text{ESS}}^\phi \cdot \mathbf{I}_T] \\ [-\frac{\tau \cdot \sigma_{k,\text{ESS}}}{\bar{E}_{k,\text{ESS}}} \cdot \mathbf{L}_T, \mathbf{0}_{T \times T}] \\ [\frac{\tau \cdot \sigma_{k,\text{ESS}}}{\bar{E}_{k,\text{ESS}}} \cdot \mathbf{L}_T, \mathbf{0}_{T \times T}] \end{bmatrix},$$

$$\mathbf{b}_{k,\text{ESS}} = \begin{bmatrix} \bar{p}_{k,\text{ESS}}^\phi \cdot \mathbf{1}_T \\ \bar{p}_{k,\text{ESS}}^\phi \cdot \mathbf{1}_T \\ n_{k,\text{ESS}}^\phi \cdot \mathbf{1}_T \\ n_{k,\text{ESS}}^\phi \cdot \mathbf{1}_T \\ n_{k,\text{ESS}}^\phi \cdot \mathbf{1}_T \\ n_{k,\text{ESS}}^\phi \cdot \mathbf{1}_T \\ \bar{e}_{k,\text{ESS}}' \cdot \mathbf{1}_T \\ \underline{e}_{k,\text{ESS}}' \cdot \mathbf{1}_T \end{bmatrix},$$

$$\bar{e}_{k,\text{ESS}}' = \bar{e}_{k,\text{ESS}} - e_{k,\text{ESS}}^0,$$

$$\underline{e}_{k,\text{ESS}}' = e_{k,\text{ESS}}^0 - \underline{e}_{k,\text{ESS}}.$$

For simplicity, it is assumed that the maximum charging rate of an ESS is equal to its maximum discharging rate, and there is no power loss in the charging or discharging process. Note that the initial state of charge is known before the flexibility region evaluation is implemented because the ESSs here are assumed to be only used for power flexibility

support service.

DCLs

The index set of buses connected with DCLs is denoted as \mathcal{N}_{DCL} . For a DCL installed at phase $\phi \in \Phi_Y \cup \Phi_\Delta$ and bus $k \in \mathcal{N}_{\text{DCL}}$, its active power injections at time t are constrained by

$$-\bar{p}_{k,\text{DCL}}^\phi \leq p_{k,\text{DCL}}^{\phi,t} \leq -\underline{p}_{k,\text{DCL}}^\phi, \quad \forall t \in \{1, \dots, T\} \quad (5.10)$$

where $\underline{p}_{k,\text{DCL}}^\phi$ ($\bar{p}_{k,\text{DCL}}^\phi$) is the minimum (maximum) active power demand. Constraint Equation 5.10 is active power limit. Define the vector $\mathbf{x}_{\text{DCL}} := [(p_{k,\text{DCL}}^{\phi,t})_{t \in \{1, \dots, T\}}]^\top$. According to Equation 5.10, the active power region of this DCL over a whole aggregation period can be written in a compact way as

$$\mathbf{A}_{k,\text{DCL}} \mathbf{x}_{k,\text{DCL}} \leq \mathbf{b}_{k,\text{DCL}} \quad (5.11)$$

where

$$\mathbf{A}_{k,\text{DCL}} = \begin{bmatrix} \mathbf{I}_T \\ -\mathbf{I}_T \end{bmatrix}, \quad \mathbf{b}_{k,\text{DCL}} = \begin{bmatrix} -\underline{p}_{k,\text{DCL}}^\phi \cdot \mathbf{1}_T \\ \bar{p}_{k,\text{DCL}}^\phi \cdot \mathbf{1}_T \end{bmatrix}.$$

5.1.2 Network Model

In this subsection, a linear power flow model developed in [88] is used to calculate the system voltage magnitudes, and the total three phase net active and reactive power injections at the substation. The system voltage magnitudes at time t can be written as

$$\mathbf{v}^t = \sum_{k \in \mathcal{N} \setminus \{0\}} \mathbf{a}_{k,p}^\phi p_k^{\phi,t} + \sum_{k \in \mathcal{N} \setminus \{0\}} \mathbf{a}_{k,q}^\phi q_k^{\phi,t} + \mathbf{a}_0, \quad \forall t \in \{1, \dots, T\} \quad (5.12)$$

where

- $\mathbf{v}^t \in \mathbb{R}^{3N}$ collects the voltage magnitudes from all phases of all buses at time t ,
- $p_k^{\phi,t}(q_k^{\phi,t})$ is the total active (reactive) power injection at phase ϕ at bus k at time t ,
- $\mathbf{a}_{k,p}^\phi(\mathbf{a}_{k,q}^\phi)$ is a column vector of system parameters corresponding to the total active (reactive) injections at phase ϕ at bus k , and
- \mathbf{a}_0 is a column vector of system parameters relative to the zero-load voltage.

Since

$$\begin{aligned} p_k^{\phi,t} &= p_{k,\text{PV}}^{\phi,t} + p_{k,\text{ESS}}^{\phi,t} + p_{k,\text{DCL}}^{\phi,t} + \hat{p}_{k,\text{L}}^{\phi,t}, \\ q_k^{\phi,t} &= q_{k,\text{PV}}^{\phi,t} + q_{k,\text{ESS}}^{\phi,t} + q_{k,\text{DCL}}^{\phi,t} + \hat{q}_{k,\text{L}}^{\phi,t}, \end{aligned}$$

and the respective device power injection is set as 0 when the specific type of device is not present, the \mathbf{v}_t can be written as

$$\begin{aligned} \mathbf{v}^t &= \sum_{k \in \mathcal{N}_{\text{PV}}} \mathbf{a}_{k,p}^\phi p_{k,\text{PV}}^{\phi,t} + \mathbf{a}_{k,q}^\phi q_{k,\text{PV}}^{\phi,t} + \sum_{k \in \mathcal{N}_{\text{ESS}}} \mathbf{a}_{k,p}^\phi p_{k,\text{ESS}}^{\phi,t} \\ &\quad + \mathbf{a}_{k,q}^\phi q_{k,\text{ESS}}^{\phi,t} + \sum_{k \in \mathcal{N}_{\text{DCL}}} \mathbf{a}_{k,p}^\phi p_{k,\text{DCL}}^{\phi,t} + \mathbf{a}_{k,q}^\phi q_{k,\text{DCL}}^{\phi,t} + \mathbf{a}^t, \quad \forall t \in \{1, \dots, T\} \end{aligned} \quad (5.13)$$

where

- $\mathbf{a}^t = \mathbf{a}_0 + \sum_{k \in \mathcal{N}_{\text{L}}} \mathbf{a}_{k,p}^\phi \hat{p}_{k,\text{L}}^{\phi,t} + \mathbf{a}_{k,q}^\phi \hat{q}_{k,\text{L}}^{\phi,t}$, $\hat{p}_{k,\text{L}}^{\phi,t}(\hat{q}_{k,\text{L}}^{\phi,t})$ is the expected active (reactive) power injection of an uncontrollable load, and
- \mathcal{N}_{L} is the index set of buses connected with uncontrollable loads.

Assume that DCLs and uncontrollable loads have the fixed power factor angles, and then $q_{k,\text{DCL}}^{\phi,t} = p_{k,\text{DCL}}^{\phi,t} \cdot \tan \theta_{k,\text{DCL}}$ and $\hat{q}_{k,\text{L}}^{\phi,t} = \hat{p}_{k,\text{L}}^{\phi,t} \cdot \tan \theta_{k,\text{L}}$.

The system voltage magnitudes are constrained by

$$\underline{\mathbf{v}} \leq \mathbf{v}^t \leq \overline{\mathbf{v}}, \quad \forall t \in \{1, \dots, T\} \quad (5.14)$$

where $\underline{\mathbf{v}}(\overline{\mathbf{v}})$ is the lower (upper) bound of system voltage magnitudes. In addition, similar to Equation 5.13, the total three phase net active and reactive power injections at the substation at time t can be written as

$$\begin{aligned} p_0^t = & \sum_{k \in \mathcal{N}_{PV}} b_{k,p}^\phi p_{k,PV}^{\phi,t} + b_{k,q}^\phi q_{k,PV}^{\phi,t} + \sum_{k \in \mathcal{N}_{ESS}} b_{k,p}^\phi p_{k,ESS}^{\phi,t} \\ & + b_{k,q}^\phi q_{k,ESS}^{\phi,t} + \sum_{k \in \mathcal{N}_{DCL}} b_{k,p}^\phi p_{k,DCL}^{\phi,t} + b_{k,q}^\phi q_{k,DCL}^{\phi,t} + b^t, \quad \forall t \in \{1, \dots, T\} \end{aligned} \quad (5.15)$$

$$\begin{aligned} q_0^t = & \sum_{k \in \mathcal{N}_{PV}} c_{k,p}^\phi p_{k,PV}^{\phi,t} + c_{k,q}^\phi q_{k,PV}^{\phi,t} + \sum_{k \in \mathcal{N}_{ESS}} c_{k,p}^\phi p_{k,ESS}^{\phi,t} \\ & + c_{k,q}^\phi q_{k,ESS}^{\phi,t} + \sum_{k \in \mathcal{N}_{DCL}} c_{k,p}^\phi p_{k,DCL}^{\phi,t} + c_{k,q}^\phi q_{k,DCL}^{\phi,t} + c^t, \quad \forall t \in \{1, \dots, T\} \end{aligned} \quad (5.16)$$

where

- $b^t = b_0 + \sum_{k \in \mathcal{N}_L} b_{k,p}^\phi \hat{p}_{k,L}^{\phi,t} + b_{k,q}^\phi \hat{q}_{k,L}^{\phi,t}$,
- $c^t = c_0 + \sum_{k \in \mathcal{N}_L} c_{k,p}^\phi \hat{p}_{k,L}^{\phi,t} + c_{k,q}^\phi \hat{q}_{k,L}^{\phi,t}$, and
- $b_0, b_{k,p}^\phi, b_{k,q}^\phi, c_0, c_{k,p}^\phi, c_{k,q}^\phi$ are system parameters.

If $p_0^t(q_0^t) > 0$, the distribution system absorbs the redundant energy from the transmission system. If $p_0^t(q_0^t) < 0$, the distribution system release the redundant energy to the transmission system.

Note that the system parameters corresponding to equivalent single-phase power injections of three-phase electric devices in Equation 5.13, Equation 5.15, and Equation 5.16 should be $\sum_\phi \mathbf{m}_{k,p}^\phi, \sum_\phi b_{k,p}^\phi, \sum_\phi c_{k,p}^\phi$ for active power injections and $\sum_\phi \mathbf{m}_{k,q}^\phi, \sum_\phi b_{k,q}^\phi, \sum_\phi c_{k,q}^\phi$ for reactive power injections. In essence, the power flow model used here

can be viewed as a linear interpolation between two power flow solutions: the given operational point and a known operational point with no power injection. The [88] validates that this linear model can provide an accurate approximation of unbalanced power flow, which has been applied in the existing papers [64, 89] on power flexibility aggregation.

5.1.3 Uncertainty Propagation Model

Denote the actual available maximum active power of PV inverters and uncontrollable load at time t as $\bar{p}_{k,\text{PV}}^{\phi,t}$ and $p_{k,\text{L}}^{\phi,t}$, respectively. The forecast errors of $\bar{p}_{k,\text{PV}}^{\phi,t}$ and $p_{k,\text{L}}^{\phi,t}$ constitute random variables and introduce the uncertainties to the system, and are denoted as $\xi_{k,\text{PV}}^{\phi,t}$ and $\xi_{k,\text{L}}^{\phi,t}$, respectively. When $\xi_{k,\text{PV}}^{\phi,t}(\xi_{k,\text{L}}^{\phi,t}) > 0$, it means that $\bar{p}_{k,\text{PV}}^{\phi,t}(p_{k,\text{L}}^{\phi,t})$ is underestimated. When $\xi_{k,\text{PV}}^{\phi,t}(\xi_{k,\text{L}}^{\phi,t}) < 0$, it means that $\bar{p}_{k,\text{PV}}^{\phi,t}(p_{k,\text{L}}^{\phi,t})$ is overestimated.

If the actual value is considered, the $\hat{\bar{p}}_{k,\text{PV}}^{\phi,t}$ is replaced by $\bar{p}_{k,\text{PV}}^{\phi,t}$ in Equation 5.1, and extra constraints for PV inverters are written as

$$p_{k,\text{PV}}^{\phi,t} \leq \bar{p}_{k,\text{PV}}^{\phi,t}, \quad \forall t \in \{1, \dots, T\} \quad (5.17)$$

where $\bar{p}_{k,\text{PV}}^{\phi,t} = \hat{\bar{p}}_{k,\text{PV}}^{\phi,t} + \xi_{k,\text{PV}}^{\phi,t}$.

If the actual value of $p_{k,\text{L}}^{\phi,t}$ is considered, the $\hat{p}_{k,\text{L}}^{\phi,t}$ is replaced by $p_{k,\text{L}}^{\phi,t}$ in Equation 5.14, and extra constraints for system voltage magnitudes are written as

$$\underline{\mathbf{v}} \leq \mathbf{v}^t + \sum_{k \in \mathcal{N}_{\text{L}}} (\mathbf{a}_{k,\text{p}}^{\phi} + \tan \theta_{k,\text{L}} \cdot \mathbf{a}_{k,\text{q}}^{\phi})(p_{k,\text{L}}^{\phi,t} - \hat{p}_{k,\text{L}}^{\phi,t}) \leq \bar{\mathbf{v}}, \quad \forall t \in \{1, \dots, T\} \quad (5.18)$$

where $p_{k,\text{L}}^{\phi,t} = \hat{p}_{k,\text{L}}^{\phi,t} + \xi_{k,\text{L}}^{\phi,t}$.

5.2 Flexibility Region Construction

In this section, based on the system model in Section 2.1, three-type non-feeder-level power flexibility regions (i.e., device-level, network-level and uncertainty-level power

flexibility regions) are defined and constructed first. Through a coordinate transformation, two-type feeder-level power flexibility regions (i.e., deterministic and stochastic power flexibility regions) are then derived.

5.2.1 Non-Feeder-Level Power Flexibility Region

Define the vector that collects $n = (2n_{\text{PV}} + 2n_{\text{ESS}} + n_{\text{DCL}}) \cdot T$ controllable power injections of all the DERs over a whole aggregation period $\mathbf{x} := [(\mathbf{x}_{k,\text{PV}}^\top)_{k \in \mathcal{N}_{\text{PV}}}, (\mathbf{x}_{k,\text{ESS}}^\top)_{k \in \mathcal{N}_{\text{ESS}}}, (\mathbf{x}_{k,\text{DCL}}^\top)_{k \in \mathcal{N}_{\text{DCL}}}]^\top$ where $n_{\text{PV}} = |\mathcal{N}_{\text{PV}}|$, $n_{\text{ESS}} = |\mathcal{N}_{\text{ESS}}|$, and $n_{\text{DCL}} = |\mathcal{N}_{\text{DCL}}|$. Define the vector that collects $n_\xi = (n_{\text{PV}} + n_{\text{L}}) \cdot T$ random variables $\vec{\xi} := [(\vec{\xi}_{k,\text{PV}}^\top)_{k \in \mathcal{N}_{\text{PV}}}, (\vec{\xi}_{k,\text{L}}^\top)_{k \in \mathcal{N}_{\text{L}}}]^\top$ where

$$\begin{aligned} n_{\text{L}} &= |\mathcal{N}_{\text{L}}|, \\ \vec{\xi}_{k,\text{PV}} &:= [(\xi_{k,\text{PV}}^{\phi,t})_{t \in \{1, \dots, T\}}]^\top, \\ \vec{\xi}_{k,\text{L}} &:= [(\xi_{k,\text{L}}^{\phi,t})_{t \in \{1, \dots, T\}}]^\top. \end{aligned}$$

According to Equation 5.4, Equation 5.9, and Equation 5.11, the device-level power flexibility region is defined as a power region of \mathbf{x} where individual device constraints are met, which can be written in a compact way as

$$\mathbf{A}_{\text{DEV}} \mathbf{x} \leq \mathbf{b}_{\text{DEV}} \quad (5.19)$$

where

$$\mathbf{A}_{\text{DEV}} = \begin{bmatrix} \mathbf{bdiag}(\mathbf{A}_{k,\text{PV}}) & \mathbf{0}_{4Tn_{\text{PV}} \times 2Tn_{\text{ESS}}} & \mathbf{0}_{4Tn_{\text{PV}} \times Tn_{\text{DCL}}} \\ \mathbf{0}_{8Tn_{\text{ESS}} \times 2Tn_{\text{PV}}} & \mathbf{bdiag}(\mathbf{A}_{k,\text{ESS}}) & \mathbf{0}_{8Tn_{\text{ESS}} \times Tn_{\text{DCL}}} \\ \mathbf{0}_{2Tn_{\text{DCL}} \times 2Tn_{\text{PV}}} & \mathbf{0}_{2Tn_{\text{DCL}} \times 2Tn_{\text{ESS}}} & \mathbf{bdiag}(\mathbf{A}_{k,\text{DCL}}) \end{bmatrix},$$

and

$$\mathbf{b}_{\text{DEV}} = [(\mathbf{b}_{k,\text{PV}}^\top)_{k \in \mathcal{N}_{\text{PV}}}, (\mathbf{b}_{k,\text{ESS}}^\top)_{k \in \mathcal{N}_{\text{ESS}}}, (\mathbf{b}_{k,\text{DCL}}^\top)_{k \in \mathcal{N}_{\text{DCL}}}]^\top.$$

According to Equation 5.14, the network-level power flexibility region is defined as a power region of \mathbf{x} where system voltage constraints are met, which can be written in a compact way as

$$\mathbf{A}_{\text{NET}}\mathbf{x} \leq \mathbf{b}_{\text{NET}} \quad (5.20)$$

where

$$\begin{aligned} \mathbf{A}_{\text{NET}} &= \begin{bmatrix} -\mathbf{A} \\ \mathbf{A} \end{bmatrix}, \\ \mathbf{b}_{\text{NET}} &= \begin{bmatrix} -\mathbf{1}_T \otimes \underline{\mathbf{v}} + \mathbf{a} \\ \mathbf{1}_T \otimes \bar{\mathbf{v}} - \mathbf{a} \end{bmatrix}, \\ \mathbf{A} &= \begin{bmatrix} \mathbf{A}^1 & \dots & \mathbf{A}^t & \dots & \mathbf{A}^T \end{bmatrix}^\top, \\ \mathbf{a} &= [(\mathbf{a}^{t^\top})_{t \in \{1, \dots, T\}}]^\top. \end{aligned}$$

Denote \mathbf{A}^t 's j -th column and \mathbf{x} 's j -th element as the vector \mathbf{A}_j^t and the scalar \mathbf{x}_j , respectively. Then \mathbf{A}^t can be achieved based on the following rule

$$\mathbf{A}_j^t = \begin{cases} \mathbf{a}_{k,p}^\phi & \text{if } \mathbf{x}_j = p_{k,(\cdot)}^{\phi,t} \\ \mathbf{a}_{k,q}^\phi & \text{if } \mathbf{x}_j = q_{k,(\cdot)}^{\phi,t} \\ \mathbf{0}_{3N} & \text{if otherwise} \end{cases}.$$

According to Equation 5.17 and Equation 5.18, the uncertainty-level power flexibility region is defined as a power region of \mathbf{x} where uncertainty-relative constraints are met, which can be written in a compact way as

$$\mathbf{A}_{\text{UNC}}\mathbf{x} + \mathbf{G}_{\text{UNC}}\vec{\xi} \leq \mathbf{b}_{\text{UNC}} \quad (5.21)$$

where

$$\begin{aligned}
\mathbf{A}_{\text{UNC}} &= \begin{bmatrix} \mathbf{A}_{\text{DEV}}^{\xi} \\ \mathbf{A}_{\text{NET}}^{\xi} \end{bmatrix}, \mathbf{G}_{\text{UNC}} = \begin{bmatrix} \mathbf{G}_{\text{DEV}}^{\xi} \\ \mathbf{G}_{\text{NET}}^{\xi} \end{bmatrix}, \mathbf{b}_{\text{UNC}} = \begin{bmatrix} \mathbf{b}_{\text{DEV}}^{\xi} \\ \mathbf{b}_{\text{NET}}^{\xi} \end{bmatrix}, \\
\mathbf{A}_{\text{DEV}}^{\xi} &= \begin{bmatrix} \mathbf{bdiag}([\mathbf{I}_T, \mathbf{0}_{T \times T}]) & \mathbf{0}_{T n_{\text{PV}} \times 2T n_{\text{ESS}}} & \mathbf{0}_{T n_{\text{PV}} \times T n_{\text{DCL}}} \end{bmatrix}, \\
\mathbf{G}_{\text{DEV}}^{\xi} &= \begin{bmatrix} \mathbf{bdiag}(-\mathbf{I}_T) & \mathbf{0}_{T n_{\text{PV}} \times T n_L} \end{bmatrix}, \\
\mathbf{b}_{\text{DEV}}^{\xi} &= [(\hat{\mathbf{p}}_{k,\text{PV}}^{\phi^\top})_{k \in \mathcal{N}_{\text{PV}}}]^\top, \\
\mathbf{A}_{\text{NET}}^{\xi} &= \mathbf{A}_{\text{NET}}, \\
\mathbf{G}_{\text{NET}}^{\xi} &= \begin{bmatrix} -\mathbf{G} & \mathbf{G} \end{bmatrix}^\top, \\
\mathbf{b}_{\text{NET}}^{\xi} &= \mathbf{b}_{\text{NET}}, \\
\mathbf{G} &= \begin{bmatrix} \mathbf{G}^1 \dots \mathbf{G}^t \dots \mathbf{G}^T \end{bmatrix}^\top.
\end{aligned}$$

Denote \mathbf{G}^t 's j -th column and $\vec{\xi}$'s j -th element as the vector \mathbf{G}_j^t and the scalar ξ_j , respectively. Then \mathbf{G}^t can be achieved based on the following rule

$$\mathbf{G}_j^t = \begin{cases} \mathbf{a}_{k,\text{p}}^{\phi} + \tan \theta_{k,\text{L}} \cdot \mathbf{a}_{k,\text{q}}^{\phi} & \text{if } \xi_j = \xi_{k,\text{L}}^{\phi,t} \\ \mathbf{0}_{3N} & \text{if otherwise} \end{cases}.$$

5.2.2 Feeder-Level Power Flexibility Region

Denote the vector that collects $2T$ controllable power injections at the substation over a whole aggregation period $\mathbf{y} := [(\mathbf{y}^{t^\top})_{t \in \{1, \dots, T\}}]^\top$ where $\mathbf{y}^t := [p_0^t, q_0^t]^\top$. If the actual value is considered, the $\hat{p}_{k,\text{L}}^{\phi,t}$ is replaced by $p_{k,\text{L}}^{\phi,t}$ in Equation 5.15 and Equation 5.16, and then the p_0^t and q_0^t can be written as

$$\begin{aligned}
p_0^t &= \mathbf{b}_x^{t^\top} \mathbf{x} + \mathbf{b}_\xi^{t^\top} \vec{\xi} + b^t \\
q_0^t &= \mathbf{c}_x^{t^\top} \mathbf{x} + \mathbf{c}_\xi^{t^\top} \vec{\xi} + c^t
\end{aligned} \tag{5.22}$$

Denote \mathbf{b}_x^t 's, \mathbf{b}_ξ^t 's, \mathbf{c}_x^t 's and \mathbf{c}_ξ^t 's j -th element as the scalar $\mathbf{b}_{x,j}^t$, $\mathbf{b}_{\xi,j}^t$, $\mathbf{c}_{x,j}^t$ and $\mathbf{c}_{\xi,j}^t$, respectively. Then \mathbf{b}_x^t , \mathbf{b}_ξ^t , \mathbf{c}_x^t and \mathbf{c}_ξ^t can be achieved based on the following rules

$$\mathbf{b}_{x,j}^t = \begin{cases} b_{k,p}^\phi & \text{if } \mathbf{x}_j = p_{k,(\cdot)}^{\phi,t} \\ b_{k,q}^\phi & \text{if } \mathbf{x}_j = q_{k,(\cdot)}^{\phi,t}, \\ 0 & \text{if otherwise} \end{cases}$$

$$\mathbf{b}_{\xi,j}^t = \begin{cases} b_{k,p}^\phi + \tan \theta_{k,L} \cdot b_{k,q}^\phi & \text{if } \vec{\xi}_j = \xi_{k,L}^{\phi,t} \\ 0 & \text{if otherwise} \end{cases},$$

$$\mathbf{c}_{x,j}^t = \begin{cases} c_{k,p}^\phi & \text{if } \mathbf{x}_j = p_{k,(\cdot)}^{\phi,t} \\ c_{k,q}^\phi & \text{if } \mathbf{x}_j = q_{k,(\cdot)}^{\phi,t}, \\ 0 & \text{if otherwise} \end{cases}$$

and

$$\mathbf{c}_{\xi,j}^t = \begin{cases} c_{k,p}^\phi + \tan \theta_{k,L} \cdot c_{k,q}^\phi & \text{if } \vec{\xi}_j = \xi_{k,L}^{\phi,t} \\ 0 & \text{if otherwise} \end{cases}.$$

According to Equation 5.22, the \mathbf{y} can be written as

$$\mathbf{y} = \mathbf{H}_x \mathbf{x} + \mathbf{H}_\xi \vec{\xi} + \mathbf{h} \quad (5.23)$$

where

$$\mathbf{H}_x = \begin{bmatrix} \mathbf{H}_x^1 \\ \vdots \\ \mathbf{H}_x^t \\ \vdots \\ \mathbf{H}_x^T \end{bmatrix}, \mathbf{H}_\xi = \begin{bmatrix} \mathbf{H}_\xi^1 \\ \vdots \\ \mathbf{H}_\xi^t \\ \vdots \\ \mathbf{H}_\xi^T \end{bmatrix}, \mathbf{h} = \begin{bmatrix} \mathbf{h}^1 \\ \vdots \\ \mathbf{h}^t \\ \vdots \\ \mathbf{h}^T \end{bmatrix},$$

$$\mathbf{H}_x^t = \begin{bmatrix} \mathbf{b}_x^{t^\top} \\ \mathbf{c}_x^{t^\top} \end{bmatrix}, \mathbf{H}_\xi = \begin{bmatrix} \mathbf{b}_\xi^{t^\top} \\ \mathbf{c}_\xi^{t^\top} \end{bmatrix}, \mathbf{h} = \begin{bmatrix} b^t \\ c^t \end{bmatrix}.$$

Since $\mathbf{x} \in \mathbb{R}^n$, every \mathbf{x} can be expressed as a linear combination of n linearly independent vectors

$$\mathbf{x} = \mathbf{W}_e \mathbf{e} + \mathbf{W}_f \mathbf{f} = \sum_{i=1}^{2T} e_i \mathbf{w}_i + \sum_{j=1}^{n-2T} f_j \mathbf{w}_{j+2T} \quad (5.24)$$

where

- $\mathbf{W}_e = [\mathbf{w}_1, \dots, \mathbf{w}_{2T}] \in \mathbb{R}^{n \times 2T}$, $\mathbf{w}_i \in \mathbb{R}^n$,
- $\mathbf{e} = [e_1, \dots, e_{2T}]^\top \in \mathbb{R}^{2T}$,
- $\mathbf{W}_f = [\mathbf{w}_{2T+1}, \dots, \mathbf{w}_n] \in \mathbb{R}^{n \times (n-2T)}$, and
- $\mathbf{f} = [f_1, \dots, f_{n-2T}]^\top \in \mathbb{R}^{n-2T}$.

Also since the rows of \mathbf{H}_x are linearly independent, the number of bases for the null-space of \mathbf{H}_x is $n - 2T$. Then $\mathbf{w}_1, \dots, \mathbf{w}_n$ can be designed as orthogonal bases of \mathbb{R}^n such that $\mathbf{w}_{2T+1}, \dots, \mathbf{w}_n$ span the null-space of \mathbf{H}_x . Specially, \mathbf{W}_e and \mathbf{W}_f can be achieved by using Gram-Schmidt [90] to orthonormalize the bases of the row-space and null-space of \mathbf{H}_x , respectively.

By inserting Equation 5.24 into Equation 5.23, \mathbf{y} can be rewritten in the new coordinates as

$$\mathbf{y} = \mathbf{H}_x (\mathbf{W}_e \mathbf{e} + \mathbf{W}_f \mathbf{f}) + \mathbf{H}_\xi \vec{\xi} + \mathbf{h} = \hat{\mathbf{H}}_x \mathbf{e} + \mathbf{H}_\xi \vec{\xi} + \mathbf{h} \quad (5.25)$$

where $\hat{\mathbf{H}}_x = \mathbf{H}_x \mathbf{W}_e \in \mathbb{R}^{2T \times 2T}$ is invertible, and $\mathbf{H}_x \mathbf{W}_f$ vanishes because \mathbf{W}_f spans the null-space of \mathbf{W}_x . Then, \mathbf{e} can be expressed as

$$\mathbf{e} = \hat{\mathbf{H}}_x^{-1} (\mathbf{y} - \mathbf{H}_\xi \vec{\xi} - \mathbf{h}) \quad (5.26)$$

Using Equation 5.26 to substitute for \mathbf{x} in Equation 5.19 - Equation 5.21, the feeder-level power flexibility region is defined as a feasible power region of \mathbf{y} where any point can be fulfilled by the dispatch of DERs (i.e., \mathbf{x}) without violating their individual operation constraints, which can be written in a compact way as

$$\mathbf{A}_y \mathbf{y} + \mathbf{A}_\xi \vec{\xi} + \mathbf{A}_f \mathbf{f} \leq \mathbf{b}_y \quad (5.27)$$

where

$$\begin{aligned} \mathbf{A}_y &= \begin{bmatrix} \mathbf{A}_{\text{DEV}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \\ \mathbf{A}_{\text{NET}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \\ \mathbf{A}_{\text{UNC}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \end{bmatrix}, \\ \mathbf{A}_\xi &= \begin{bmatrix} \mathbf{A}_{\text{DEV}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \mathbf{H}_\xi \\ \mathbf{A}_{\text{NET}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \mathbf{H}_\xi \\ \mathbf{G}_{\text{UNC}} - \mathbf{A}_{\text{UNC}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \mathbf{H}_\xi \end{bmatrix}, \\ \mathbf{A}_f &= \begin{bmatrix} \mathbf{A}_{\text{DEV}} \mathbf{W}_f \\ \mathbf{A}_{\text{NET}} \mathbf{W}_f \\ \mathbf{A}_{\text{UNC}} \mathbf{W}_f \end{bmatrix}, \\ \mathbf{b}_y &= \begin{bmatrix} \mathbf{b}_{\text{DEV}} + \mathbf{A}_{\text{DEV}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \mathbf{h} \\ \mathbf{b}_{\text{NET}} + \mathbf{A}_{\text{NET}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \mathbf{h} \\ \mathbf{b}_{\text{UNC}} + \mathbf{A}_{\text{UNC}} \mathbf{W}_e \hat{\mathbf{H}}_x^{-1} \mathbf{h} \end{bmatrix}. \end{aligned}$$

Denote the region constituted by Equation 5.27 as \mathcal{D}_y . When $\vec{\xi} = \mathbf{0}_{n_\xi}$, the \mathcal{D}_y is called a deterministic feeder-level power flexibility region. When $\vec{\xi}$ follows some unknown distribution $p(\vec{\xi})$, the \mathcal{D}_y is called a stochastic feeder-level power flexibility region, and this region is uncertain. In addition, this feeder-level power flexibility region can be treated as an aggregation of non-feeder-level power flexibility regions. As for a polyhedral system model, the feeder-level power flexibility region can be explicitly characterized as another polyhedron via coordinate transformation.

5.3 Inner Approximation Framework

Although the above-constructed feeder-level power flexibility region \mathcal{D}_y has an explicit form, with uncertainties $\vec{\xi}$ and extra variables \mathbf{f} , it is complex and intractable for transmission system operators to procure or use such an exact representation. Instead, its concise and efficient representation is desired. Therefore, this work aims to identify an inner approximated region of \mathcal{D}_y with T time-decoupling two-dimensional Euclidean balls denoted as $\hat{\mathcal{D}}_y$, which can be parameterized as follows

$$\hat{\mathcal{D}}_y := \prod_{t=1}^T B(\mathbf{x}_c^t, r^t) \quad (5.28)$$

where $B(\mathbf{x}_c^t, r^t) := \{\mathbf{y}^t | \mathbf{y}^t = \mathbf{x}_c^t + r^t \cdot \mathbf{u}^t, \|\mathbf{u}^t\|_2 \leq 1\}$ is a ball in the (p_0^t, q_0^t) -plane with the center \mathbf{x}_c^t and radius r^t .

In this section, a data-driven inner approximation framework is developed, as shown in Figure 5.2. In the proposed framework, the inner approximated region $\hat{\mathcal{D}}_y$ is found by solving a deterministic Chebyshev centering optimization problem. To mitigate the adverse impact of the forecast error uncertainty on the approximated results, a chance-constrained Chebyshev centering optimization problem is then formulated. Finally, a data-driven solution is proposed by converting the original inner-approximation problem to a hyperplane learning problem.

5.3.1 Problem Formulation

When the forecast error uncertainty does not exist, i.e., $\vec{\xi} = \mathbf{0}_{n_\xi}$, to aggregate power flexibility at the feeder as much as possible, the approximated region (i.e., multiple

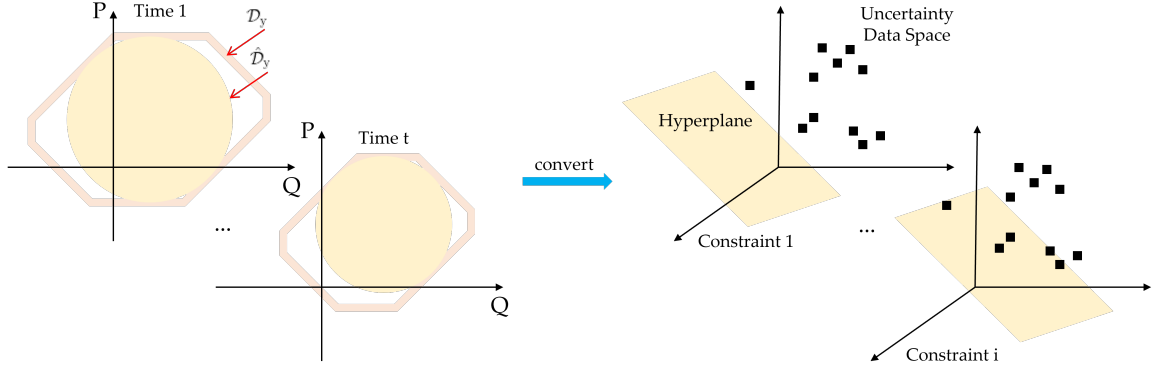


Figure 5.2: Illustration of the proposed inner approximation framework.

maximum inscribed balls) is pursued here by solving the following optimization problem

$$\begin{aligned} \max_{\mathbf{x}_c, \mathbf{r} \geq 0, \mathbf{f}} \quad & \|\mathbf{r}\|_1 \quad \text{s.t.} \\ & \mathbf{A}_y \mathbf{x}_c + \|\mathbf{A}_y\|_* \mathbf{r} + \mathbf{A}_f \mathbf{f} - \mathbf{b}_y \leq 0 \end{aligned} \quad (5.29)$$

where $\mathbf{x}_c := [(\mathbf{x}_c^t)_{t \in \{1, \dots, T\}}]^\top$, $\mathbf{r} := [(r^t)_{t \in \{1, \dots, T\}}]^\top$, $\|\cdot\|_1$ is the L1 norm, $\|\cdot\|_*$ is defined as an operator to find the Euclidean norm of every two entries in a row corresponding to the same r^t .

The problem Equation 5.29 aims at maximizing the sum of volume of all balls while keeping any points of $\hat{\mathcal{D}}_y$ on the boundary within \mathcal{D}_y , which can be considered as a variant of the Chebyshev centering optimization problem [91] for multiple Chebyshev centers (i.e., ball centers in our case). Note that it is a linear programming problem, and thus can be solved by off-the-shelf solvers. After solving this problem, multiple ball-shape regions can be utilized to evaluate the feasible power flexibility supply ability of DERs. Moreover, the whole distribution system can be treated as a special inverter with time-variant capacity curves through this approximation.

When the forecast error uncertainty exists, i.e., $\vec{\xi} \neq \mathbf{0}_{n_\xi}$, the approximated region found by solving the problem Equation 5.29 may encounter the safety issue, i.e., some infeasible power flexibility points may appear. To mitigate this issue, the problem Equation 5.29 is

reformulated into a chance-constrained Chebyshev centering optimization problem

$$\begin{aligned} & \max_{\mathbf{x}_c, \mathbf{r} \geq 0, \mathbf{f}} \|\mathbf{r}\|_1 \quad \text{s.t.} \\ & \Pr\{\mathbf{A}_{y,i}\mathbf{x}_c + \|\mathbf{A}_{y,i}\|_*\mathbf{r} + \mathbf{A}_{f,i}\mathbf{f} + \mathbf{A}_{\xi,i}\vec{\xi} - \mathbf{b}_{y,i} \leq 0\} \geq 1 - \epsilon, \quad \forall i \end{aligned} \quad (5.30)$$

where

- Row vectors $\mathbf{A}_{y,i}$, $\mathbf{A}_{f,i}$, and $\mathbf{A}_{\xi,i}$ are the i -th row of matrix \mathbf{A}_y , \mathbf{A}_f , and \mathbf{A}_ξ , respectively
- $\mathbf{b}_{y,i} \in \mathbb{R}$ is the i -th element of vector \mathbf{b}_y , and
- $\epsilon \in (0, 1]$ is the maximum allowable violation probability.

The problem Equation 5.30 is an individual chance-constrained programming problem which ensures individual constraint is met with high probability $1 - \epsilon$. The main challenge in solving this problem is that the probability distribution $p(\vec{\xi})$ is rarely available in practice. Even with the prior knowledge on the probability distribution, calculating the probability involves multivariate integration, which is NP-hard. In this framework, assume that the partial knowledge on the probability distribution, i.e., the uncertainty data, is available. A computationally-tractable and data-driven solution is developed in next subsection.

5.3.2 Data-driven Solution

Relying on some specific probability distribution, such as Gaussian etc., can lead to sub-optimal results. Therefore, instead of using a single type of probability distribution function, this framework uses a family of probability distribution functions defined in an ambiguity set where all of them share the same first-order and second-order moment information, i.e., mean vector $\vec{\mu}$ and covariance matrix Σ

$$\mathcal{D}_\xi := \{p(\vec{\xi}) | \mathbb{E}_{p(\vec{\xi})}[\vec{\xi}] = \vec{\mu}, \mathbb{E}_{p(\vec{\xi})}[\vec{\xi}\vec{\xi}^\top] = \Sigma\} \quad (5.31)$$

where $\vec{\mu}$ and Σ can be estimated from m historical uncertainty realizations $\{\vec{\xi}_i\}_{i=1}^m$ as

$$\begin{aligned}\vec{\mu} &= \frac{1}{m} \sum_{i=1}^m \vec{\xi}_i \\ \Sigma &= \frac{1}{m} \sum_{i=1}^m (\vec{\xi}_i - \vec{\mu})(\vec{\xi}_i - \vec{\mu})^\top\end{aligned}\tag{5.32}$$

Then the problem Equation 5.30 can be further reformulated as

$$\begin{aligned}\max_{\mathbf{x}_c, \mathbf{r} \geq 0, \mathbf{f}} \quad & \|\mathbf{r}\|_1 \quad \text{s.t.} \\ \inf_{p(\vec{\xi}) \in \mathcal{D}_\xi} \quad & \mathbf{Pr}\{\mathbf{A}_{y,i}\mathbf{x}_c + \|\mathbf{A}_{y,i}\|_*\mathbf{r} + \mathbf{A}_{f,i}\mathbf{f} + \mathbf{A}_{\xi,i}\vec{\xi} - \mathbf{b}_{y,i} \leq 0\} \geq 1 - \epsilon, \quad \forall i\end{aligned}\tag{5.33}$$

The problem Equation 5.33 enforces the satisfaction of the chance constraints for all the probability distribution functions in the ambiguity set.

Take probability constraint i as an example, and it can be viewed as a subspace of \mathbb{R}^{n_ξ} separated by a hyperplane with slope $\mathbf{A}_{\xi,i}$ and intercept b_i in the uncertainty data space

$$\mathbf{A}_{\xi,i}\vec{\xi} = b_i\tag{5.34}$$

where $b_i = \mathbf{b}_{y,i} - \mathbf{A}_{y,i}\mathbf{x}_c - \|\mathbf{A}_{y,i}\|_*\mathbf{r} - \mathbf{A}_{f,i}\mathbf{f}$. As for hyperplane Equation 5.34, the slope $\mathbf{A}_{\xi,i}$ is known, and the intercept b_i is the only parameter pending to be solved.

Recall the result of Bertsimas and Sethuraman [92],

$$\sup \mathbf{Pr}\{\mathbf{A}_{\xi,i}\vec{\xi} \geq b_i\} = \frac{1}{1 + d^2}\tag{5.35}$$

where

$$d^2 = \inf_{\mathbf{A}_{\xi,i}\vec{\xi} \geq b_i} (\vec{\xi} - \mu)^\top \Sigma^{-1} (\vec{\xi} - \mu).$$

The d^2 can also be written as

$$d^2 = \inf_{\mathbf{c}^\top \mathbf{w} \geq f} \mathbf{w}^\top \mathbf{w} \quad (5.36)$$

where $\mathbf{w} = \Sigma^{-\frac{1}{2}}(\vec{\xi} - \mu)$, $\mathbf{c}^\top = \mathbf{A}_{\xi,i}\Sigma^{\frac{1}{2}}$, and $f = b_i - \mathbf{A}_{\xi,i}\mu$.

First notice that if $f \leq 0$, then $d^2 = 0$ when $\vec{\xi} = \mu$, which is certainly the optimum because $d^2 \geq 0$ and the infimum is taken. At this time, $\sup \Pr\{\mathbf{A}_{\xi,i}\vec{\xi} \geq b_i\} = 1$, and thus there is no optimization space for the hyperplane $\mathbf{c}^\top \mathbf{w} = f$ to mitigate constraint violation risk. If $f \geq 0$, then d^2 is the square of the minimum distance from the origin to the hyperplane $\mathbf{c}^\top \mathbf{w} = f$, which can be expressed as follows

$$d^2 = \frac{f^2}{\mathbf{c}^\top \mathbf{c}} = \frac{(b_i - \mathbf{A}_{\xi,i}\mu)^2}{\mathbf{A}_{\xi,i}\Sigma\mathbf{A}_{\xi,i}^\top} \quad (5.37)$$

According to Equation 5.35 and Equation 5.37, any probability constraint in problem Equation 5.33 is equivalent to

$$\sup \Pr\{\mathbf{A}_{\xi,i}\vec{\xi} \geq b_i\} = \frac{1}{1 + d^2} = \frac{\mathbf{A}_{\xi,i}\Sigma\mathbf{A}_{\xi,i}^\top}{\mathbf{A}_{\xi,i}\Sigma\mathbf{A}_{\xi,i}^\top + (b_i - \mathbf{A}_{\xi,i}\mu)^2} \leq \epsilon \quad (5.38)$$

The Equation 5.38 can be rewritten as

$$b_i \geq \underbrace{\mathbf{A}_{\xi,i}\vec{\mu}}_{\text{bound}} + \kappa(\epsilon)\sqrt{\mathbf{A}_{\xi,i}\Sigma\mathbf{A}_{\xi,i}^\top} \quad (5.39)$$

where $\kappa(\epsilon) = \sqrt{\frac{1-\epsilon}{\epsilon}}$ is a coefficient to control the robustness against the forecast error uncertainty. As the size of uncertainty dataset m is sufficiently large, according to central limit theorem, the sample mean vector and covariance matrix are close to the real mean vector and covariance matrix, and the bound is valid with relatively small estimation errors. According to Equation 5.39, multiple hyperplanes can be simultaneously learned

by solving the following deterministic linear programming problem

$$\begin{aligned} & \max_{\mathbf{x}_c, \mathbf{r} \geq 0, \mathbf{f}} \|\mathbf{r}\|_1 \quad \text{s.t.} \\ & \underbrace{\mathbf{A}_{y,i}\mathbf{x}_c + \|\mathbf{A}_{y,i}\|_*\mathbf{r} + \mathbf{A}_{f,i}\mathbf{f} - \mathbf{b}_{y,i}}_{\text{expected constraint terms}} + \underbrace{\mathbf{A}_{\xi,i}\mu + \kappa(\epsilon)\sqrt{\mathbf{A}_{\xi,i}\Sigma\mathbf{A}_{\xi,i}^\top}}_{\text{error compensation terms}} \leq 0, \quad \forall i \end{aligned} \quad (5.40)$$

From the problem Equation 5.40, it can be observed that solving the original inner approximation problem is to learn multiple hyperplanes (i.e., finding multiple intercepts) in an unsupervised way, which can classify the majority of uncertainty data as the “constraint feasible” label while regularizing partial hyperplane parameters. In addition, compared with the problem Equation 5.29, the problem Equation 5.40 has extra forecast error compensation terms to hedge against the risk of constraint violation due to the inherent uncertainty in the expected constraint terms.

5.4 Numerical Simulation

In this section, the proposed inner approximation framework are numerically verified in an artificially-designed system for illustrative purposes. Similar to paper [93], a balanced network with a single three-phase PQ bus (with index 1) connected to the slack bus via a transmission line is considered. The line admittance matrix is given as follows, in p.u.:

$$\begin{bmatrix} 7 - 12j & -1 + 2j & -1 + 2j \\ -1 + 2j & 7 - 12j & -1 + 2j \\ -1 + 2j & -1 + 2j & 7 - 12j \end{bmatrix}. \quad (5.41)$$

In this example, there are three dispatchable DERs: PV, ESS and DCL, and an uncontrollable load installed at the bus 1, as shown in Figure 5.3. For simplicity, they are all three-phase and purely wye-connected, except one single-phase DCL. Detailed device model parameters can be found in Table 5.1.

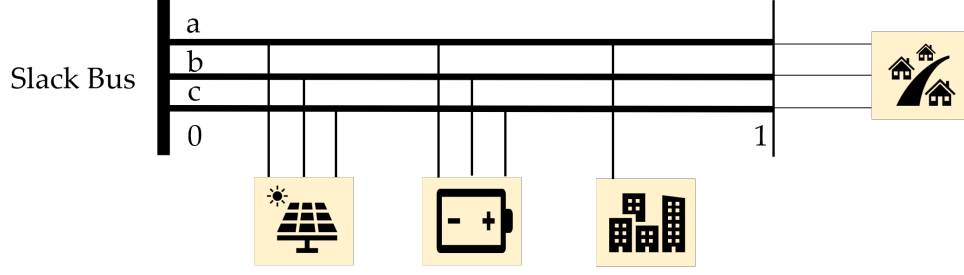


Figure 5.3: Illustration of the artificially-designed system.

Table 5.1: Parameters of the device model

Parameters	Value	Parameters	Value
$\theta_{1,\text{PV}}/\theta_{1,\text{DCL}}/\theta_{1,\text{L}}$	0.3176 rad	$\underline{e}_{1,\text{ESS}}$	0.1
$\bar{p}_{1,\text{ESS}}^\phi$	0.5 p.u.	$\bar{e}_{1,\text{ESS}}$	0.9
$s_{1,\text{ESS}}^\phi$	0.6 p.u.	$\sigma_{1,\text{ESS}}$	3
$\bar{E}_{1,\text{ESS}}$	10 p.u.	$\underline{p}_{1,\text{DCL}}^a$	0.2 p.u.
$e_{1,\text{ESS}}^0$	0.5	$\bar{p}_{1,\text{DCL}}^a$	0.6 p.u.

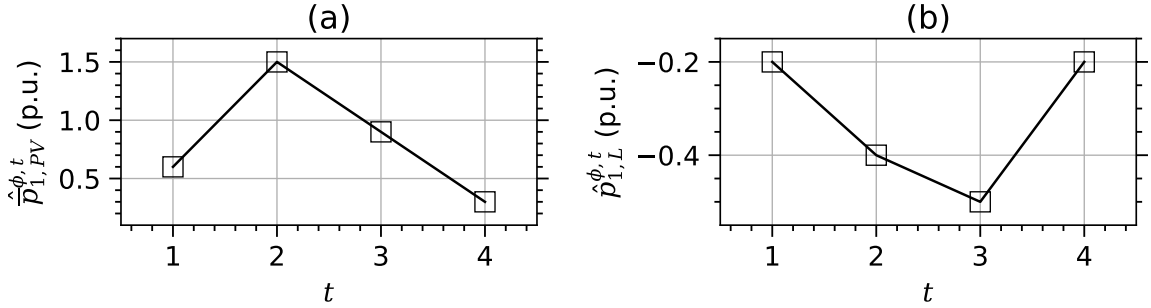


Figure 5.4: Illustration of PV and uncontrollable load data.

Moreover, the length of the aggregation time horizon is set as 4 with the granularity of 2 hours. The lower and upper limits of voltage magnitudes are set as 0.95 and 1.05, respectively. The expected available maximum active power of PV inverter and the expected active power injection of uncontrollable load are shown in Figure 5.4.

The proposed inner approximation framework is implemented in Python with CVXPY solvers for the problem Equation 5.29, Equation 5.40 and Equation 5.44.

In order to evaluate the approximated region, four types of metrics are defined. The first metric is the negative value of the optimization objective function $\|\mathbf{r}\|_1$, which reflects the optimality of the solution for the entire aggregation period. The second metric is the

radius r^t , which reflects the optimality of the solution at time t . The third metric is the active-power / reactive-power / capacity flexibility contribution rate $\gamma_p^t / \gamma_q^t / \gamma_s^t \in [0, 1]$, which reflects the level of the contribution of flexibility at time t in the entire aggregation period, and is respectively defined as

$$\begin{aligned}\gamma_p^t &= \frac{S_{p,+}^t}{\sum_{t=1}^T S_{p,+}^t}, \quad \forall t \\ \gamma_q^t &= \frac{S_{q,+}^t}{\sum_{t=1}^T S_{q,+}^t}, \quad \forall t \\ \gamma_s^t &= \frac{S_{s,+}^t}{\sum_{t=1}^T S_{s,+}^t}, \quad \forall t\end{aligned}\tag{5.42}$$

where

- $S_{p,+}^t$ is the area of the active-reactive flexibility region at time t in the left half plane of the y-axis,
- $S_{q,+}^t$ is the area of the active-reactive flexibility region at time t in the lower half plane of the x-axis, and
- $S_{s,+}^t$ is the area of the active-reactive flexibility region at time t .

The fourth metric is the violation risk level $\hat{\epsilon}_t \in [0, 1]$, which reflects the level of the robustness of the flexibility region to the forecast error uncertainty at time t , and is defined as

$$\hat{\epsilon}_t = 1 - \frac{\sum_{i=1}^m 1_{\{\hat{D}_y^t \subset \hat{D}_{\xi_i}^t\}}}{m}\tag{5.43}$$

where

- $\hat{D}_y^t := B(\mathbf{x}_c^t, r^t)$ is the approximated flexibility region at time t solved by the problem Equation 5.29 or Equation 5.40, and

- $\hat{D}_{\xi_i}^t$ is one of possible approximated stochastic flexibility regions at time t solved by the problem formulated by

$$\begin{aligned} & \max_{\mathbf{x}_c, \mathbf{r} \geq 0, \mathbf{f}} \|\mathbf{r}\|_1 \quad \text{s.t.} \\ & \mathbf{A}_{y,i} \mathbf{x}_c + \|\mathbf{A}_{y,i}\|_* \mathbf{r} + \mathbf{A}_{f,i} \mathbf{f} + \mathbf{A}_{\xi,i} \vec{\xi}_i - \mathbf{b}_{y,i} \leq 0 \end{aligned} \quad (5.44)$$

Note that the process of solving the problem Equation 5.44 with different uncertainty realizations is a Monte Carlo method to approximate the stochastic power flexibility regions. If the approximated deterministic flexibility region at time t can be included in the approximated stochastic flexibility region at time t , it means that the approximated deterministic flexibility region at time t is robust to the uncertainty realization $\vec{\xi}_i$. The more uncertainty realizations this region is robust to at time t , the lower the corresponding violation risk level will become.

5.4.1 Deterministic Region Approximation

After solving the problem Equation 5.29, the approximated region is shown in Figure 5.5. At each time t , there is a ball-shaped region where any point (i.e., small white circle) can be fulfilled by appropriately dispatching controllable DERs. Specifically,

- At $t = 1$, the minimum load demand is smaller than the maximum generation supply, and thus the DERs can provide capacity support to the transmission system (i.e., the region intersects the red rectangle region);
- At $t = 2$, $\hat{p}_{1,\text{PV}}^{\phi,t}$ encounters a large increase in its value, and thus the maximum generation supply increases and the DERs can provide more capacity support (i.e., the radius is increased and the ball center moves to the left);
- At $t = 3$, $\hat{p}_{1,\text{L}}^{\phi,t}$ encounters a large increase in its value, and thus the minimum load demand increases and the ability of providing the flexibility is undermined (i.e., the radius is decreased and the ball center moves to the right);

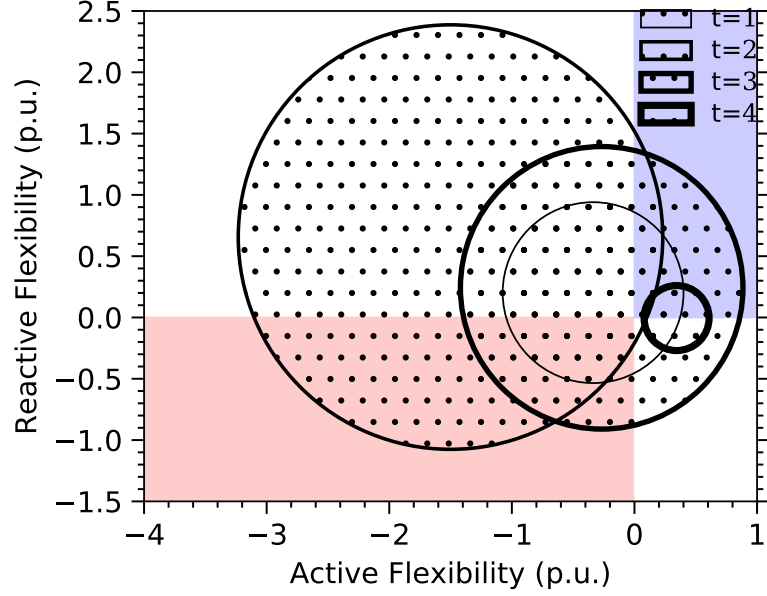


Figure 5.5: Illustration of the approximated deterministic region. The points in the red rectangle region can provide the capacity support to the transmission system while the ones in the blue rectangle region require the capacity support from the transmission system.

- At $t = 4$, $\hat{p}_{1,PV}^{\phi,t}$ encounters a large decrease in its value, and thus the maximum generation supply decreases and it is less than the minimum load demand and the DERs can provide less capacity support (i.e., the radius is further decreased and the ball center moves to the right and the region only intersects the blue rectangle region).

As shown in Figure 5.6, the contribution of the active-power / reactive-power / capacity flexibility at $t = 2$ is the highest while the contribution of the active-power / reactive-power / capacity flexibility at $t = 4$ is the lowest. In addition, it can be observed that compared with γ_p^t and γ_s^t , γ_q^t has less value variance in the entire aggregation period because the designed system has certain power factor limit.

The values of PV data are proportionally changed by multiplying a coefficient called PV penetration level. As shown in Figure 5.7, as the PV penetration level increases (i.e., more and more PVs are integrated to the system), the objective function value gradually increases until saturation because network constraints (i.e., voltage constraints) limit the capacity of flexibility support. As shown in Figure 5.8, as the PV penetration level increases, the

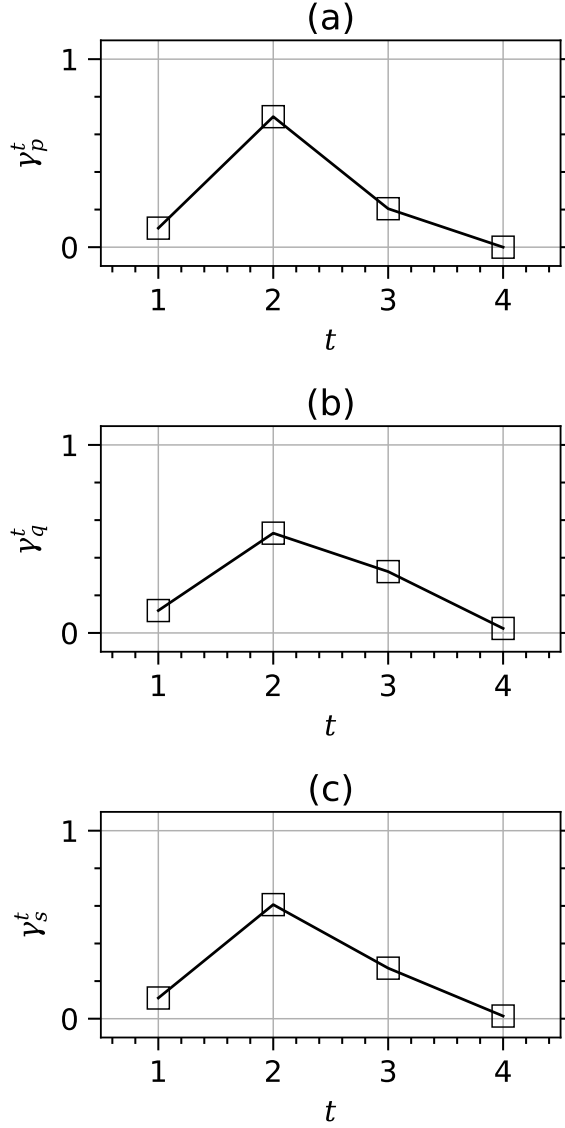


Figure 5.6: Illustration of the flexibility metrics: γ_p^t , γ_q^t and γ_s^t in the entire aggregation period.

contribution variance at different time steps is reduced for γ_p^t because at each time step there is sufficient flexibility support. The γ_q^t and γ_s^t have the similar contribution mode with the highest value at $t = 2$ and the lowest value at $t = 4$.

Given an uncertainty realization, the actual approximated region can be found by solving problem Equation 5.44. As shown in Figure 5.9, some flexibility points in the expected approximated region are outside the actual approximated region. So there exists high constraint violation risks when the forecast error uncertainty is not considered in the

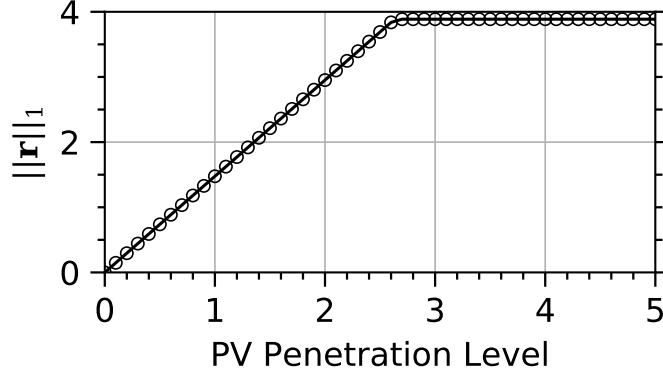


Figure 5.7: Illustration of the effect of PV penetration level on the optimization results.

problem Equation 5.29.

5.4.2 Deterministic Region Approximation Under Uncertainty

This subsection verifies how effective the proposed inner approximation framework is to mitigate the risk of constraint violation caused by forecast error uncertainty. The 1000 uncertainty data points is randomly sampled from a Mixture Gaussian distribution, i.e., $\omega_1 \mathcal{N}(\vec{\mu}_1, \Sigma_1) + \omega_2 \mathcal{N}(\vec{\mu}_2, \Sigma_2)$ where

$$\omega_1 = 0.6$$

$$\omega_2 = 0.4$$

$$\vec{\mu}_1 = [0.005, 0.005, 0.005, 0.005, 0.006, 0.006, 0.006, 0.006]^\top$$

$$\vec{\mu}_2 = [-0.003, -0.003, -0.003, -0.003, -0.005, -0.005, -0.005, -0.005]^\top$$

$$\Sigma_1 = \mathbf{diag}([0.0001, 0.0001, 0.0001, 0.0001, 0.0001, 0.0001, 0.0001, 0.0001])$$

$$\Sigma_2 = \mathbf{diag}([0.0002, 0.0002, 0.0002, 0.0002, 0.0002, 0.0002, 0.0002, 0.0002])$$

By solving the problem Equation 5.40 with $\epsilon = 0.2$, a robust deterministic approximated region can be found. As shown in Figure 5.10, the grey, black and red regions are the stochastic, deterministic and robust deterministic approximated regions, respectively. Compared with the black region, at each time moment, the red region is

contained in the black region while it covers less points from the grey region.

Therefore, the proposed inner approximation framework is valid to mitigate the adverse impact of the forecast error uncertainty. It can be seen in Figure 5.11 that as the value of ϵ decreases, the optimized r^t (i.e., black squares) will gradually decrease, and the solution become more conservative (i.e., below the black dash line corresponding to the problem Equation 5.29) while the robustness metric \hat{e}_t will increase first and then decrease, and the robustness to forecast error uncertainty is improved in the solution of the problem Equation 5.40 (i.e., above the red dash line corresponding to the problem Equation 5.29).

5.5 Conclusion

This chapter proposes an inner approximation framework for flexibility aggregation under uncertainty based on a chance-constrained Chebyshev centering optimization problem. The formulated problem can be solved by simultaneously and unsupervisedly learning multiple hyperplanes in a data-driven manner. The proposed framework can effectively reduce the risk of constraint violation due to forecast error uncertainty.

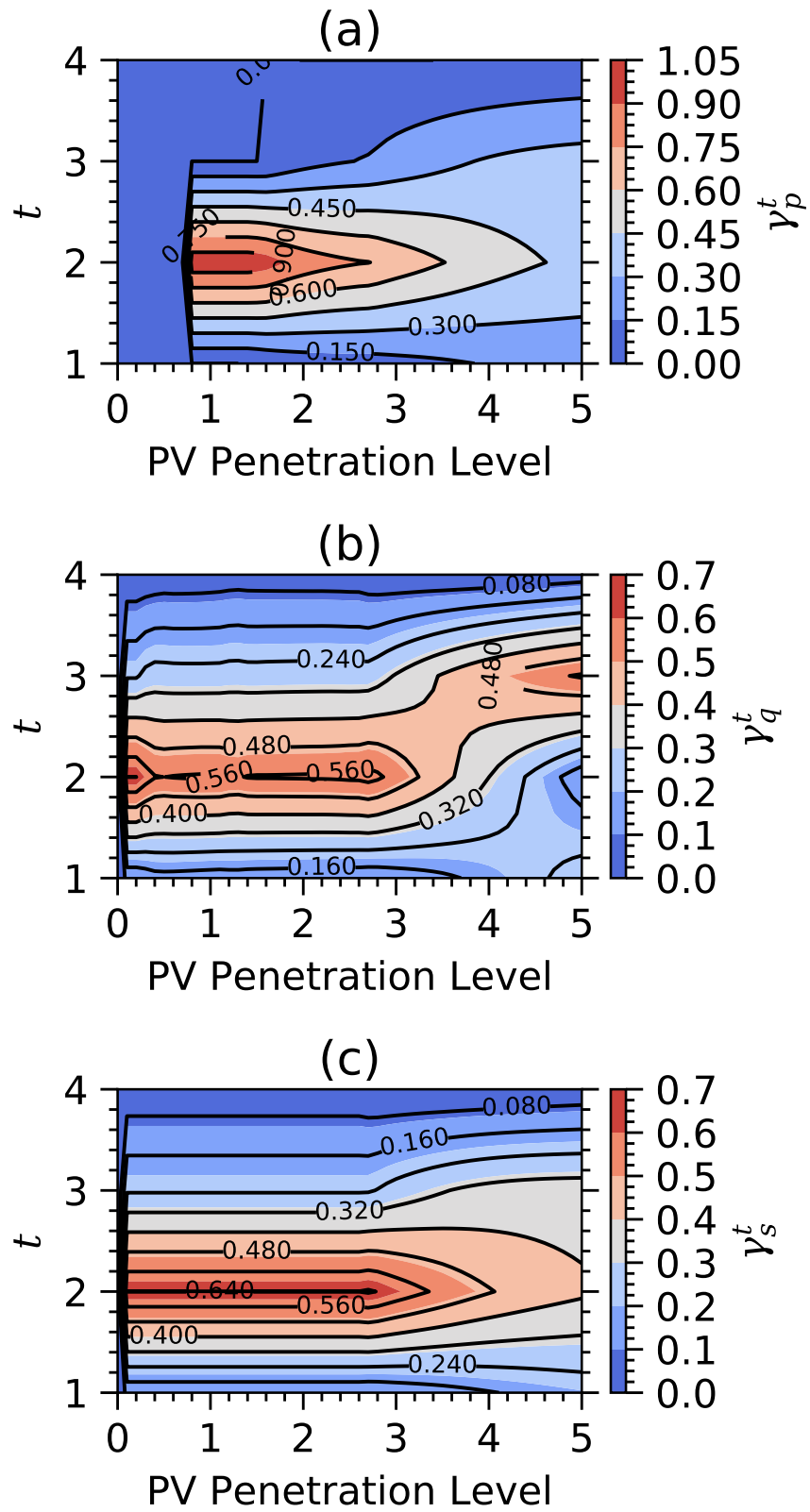


Figure 5.8: Illustration of the effect of PV penetration level on the approximated region.

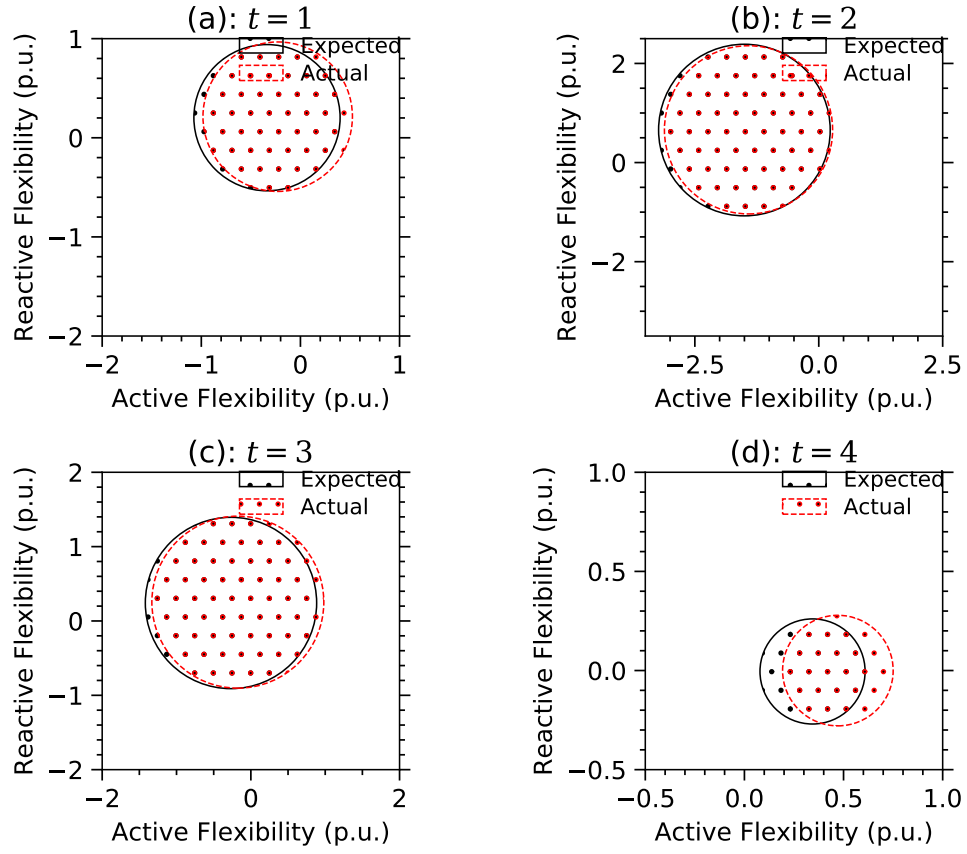


Figure 5.9: Illustration of the approximated deterministic region without considering forecast error uncertainty.

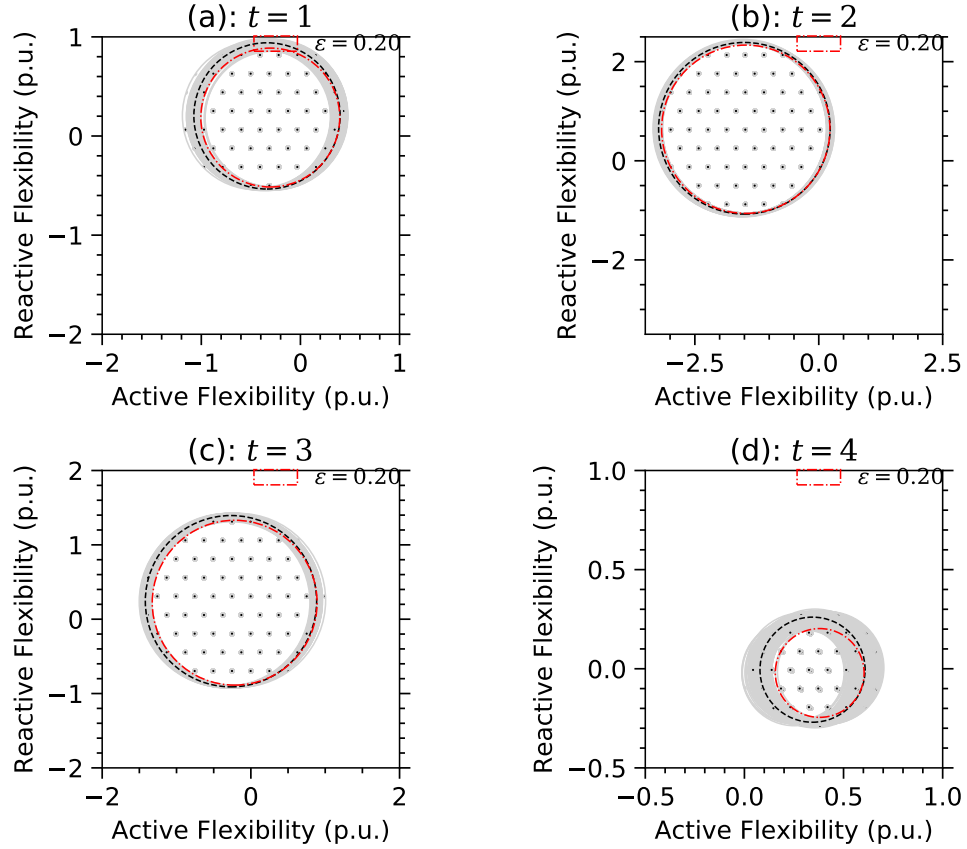


Figure 5.10: Illustration of the robust deterministic approximated region.

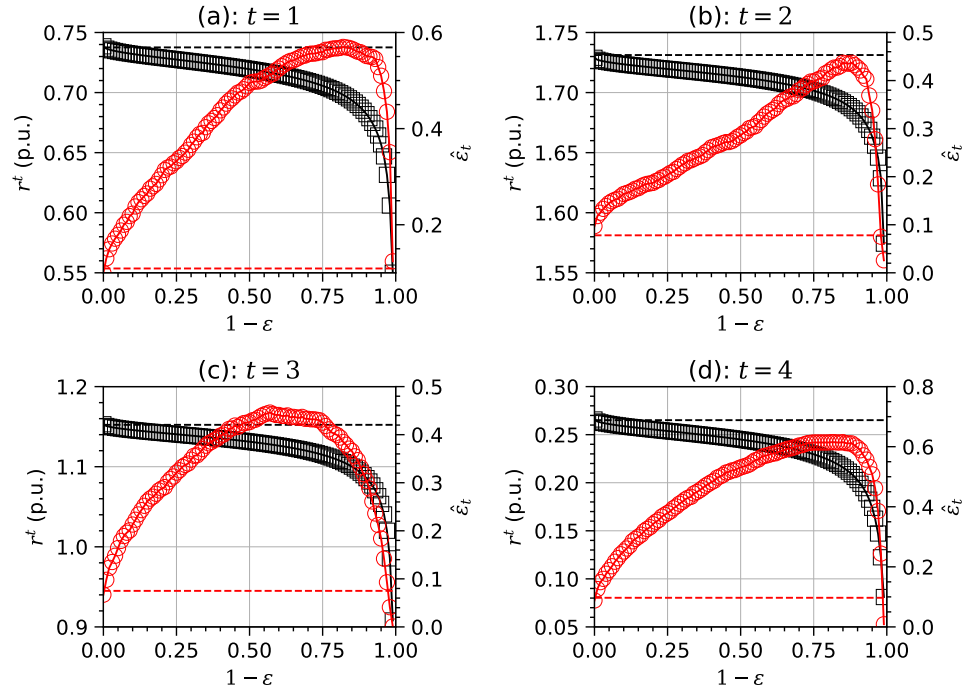


Figure 5.11: Illustration of the effect of ε on the optimality and robustness.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

This dissertation investigates three representative operation problems for power distribution systems with high penetration of DERs, where the conventional operation methods are experiencing unprecedented challenges due to a high level of uncertainties. On one hand, it is extremely difficult to model uncertainties. On the other hand, it is almost intractable to make an optimal decision making under uncertainties. With the tremendous amount of available uncertainty data and the vast success of machine learning in the data-intensive industry, machine learning is resorted to develop novel data-driven operation methods so as to handle uncertainties in the operation problems. The conclusions and main contributions are summarized as follows:

First, a residential EV charging load profile identification problem is studied. The residential EV charging load is an important uncertainty source in the demand side. Different charging habits (i.e., charging start time, charging end time, charging rate etc.) of customers lead to huge difference among EV charging load profiles. Meanwhile, this uncertainty source is behind the meter, and the only information relative to EV charging load demand is the aggregated power consumption, which exacerbates the difficulty of the problem. The conventional methods such as Hidden Markov Model is weak to model residential EV charging load profile given the aggregated power consumption profile with the low identification accuracy. This dissertation reformulates this problem as a time-series classification problem, and a Deep Generative Model: a representation layer embedded Markov chain is proposed. The proposed model keeps the Markov property of Hidden Markov Model. It can learn the joint probability distribution of EV charging load profile

from the uncertainty data. Thanks to the powerful representation ability of deep neural networks to capture abstract but useful features, the proposed model achieves improved identification accuracy compared with Hidden Markov Model, which can help to enhance the situational awareness.

Second, a real-time voltage regulation problem in a PV penetrated distribution system is studied. The PV as an important uncertainty source in the generation side is exacerbating the voltage quality issue. Specifically, the fluctuation of PV generation degrades the voltage stability and causes the voltage deviation as well as violation. The existing deterministic voltage regulation methods are extremely difficult to quickly respond to uncertainty and make an optimal decision to mitigate its adverse impact. This dissertation reformulates this problem as a Markov Decision Process problem, and a Deep Reinforcement Learning based method is proposed to solve it. The proposed method can learn a near-optimal strategy from the uncertainty data so as to autonomously schedule Energy Storage Systems to mitigate a long-term voltage deviation. In addition, there is no need to have the uncertainty model for decision making, which largely boosts the tractability of the problem. Compared with a deterministic quadratic programming based strategy, the proposed method can reach a near-optimal optimization level and respond to uncertainty in real-time, which can help to support the decision making under uncertainty.

Finally, an ahead-of-time deterministic active-reactive power flexibility region estimation problem with forecast error uncertainty is studied. The forecast error is an important uncertainty source in the control side. The inaccurate forecast model brings large forecast errors, threatening the reliability and security of time-ahead operation. That is, the estimated region contains some infeasibility operation points, and increases the risk of operation constraint violation and failure of flexibility aggregation. This dissertation explicitly constructs a stochastic flexibility region using a coordinate transformation. An inner approximation framework is proposed based on a chance-constrained Chebyshev centering optimization problem, which can be solved by learning multiple hyperplanes

from the uncertainty data in an unsupervised way so as to classify most of uncertainty realizations as the constraint feasibility label. Compared with a deterministic Chebyshev centering optimization solution, the proposed method has better robustness to forecast error uncertainty at the expense of losing a partial of optimality, which can help to strengthen the flexibility aggregation under uncertainty.

6.2 Future Work

There are some other interesting and relevant topics that could be considered for further investigations.

Residential rooftop PV has been widely integrated into distribution systems. In most cases, PV generations are also installed behind-the-meter (BTM), and only the net demand is measured. So the BTM PV generation is unknown to utilities. Identifying solar generation from net demand is critical for improving grid-edge visibility. Therefore, the future investigation could be done on how to extend the proposed Deep Generative Model in Chapter Three to the behind-the-meter PV identification problem. Meanwhile, the proposed model can be modified to use the widely available high-resolution data for training, which may further improve the classification accuracy and enhance the utilities' situational awareness of grid-edge resource.

For the proposed method in Chapter Four, one DRL agent is trained to control all energy storage systems. In order to improve the scalability, the future work could be conducted on using multi-agent DRL techniques to train multiple agents for voltage regulation. Meanwhile, it is interesting to using DRL to coordinate ESSs with other types of voltage regulation devices such as voltage regulators and capacitors. For the proposed method in Chapter Five, the impact of forecast error uncertainty can be mitigated from the estimation step. In future work, the inner approximation framework can be further modified, and allow some controllable DERs to fully erase the impact of forecast error uncertainty.

REFERENCES

- [1] U.S. Energy Information Admin., *Annual Energy Outlook*, 2021.
- [2] ———, *Annual Energy Outlook*, 2019.
- [3] D. Gohlke and Y. Zhou, “Impacts of electrification of light-duty vehicles in the united states, 2010-2017,” Argonne National Lab, Argonne, IL, Tech. Rep., 2018.
- [4] P. G. Pereirinha, M. Gonzalez, I. Carrilero, D. Ansean, J. Alonso, and J. C. Viera, “Main trends and challenges in road transportation electrification,” *Transport. Res. Procedia*, vol. 33, pp. 235–242, 2018.
- [5] A. Dubey and S. Santoso, “Electric vehicle charging on residential distribution systems: Impacts and mitigations,” *IEEE Access*, vol. 3, pp. 1871–1893, 2015.
- [6] S. Huang and Q. Wu, “Dynamic tariff-subsidy method for pv and v2g congestion management in distribution networks,” *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 5851–5860, Sep. 2019.
- [7] B. Zhang and M. Kezunovic, “Impact on power system flexibility by electric vehicle participation in ramp market,” *IEEE Trans. Smart Grid*, vol. 7, no. 3, pp. 1285–1294, May 2016.
- [8] Z. Liu, Q. Wu, K. Ma, M. Shahidehpour, Y. Xue, and S. Huang, “Two-stage optimal scheduling of electric vehicle charging based on transactive control,” *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2948–2958, 2019.
- [9] K. Qian, C. Zhou, M. Allan, and Y. Yuan, “Modeling of load demand due to ev battery charging in distribution systems,” *IEEE Trans. Power Syst.*, vol. 26, no. 2, pp. 802–810, May 2011.
- [10] D. Tang and P. Wang, “Nodal impact assessment and alleviation of moving electric vehicle loads: From traffic flow to power flow,” *IEEE Trans. Power Sys.*, vol. 31, no. 6, pp. 4231–4242, 2016.
- [11] Y. Cao *et al.*, “An optimized ev charging model considering tou price and soc curve,” *IEEE Trans. Smart Grid*, vol. 3, no. 1, pp. 388–393, 2011.
- [12] S. Shao, M. Pipattanasomporn, and S. Rahman, “Grid integration of electric vehicles and demand response with customer choice,” *IEEE Trans. Smart Grid*, vol. 3, no. 1, pp. 543–550, 2012.

- [13] J. Zheng, X. Wang, K. Men, C. Zhu, and S. Zhu, "Aggregation model-based optimization for electric vehicle charging strategy," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 1058–1066, Jun. 2013.
- [14] A. Y. Lam, K.-C. Leung, and V. O. Li, "Capacity estimation for vehicle-to-grid frequency regulation services with smart charging mechanism," *IEEE Trans. Smart Grid*, vol. 7, no. 1, pp. 156–166, 2015.
- [15] M. G. Flammini, G. Pretico, A. Julea, G. Fulli, A. Mazza, and G. Chicco, "Statistical characterisation of the real transaction data gathered from electric vehicle charging stations," *Electric Power Systems Research*, vol. 166, pp. 136–150, 2019.
- [16] E. Xydias, C. Marmaras, L. M. Cipcigan, N. Jenkins, S. Carroll, and M. Barker, "A data-driven approach for characterising the charging demand of electric vehicles: A UK case study," *Applied energy*, vol. 162, pp. 763–771, 2016.
- [17] Y. B. Khoo, C.-H. Wang, P. Paevere, and A. Higgins, "Statistical modeling of electric vehicle electricity consumption in the Victorian EV trial, Australia," *Transport. Res. Part D*, vol. 32, pp. 263–277, 2014.
- [18] E. C. Kara, J. S. Macdonald, D. Black, M. Berges, G. Hug, and S. Kiliccote, "Estimating the benefits of electric vehicle smart charging at non-residential locations: A data-driven approach," *Applied Energy*, vol. 155, pp. 515–525, 2015.
- [19] S. Wang, L. Du, J. Ye, and D. Zhao, "Robust identification of EV charging profiles," in *2018 IEEE Transportation Electrification Conference and Expo (ITEC)*, IEEE, 2018, pp. 1–6.
- [20] Z. Zhang *et al.*, "Training-free non-intrusive load monitoring of electric vehicle charging with low sampling rate," in *40th Annual Conference of the IEEE Industrial Electronics Society*, Oct. 2014, pp. 5419–5425.
- [21] A. A. Munshi and Y. A.-R. I. Mohamed, "Unsupervised nonintrusive extraction of electrical vehicle charging load patterns," *IEEE Trans Ind. Informat.*, vol. 15, no. 1, pp. 266–279, 2019.
- [22] Q. Dang, Y. Huo, and C. Sun, "Privacy preservation needed for smart meter system: A methodology to recognize electric vehicle (EV) models," in *2018 IEEE ISGT-ASIA*, IEEE, 2018, pp. 1016–1020.
- [23] H. Zhao, X. Yan, and L. Ma, "Training-free non-intrusive load extracting of residential electric vehicle charging loads," *IEEE Access*, vol. 7, pp. 117044–117053, 2019.

- [24] O. Parson, S. Ghosh, M. Weal, and A. Rogers, “Non-intrusive load monitoring using prior models of general appliance types,” in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.
- [25] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han, “Unsupervised disaggregation of low frequency power measurements,” in *Proceedings of the 2011 SIAM International Conference on Data Mining*, 2011, pp. 747–758.
- [26] J. Z. Kolter and T. Jaakkola, “Approximate inference in additive factorial hmms with application to energy disaggregation,” in *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, N. D. Lawrence and M. Girolami, Eds., ser. Proceedings of Machine Learning Research, vol. 22, La Palma, Canary Islands: PMLR, Apr. 2012, pp. 1472–1482.
- [27] B. Wang, H. Dong, A. P. Boedihardjo, F. Chen, and C.-T. Lu, “A hierarchical probabilistic model for low sample rate home-use energy disaggregation,” in *Proceedings of the 2013 SIAM International Conference on Data Mining*, 2013, pp. 704–712.
- [28] Y. Li, Z. Peng, J. Huang, Z. Zhang, and J. H. Son, “Energy disaggregation via hierarchical factorial hmm,” in *Proceedings of the 2nd International Workshop on Non-Intrusive Load Monitoring (NILM)*. *Google Scholar*, 2014.
- [29] H. Sun *et al.*, “Review of challenges and research opportunities for voltage control in smart grids,” *IEEE Trans. Power Sys.*, vol. 34, no. 4, pp. 2790–2801, Jul. 2019.
- [30] S. Karagiannopoulos, C. Mylonas, P. Aristidou, and G. Hug, “Active distribution grids providing voltage support: The swiss case,” *IEEE Trans. Smart Grid*, vol. 12, no. 1, pp. 268–278, 2021.
- [31] S. Wang, L. Du, and Y. Li, “Decentralized Volt/Var control of EV charging station inverters for voltage regulation,” in *2020 IEEE Transportation Electrification Conference & Expo (ITEC)*, 2020, pp. 604–608.
- [32] Y. Wang, M. H. Syed, E. Guillo-Sansano, Y. Xu, and G. M. Burt, “Inverter-based voltage control of distribution networks: A three-level coordinated method and power hardware-in-the-loop validation,” *IEEE Trans. Sustain. Energy*, vol. 11, no. 4, pp. 2380–2391, 2019.
- [33] S. Wang, L. Du, J. Ye, and D. Zhao, “A deep generative model for non-intrusive identification of ev charging profiles,” *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 4916–4927, 2020.

- [34] L. Wang, F. Bai, R. Yan, and T. K. Saha, “Real-time coordinated voltage control of pv inverters and energy storage for weak networks with high pv penetration,” *IEEE Trans. Power Sys.*, vol. 33, no. 3, pp. 3383–3395, 2018.
- [35] F. Bai, R. Yan, T. K. Saha, and D. Eghbal, “An excessive tap operation evaluation approach for unbalanced distribution networks with high pv penetration,” *IEEE Trans. Sustain. Energy*, vol. 12, no. 1, pp. 169–178, 2021.
- [36] D. Cao *et al.*, “A multi-agent deep reinforcement learning based voltage regulation using coordinated pv inverters,” *IEEE Trans. Power Sys.*, vol. 35, no. 5, pp. 4120–4123, 2020.
- [37] S. Ghosh, S. Rahman, and M. Pipattanasomporn, “Distribution voltage regulation through active power curtailment with pv inverters and solar generation forecasts,” *IEEE Trans. Sustain. Energy*, vol. 8, no. 1, pp. 13–22, 2017.
- [38] M. Zeraati, M. E. H. Golshan, and J. M. Guerrero, “Distributed control of battery energy storage systems for voltage regulation in distribution networks with high pv penetration,” *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3582–3593, 2018.
- [39] J. Krata and T. K. Saha, “Real-time coordinated voltage support with battery energy storage in a distribution grid equipped with medium-scale pv generation,” *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 3486–3497, 2018.
- [40] Y. Guo *et al.*, “Mpc-based coordinated voltage regulation for distribution networks with distributed generation and energy storage system,” *IEEE Trans. Sustain. Energy*, vol. 10, no. 4, pp. 1731–1739, 2018.
- [41] P. Yu, C. Wan, Y. Song, and Y. Jiang, “Distributed control of multi-energy storage systems for voltage regulation in distribution networks: A back-and-forth communication framework,” *IEEE Trans. Smart Grid*, 2020.
- [42] B. Park and M. M. Olama, “A model-free voltage control approach to mitigate motor stalling and fidvr for smart grids,” *IEEE Trans. Smart Grid*, vol. 12, no. 1, pp. 67–78, 2021.
- [43] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
- [44] H. Xu, A. D. Dominguez-Garcia, and P. W. Sauer, “Optimal tap setting of voltage regulation transformers using batch reinforcement learning,” *IEEE Trans. Power Sys.*, vol. 35, no. 3, pp. 1990–2001, 2020.

- [45] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Trans. Power Deli.*, vol. 4, no. 2, pp. 1401–1407, 1989.
- [46] —, "Optimal capacitor placement on radial distribution systems," *IEEE Trans. Power Deli.*, vol. 4, no. 1, pp. 725–734, 1989.
- [47] Z. Zhang, D. Zhang, and R. C. Qiu, "Deep reinforcement learning for power system applications: An overview," *CSEE J. Power & Energy Sys.*, vol. 6, no. 1, pp. 213–225, 2019.
- [48] Q. Huang, R. Huang, W. Hao, J. Tan, R. Fan, and Z. Huang, "Adaptive power system emergency control using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1171–1182, 2020.
- [49] W. Wang, N. Yu, Y. Gao, and J. Shi, "Safe off-policy deep reinforcement learning algorithm for Volt-VAR control in power distribution systems," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3008–3018, 2020.
- [50] Y. Zhang, X. Wang, J. Wang, and Y. Zhang, "Deep reinforcement learning based Volt-Var optimization in smart distribution systems," *IEEE Trans. Smart Grid*, vol. 12, no. 1, pp. 361–371, 2021.
- [51] Q. Yang, G. Wang, A. Sadeghi, G. B. Giannakis, and J. Sun, "Two-timescale voltage control in distribution grids using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2313–2323, 2020.
- [52] J. Duan *et al.*, "Deep-reinforcement-learning-based autonomous voltage control for power grid operations," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 814–817, 2020.
- [53] H. Xu, A. Domínguez-García, V. Veeravalli, and P. Sauer, "Data-driven voltage regulation in radial power distribution systems," *IEEE Transactions on Power Systems*, vol. 35, no. 3, pp. 2133–2143, 2020.
- [54] L. Wang, R. Yan, F. Bai, T. Saha, and K. Wang, "A distributed inter-phase coordination algorithm for voltage control with unbalanced pv integration in lv systems," *IEEE Trans. Sustain. Energy*, vol. 11, no. 4, pp. 2687–2697, 2020.
- [55] A. Bedawy, N. Yorino, K. Mahmoud, Y. Zoka, and Y. Sasaki, "Optimal voltage control strategy for voltage regulators in active unbalanced distribution systems using multi-agents," *IEEE Trans. Power Sys.*, vol. 35, no. 2, pp. 1023–1035, 2020.
- [56] G. C. Karyonidis *et al.*, "Distributed reactive power control scheme for the voltage regulation of unbalanced lv grids," *IEEE Trans. Sustain. Energy*, pp. 1–1, 2020.

- [57] K. Alzaareer, M. Saad, H. Mehrjerdi, D. Asber, and S. Lefebvre, "Development of new identification method for global group of controls for online coordinated voltage control in active distribution networks," *IEEE Trans. Smart Grid*, vol. 11, no. 5, pp. 3921–3931, 2020.
- [58] Z. Li, Q. Guo, H. Sun, and J. Wang, "Coordinated economic dispatch of coupled transmission and distribution systems using heterogeneous decomposition," *IEEE Trans. Power Syst.*, vol. 31, no. 6, pp. 4817–4830, 2016.
- [59] C. Lin, W. Wu, X. Chen, and W. Zheng, "Decentralized dynamic economic dispatch for integrated transmission and active distribution networks using multi-parametric programming," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4983–4993, 2017.
- [60] C. Lin *et al.*, "Decentralized reactive power optimization method for transmission and distribution networks accommodating large-scale DG integration," *IEEE Trans. Sustain. Energy*, vol. 8, no. 1, pp. 363–373, 2016.
- [61] L. Zhao, W. Zhang, H. Hao, and K. Kalsi, "A geometric approach to aggregate flexibility modeling of thermostatically controlled loads," *IEEE Trans. Power Syst.*, vol. 32, no. 6, pp. 4721–4731, 2017.
- [62] E. Polymeneas and S. Meliopoulos, "Aggregate modeling of distribution systems for multi-period OPF," in *2016 Power Systems Computation Conf. (PSCC)*, IEEE, 2016, pp. 1–8.
- [63] X. Chen, E. Dall’Anese, C. Zhao, and N. Li, "Aggregate power flexibility in unbalanced distribution systems," *IEEE Trans. Smart Grid*, vol. 11, no. 1, pp. 258–269, 2019.
- [64] X. Chen and N. Li, "Leveraging two-stage adaptive robust optimization for power flexibility aggregation," *IEEE Trans. Smart Grid*, 2021.
- [65] B. Cui, A. Zamzam, and A. Bernstein, "Network-cognizant time-coupled aggregate flexibility of distribution systems under uncertainties," *IEEE Control Syst. Lett.*, vol. 5, no. 5, pp. 1723–1728, 2020.
- [66] S. Wang and W. Wu, "Stochastic flexibility evaluation for virtual power plant by aggregating distributed energy resources," *arXiv preprint arXiv:2006.16170*, 2020.
- [67] M. Zhang, X. Zhu, and N. Lu, "A data-driven probabilistic-based flexibility region estimation method for aggregated distributed energy resources," *arXiv preprint arXiv:2110.07406*, 2021.
- [68] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain.," *Psychological review*, vol. 65, no. 6, p. 386, 1958.

- [69] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.
- [70] Z. Zhang, “Improved adam optimizer for deep neural networks,” in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, IEEE, 2018, pp. 1–2.
- [71] V. Mnih *et al.*, “Playing atari with deep reinforcement learning,” *arXiv preprint arXiv:1312.5602*, 2013.
- [72] T. P. Lillicrap *et al.*, “Continuous control with deep reinforcement learning,” *arXiv preprint arXiv:1509.02971*, 2015.
- [73] V. Hoffmann, B. I. Fesche, K. Ingebrigtsen, I. N. Christie, and M. Punnerud Engelstad, “Automated detection of electric vehicles in hourly smart meter data,” *IEEE Access*, 2019.
- [74] Z. Ghahramani, “An introduction to hidden markov models and bayesian networks,” in *Hidden Markov models: applications in computer vision*, World Scientific, 2001, pp. 9–41.
- [75] C. M. Bishop, *Pattern recognition & machine learning*. Springer, 2006.
- [76] D. P. Kingma and M. Welling, “Stochastic gradient vb and the variational auto-encoder,” in *Second International Conference on Learning Representations, ICLR*, vol. 19, 2014.
- [77] D. I. J. Im, S. Ahn, R. Memisevic, and Y. Bengio, “Denoising criterion for variational auto-encoding framework,” in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
- [78] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, “Deep learning for time series classification: A review,” *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, 2019.
- [79] J. Kelly and W. Knottenbelt, “Neural nilm: Deep neural networks applied to energy disaggregation,” in *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, ACM, 2015, pp. 55–64.
- [80] C. Zhang, J. Butepage, H. Kjellstrom, and S. Mandt, “Advances in variational inference,” *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.
- [81] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [82] *Pecan street database*, <http://www.pecanstreet.org>.

- [83] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the 13th international conf. artificial intelligence and statistics*, 2010, pp. 249–256.
- [84] C. Wu, G. Hug, and S. Kar, “Smart inverter for voltage regulation: Physical and market implementation,” *IEEE Trans. Power Sys.*, vol. 33, no. 6, pp. 6181–6192, Nov. 2018.
- [85] T. Haarnoja *et al.*, “Soft actor-critic algorithms and applications,” *arXiv preprint arXiv:1812.05905*, 2018.
- [86] R. C. Dugan, W. H. Kersting, S. Carneiro, R. F. Arritt, and T. E. McDermott, “Roadmap for the iee pes test feeders,” in *2009 IEEE/PES Power Systems Conference and Exposition*, IEEE, 2009, pp. 1–4.
- [87] Q. Yang, G. Wang, A. Sadeghi, G. B. Giannakis, and J. Sun, “Two-timescale voltage control in distribution grids using deep reinforcement learning,” *IEEE Transactions on Smart Grid*, vol. 11, no. 3, pp. 2313–2323, 2020.
- [88] A. Bernstein and E. Dall’Anese, “Linear power-flow models in multiphase distribution networks,” in *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, IEEE, 2017, pp. 1–6.
- [89] B. Cui, A. Zamzam, and A. Bernstein, “Network-cognizant time-coupled aggregate flexibility of distribution systems under uncertainties,” in *2021 American Control Conference (ACC)*, IEEE, 2021, pp. 4178–4183.
- [90] J. Hefferon, “Linear algebra third edition,” 2018.
- [91] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.
- [92] D. Bertsimas and J. Sethuraman, “Moment problems and semidefinite optimization,” in *Handbook of semidefinite programming*, Springer, 2000, pp. 469–509.
- [93] A. Bernstein *et al.*, “Load flow in multiphase distribution networks: Existence, uniqueness, non-singularity and linear models,” *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 5832–5843, 2018.

PUBLICATIONS

The following peer-reviewed papers have been published in IEEE transactions and conference proceedings and constitute the main body of this dissertation. Specifically, Chapters 3 and 4 are largely published as the following first (J1) and second (J2) journal papers, respectively; Chapter 5 is partially submitted as the C1 conference paper and partially will be included in J3 journal paper, which is under preparation and will be submitted soon.

Journal Papers

- J1. **S. Wang**, L. Du, J. Ye, and D. Zhao, “*A Deep Generative Model for Non-intrusive Identification of EV Charging Profiles*”, IEEE Transactions on Smart Grid, vol. 11, no. 6, pp. 4916-4927, Nov. 2020.
- J2. **S. Wang**, L. Du, X. Fan, and Q. Huang, “*Deep Reinforcement Scheduling of Energy Storage Systems for Real-time Voltage Regulation in Unbalanced LV Networks with High PV Penetration*”, IEEE Transactions on Sustainable Energy, vol. 12, no. 4, pp. 2342-2352, Oct. 2021.
- J3. **S. Wang**, L. Du, “*Data-Driven Inner Approximation Framework for Ahead-of-Time Deterministic Power Flexibility Region Evaluation with Robustness to Forecast Uncertainty*”, IEEE Transactions on Industrial Informatics. To be submitted, 2022.

Conference Papers

- C1. **S. Wang**, B. Cui, and L. Du, “*An Efficient Power Flexibility Aggregation Framework via Coordinate Transformation and Chebyshev Centering Optimization*”, Accepted, IEEE Power & Energy Society General Meeting (PES-GM) 2022.
- C2. **S. Wang**, L. Du, and Y. Li, “*Decentralized Volt/Var Control of EV Charging Station Inverters for Voltage Regulation*”, In Proceedings of IEEE Transportation

Electrification Conf. & Expo. (ITEC) 2020.

- C3. **S. Wang**, L. Du, and Q. Zhou, “A *Semi-Supervised Deep Transfer Learning Architecture for Energy Disaggregation*”, In Proceedings of IEEE Power & Energy Society General Meeting (PES-GM) 2019.
- C4. **S. Wang**, L. Du, J. Ye, and D. Zhao, “*Robust Identification of EV Charging Profiles*”, In Proceedings of Transportation Electrification Conf. & Expo. (ITEC) 2018.

Other Papers

The following peer-reviewed papers have been published in IEEE transactions and conference proceedings during my Ph.D. study at Temple University. They are not directly included in this dissertation.

- J4. **S. Wang**, L. Du, Y. Li, and R. Fan, “*Stochastically Stable Synchronous Learning for EV Aggregators Considering Their Collective Age of Information*”, in press, IEEE Trans. Transportation Electrification, 2021.
- J5. **S. Wang** et al, “A *Data-driven Multi-agent Autonomous Voltage Control Framework Using Deep Reinforcement Learning*”, IEEE Transactions on Power Systems, vol. 35, no. 6, pp. 4644-4654, Nov. 2
- J6. **S. Wang**, L. Du, J. Ye, and L. He, “*Non-cooperative Social Welfare Optimization with Resiliency Against Network Anomaly*”, IEEE Transactions on Industrial Informatics, vol. 16, no. 4, pp. 2403-2412, April 2020.
- C5. Y. Du, **S. Wang**, X. Lu, L. Du, Y. Wang, B. Leao, and S. Suresh, “*Physics-Based Feature Extraction from Bulk Time-Series PMU Datasets for Event Detection*”, In Proceedings of IEEE Power & Energy Society General Meeting (PES-GM) 2021.

- C6. **S. Wang**, H. Sun, J. Guo, K. K. Jin, and D. Nikovski, “*Solving Bernoulli Bandit Problems for Weather-relative Overhead Distribution Line Failures Forecasting*”, In Proceedings of IEEE Power & Energy Society General Meeting (PES-GM) 2021.
- C7. D. Sun, **S. Wang**, L. Du, and X. Lu, “*Adaptive Control of Grid-Interactive Inverters with Data-driven Harmonic State Space Modeling*”, In Proceedings of IEEE Energy Conversion Congress and Exposition (ECCE) 2021.
- C8. **S. Wang**, D. Sun, L. Du, and J. Ye, “*Noncooperative Distributed Social Welfare Optimization with EV Charging Response*”, In Proceedings of IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society.