Flexibility-Driven Operation for Low-Carbon Power and Energy Systems

Yuchong Huo

Department of Electrical & Computer Engineering
McGill University
Montréal, Québec, Canada

August 2021

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Doctor of Philosophy.

Abstract

The integration of higher shares of renewable generation is essential for decarbonizing electricity generation. However, the main challenge of integrating renewable energy sources into a power system is the management of the increased disturbances in power balancing. These disturbances are caused by the inherent variability and uncertainty of renewable energy sources, which can put significant stress on reserve requirements in the system. Traditional power system operation paradigms are becoming less capable of handling this challenge, and this has spurred interest in studying the concept of power system flexibility.

Flexibility-based operational planning algorithms typically rely on robust optimization to offer guarantees on the ability of the operator to meet a wide array of possible scenarios. The main downside of these approaches is their conservative results whose operating costs and/or carbon footprint may be sub-economical. Such results come by because these approaches immunize their solutions for the required level of security against realizations of potential events within their uncertainty set. Moreover, these approaches also often ignore the inherent time and spatial couplings of wind and solar generation variability. To tackle this issue, this thesis proposes a modeling technique for uncertainty sets which is called the spatio-temporal flexibility requirement envelope. It reduces the over-conservatism of the robust solution by comprehensively capturing and representing the temporal trends and spatial correlation of multisite renewable generation and load demand. A mathematical program is also developed for applying this envelope to power system unit commitment and dynamic dispatch through projections of the spatio-temporal envelopes, where we mainly focus on microgrid type power systems. We illustrate the effectiveness of the spatio-temporal flexibility requirement envelopes through several case studies.

Furthermore, flexibility-based operational planning approaches are usually formulated based on model predictive control paradigms, which requires the online solution of a mixed-

integer optimization problem at each sampling time. Such approaches are not amenable to most remote microgrid and practical field microgrid implementations, where controls are typically implemented by industrial controllers with limited computational power and the dispatch algorithm faces stringent execution time for real time operation. To tackle this challenge, in this thesis we also develop rigorous machine learning approaches for simplifying and accelerating the flexibility-based microgrid dispatch algorithms, so that they can be implemented in practical settings for real time operation. The proposed machine learning approaches are able to preserve as much as possible the control performance obtained by full mixed-integer optimization. At the same time, they can provide feasible dispatch decisions. We conduct comprehensive performance evaluations to demonstrate the effectiveness of the proposed machine learning approaches.

Abrégé

L'intégration de parts plus élevées de production d'énergie renouvelable est essentielle pour décarboner la production d'électricité. Cependant, le principal défi de l'intégration des ressources d'énergie renouvelable dans un système électrique est la gestion des perturbations accrues dans l'équilibrage de la puissance. Ces perturbations sont causées par la variabilité et l'incertitude inhérentes aux ressources d'énergie renouvelable, qui peuvent exercer une pression considérable sur les besoins en réserves du système. Les paradigmes traditionnels de fonctionnement des systèmes électriques sont de moins en moins capables de relever ce défi, ce qui a suscité l'intérêt pour l'étude du concept de flexibilité du système électrique.

Les algorithmes de planification opérationnelle basés sur la flexibilité reposent généralement sur une optimisation robuste pour offrir des garanties sur la capacité de l'opérateur à répondre à un large éventail de scénarios possibles. Le principal inconvénient de ces approches est leurs résultats prudents dont les coûts d'exploitation et / ou l'empreinte carbone peuvent être sous-économiques. De tels résultats viennent du fait que ces approches immunisent leurs solutions pour le niveau de sécurité requis contre la réalisation d'événements potentiels dans leur ensemble d'incertitudes. De plus, ces approches ignorent souvent les couplages temporels et spatiaux inhérents à la variabilité de la production éolienne et solaire. Pour aborder ce problème, cette thèse propose une technique de modélisation des ensembles d'incertitudes qui est appelée enveloppe d'exigence de flexibilité spatio-temporelle. Il réduit le sur-conservatisme de la solution robuste en capturant et en représentant de manière exhaustive les tendances temporelles et la corrélation spatiale de la production renouvelable multisite et de la demande de charge. Un programme mathématique est également développé pour appliquer cette enveloppe à l'engagement des unités de réseau électrique et à la répartition dynamique via des

projections des enveloppes spatio-temporelles, où nous nous concentrons principalement sur les systèmes d'alimentation de type microréseaux. Nous illustrons l'efficacité des enveloppes d'exigence de flexibilité spatio-temporelle à travers plusieurs études de cas.

En outre, les approches de planification opérationnelle basées sur la flexibilité sont généralement formulées sur la base de paradigmes de contrôle prédictif de modèle, ce qui nécessite la solution en ligne d'un problème d'optimisation en nombres entiers mixtes à chaque pas de temps. De telles approches ne se prêtent pas à la plupart des implémentations de microréseaux isolés et de microréseaux de terrain pratiques, où les contrôles sont généralement mis en œuvre par des contrôleurs industriels avec une puissance de calcul limitée et l'algorithme de répartition fait face à un temps d'exécution strict pour un fonctionnement en temps réel. Pour relever ce défi, dans cette thèse, nous développons également des approches d'apprentissage automatique rigoureuses pour simplifier et accélérer les algorithmes de répartition des microréseaux basés sur la flexibilité, afin qu'ils puissent être mis en œuvre dans des paramètres pratiques pour un fonctionnement en temps réel. Les approches d'apprentissage automatique proposées sont capables de préserver autant que possible les performances de contrôle obtenues par l'optimisation complète des entiers mixtes. En même temps, ils peuvent fournir des décisions d'expédition réalisables. Nous effectuons des évaluations complètes des performances pour démontrer l'efficacité des approches d'apprentissage automatique proposées.

Acknowledgements

Firstly, I would like to express my deep and sincere gratitude to my supervisor Prof. François Bouffard for the continuous support of my Ph.D. study and related research, for his patience, motivation, and immense knowledge. Beyond any doubt, this Ph.D. would not have been possible without his support.

I would also like to thank my co-supervisor, Professor Géza Joós, for his feedback, helpful comments, and financial support.

Besides my supervisors, I would like to thank the rest of my thesis committee: Prof. Dennis Giannacopoulos, Prof. Aditya Mahajan, and Prof. Ioannis Psaromiligkos, for their comments and encouragement.

A very special appreciation also goes to all my friends and colleagues at McGill power laboratories. I also wish to thank Mr. Dave Turcotte of Natural Resources Canada for his assistance in obtaining high-resolution solar radiation data.

This thesis has been made possible with the financial support I received over the course of my Ph.D. studies. I thus would like to acknowledge McGill University for the McGill Engineering Doctoral Award. I also acknowledge the funding from the Natural Science and Engineering Research Council of Canada, Ottawa, ON and from InnovÉÉ, Montreal, QC.

Last but not the least, I would like to thank my parents for supporting me spiritually throughout my life.

Preface

My Ph.D. work has resulted in six first-authored publications: three journal papers and three conference papers (as listed below). My main contributions include: identifying problem definition, designing and developing the methodology, executing simulations and analysing the results, and composing thesis article. Co-authors and their contributions are also listed below.

Journal papers

- Y. Huo, F. Bouffard and G. Joós, "Spatio-Temporal Flexibility Management in Low-Carbon Power Systems," in *IEEE Transactions on Sustainable Energy*, vol. 11, no. 4, pp. 2593-2605, Oct. 2020, doi: 10.1109/TSTE.2020.2967428.

Prof. François Bouffard performed in-depth proofreading, refined the writing of the paper, assisted in addressing some of the reviewer's questions, and provided necessary computing hardware and software. Prof. Géza Joós provided comments on the paper structure and approach. This research work is included in Chapter 2 of this thesis.

- Y. Huo, F. Bouffard and G. Joós, "Decision tree-based optimization for flexibility management for sustainable energy microgrids," *Applied Energy*, vol. 290, May 2021. https://doi.org/10.1016/j.apenergy.2021.116772.

Prof. François Bouffard initiated the conceptual design of the control approach, provided comments on the paper structure and approach, refined the writing of the paper, and assisted in addressing some of the reviewer's questions. Prof. Géza Joós provided comments on the paper structure and approach, and provided access to real-time simulation tools and hardware. This research work is included in Chapter 3 of this thesis.

- Y. Huo, F. Bouffard and G. Joós, "Integrating Learning and Explicit Model Predictive Control for Unit Commitment in Microgrids," *Applied Energy*, Under review.

Prof. François Bouffard provided comments on the paper structure and approach, performed in-depth proofreading, and refined the writing of the paper. Prof. Géza Joós provided comments on the approach. This research work is included in Chapter 4 of this thesis.

Conference papers

- Y. Huo, F. Bouffard and G. Joos, "Flexibility Management Considering Transmission Limit and Correlation of Wind Power," *2019 IEEE Power & Energy Society General Meeting (PESGM)*, 2019, pp. 1-5, doi: 10.1109/PESGM40551.2019.8973550.

- Y. Huo, F. Bouffard and G. Joós, "Flexibility Envelopes for Distribution Networks," *2018 IEEE Power & Energy Society General Meeting (PESGM)*, Portland, OR, 2018, pp. 1-5, doi: 10.1109/PESGM.2018. 8586668.

Prof. François Bouffard performed in-depth proofreading and refined the writing of the two conference papers. Prof. Géza Joós provided financial assistance for attending the conferences. Part of the research work in the two conference papers is included in Chapter 2 of this thesis.

My Ph.D. work led to another first-authored conference paper. Its topic is only tangentially related to the theme of the thesis. Thus I chose not to include it in my thesis.

- Y. Huo, F. Bouffard and G. Joós, "An energy management approach for electric vehicle fast charging station," *2017 IEEE Electrical Power and Energy Conference (EPEC)*, Saskatoon, SK, 2017, pp. 1-6, doi: 10.1109/EPEC.2017.8286143.

Prof. François Bouffard performed in-depth proofreading and refined the writing of the conference paper. Prof. Géza Joós provided financial assistance for attending the conference.

Contents

List of Figures

List of Tables

List of Acronyms

CEER	council of european energy regulators.
CPU	central processing unit.
CR	critical region.
DER	distributed energy resources.
DNN	deep neural network.
DT	decision tree.
EMPC	explicit model predictive control.
EMS	energy management system.
FPGA	field-programmable gate array.
IEA	international energy agency.
IRENA	international renewable energy agency.
MG	microgrid.
MILP	mixed integer linear program.
MIP	mixed integer programming.
ML	machine learning.
MPC	model predictive control.
MSE	mean squared error.
NN	neural network.
PC	principal component.
PCA	principal component analysis.
PLC	programmable logic controller.
POI	point of interconnection.
PV	Photovoltaic.
ReLU	rectified linear unit.

RES	renewable energy source.
UC	unit commitment.
VF	value function.

List of Symbols

The main symbols used in the thesis are defined here. Further symbols will be defined as required.

Sets

Ξ_e	Set of energy storage assets.
Ξ_g	Set of conventional generators.
Ξ_H	Set of intra-hourly receding-horizon time steps.
Ξ_n	Set of buses.
Ξ_q	Set of movement directions in a flexibility requirement envelope.

Indices

τ	Index of intra-hourly receding-horizon time step.
i	Index of dispatchable generators.
j	Index of storage assets.
n	Index of buses.
q	Index of movement directions in a flexibility requirement envelope (up: $\uparrow$, down: $\downarrow$).
t	Index of time.
t_0	Current time.

Variables

$(\cdot)^*$	The optimal solution to an optimization problem.
$(\cdot)^\intercal$	The transpose of a vector or matrix.
$\delta_c(t)$	Total renewable generation curtailment in the system at time t.
$\delta_c^n(t)$	Renewable generation curtailment at bus n at time t.
$d_c(t)$	Total load curtailment in the system at time t.
$d_c^n(t)$	Load curtailment at bus n at time t.
$p_e^j(t)$	Output of energy storage j at time t.
$p_e^{j\pm}(t)$	Positive $(+)$ and negative $(-)$ parts of output of energy storage j at time t.
$p_g^i(t)$	Output of dispatchable generator i at time t.
$s_e^j(t)$	Energy level of storage j at the end of time t.
$u_e^{j\pm}(t)$	Commitment status (binary) of discharging $(+)$/charging $(-)$ of storage j at time t.
$u_g^i(t)$	Commitment status (binary) of dispatchable generator i at time t.

Parameters

$\delta(t)$	Total renewable generation in the system at time t.
$\delta_n^s(t)$	Output of the solar farm at bus n at time t.
$\delta_n^w(t)$	Output of the wind farm at bus n at time t.
η_c^j, η_d^j	Charging / discharging efficiency of storage j.
$\hat{d}(t)$	Net load in the system at time t, $\hat{d}(t) = d(t) - \delta(t)$.
$\overline{(\cdot)}, \underline{(\cdot)}$	The maximum / minimum limit of the variable.
a_g^i, b_g^i, c_g^i	Cost parameters of generator i.

$C_i(\cdot)$	Cost function of generator i.
$e_A^{\uparrow}$,$e_A^{\downarrow}$	System-wide flexibility requirement envelope.
$e_{F_l}^{\uparrow}$,$e_{F_l}^{\downarrow}$	Flexibility requirement envelope applying to transmission line l.
$p_{poi}(t)$	Contractual value of power exchange at POI at time t.
$r_{g\mathrm{dn}}^i$	Downward ramping limits of generator i.
$r_{g\mathrm{sd}}^i$	Shut-down ramping limits of generator i.
$r_{g\mathrm{su}}^i$	Start-up ramping limits of generator i.
$r_{g\mathrm{up}}^i$	Upward ramping limits of generator i.
T_Δ	Duration between two receding-horizon time steps.
T_H	Length of the forward-looking horizon.
$\mathbf{E}(t_0)$	Spatio-temporal flexibility requirement envelope at current time t_0.
$d_n(t)$	Load at bus n at time t.
$d(t)$	Total load in the system at time t.

Chapter 1

Introduction

Today, the installation of renewable energy source (RES) is growing at fast rate globally. The International Energy Agency (IEA) forecasts that renewable capacity additions are on track for a record expansion of nearly 10% in 2021 worldwide. In the next five years, the generation of renewable electricity will expand almost 50% to almost 9745 TWh globally, which is equivalent to the combined demand of China and the European Union. By 2025, the share of renewable electricity in total electricity generation is forecast to be 33%, surpassing the coal-fired electricity generation [1].

However, many problems can arise from the integration of large quantities of renewable generation into the grid. Most RES such as wind generators and photovoltaics (PVs) cannot be predicted with perfect accuracy because of their intermittent and stochastic nature originating from weather conditions [2]. These inherent characteristics of renewable generation inevitably increase the variability and uncertainty in power systems and threaten the secure and reliable operation of the system. Consequently, this prompted the recent emergence of the studies of power system operational flexibility in both academia and industry [3], [4].

Nowadays the implementation of microgrid (MG) is recognized as an attractive and cost effective way of managing flexibility in power systems [5]. It provides a platform for leveraging the inherent flexibility in distributed energy resources (DERs) and meeting the flexibility requirement arising from increased RES penetration. In this thesis, we mainly focus on MG type power systems, while the transmission systems are also included.

Traditionally, energy dispatch in practical MG is executed using rule-based strategies.

Although rule-based strategies are effective at providing feasible dispatch instructions, they are inadequate in handling the volatility brought about by renewable generation, while also they cannot exploit flexible dispatchable assets, like energy storage, to their fullest capacity. Moreover, the statistical characteristics such as spatial correlation and diurnal or seasonal trends of renewable generation are usually ignored when managing flexibility in MGs, which could result in overly conservative scheduling results. To tackle these challenges, in this thesis we develop new paradigms for (1) short-term flexibility requirement assessment induced by variable demand and renewable generation, and (2) systematic approaches for MG dispatch capable of optimizing MG dispatchble asset flexibility and maximize variable renewable generation intake.

1.1 Characterization of Operational Flexibility

1.1.1 Definition of Power System Operational Flexibility

The concept of power system operational flexibility has been introduced formally only recently. Therefore, there is still no universal and common definition for operational flexibility yet [6]. Different academic and industrial researchers are proposing definitions based on different context and perceived challenges, which we discuss shortly here.

From the point of view of power system operation, operational flexibility is defined as "the capability of the power system to follow a schedule that continuously attains active power balance and to contain injection deviations from this schedule in order to maintain a secure operating state" [7]. In this context the security of a power system refers to "the degree of risk in its ability to survive imminent disturbances (contingencies) without interruption of customer service" [7]. A definition in [8] states that operational flexibility is the "the capacity of the electricity system to respond to changes that may affect the balance of supply and demand at all times". In [9], operational flexibility is also described as "the capability of a power system to cope with the variability and uncertainty that variable renewable energy generation introduces into the system in different time scales, from the very short term to the long term, avoiding curtailment of variable renewable energy and reliably supplying all the demanded energy to customers".

From the technical side, operational flexibility is required to modify the electricity

production and consumption of a power system in response to variability, expected or otherwise [10].

A techno-economic definition by [11] states that, "flexibility is the ability of a power system to cope with variability and uncertainty in both generation and demand, while maintaining a satisfactory level of reliability at a reasonable cost, over different time horizons". Reference [12] also defines operational flexibility as "the relevant characteristics of a power system that facilitates the reliable and cost-effective management of variability and uncertainty in both supply and demand" from the the economic viewpoint.

According to the above definitions, a power system should have sufficient operational flexibility to maintain economic and secure operation while coping with increasing levels of intermittency imposed by the wider integration of non-dispatchable renewables.

1.1.2 Flexibility Requirements

The expression "flexibility requirement" is used in this thesis to denote the generalized reserve requirements induced by the intermittent renewable generation and uncertain load in power system. Generally, the flexibility requirement of a power system is dictated by the amount of variability and uncertainty in net load (i.e. load less RES output) [4].

Variability arises because the net load has a maximum output limit that changes with time [6]. Historically, grid operators only need to adjust the production of the generators to follow the load demand. With the growing grid integration of RES, their intermittent nature introduces additional variability, which increases the flexibility requirements in the power system.

Uncertainty arises because this maximum limit of the net load cannot be predicted with perfect accuracy [6]. Operational flexibility is required to compensate these uncertainties and real time generation-demand mismatches. Large forecast uncertainties could lead to extra charges as generation capacities might have to be reserved or costly re-dispatches are needed [7].

Moreover, because of the variability and uncertainty of renewable generation and uncertain loads in power system, transmission line flows can deviate largely from their scheduled flow. If the transmission grid is operating close to the limit, enough operational flexibility will be required to alleviate possible congestion.

1.1.3 Resources of Operational Flexibility

The flexibility requirements described above have to be properly handled using a variety of flexibility resources, such as conventional controllable power plants, energy storage, and other sources of operational flexibility, as illustrated in Figure 1.1.

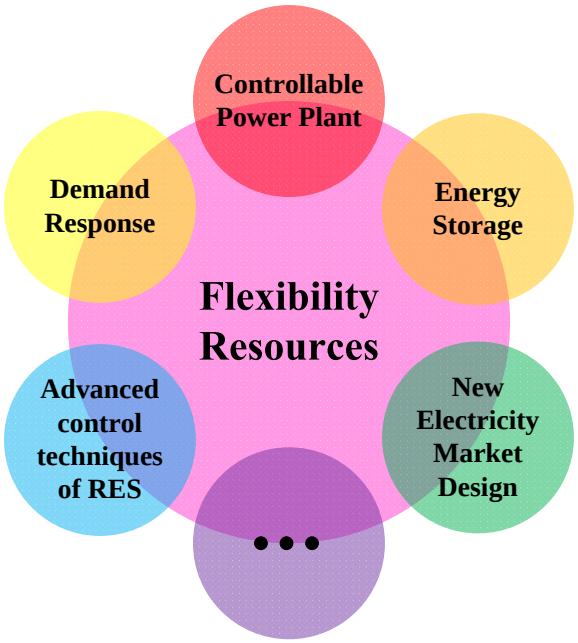

Figure 1.1: Sources of operational flexibility in a power system.

The operational flexibility that can be derived from a conventional controllable power plant (e.g., coal-fired, hydroelectric and nuclear power plants) is mainly determined by its capacity, ramp rate, and startup/shutdown time [13]. Generators with different levels of operational flexibility play different roles in meeting demand. For example, nuclear power plants may take many hours to start up and shut down and are usually built to operate at rated power as base load, while hydroelectric and gas turbine power plants have greater flexibility and can operate as load following or peaking power plants [7]. Manufacturers are always exploiting more flexible units such as those with improved ramping capabilities, low minimum generation constraints and high efficiencies [2], [13].

Storage is another important source of operational flexibility. Integrating energy storage system with the high penetration of renewable generation provides many benefits. Energy

storage can absorb the surplus renewable generation and then redistribute it during periods of high electricity demand, which reduces the curtailment of renewable generation and allows the energy to be utilized when needed [2]. Different types of energy storage technologies can be chosen in different situations. For example, batteries, ultracapacitors, superconducting inductors, flywheels and compressed air can be used for regulation, while pumped hydro can be used for day-night arbitrage [13]. A comparison of different energy storage technologies in terms of discharge time and nominal capacity can be found in [14]. Moreover, it is anticipated that the capital costs of storage will fall rapidly in the future, as storage technology develops [2].

There are many other types of flexibility resources such as demand response [15–17], new electricity market design [18–20], and advanced control techniques of RES [21]. These technologies will not be discussed in this thesis. A comprehensive overview of these technologies can be found in [2] and [3].

1.2 State of the Art on the Management of Operational Flexibility

Managing flexibility can be seen as a process that optimally matches flexibility resources with flexibility requirements. It is worth noting that studies on management of operational flexibility can be classified into two types: short-term flexibility planning and long-term flexibility planning. The objective of short-term flexibility planning is to ensure economical and secure operations of a power system by committing the right flexibility resources and prepositioning them properly ahead of time [13], as shown in Figure 1.2. This is generally done through unit commitment and dynamic economic dispatch.

On the other hand, long-term flexibility planning [22–25] focuses on the changes in the generation mix, legislation policies, and consumption patterns over several months or years [2]. In this thesis, we only discuss short-term planning of operational flexibility. The long-term flexibility planning is left for future research.

Previous studies used stochastic optimization methods and robust optimization approaches for short-term planning of operational flexibility [3]. The former models the flexibility requirements with probability distributions, which can effectively reduce

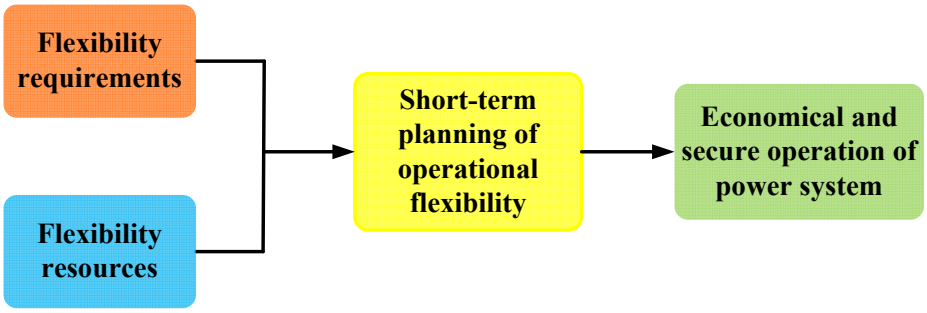

Figure 1.2: Illustration of short-term flexibility planning in power system.

operational costs. On the other hand, stochastic methods do suffer from computational intractability caused by having to consider increasingly large numbers of scenarios [26], [27]. In response to the shortfalls of stochastic optimization, robust optimization approaches have gained more attention and been used extensively in short-term planning of operational flexibility for their tractable computational costs and their ability to provide certainty to operators. [1] Therefore, we mainly consider robust approaches for the short-term planning of operational flexibility in this thesis.

1.2.1 Quantification of Flexibility Requirements

When carrying out short-term planning of operational flexibility, we need to first quantify the flexibility requirements in the system. The robust approaches use uncertainty sets to describe the flexibility requirements of a power system [28]. Usually, uncertainty sets consider worst-case or highly conservative realizations of uncertain renewable generation and load.

Many different modeling techniques for uncertainty sets have been proposed in last few years. References [6], [29] proposed the concept of the flexibility requirement envelopes to capture the intra-hourly spectrum of reserve requirements entailed by variable renewable generation. Box shaped envelopes were developed in [30] and [31] to quantify flexibility requirements for a range of intra-hourly duration.

However, the studies presented above only built uncertainty sets for the univariate system net load. When modeling geographically-distributed renewable generation or load demand, the uncertainty sets take the form of a multivariate envelope. The authors of [32] and [33]

[1] A detailed comparison of robust and stochastic approaches can be found in [26].

focus on capturing the spatial correlation of multisite renewable generation with ellipsoidal uncertainty sets. Jiang *et al.* [34] describes the uncertainty set by a parallelotope which is constructed by the intersection of a family of ellipsoids. Frameworks for generating and evaluating uncertainty sets in the form of multivariate polytopes were developed by [35] and [5]. A multivariate prediction interval was studied in [36] for multisite wind generation.

1.2.2 Flexibility Management in Bulk Power Systems

In order to properly match flexibility resources with flexibility requirements, the system operator must validate whether the flexibility resources are capable of providing power setpoint adjustments (both upward and downward), within the time scales and power volumes entailed by the realization of the net load variability and uncertainty, and within the transmission capacity of the network [13]. The related costs of dispatching flexibility resources are also optimized in this process.

Flexibility management in bulk power systems is a complex optimization problem which involves unit commitment and dynamic dispatch. The typical inputs of the optimization model include the quantified flexibility requirements and the parameters of flexibility resources (e.g., capacity, ramp rate and startup/shutdown time).

References [6] and [29] first characterized the flexibility potential dynamics of each flexibility resource by maximizing its upward and downward deviation from a scheduled output, subject to power capacity, ramping and energy constraints. Then, the short-term planning of operational flexibility was carried out by ensuring that the aggregate flexibility potential of all the resources is able to enclose the flexibility requirement envelope over the planning horizon. The authors of [37] studied the coordination of available operational flexibility between different TSOs in a multi-area power system. The short-term planning of operational flexibility was formulated as a mixed-integer linear programming (MILP) and calculated based on computational geometry which considers location and availability of reserves, transmission constraints, interdependencies of tie-line flows between different areas, and the $N-1$ security criterion.

A framework for developing a composite metric that assesses the flexibility of conventional generators was proposed in [38]. Six technical characteristics of conventional generators were used as indicators in the assessment. These indicators were weighted using a fuzzy analytic hierarchy process in order to reflect their relative importance in the supply of flexibility. A

similar framework was also proposed in [39]. Other works, for example [40] and [41], modeled the flexibility that can be derived from the emerging renewable generation technologies to enhance the system's flexibility and to gain higher profits for the system operator.

1.2.3 Flexibility Management in Microgrids

MGs are electricity distribution systems containing loads and DERs that can be operated in a controlled, coordinated way either while connected to the main power network or while islanded [42]. The key feature of a MG is its ability to utilize available DERs to attain certain goals or objectives through a MG controller which belongs to the operator of that MG [42], as is shown in Figure 1.3. The energy management system (EMS) is the central part of the MG controller.

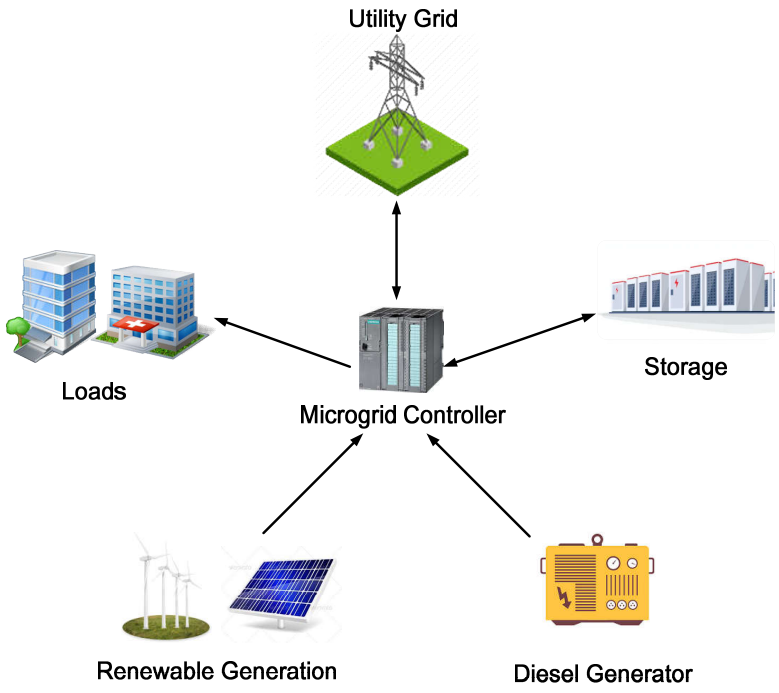

Figure 1.3: An example of MG structure.

Renewable generation is prevalent in MG [43]. The RES in MG will create large and sudden ramping events and reduce the available lead-time of operational planning [5]. This has increased flexibility requirements in MG.

The main responsibility of the MG controller is to manage the flexibility available from DERs and controllable loads so as to balance the power production and energy consumption. Reliable and cost effective solutions can be found through short-term planning of operational flexibility, which consists of MG unit commitment and economic dispatch. For example, Holjevac and others [44] provided a centralized framework for comprehensive operational flexibility evaluation of multi-energy MG. This was done by incorporating MILP model for annual simulations and expanding it with model predictive control (MPC) algorithm for short-term daily operational analyses. A rolling-horizon algorithm for management of operational flexibility in MG was proposed in [45], where unit commitment (UC) is solved repeatedly to adjust the set points of energy resources based on updated information of load and renewable generation. The authors of [46] developed a rolling horizon scheme which simultaneously exploits the operational flexibility of energy production and energy demand. A two-layer model predictive controller was proposed in [47] to schedule and extract the existing flexibility in the energy resources of MG. A MPC-based framework for calculating the optimal economic dispatch of combined heat and power MG was developed in [48], which used affine arithmetic to model the flexibility requirement of renewable generation. The authors of [49] and [50] developed MPC-based approaches for real-time flexibility management in the context of a MG, while taking into account the uncertainties of electricity prices. A MPC strategy for extracting flexibility in battery energy storage systems in a DC microgrid was presented in [51]. The authors of [52] studied the flexibility requirement of aggregated electric vehicles and proposed a two-layer MPC approach for the management of operational flexibility in a MG containing electric vehicles. A MPC-based approach for managing distributed energy resources of a MG was presented in [53], where the flexibility requirements arising from uncertain electricity prices, uncertain loads and fluctuating renewable energy were taken into consideration. In [54], a MPC-based strategy was developed to perform real-time flexibility management of an isolated microgrid to maximize the power delivery from renewable generation to commercial buildings. A reserve assessment method was developed in [5] for distribution system, where the flexibility of dispatchable generators and the power exchange with the utility grid were modeled as polytopes. The operational cost of the system was minimized by allocating the polytopes while considering the resource location and the network constraints. A systematic approach for harnessing the flexibility benefits

of MG business cases was provided in [55]. Reference [42] designed a centralized EMS considering different objectives of MG, such as minimized cost, reduction in peak power, power smoothing, greenhouse gas emission reduction, and increased reliability of service. Multi-objective optimization technique was employed to solve the power dispatch optimization problem.

Energy storage plays an integral role in the management of generation and loads in a MG and thus is a critical component in the development of MG dispatch algorithms. Storage can provide services like demand shift and peak load shaving, which increases the flexibility in MG operation. It can also delay the startup of expensive generation sources such as diesel generators, which reduces the operation cost of MG. Reference [56] proposed an model employing a backcasting algorithm to estimate the net value of stored energy in an energy storage system. The current cost of energy was then compared against this net value to determine how the energy storage could perform arbitrage in MG. A model that facilitated the seamless integration of energy storage systems with conventional generators was developed in [29]. It allowed the energy levels of storage systems to be prepositioned beforehand to optimize their potential deployment over a forward-looking horizon.

1.3 Gaps in the State of the Art

The robust optimization approaches are employed by the various flexibility management methods presented above. Although robust approaches secure the system according to very stringent reserve requirements, it is easy to have sub-economical results due to having to hedge against highly improbable potential events [57]. Intensive research activity is investigating solutions to alleviate this problem. It is found that the conservativeness of a robust solution is directly linked to the size of the uncertainty set [35]. In addition, [33] suggests that considering statistical characteristics such as spatial correlation and diurnal or seasonal trends of renewable generation can effectively improve the accuracy of uncertainty sets and reduce its size while keeping the predefined coverage rates. However, existing works such as [37] and [27] overlook these statistical properties, while works such as [34] and [5] only consider spatial correlation. At the same time, other existing works such as [33], [35] and [32] merely build mathematical models for uncertainty sets. The authors did not discuss how to apply these models to the security-constrained unit

commitment and dynamic dispatch problems in power systems, which are the central parts of the short-term flexibility planning.

Moreover, when we implement flexibility-based operational planning paradigms in MG, another problem arises. Controls in MG are typically implemented by programmable logic controller (PLC) or other industrial controllers. Figure 1.4 shows an example of PLC which is adopted from [58]. PLCs are programmable using ladder logic which very much looks like decision tree (DT). Usually, PLCs have limited computational power in terms of memory and speed capabilities [59].

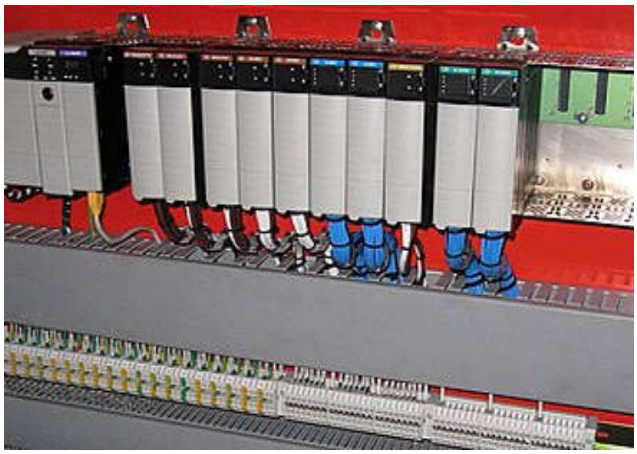

Figure 1.4: An example of PLC [58].

At the same time, the MG dispatch algorithm faces stringent execution time for real time operation, often in remote locations. Although existing studies such as [5, 44–47] have shown that flexibility-based operational planning paradigms can contribute a great deal to the economic and secure operation of MG, their models are usually formulated based on MPC paradigm (or receding horizon scheme), which requires the online solution of a mixed integer programming (MIP) at each sampling time of the controller. Traditional methods for solving MIP problems require advanced iterative optimization algorithms and significant computational power in terms of CPU and memory [60], which may be difficult for real time implementation in a MG controller with limited computational power. Resolving these issues calls for approaches capable of simplifying and accelerating the traditional flexibility-based operational planning paradigms so that they can be implemented in a practical industrial controller for real time operation.

1.4 Thesis Contribution

This thesis contributes to filling the research gaps presented above. We first propose a modeling technique for uncertainty sets which we call *spatio-temporal flexibility requirement envelopes* as a natural extension of the original proposal of [6]. It seeks to model uncertainty sets in a more precise manner by representing the temporal trends and the spatial correlation of multisite renewable generation and load demand. We then propose a robust optimization based framework for applying this envelope to power system unit commitment and economic dispatch. Compared to previous work, our proposed method has the following features. First, it extends the notion of flexibility requirement envelopes proposed in [6] and [29] by comprehensively capturing the temporal trends and correlation of multiple site renewable generation and loads in power system. Second, this approach is applied in robust security-constrained unit commitment and dynamic dispatch problems in power systems, capturing the effects of space-correlated renewable generation assets on the transmission network and its constraints. The energy scheduling framework is formulated as a mixed integer program.

Furthermore, this thesis attempts to implement the flexibility-based operational planning paradigms in a MG controller with limited computational power for real time MG energy management. This is realized by leveraging the power of machine learning (ML) techniques.

We first study a simple case where the MG has a maximum of one dispatchable generator. We formulate the MG dispatch problem with the concept of flexibility requirement envelopes, which is in the form of a MPC problem and requires the solution of a MIP at each time step. A rigorous DT-based learning framework is constructed to approximate the mapping between the inputs and outputs of the MIP, where the operating constraints of MG, uncertainties in net load, the commitment status (ON/OFF) of a dispatchable generator, and an energy storage system are considered systematically. We then directly apply the trained DT to predict dispatch decisions for the MG with given inputs, which is an open-loop process and thus computationally light and promising to enable real time implementation in a computationally limited MG controller. Also, the DT-based MG dispatch strategy is capable of maintaining adequate performance even in cases where MG assets are down or degraded.

Moreover, we consider the general case where the MG has multiple dispatchable

generators [2]. The UC problem of MG is again formulated with the concept of flexibility requirement envelopes. We develop a rigorous approach which combines k-means clustering, deep neural network (DNN) classification and explicit model predictive control (EMPC) for implementing the flexibility-based UC paradigms in a MG controller with limited computational power. The computational complexity of the proposed approach can be adjusted to meet the hardware limitations of any given MG controller, while preserving as much as possible the optimality of the full-fidelity EMPC. This enables the online real time implementation of the proposed approach. Moreover, the proposed approach is able to handle the variables and constraints of the original UC problem systematically, which guarantees the feasibility of its output UC schedules.

1.5 Claim of Originality

This thesis establishes the following distinct contributions to the field of operational flexibility management in power systems.

1. This thesis proposes the "spatio-temporal flexibility requirement envelope" that comprehensively captures the temporal trends and correlation of multiple site renewable generation and loads for short-term planning of operational flexibility in power system. We also propose a robust optimization based framework for applying this envelope to power system unit commitment and economic dispatch.

2. A rigorous DT-based learning framework for real time implementation of flexibility-based MG dispatch paradigm in a computationally limited MG controller is developed. Here, we assume that the MG has a maximum of one dispatchable generator. In this framework, the operating constraints of MG, uncertainties in net load, the commitment status (ON/OFF) of a dispatchable generator, and an energy storage system are considered systematically.

3. A rigorous approach for implementing flexibility-based UC on multiple dispatchable generators in a MG controller with limited computational power is developed for real

[2]MGs can have various sizes ranging from small residential MGs (e.g., a building, a hospital) to large-scale MGs (e.g., a city). For large-scale MGs, it is possible that they have multiple dispatchable generators.

time MG operation. This approach is able to handle the variables and constraints of the original UC problem systematically. At the same time, the computational complexity of this approach is tunable by accepting a certain level of suboptimality.

1.6 Thesis Outline

Chapter 2: Spatio-Temporal Flexibility Management in Power Systems

In this chapter, the spatio-temporal flexibility requirement envelope is modeled. This chapter first introduces the modeling methodology of spatio-temporal flexibility requirement envelopes. Next, the projection of this envelope is presented. Afterwards, we introduce the energy scheduling framework for deploying operating reserves to satisfy quantified flexibility requirements. We showcase the use and advantages of spatio-temporal flexibility requirement envelopes and their associated scheduling approach in a microgrid and on a modified IEEE Reliability Test System.

Chapter 3: Decision tree-based Flexibility Management for Microgrids

In this chapter, a rigorous DT-based learning framework for implementing flexibility-based MG dispatch paradigm in a MG controller is modeled. This chapter first formulates the MG dispatch problem with the concept of flexibility requirement envelopes. Here, we assume that the MG has a maximum of one dispatchable generator. Next, the method for building training data set is presented. From the training data generated, we first grow a large DT, and then prune it back in order to obtain a subtree, whose size is more reasonable for a practical implementation. We also introduce the method for testing the DT-based MG dispatch algorithm. A comprehensive performance evaluation is conducted to demonstrate the effectiveness of the proposed DT-based approach under various scenarios.

Chapter 4: Integrating Learning and EMPC for UC in Microgrids

In this chapter, a rigorous approach which combines k-means clustering, DNN classification and EMPC for implementing the flexibility-based UC paradigms in a MG controller is modeled. This chapter first formulates the UC problem of MG with the concept of flexibility requirement envelopes. The MG has multiple dispatchable generators. We also

develop the EMPC formulation of the original UC problem. Next, the method for preparing training data set is presented. A two-stage learning based approach is developed afterwards. In this approach, the first stage performs k-means clustering and the second stage performs DNN classification. The testing method of the proposed approach is also introduced. A case study is conducted to demonstrate the effectiveness of the proposed approach under various scenarios.

Chapter 5: Conclusion

In this chapter, we summarize the key achievements of this thesis and discuss several research directions for future research.

Appendices

Appendix A provides detailed information of time series Gaussianization and principal component analysis (PCA), which is used when we build the spatio-temporal flexibility requirement envelopes in Chapter 2.
Appendix B summarizes the method for building flexibility requirement envelopes previously proposed in [6].
Appendix C formulates the microgrid dispatch problem (with one dispatchable generator only) based on the concept of flexibility requirement envelope.
Appendix D gives a brief introduction about how to grow large decision trees with training data. It also introduces briefly the cost complexity pruning method, which is used for pruning large decision trees.
Appendix E formulates the microgrid UC problem (with multiple dispatchable generators) based on the concept of flexibility requirement envelope.
Appendix F gives a brief introduction of the "adaptive contextual learning based microgrid UC strategy" which was previously proposed in [61].

Chapter 2

Spatio-Temporal Flexibility Management in Power Systems

The deepening penetration of renewable power generation is challenging how the minute balancing of supply and demand is carried out by power system operators. In Chapter 1, we reviewed the state of the art on short-term planning of operational flexibility. To summarize, short-term flexibility planning algorithms typically rely on robust optimization to offer guarantees on the ability of the operator to meet a wide array of possible scenarios. The main drawback of these approaches is their conservative results whose operating costs and/or carbon footprint may be sub-economical.

In this chapter, we seek to reduce the conservativeness of the robust solution by proposing the concept of *spatio-temporal flexibility requirement envelopes*. Please note that the spatio-temporal flexibility requirement envelope developed in this chapter is a natural extension of the *flexibility requirement envelope* proposed in [6]. In [6], the spatial correlation between different renewable energy resources was ignored. Moreover, the output of renewable generation was assumed to be a stationary random process. Although the original proposal of [6] can secure the system according to very stringent reserve requirements, it is easy to have sub-economical results due to having to hedge against highly improbable potential events. It is found in [35] that the conservativeness of a robust solution is directly linked to the size of the uncertainty set. Moreover, [33] suggests that considering statistical characteristics such as spatial correlation and diurnal or seasonal trends of renewable generation can effectively improve the accuracy of the uncertainty set

and reduce its size while keeping the predefined coverage rates. Therefore, the spatio-temporal flexibility requirement envelope proposed in this chapter seeks to model the flexibility requirements in a more precise manner and reduce the over-conservatism of the energy scheduling results by representing the temporal trends and the spatial correlation of multisite renewable generation and load demand. A mathematical program for energy scheduling is also developed using the projections of this envelope. We showcase the use and advantages of spatio-temporal flexibility requirement envelopes and their associated scheduling approach in a microgrid and on a modified IEEE Reliability Test System. An overview of our proposed method is shown in Figure 2.1.

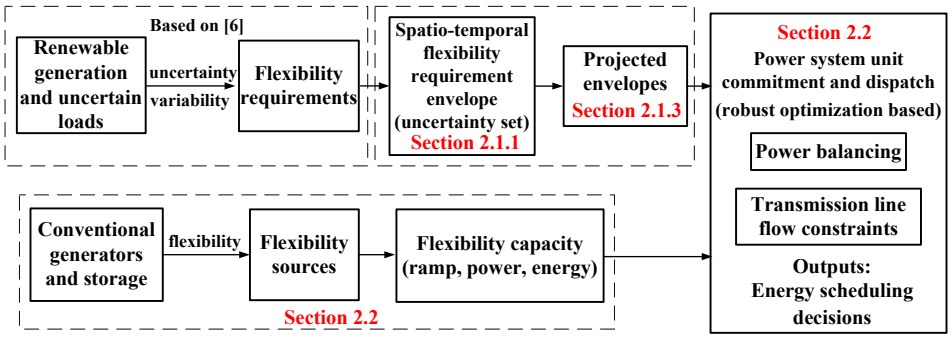

Figure 2.1: Overview of the method proposed in this chapter.

2.1 Modeling of Spatio-temporal Flexibility Requirements

2.1.1 Spatio-temporal Flexibility Requirement Envelope

The procedure for building a spatio-temporal flexibility requirement envelope from the current time t_0 is introduced next. The inputs of the process are time series of wind speed, solar power and uncertain load demand which contain their current and historical data. All the input time series have the same length of T_S (e.g., one year), and they are synchronized and evenly spaced in time (e.g., five-minute intervals).

Step 1: *(Temporal Detrending):* For the input solar power time series $\{\delta_n^s(t)\}_{t=t_0-T_S+1}^{t_0}$ at bus

n during daytime, we compute $\delta_n^{s\prime}(t)$, the detrended solar power time series [62]

$$\delta_n^{s\prime}(t) = \frac{\delta_n^s(t)}{\delta_n^{CS}(t)} \tag{2.1}$$

where $\delta_n^{CS}(t)$ is the output of solar power under the clear-sky irradiance (theoretical and deterministic maximum of solar power caused by the periodical movements of the Earth and the Sun) at bus n at time t. Similarly, we calculate $\omega_n'(t)$, the detrended wind speed or uncertain load demand time series at bus n [63]:

$$\omega_n'(t) = \frac{\omega_n(t) - \mu_n^M(t)}{\sigma_n^M(t)} \tag{2.2}$$

where $\omega_n(t)$ $(t = t_0, t_0 - 1, \ldots, t_0 - T_S + 1)$ is the input original wind speed or uncertain load demand time series, and $\mu_n^M(t)$ and $\sigma_n^M(t)$ are its estimated mean and standard deviation calculated following the method used in [64]:

$$\mu_n^M(t) = \frac{1}{M} \sum_{k=0}^{M-1} \omega_n(t-k) \tag{2.3}$$

$$\sigma_n^M(t) = \sqrt{\frac{1}{M-1} \sum_{k=0}^{M-1} \left(\omega_n(t-k) - \mu_n^M(t)\right)^2} \tag{2.4}$$

where M is the length of the moving time window for calculating $\mu_n^M(t)$ and $\sigma_n^M(t)$.

Step 2: *(Spatial Decorrelation):* In this step, the detrended time series $\omega_n'(t)$ and $\delta_n^{s\prime}(t)$ are transformed into Gaussian time series onto which principal component analysis (PCA) is performed. The goal here is to decompose the factors driving the variability processes into independent variability components. The details of the transformation are provided in Appendix A.

Step 3: *(Principal Component-Wise Flexibility Requirement Envelope Construction):* For each principal component (PC), we build a flexibility requirement envelope $\{\varepsilon_r^{\downarrow}(\tau), \varepsilon_r^{\uparrow}(\tau)\}_{\tau \in \Xi_H}$ using the method presented in [6]. Here $\{\varepsilon_r^{\downarrow}(\tau), \varepsilon_r^{\uparrow}(\tau)\}_{\tau \in \Xi_H}$ is the flexibility requirement envelope for the rth $(1 \leq r \leq R)$ PC. The details of the procedure for building $\{\varepsilon_r^{\downarrow}(\tau), \varepsilon_r^{\uparrow}(\tau)\}_{\tau \in \Xi_H}$ is explained in Appendix B.

Step 4: *(Envelope Construction in the PC Domain):* Construct a R-dimensional hypercube $\mathbf{E}(t_0, \tau)$ at present time t_0 for all the forward-looking times $\tau \in \Xi_H$ using the flexibility requirement envelope of all the R PCs generated in Step 3. Because the PCs are independent from each other, we use the Cartesian product:

$$\begin{aligned}\mathbf{E}(t_0, \tau) = &\{Z_1(t_0) + \varepsilon_1^{\uparrow}(\tau), Z_1(t_0) + \varepsilon_1^{\downarrow}(\tau)\}\times \\ &\{Z_2(t_0) + \varepsilon_2^{\uparrow}(\tau), Z_2(t_0) + \varepsilon_2^{\downarrow}(\tau)\}\times \\ &\cdots \\ &\times\{Z_R(t_0) + \varepsilon_R^{\uparrow}(\tau), Z_R(t_0) + \varepsilon_R^{\downarrow}(\tau)\}\end{aligned} \tag{2.5}$$

where $Z_r(t_0)$ $(1 \leq r \leq R)$ is the value of the time series of rth PC at current time t_0, which was obtained in Step 2.

Step 5: *(Re-Inserting Spatial Dependence):* Reconstruct the hypercube $\mathbf{E}(t_0, \tau)$ $(\tau \in \Xi_H)$ by inverting the PCA transform using (A.3), following by reverting (A.1).

Step 6: *(Re-Trending):* Reconstruct the $\mathbf{E}(t_0, \tau)$ $(\tau \in \Xi_H)$ obtained in the last step by inverting the transform in (2.1) and (2.2) using $\delta_n^{CS}(t_0)$, $\mu_n^M(t_0)$, and $\sigma_n^M(t_0)$.

Step 7: *(Wind Power Curve):* Transform the wind speed into its corresponding wind power generation. We follow the method in [33] and [63]. The aggregate power curve for each wind farm is used moving forward; this curve can be estimated using observed wind power-wind speed pairs from wind farms.

2.1.2 Discussion of the Methodology

The output of the procedure described above is a multidimensional envelope, $\mathbf{E}(t_0, \tau)$ $(\tau \in \Xi_H)$. It encompasses the vast majority of possible joint realizations of wind power, solar power, and uncertain load looking ahead τ units of time later, as seen from the current time t_0. Connecting the elements of $\mathbf{E}(t_0, \tau)$ at all τ results in the spatio-temporal flexibility requirement envelope $\mathbf{E}(t_0)$. Its shape may change when t_0 moves forward due to the non-stationary nature of renewable generation and load.

The envelopes in Figure 2.2 show an example where y_{n_1} and y_{n_2} represent the power outputs of two uncertainty sources located at buses n_1 and n_2. They could be wind power,

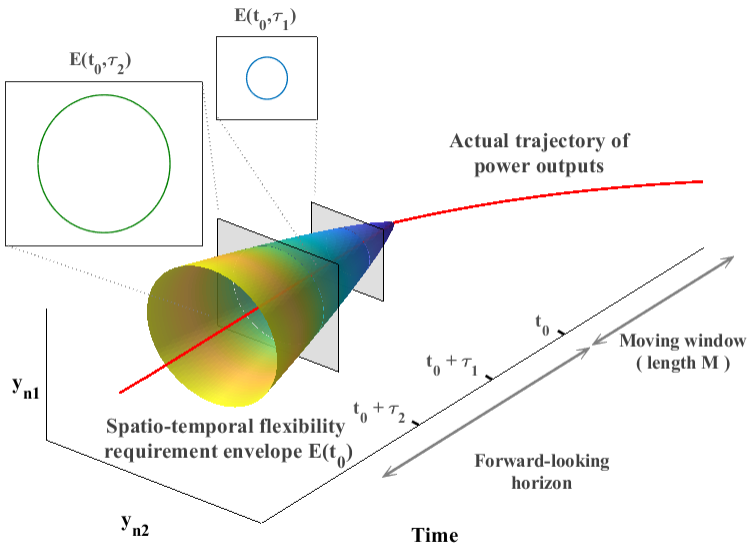

Figure 2.2: Spatio-temporal flexibility requirement envelope obtained at t_0.

solar power or uncertain load. The envelope is computed at present time t_0 ($\tau = 0$) over a forward-looking horizon of length T_H future time steps. The time window with length M for detrending is located ahead of t_0. By inspection of $\mathbf{E}(t_0, \tau_1)$ and $\mathbf{E}(t_0, \tau_2)$ in Figure 2.2, we can find that the size of the envelope increases with τ. This is because the uncertainty in forecast increases with the look-ahead time.

In the proposed methodology, Step 1 extracts and removes the temporal trends of the input time series. The temporal trends are removed because the application of PCA in Step 2 requires that the input time series be stationary. The removed trends are re-incorporated into the envelope in Step 6.

With (2.2), wind power and load may display diurnal, seasonal and other cyclical trends. The resulting stationarity of the detrended time series $\omega'_n(t)$ is assessed by the augmented Dickey-Fuller (ADF) test [63]. A proper value of the window size M should be determined so that the resulting time series pass the ADF and resulting time series are stationary. Empirical results (see Section 2.3.2) show that M has an impact on the predictive performance of the spatio-temporal envelope and its performance in power system energy management.

Step 2 captures the spatial correlation of all the input time series. The spatial dependency is removed in this step, and the resulting principal components are independent of each other. Thus, there is no need to consider the impact of other PCs when we build flexibility

requirement envelopes for each PC in Step 3, and when we combine them in Step 4. This is a feature which greatly simplifies the process. The spatial dependency structure is re-inserted into the envelope in Step 5. This guarantees that the spatio-temporal flexibility requirement envelopes capture the spatial correlation between all of the input time series.

2.1.3 Projection

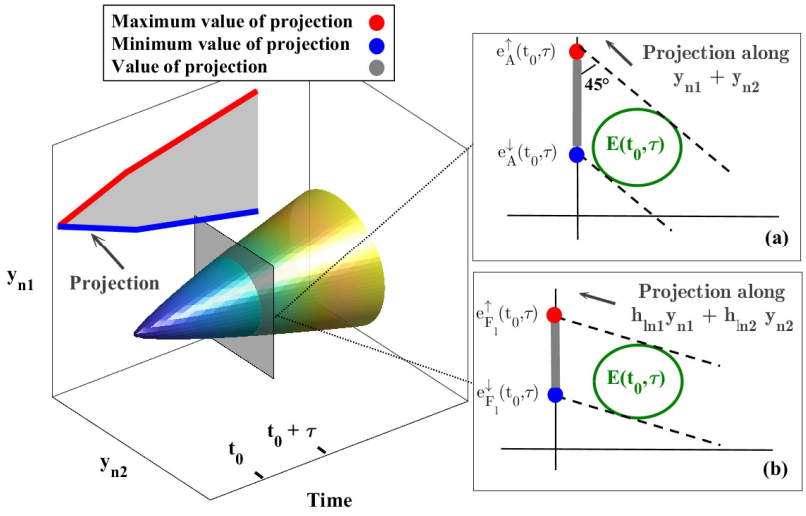

Figure 2.3: Projection of spatio-temporal flexibility requirement envelope. Connecting the maximum (red point) and minimum (blue point) values of the projection at all τ and we will get the upward branch (red line) and downward branch (blue line) of a two dimensional projected envelope.

When carrying out power system unit commitment and dynamic dispatch, we are interested in quantifying the impact of the uncertainties of renewable generation and loads on the system net load and on the flows in transmission lines. This information is necessary to pre-position dispatchable resources to guarantee secure and economic operation. In this section, we show that the impact of uncertainties on the net load, which translates to transmission line flows, can be quantified by projecting spatio-temporal flexibility requirement envelopes. This idea is illustrated schematically in Figure 2.3.

From Figure 2.3 (a), we find that the maximum and minimum values of the aggregate output of y_{n_1} and y_{n_2} can be obtained by maximizing and minimizing the projection of

$\mathbf{E}(t_0, \tau)$ onto the vertical axis along the direction of $y_{n_1} + y_{n_2}$. If we connect the maximum and the minimum of the projection at all τ, we will get a two-dimensional envelope $\{e_A^{\uparrow}(t_0, \tau), e_A^{\downarrow}(t_0, \tau)\}_{\tau \in \Xi_H}$ which encompasses the vast majority (e.g., 95 %) of possible realizations of $y_{n_1} + y_{n_2}$ for $\tau \in \Xi_H$, as seen from the present time t_0. Mathematically, the projected envelope $\{e_A^{\uparrow}(t_0, \tau), e_A^{\downarrow}(t_0, \tau)\}_{\tau \in \Xi_H}$ is obtained moving forward in $\tau \in \Xi_H$ by solving:

$$e_A^{\uparrow}(t_0, \tau) = \max_{\mathbf{Y} \in \mathbf{E}(\mathbf{t_0}, \tau)} (\mathbf{1} \cdot \mathbf{Y}) \tag{2.6}$$

$$e_A^{\downarrow}(t_0, \tau) = \min_{\mathbf{Y} \in \mathbf{E}(\mathbf{t_0}, \tau)} (\mathbf{1} \cdot \mathbf{Y}) \tag{2.7}$$

Here, $\mathbf{Y}$ is the column vector consisting of active power injections from the uncertainty sources (including wind power, solar power and uncertain loads) at each bus. $\mathbf{1}$ is a row vector of ones of the same length as $\mathbf{Y}$.

At the same time, the impact of y_{n_1} and y_{n_2} on the power flow in transmission line l is found by calculating $h_{ln_1} y_{n_1} + h_{ln_2} y_{n_2}$ whose maximum/minimum values can also be quantified by projection (along the direction of $h_{ln_1} y_{n_1} + h_{ln_2} y_{n_2}$), as is shown in Figure2.3 (b)). Here, h_{ln_1} and h_{ln_2} are Power Transfer Distribution Factors (PTDF) describing the sensitivities of power flow on line l to the changes of active power injection at buses n_1 and n_2. Again, by connecting the maximum and the minimum values of the projection at all τ, we can get a two-dimensional envelope $\{e_{F_l}^{\uparrow}(t_0, \tau), e_{F_l}^{\downarrow}(t_0, \tau)\}_{\tau \in \Xi_H}$ that encompasses the vast majority of possible realizations of $h_{ln_1} y_{n_1} + h_{ln_2} y_{n_2}$ for the forward-looking horizon, as seen from t_0. The projected envelope $\{e_{F_l}^{\uparrow}(t_0, \tau), e_{F_l}^{\downarrow}(t_0, \tau)\}_{\tau \in \Xi_H}$ is computed moving forward in $\tau \in \Xi_H$ as follows

$$e_{F_l}^{\uparrow}(t_0, \tau) = \max_{\mathbf{Y} \in \mathbf{E}(\mathbf{t_0}, \tau)} (\mathbf{H}_l \cdot \mathbf{Y}) \tag{2.8}$$

$$e_{F_l}^{\downarrow}(t_0, \tau) = \min_{\mathbf{Y} \in \mathbf{E}(\mathbf{t_0}, \tau)} (\mathbf{H}_l \cdot \mathbf{Y}) \tag{2.9}$$

The row vector $\mathbf{H}_l$ consists of PTDFs that describe the change of the power flow in line l with respect to the active power injection at each bus.

Figure 2.4 shows the projected envelope $\{e_A^{\uparrow}(t_0, \tau), e_A^{\downarrow}(t_0, \tau)\}_{\tau \in \Xi_H}$. It aggregates the output uncertainty arising from each uncertainty source and encompasses the vast majority of their possible realizations moving forward from present time t_0. The slope of this

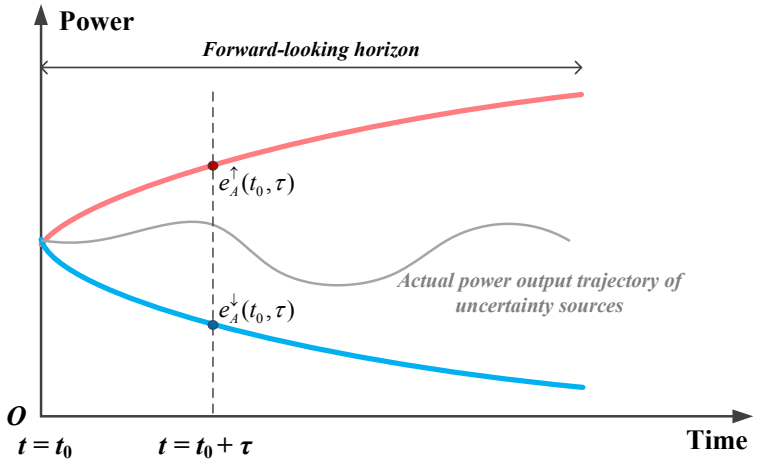

Figure 2.4: Schematic illustration of the projected envelope $\{e_A^{\uparrow}(t_0,\tau), e_A^{\downarrow}(t_0,\tau)\}_{\tau\in\Xi_H}$.

envelope encompasses the vast majority of possible transitions of the aggregate output trajectory of all the uncertainty sources, which is associated with reserve ramping requirements. At the current time t_0 where $\tau = 0$, the envelope collapses to the current operating point: $e_A^{\uparrow}(t_0,0) = e_A^{\downarrow}(t_0,0) = \sum_n (d_n(t_0) - \delta_n^w(t_0) - \delta_n^s(t_0))$. Also, the area between the envelope and the time axis implicitly indicates the operating reserve needed in the form of energy.

It is worth noting that $\{e_{F_l}^{\uparrow}(t_0,\tau), e_{F_l}^{\downarrow}(t_0,\tau)\}_{\tau\in\Xi_H}$ has a shape similar to the envelope in Figure 2.4. Because of the variability and uncertainty of renewable generation and loads, transmission line flows can deviate from their scheduled flow. The flexibility requirement envelope associated with transmission lines $\{e_{F_l}^{\uparrow}(t_0,\tau), e_{F_l}^{\downarrow}(t_0,\tau)\}_{\tau\in\Xi_H}$ encompasses the vast majority of possible transmission line flow deviations from the scheduled power flows at the current time t_0. Because transmission lines all have limited capacities, the system operator needs to procure enough operating reserve (from dispatchable generation and responsive demand) to hedge also against possible future congestion. Moreover, at t_0 ($\tau = 0$), we also find that $e_{F_l}^{\uparrow}(t_0,0) = e_{F_l}^{\downarrow}(t_0,0) = \sum_n h_{ln} (d_n(t_0) - \delta_n^w(t_0) - \delta_n^s(t_0))$, the line flow in line l at the current time t_0.

2.2 Spatio-temporal Operational Planning of Operating Reserve

2.2.1 Mathematical Program

The following mathematical program is developed for receding-horizon power system unit commitment and dynamic dispatch where the envelopes quantifying spatio-temporal flexibility requirements are used to drive the current and forward-looking decisions. The variables $(\cdot)(\tau, q)$ $(\tau \in \Xi_H,\ q \in \{\uparrow, \downarrow\})$ are forward-looking decisions, as driven by the flexibility requirement envelopes $\{e_A^{\uparrow}(t_0,\tau), e_A^{\downarrow}(t_0,\tau)\}_{\tau\in\Xi_H}$ and $\{e_{F_l}^{\uparrow}(t_0,\tau), e_{F_l}^{\downarrow}(t_0,\tau)\}_{\tau\in\Xi_H}$.

Objective

The objective (2.10) is to minimize the cost of dispatchable generation $(C_i(\cdot))$ at the current time t_0 $(\tau = 0)$, while considering the forward-looking horizon $t_0 + \tau$ for $\Xi_H = \{0, 1, 2, \cdots, T_H\}$, where T_H is the length of horizon. We perform this minimization also considering the flexibility requirement envelope directions, $\Xi_q = \{\uparrow, \downarrow\}$, penalties for load and renewable generation curtailment π^d and π^δ and a discounting factor $\gamma(\tau) \leq 1$ which weighs the forward-looking steps against the current-time costs. $\gamma(\tau)$ also reflects that near-term flexibility is more valuable than long-term flexibility [6].

$$\min \sum_{\tau\in\Xi_H, q\in\Xi_q} \gamma(\tau) \left[\sum_{i\in\Xi_g} C_i(p_g^i(\tau,q), u_g^i(\tau,q)) + \sum_{n\in\Xi_n} \left(\pi^d d_c^n(\tau,q) + \pi^\delta \delta_c^n(\tau,q) \right) \right] \tag{2.10}$$

The optimization is subject to:

Power balance for up-going and down-going envelopes $q \in \Xi_q$ and all forward-looking times $\tau \in \Xi_H$

$$e_A^q(t_0,\tau) - \sum_{i\in\Xi_g} p_g^i(\tau,q) - \sum_{j\in\Xi_e} \left(p_e^{j+}(\tau,q) - p_e^{j-}(\tau,q) \right) + \sum_{n\in\Xi_n} \left(\delta_c^n(\tau,q) - d_c^n(\tau,q) \right) = 0 \tag{2.11}$$

Transmission capacity constraints of line l for up-going and down-going envelopes $q \in \Xi_q$ and all forward-looking times $\tau \in \Xi_H$

$$-f_l^{\max} - e_{F_l}^q(t_0,\tau) \leq f_l(\tau,q) \leq f_l^{\max} - e_{F_l}^q(t_0,\tau) \tag{2.12}$$

where

$$f_l(\tau,q)=\sum_{n\in\Xi_n}h_{ln}\Big[\sum_{i\in\mathcal{N}(n)}p_g^i(\tau,q)+\sum_{j\in\mathcal{N}(n)}\Big(p_e^{j+}(\tau,q)-p_e^{j-}(\tau,q)\Big)+d_c^n(\tau,q)-\delta_c^n(\tau,q)\Big] \quad (2.13)$$

and $f_l^{\max}$ is the maximum allowed power flow in line l. The summations over $\mathcal{N}(n)$ are used to map the locations of generators and storage assets onto nodes of the network.

Bounds on load and renewable generation curtailment at bus $n\in\Xi_n$ for $q\in\Xi_q$ and $\tau\in\Xi_H$

$$0\le d_c^n(\tau,q)\le d_n(t_0) \quad (2.14)$$

$$0\le \delta_c^n(\tau,q)\le \delta_n^w(t_0)+\delta_n^s(t_0) \quad (2.15)$$

Consistency of dispatch decisions and envelope tracking at current time t_0 ($\tau=0$)

We note that at the current time t_0, by necessity, we require

$$p_g^i(t_0)=p_g^i(0,\uparrow)=p_g^i(0,\downarrow) \quad (2.16)$$

$$p_e^j(t_0)=p_e^{j+}(0,\uparrow)-p_e^{j-}(0,\uparrow)=p_e^{j+}(0,\downarrow)-p_e^{j-}(0,\downarrow) \quad (2.17)$$

$$s_e^j(t_0)=s_e^j(0,\uparrow)=s_e^j(0,\downarrow) \quad (2.18)$$

Capacity and ramp constraints on dispatchable generation $i\in\Xi_g$ for $q\in\Xi_q$ and $\tau\in\Xi_H$

$$u_g^i(\tau,q)\underline{p_g^i}\le p_g^i(\tau,q)\le u_g^i(\tau,q)\overline{p_g^i} \quad (2.19)$$

$$p_g^i(\tau-1,q)-p_g^i(\tau,q)\le r_{g\mathrm{dn}}^i u_g^i(\tau,q)+r_{g\mathrm{sd}}^i(u_g^i(\tau-1,q)-u_g^i(\tau,q))+\overline{p_g^i}(1-u_g^i(\tau-1,q)) \quad (2.20)$$

$$p_g^i(\tau,q)-p_g^i(\tau-1,q)\le r_{g\mathrm{up}}^i u_g^i(\tau-1,q)+r_{g\mathrm{su}}^i(u_g^i(\tau,q)-u_g^i(\tau-1,q))+\overline{p_g^i}(1-u_g^i(\tau,q)) \quad (2.21)$$

Capacity and energy constraints on storage $j\in\Xi_e$ for $q\in\Xi_q$ and $\tau\in\Xi_H$

$$0\le p_e^{j+}(\tau,q)\le u_e^{j+}(\tau,q)\overline{p_e^j} \quad (2.22)$$

$$0\le p_e^{j-}(\tau,q)\le -u_e^{j-}(\tau,q)\underline{p_e^j} \quad (2.23)$$

$$\underline{s_e^j} \leq s_e^j(\tau, q) \leq \overline{s_e^j} \tag{2.24}$$

$$s_e^j(\tau, q) = s_e^j(\tau - 1, q) - p_e^{j+}(\tau, q) T_\Delta (\eta_d^j)^{-1} + p_e^{j-}(\tau, q) T_\Delta \eta_c^j \tag{2.25}$$

$$u_e^{j+}(\tau, q) + u_e^{j-}(\tau, q) = 1 \tag{2.26}$$

2.2.2 Discussion

Receding horizon operation

At the current time, t_0, the spatio-temporal flexibility requirement envelope and its projected envelopes are refreshed when the random process of renewable generation or uncertain load are observed. The energy scheduling problem in (2.10)–(2.26) is solved looking forward in time $\tau \in \Xi_H$. Although we are looking forward in time, only the scheduling decisions for the current time step t_0 (i.e., $\tau = 0$) is applied, while the forward-looking decisions, as driven by the flexibility requirement envelopes, are there to ensure the current-time decisions will be consistent with what *may happen* moving forward. The scheduled trajectory of dispatchable units is assumed to remain constant between each time step $t_0 \to t_0 + 1$.

If the system does not have enough flexibility to satisfy all the flexibility requirements envelopes at t_0 and the various look ahead time steps $\tau \in \Xi_H$, the system operator may have to deploy slack resources (i.e., d_c^n and δ_c^n). Slack resources are priced much higher than other dispatchable resources. Thus, these are used as the last resort when the energy scheduling problem becomes infeasible. At the beginning of next time step $t_0 + 1$, the above process is repeated, with a re-calculation of the envelopes using the latest observations of the load and renewable generation.

Quantifying the flexibility of dispatchable units

In this chapter, we consider two kinds of system flexibility sources: conventional generators and energy storage. Constraints (2.19)–(2.21) characterize the flexibility of each conventional generator in terms of power capacity and ramp rate limits. The power and energy capacity limit of storage are modeled in (2.22)–(2.23) and (2.24)–(2.26), respectively.

Deploying operating reserves

The operating reserves are planned implicitly in (2.11) and (2.12) by satisfying the flexibility requirements with flexibility sources. Specifically, (2.11) deploys the flexibility from each dispatchable unit to maintain power balance for all $\tau \in \Xi_H$ and $q \in \Xi_q$. As described in (2.11), power balancing will be maintained when the aggregate response of dispatchable units is capable of tracking all the possible realizations of net load trajectory in the forward-looking horizon, which in bounded in the envelope $\{e_A^{\uparrow}(t_0,\tau), e_A^{\downarrow}(t_0,\tau)\}_{\tau\in\Xi_H}$. Constraints in (2.12) take the transmission limits into account. Similarly, the system security will be guaranteed if the aggregate response of dispatchable units can handle all the possible transmission line flow deviations in the forward-looking horizon, which is bounded in the envelope $\{e_{F_l}^{\uparrow}(t_0,\tau), e_{F_l}^{\downarrow}(t_0,\tau)\}_{\tau\in\Xi_H}$.

The spatio-temporal flexibility requirement envelope addresses the secondary and tertiary reserve in the power system. Specifically, the first 10 minutes of the envelope in the forward-looking horizon quantifies the requirements of secondary reserve, while the rest of the envelope addresses the tertiary reserve. [1]

2.3 Case Studies

2.3.1 Simulation Setup

We illustrate the application of the spatio-temporal flexibility management approach and its use in generation scheduling problems in a microgrid and a larger transmission system.

Time Resolution

The length of each forward-looking time step is $T_\Delta = 5$ minutes, and the length of the receding horizon $T_H = 12$ (for a total forward-looking time of 60 minutes) at every scheduling time t_0.

[1] Please note that the spatio-temporal flexibility requirement envelope is not addressing the primary reserve or the contingency services.

Energy Scheduling Strategies

The mathematical program described in (2.10)–(2.26) is used to simulate our proposed energy scheduling strategy, which is a robust approach. The proposed spatio-temporal flexibility requirement envelope (STFRE) is used to model the uncertainty set.

We compare the proposed method with three robust methods, which have different uncertainty set characterization approaches

- *Polyhedral uncertainty set (POLYH):* The first uncertainty set is the prediction polyhedron proposed in [35].
- *Ellipsoidal uncertainty set (ELLIP):* The second uncertainty set is the ellipsoidal prediction region in [32].
- *Multivariate Prediction Intervals (MPI):* We also compare our proposed method with the MPI, which is a box shaped uncertainty set discussed in [35] and [36]. [2]

In [32] and [35] the parameters of POLYH and ELLIP are also updated using a moving window-based method to identify the trends of the input data, whereas the temporal trends are ignored in MPI. The authors of [32], [35] and [36] did not apply these uncertainty sets to power system energy scheduling. Thus, they will also be simulated using (2.10)–(2.26).

Performance Metrics

The energy scheduling results are assessed by three performance metrics: total generation cost (TGC), energy not served (ENS) and energy curtailed (EC)

$$TGC = \sum_{t} \sum_{i \in \Xi_g} C_i(p_g^i(t), u_g^i(t)) \tag{2.27}$$

$$ENS = T_\Delta \sum_{t} \sum_{n \in \Xi_n} d_c^n(t) \tag{2.28}$$

$$EC = T_\Delta \sum_{t} \sum_{n \in \Xi_n} \delta_c^n(t) \tag{2.29}$$

[2]The benchmark method "Multivariate Prediction Intervals (MPI)" is the multivariate version of the work proposed in [6]. In each dimension of MPI, the boundaries are obtained using the method presented in [6].

Moreover, the predictive performance of STFRE, POLYH, ELLIP and MPI are examined for different look-ahead time τ based on two metrics [33], [35]:

- *Calibration:* This measures the deviations between the observed and nominal coverage rates of the uncertainty set at look ahead time τ. In this chapter, the nominal coverage rates are all set to 95%. The observed coverage rate at τ is obtained by calculating the ratio of realizations after τ time steps which fully lie within the uncertainty set in the whole simulation period.
- *Volume:* The volume (size) of the uncertainty set V_τ at different look-ahead time τ is obtained using the Monte Carlo based method from [35]. The idea is to generate N' uniformly distributed random samples in a hyper-cube with edges of length equal to the maximum capacity of all the wind power/solar power/uncertain load. Then count the number of samples N'' that lie in the uncertainty set at τ. The volume of the uncertainty set at τ is calculated by $V_\tau = N''V_C/N'$, where V_C is the volume of the corresponding hyper-cube.

Computing Platform

All the models are solved using CPLEX on a GAMS platform [65]. The computer used is equipped with an Intel Core i5/ 3.10 GHz processor and 8 GB RAM. [3]

2.3.2 Microgrid

This first case study showcases our proposed approach in a simple grid-connected microgrid, whose single-line diagram is shown in Figure 2.5. This microgrid includes a critical load, a curtailable load, a wind farm, a solar farm, a diesel generator and storage.

For the diesel generator: $\overline{p_g} = 600$ kW, $\underline{p_g} = 60$ kW, $r_{g\text{su}} = 100$ kW/5 min, $r_{g\text{sd}} = 600$ kW/5 min, $r_{g\text{up}} = r_{g\text{dn}} = 100$ kW/5 min. We assume quadratically-varying costs $C(p_g(t), u_g(t)) = a_g p_g(t)^2 + b_g p_g(t) + c_g u_g(t)$. From [42], we take $a_g = 1.52 \times 10^{-5}$ l/kW^{2}h, $b_g = 0.02186$ l/kWh and $c_g = 41.6$ l/h. Cost of diesel fuel = \$1.30/l. For the storage:

[3] Please note that most of steps associated with constructing the spatio-temporal flexibility requirement envelope (e.g., Step 1 – Step 4 in section 2.1.1) can be done offline using a computer. The online operation of the spatio-temporal envelope only includes Step 5 – Step 7, which is computationally light and can be easily implemented in a MG controller with limited computational power.

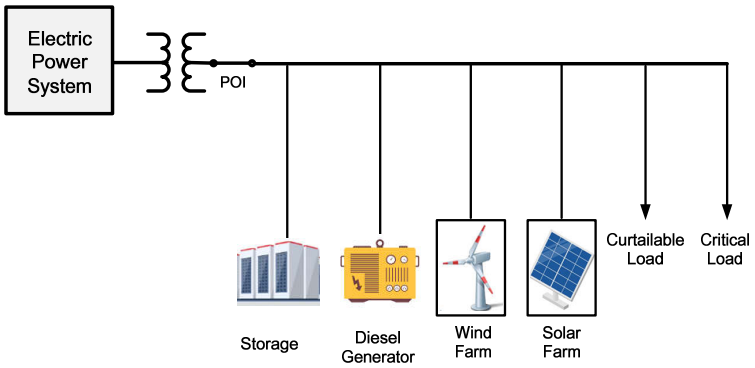

Figure 2.5: Diagram of microgrid under study.

$\overline{p_e} = 100$ kW, $\underline{p_e} = -100$ kW, $\overline{s_e} = 90$ kWh, $\underline{s_e} = 10$ kWh, $\eta_c = \eta_d = 0.963$, $\pi^d = \pi^\delta = 100$ \$/kWh. The minimum and maximum values of curtailable load are 150 kW and 450 kW, respectively. The critical load is constant at 50 kW. The maximum power output of the wind farm and the solar farm are both 125 kW. The power exchange requirement at the point of interconnection (POI) is kept constant throughout the year. We set this requirement to be the minimal net load of any given day.

Uncertainties come from wind generation, solar generation and curtailable load, for which we consider a full year worth of data records. The curtailable load data comes from a village in the Canadian province of Quebec. The wind and solar power data is taken from [66]. Network constraints are ignored here. The length of the time window M of curtailable load and wind power are set to be one hour and 24 hours, respectively. The discount rate $\gamma(\tau)$ is assumed to be one for all τ. Figure 2.6 shows the evolution of curtailable load, wind generation and solar generation in the first two weeks of January, which displays their periodic patterns. The time series are geographically independent from each other. Thus, in this case study we only focus on the temporal trends of wind power, solar power and curtailable load.

Figure 2.7 (a) and (b) shows the projected envelopes for the load generated by different uncertainty sets on two different days and times (marked with L_1 and L_2 in Figure 2.6 (a)). We observe that the envelopes generated by MPI are identical. This is because MPI ignores the temporal trends of curtailable load. However, the envelopes generated by STFRE, POLYH and ELLIP on January 1st have much smaller sizes than those of MPI, while on January 6th they have similar sizes. The reason behind this observation can be found in

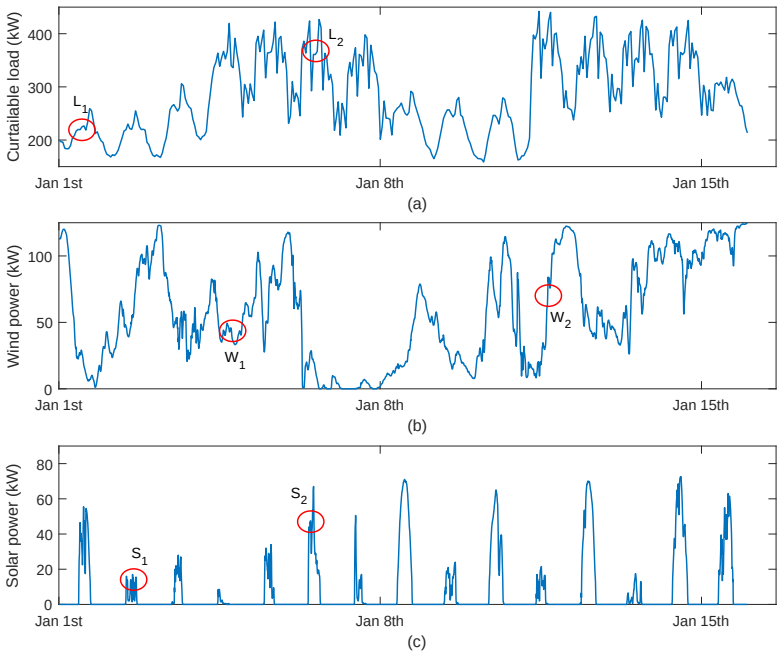

Figure 2.6: Power profile of (a) curtailable load, (b) wind generation and (c) solar generation in the first two weeks of January.

Figure 2.6 (a). The curtailable load is more fluctuating on January 6th than on January 1st. The STFRE, POLYH and ELLIP capture this feature and adjust the size of their envelopes. Though the projected envelopes generated by STFRE, POLYH and ELLIP are sometimes smaller in size, it can be found that the load trajectory always stays inside their envelopes. Similar phenomena can be observed for wind power in Figure 2.7 (c) and (d). The projected envelopes for solar power at two different times (marked with S_1 and S_2 in Figure 2.6 (c)) are shown in Figure 2.7 (e) and (f). It can be observed that the envelopes generated by STFRE are always kept under the theoretical maximum of solar power (solar power under clear-sky irradiance), which is changing all the time. However, the envelopes generated by POLYH, ELLIP and MPI cannot capture this feature, and their envelopes go beyond the theoretical maximum at 11:45 in Figure 2.7 (f), which should be impossible for solar power.

Figure 2.8 (a) reports the deviations between the observed coverage rates and nominal coverage rates of different uncertainty sets for different forward-looking times. We can see that STFRE, POLYH and ELLIP maintain a coherent calibration for all the look-ahead times, while the deviation associated with MPI is higher. Figure 2.8 (b) compares the

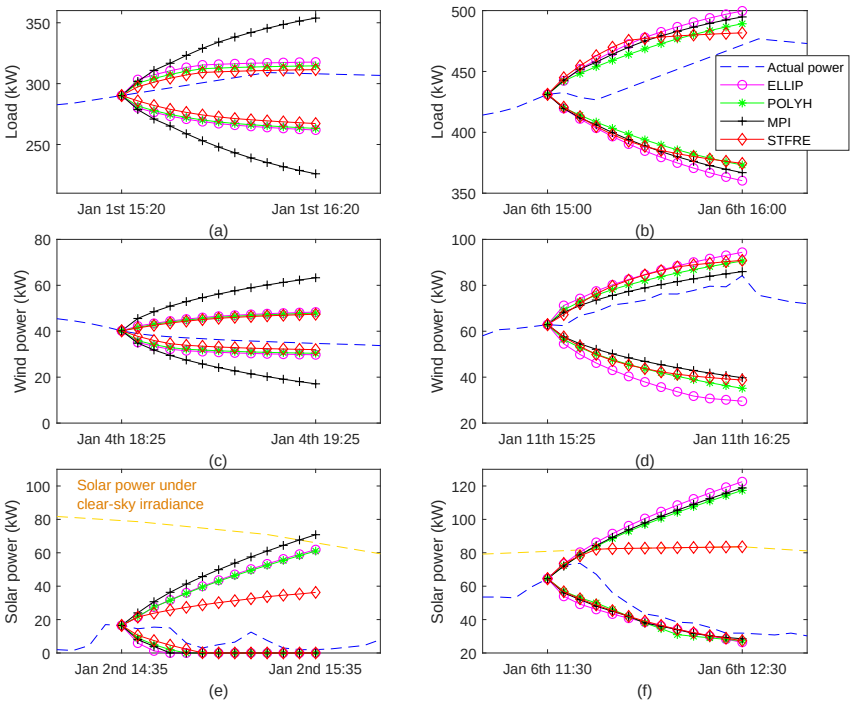

Figure 2.7: Projected envelopes for load/wind/solar generated by STFRE, ELLIP, POLYH and MPI. The load/wind/solar is more fluctuating in (b), (d) and (f). The actual power trajectories are kept inside the envelopes.

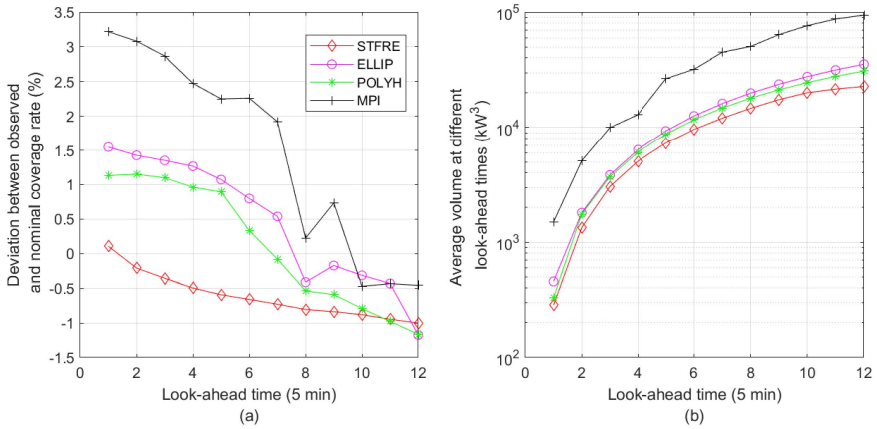

Figure 2.8: (a) Calibration results (b) Average volume of uncertainty sets at different look-ahead times in the whole simulation period (microgrid).

volumes of different uncertainty sets. As expected, MPI has the highest volume, while STFRE has the lowest volume, which all increase with the look-ahead time.

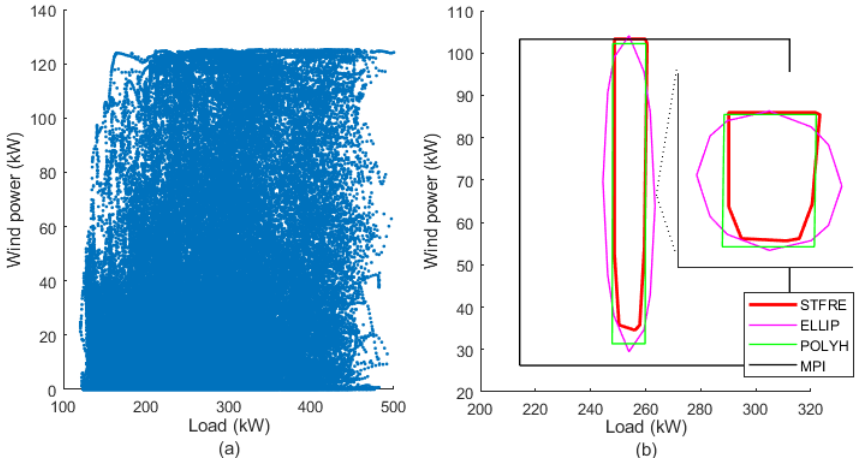

Figure 2.9: (a) Scatter plot of load and wind power. (b) The projection of uncertainty sets in the load-wind plane (generated at 21:20 on January 1st). The STFRE, ELLIP and POLYH are zoomed in a windnow on the right.

We can see that STFRE shows little advantage in comparison to POLYH and ELLIP in this case study. This can be explained by Figure 2.9. The scatter plot clearly illustrates the spatial independence between load and wind power. STFRE has a shape similar to POLYH. The size of STFRE, POLYH and ELLIP are well adjusted using a moving window, while MPI has a much larger size by ignoring temporal trends.

Table 2.1: Microgrid Dispatching Strategies' Performance

Uncertainty Sets	**Performance metrics**		
	TGC (k$)	**ENS (kWh)**	**EC (kWh)**
STFRE	334.1	0.03	0
POLYH	336.4 (+0.69%)	0.03	0
ELLIP	338.3 (+1.26%)	0.02	0
MPI	349.6 (+4.64%)	0.02	0

Next we look into the performance of the different uncertainty set characterizations when

carrying out energy dispatching for the microgrid. Table 2.1 displays the performance of energy scheduling based on the different uncertainty sets for an entire year of operation. The percentage changes in cost are taken with respect to the STFRE. All strategies have a good performance in terms of ENS and EC, but there are cost differences across strategies. The proposed STFRE has the lowest cost, while MPI has the highest one.

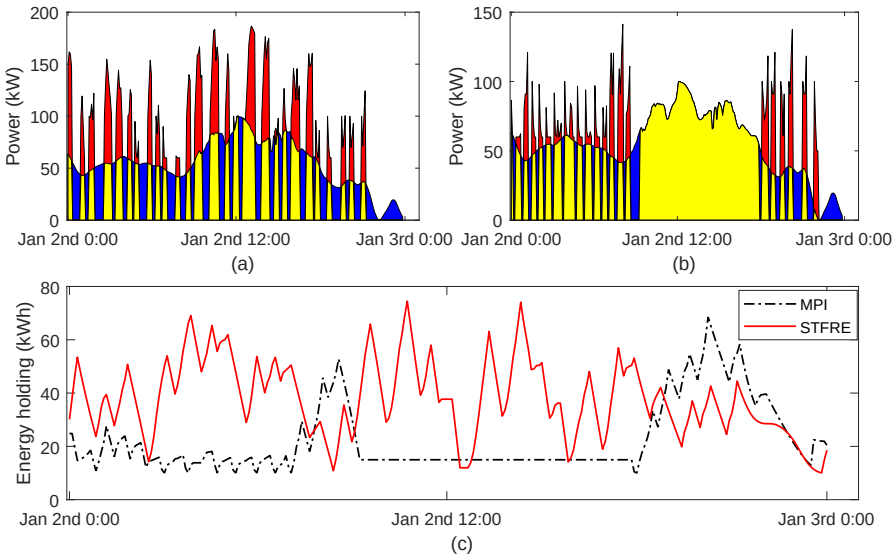

Figure 2.10: Daily evolution of the energy scheduling results associated with (a) STFRE, (b) MPI. Blue areas indicate load supply by storage, and yellow areas indicate load supply by the diesel generator. Red areas indicate diesel generator use to charge the storage asset. (c) Daily evolution of energy holding in storage.

A daily evolution of the energy scheduling results associated with STFRE and MPI on January 2nd can be seen in Figure 2.10. It can be observed that the diesel generator tends to charge the storage under STFRE. The storage is dispatched to serve the net load alone when its energy holding is high and the diesel generator is turned off. This leads to significant decrease in TGC. We can also find that the storage under MPI is not charged as frequently as with STFRE. As a result, the storage energy holding under MPI is always lower than with STFRE. Thus, the flexibility of the storage is not exploited as well by MPI in comparison to STFRE. This contributes to drive the cost higher. The cost associated with POLYH and ELLIP are just slightly higher than with STFRE.

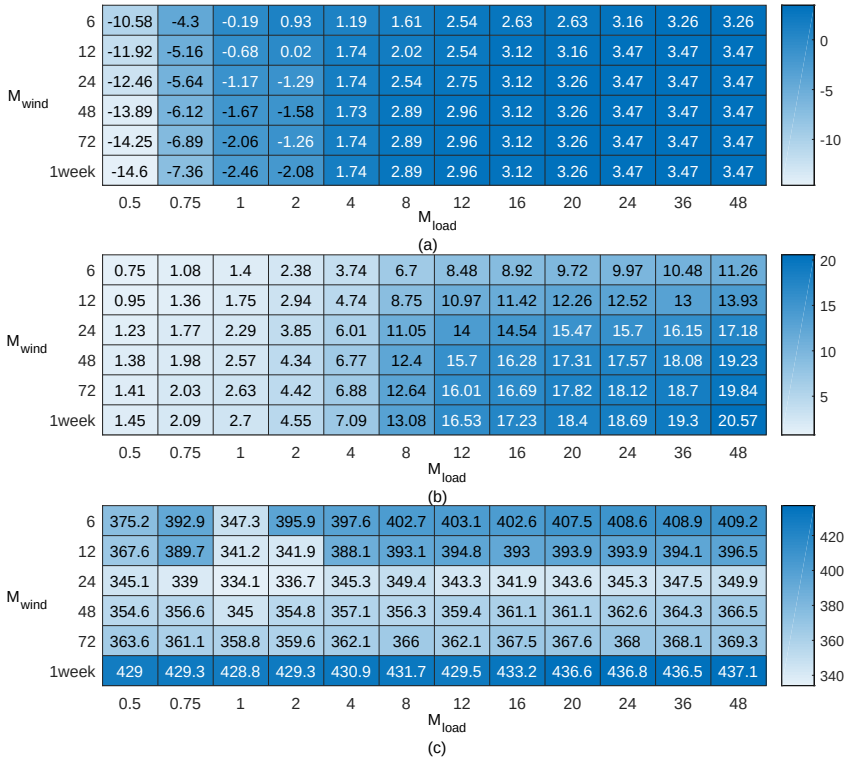

Figure 2.11: Effect of M_{load} and M_{wind} on the (a) calibration errors (%) (b) Average volume of STFRE at 60 minute look-ahead time ($\times 10^4$ kW^3) and (c) TGC (k$) of STFRE

Figure 2.11 shows how the calibration, volume and TGC of STFRE varies with the moving window size of load (M_{load}) and wind (M_{wind}). We observe that these performance metrics are more sensitive to M_{load}. Initially, TGC has a sharp decline with the increasing of M_{load} until M_{load} is equal to one hour, where the calibration errors also become quite small. The subsequent increase in TGC as M_{load} grows is indicative of the fact that the volume of STFRE keeps increasing rapidly while the observed coverage rate saturates. Finally, we choose M_{load} and M_{wind} to be one hour and 24 hour, where the TGC is minimal and the calibration error is close to zero. The corresponding detrended time series of load and wind all pass the ADF test with these window length choices.

In Figure 2.12 we validate the effectiveness of the proposed STFRE with one minute resolution datasets of renewable sources, which are taken from [67] and [68]. In general, datasets with higher resolution are able to capture the more volatile and rapid changes in

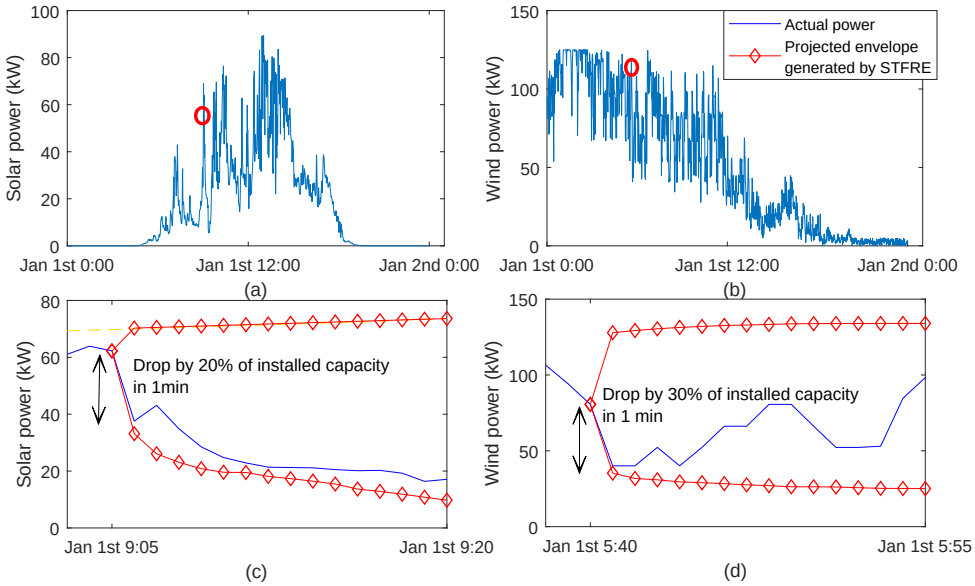

Figure 2.12: Power profile of (a) solar generation and (b) wind generation on January 1st with one minute resolution. Projected envelope for (c) solar power and (d) wind power generated by STFRE.

solar and wind power generation. For example, in Figure 2.12 (c) and (d) the solar power and wind power have dropped by 20% and 30% of their installed capacities within one minute. Still the STFRE is able to capture these extreme variabilities successfully and bound the actual power trajectories inside the envelope. Here the forward-looking time step is set to one minute. Also, we only show a look-ahead horizon of 15 minute because the envelope saturates early. The calibration error for one, 15 and 60 minute look-ahead times are -1.05%, -3.16% and -4.21% respectively. These are within high and acceptable levels of accuracy.

Fig. 2.13 shows how the discounting factor $\gamma(\tau)$ of STFRE influences the dispatch performance of our proposed strategy. It is assumed that $\gamma(\tau) = \Gamma^{\tau}$. The proposed STFRE gets the lowest cost when Γ is between 0.9 and 1. However, there is ENS and EC when Γ is lower than 1. Finally, we choose $\Gamma = 1$ considering all the three performance metrics.

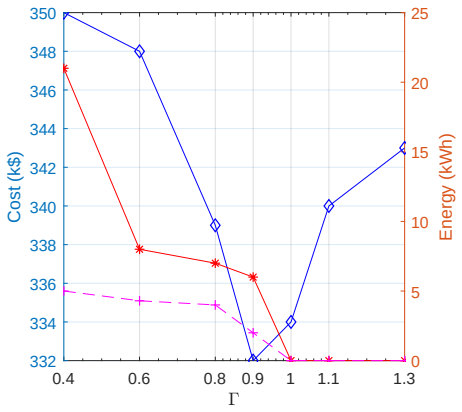

Figure 2.13: Sensitivity of the TGC, ENS and EC associated with the STFRE to the discounting factor Γ. The blue line represents the TGC. The red line and the magenta line correspond to the ENS and EC, respectively.

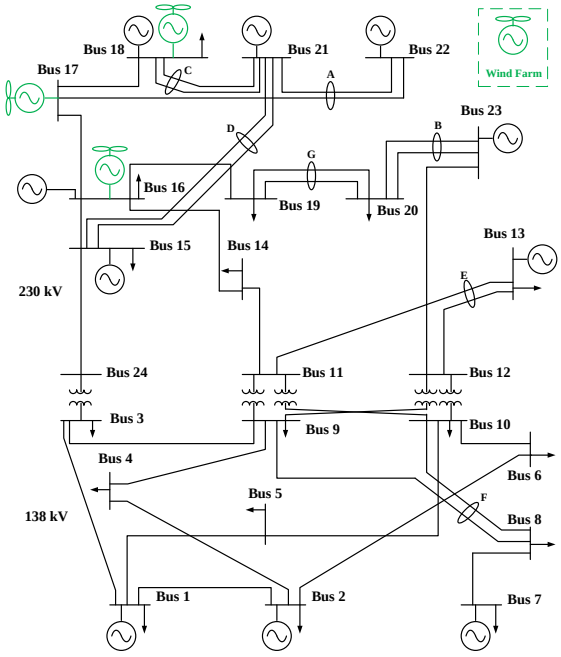

Figure 2.14: Diagram of the modified IEEE RTS system. The units of power, ramp and energy in this system are MW, MW/5min and MWh.

2.3.3 Transmission System

Next, our proposed approach is tested on a modified IEEE Reliability Test System (RTS), as shown in Figure 2.14. The data of the network, the load profile, generating unit ramp rates, minimum and maximum power outputs, minimum up and down times are all found in [69]. For convenience, start-up and shut-down ramps of units are set to be their minimum and maximum power outputs over 5 minutes, respectively. We also assume that the nuclear (U400) and hydro (U50) generators are must-run units as in [19]. The value of lost load or curtailing renewable generation is equal to $2000 per megawatt-hour. There are three wind farms each with the same power capacity, and they are installed at buses 16, 17 and 18. Wind data is taken from [66]. In Figure 2.14 we ignore the uncertainty of loads, and we assume that there are no storage assets present.

We run the energy scheduling strategies with different uncertainty sets for comparison. Here, the moving windows for all the wind sources have the same length set to 24 hours. Since temporal trends have already been discussed in the last section, here we focus more specifically on the spatial correlation between the wind farms. In this section, a whole year of operation is considered. A unit commitment for the next 24 hours is executed at the beginning of each hour. The unit commitment is held fixed for each hour, and a receding horizon economic dispatch runs for every $T_\Delta = 5$ minutes.

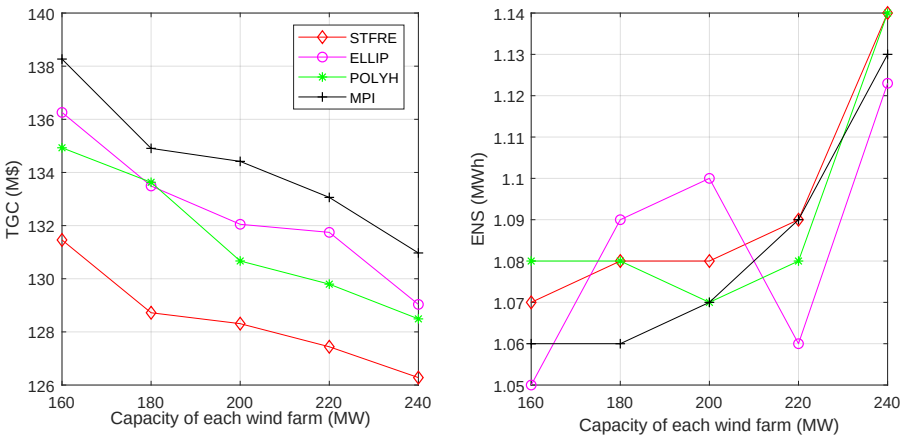

Figure 2.15: TGC and ENS of energy scheduling strategies with different uncertainty sets in one year's operation with different wind farms capacities.

Figure 2.15 displays the TGC and ENS of energy scheduling strategies as we vary

simultaneously the three wind farm capacities. As expected, STFRE always has the lowest TGC, while MPI has the highest costs again. At the same time, the differences in terms of ENS between the strategies are negligible. Please note that none of the strategies use wind curtailment; therefore, no results for EC are given.

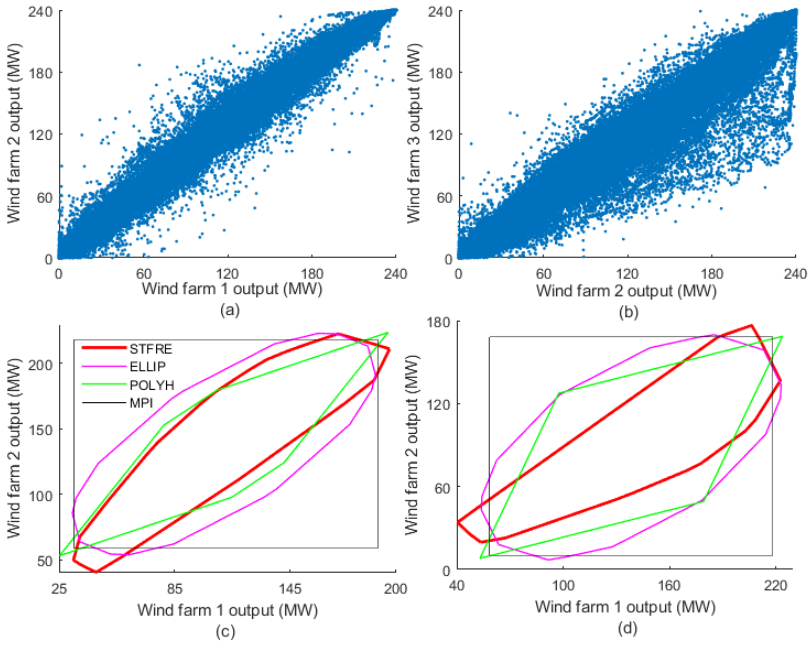

Figure 2.16: (a) and (b) are scatter plots of power outputs of three wind farms. (c) and (d) Projections of different uncertainty sets in the wind farm 1-wind farm 2 and the wind farm 2-wind farm 3 plane.

Scatter plots of the power outputs of the three wind farms (when their respective capacities are of 240 MW) are shown in Figure 2.16 (a) and (b); these clearly show their spatial correlation. In Figure 2.16 (c) and (d), we project the 60-minute uncertainty sets generated by STFRE, ELLIP, POLYH and MPI onto two planes. For each strategy, the projection is made at 8:10 AM on January 1st. By inspection of Figure 2.16, we find that STFRE can effectively capture the spatial correlation between the three wind farms. Also, it can adjust its shape according to the scatter plot region of the correlated wind power outputs, while other uncertainty sets have fixed shapes (i.e., polyhedron, ellipsoid or box). As a result, STFRE has the lowest volume among all the uncertainty sets (see Figure 2.17).

Moreover, in Figure 2.18 (a) and (b) we can see that the transmission line envelopes $\{e^{\uparrow}_{F_{18}},$

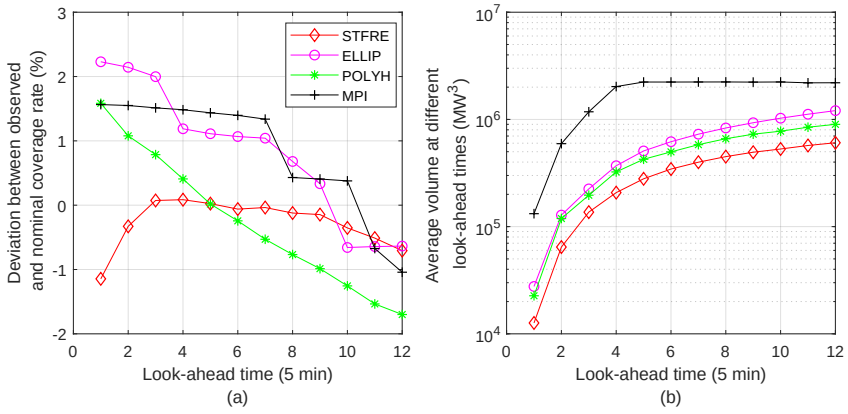

Figure 2.17: (a) Calibration results (b) Average volume of uncertainty sets at different look-ahead times over the entire simulation period (transmission system).

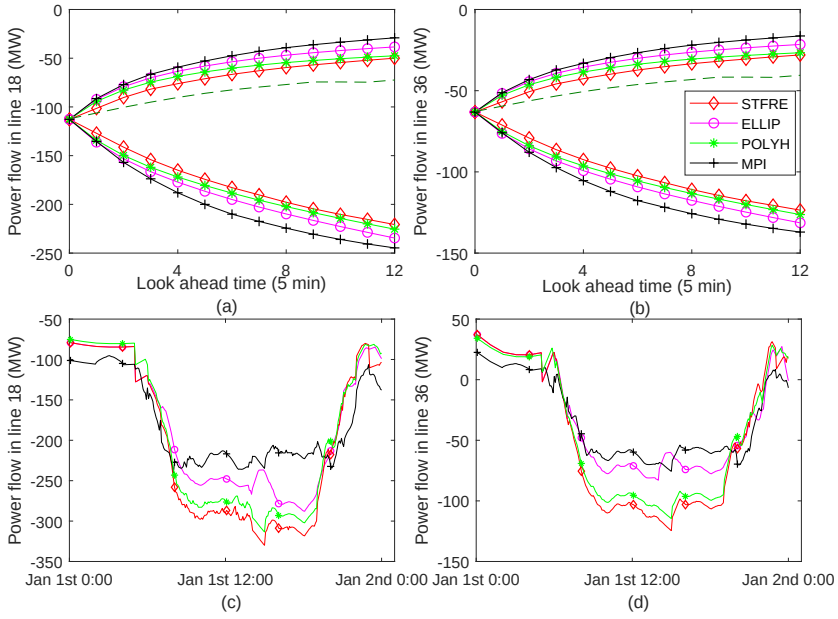

Figure 2.18: (a) Projected envelopes for transmission line $\{e^{\uparrow}_{F_{18}}, e^{\downarrow}_{F_{18}}\}$. (b) Projected envelopes for transmission line $\{e^{\uparrow}_{F_{36}}, e^{\downarrow}_{F_{36}}\}$. The dashed lines are actual trajectories. (c) Power flow in line 18 on Jan 1st. (d) Power flow in line 36 on Jan 1st.

$e^{\downarrow}_{F_{18}}\}$ and $\{e^{\uparrow}_{F_{36}}, e^{\downarrow}_{F_{36}}\}$ generated by STFRE also have smaller sizes than those generated by other uncertainty sets. Though the projected envelopes produced by STFRE in Figure

2.18 are smaller in size, it can be observed that the actual power trajectories remain inside respective envelopes. Figure 2.18 (c) and (d) show the power flows in lines 18 and 36 on January 1st when all three wind farms have a capacity of 240 MW. The power flows in line 18 and line 36 under ELLIP, POLYH and MPI are always lower than that under STFRE (negative represents the direction of power flow). This is because the larger projected envelopes produced by ELLIP, POLYH and MPI in Figure 2.18 (a) and (b) make them overestimate the potential risk of flow violations in transmission lines. Therefore, they lead to a larger reservation for safety margins. Hence, the available transmission line capacities are better exploited by STFRE than other uncertainty sets, which contributes to drive down its cost.

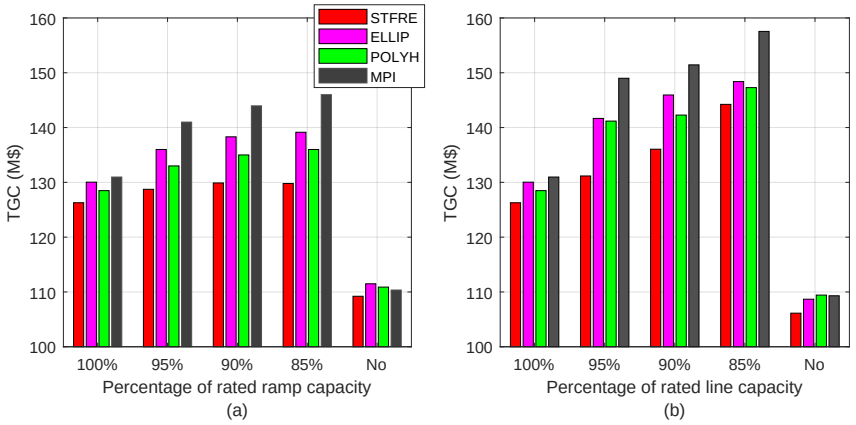

Figure 2.19: Cost performance of different strategies in one year's operation under different degrees of transmission limits and system ramp capacities.

Figure 2.19 shows the cost performance of different energy scheduling strategies under different degrees of transmission limits and system ramping capacities. The transmission capacity of each branch or the ramp capacity of each generator is set as a certain percentage of its original capacity. In the 'No case', no transmission limit or ramp limit is considered. By inspection of Figure 2.19 (a), we see that all the strategies will perform better when there are higher transmission limits. When there are no transmission limits, all strategies have similar performances. In Figure 2.19 (b), we see that lower ramp limits will lead to higher operating costs. However, in all cases the proposed STFRE leads to the most cost efficient decisions.

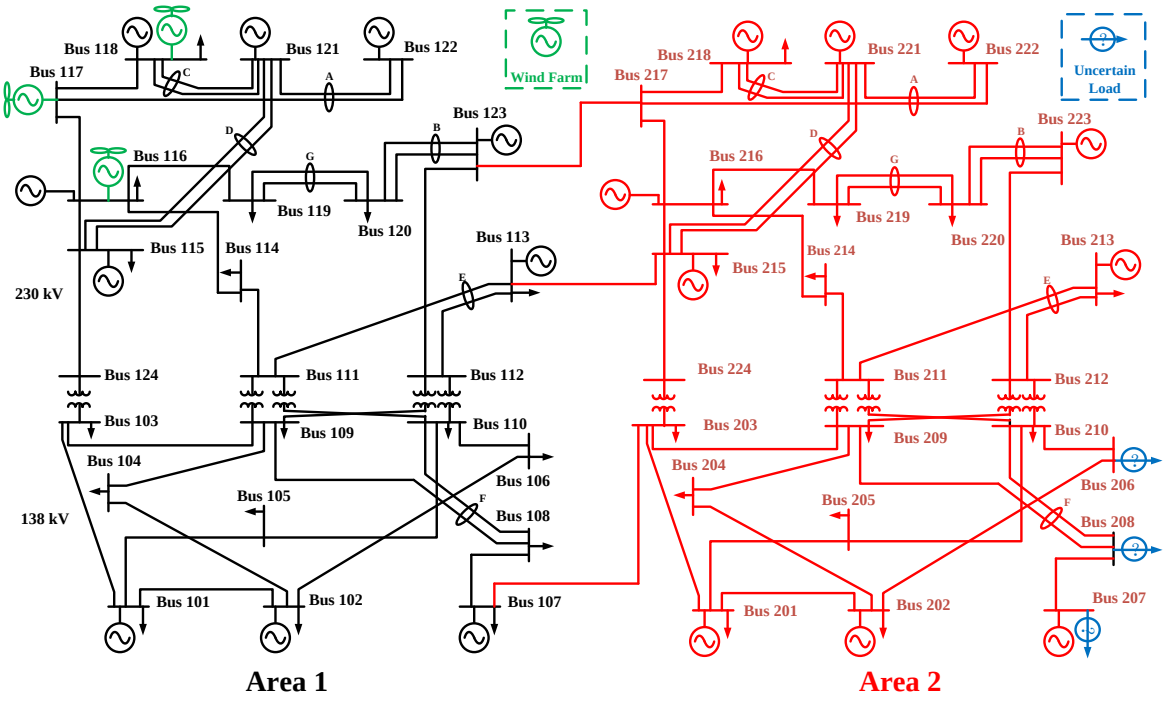

Figure 2.20: Structure of two interconnected IEEE RTS systems.

Finally, in order to evaluate the effects of the system size on the performance of the proposed approach, two IEEE RTS system are interconnected to form the larger system shown in Figure 2.20. Three transmission lines (500 MW each) are added to connect the two systems. Area 1 is the same system as the one shown in Figure 2.14. In Area 2, we replace the loads at buses 6, 7 and 8 with three correlated uncertain loads. The uncertain load data is taken from [70], and the load curves are scaled to match the maximum load at corresponding buses. Other generation and transmission parameters in each area are the same with the settings in Figure 2.14. The performance metrics of different energy scheduling strategies along with their computational complexities are listed in Table 2.2.

The proposed STFRE still has the lowest TGC [4]. All the strategies have similar good performances in terms of ENS and EC. Table 2.2 also shows the computational complexities of the different strategies, which are evaluated based on the average CPU time required to run

[4]Although the STFRE only saves 3.2% of TGC compared with POLYH, in a large power system (e.g., the system in Figure 2.20) 3.2% of TGC still means that significant generation costs and carbon emissions have been reduced.

Table 2.2: Energy Scheduling Strategies' Performance (One Month)

Uncertainty Sets	TGC (M$)	ENS (MWh)	EC (MWh)	Average CPU time per step (s)
STFRE	21.9	0.37	0	1.02 (0.43)
POLYH	22.6 (+3.2%)	0.37	0	1.01 (0.42)
ELLIP	23.0 (+5.0%)	0.36	0	1.02 (0.41)
MPI	23.5 (+7.3%)	0.36	0	1.02 (0.43)

each dispatch step (the number in brackets is the computational time for the single-area IEEE RTS system in Figure 2.14). It can be seen that all the strategies have similar computational complexities. At the same time, as all these strategies are essentially robust optimization approaches, their computational complexity increases with the size of the system. However, for both the single- and two-area test systems, the computational time of our approach is still far bellow the length of the forward-looking time step (five minutes) necessary for an eventual real-time implementation.

2.4 Summary

In this chapter, we proposed spatio-temporal flexibility requirement envelopes for managing flexibility and scheduling energy in power systems with significant variable renewable generation. Using historic generation and demand data, it can comprehensively capture and represent the temporal trends and spatial correlation of multisite renewable generation and load demand. A mathematical program is also developed for applying the proposed variability modeling approach into power system energy scheduling through projections of the spatio-temporal envelopes.

We conducted case studies to showcase the effectiveness of our approach both in microgrid context and in a larger transmission system. Results show that spatio-temporal flexibility requirement envelopes can effectively capture the temporal trends and spatial correlation of input power profiles by adjusting its shape and size, which effectively reduces the over-conservatism of the energy scheduling solution. We observed that, as a result, the flexibility

of storage in a microgrid and the full capacity of transmission lines in a transmission system can be better exploited. This leads to a decrease in total energy scheduling cost in the long run, while not incurring more load shedding or renewable generation curtailment as compared to other uncertainty set characterization methods (i.e., ellipsoidal, polyhedral and box uncertainty sets).

In the next chapter, we develop a rigorous DT-based learning framework for implementing flexibility-based MG dispatch paradigm (the MG has one dispatchable generator only) in a MG controller with limited computational power.

Chapter 3

Decision Tree-Based Flexibility Management for Microgrids

In this chapter, we apply a flexibility-based operational planning paradigm to MG energy dispatch. Here, the MG has a maximum of one dispatchable generator. As is discussed in Chapter 1, the classic energy dispatch problem with energy storage and dispatchable thermal generation assets requires the solution of mixed-integer optimization problems. Such approaches are not amenable to most remote MG and practical field MG implementations, where controls are rule-based and typically implemented by PLC. Albeit such rule-based dispatch controls are always feasible, they cannot optimize fully over the availability of renewable generation and asset capacities of MGs, especially energy storage. In this chapter we propose a systematic method to generate the MG dispatch rule base with the objective of matching as much as possible the control performance obtained by full mixed-integer optimization. To achieve this we develop a rigorous control mapping method based on DTs. The numerical results demonstrate that the DT-based dispatch strategy can provide feasible and near optimal dispatch decisions for MG. Its computational efficiency is very high, a feature promising for real time in field implementation.

3.1 Introduction

The implementation of MG is envisioned to be an effective way of integrating renewable resources to the grid. The MG controller is considered as a key element to realize this

vision [71], as it coordinates different distributed energy resources to maintain the economic and secure operation of a MG. In practice, to fully cater to the requirements of the dispatch function of a MG controller suggested in [71] and [72], a number of challenging issues arise. For example, traditional rule-based dispatch functions are becoming less capable of handling the volatility brought by increasing levels of renewable generation and handling systematically the capacities of energy storage assets; at the same time, controls in MG are typically implemented by PLC or other industrial controllers with limited computational power and the dispatch algorithm faces stringent execution time for real time operation, often in remote locations. Resolving these issues calls for alternative operational planning approaches capable of bridging the need for higher control performance and simplicity of field implementation and execution.

For years, the concept of flexibility [6] has played a crucial role in MG dispatch operations for accommodating more intermittent renewable generation. In Chapter 1, we reviewed the state of the art on the flexibility management in MGs. To summarize, recent literature has shown that flexibility-based operational planning paradigms can contribute a great deal to the economic and secure operation of MG. However, such paradigms usually require the online solution of MIP. Traditional methods for solving MIP problems require advanced iterative optimization algorithms and significant computational power in terms of CPU and memory [60], which may be difficult for implementation in practical settings.

Recently, ML techniques are beginning to emerge as promising tools for reducing the computational complexity of traditional optimization algorithms [73]. In energy scheduling, the same optimization problem is solved repeatedly, differing only in the input data, corresponding to the current and forecasted demand and renewable generation. Thus, there exists an opportunity for ML to learn and approximate the mapping between the input data of an optimization-based dispatch algorithm and its solution. The resulting ML models, once well-trained, are able to map inputs to dispatch decisions at a fraction of the computational effort of the original optimization problems [74]. Previous studies [60], [75] designed model-free reinforcement learning to represent this mapping in MGs which are partially observable.

In our study the MG controller is assumed to have full knowledge of the system, and the underlying MG dispatch optimization problem is perfectly known. The model-free reinforcement learning method cannot fully leverage the pre-existing knowledge about the

optimization model, which may result in solutions that violate system operating constraints [73], [74]. To better exploit any known structures of the energy scheduling problem, supervised learning methods were used in [76], [77] and [78]. In these papers, learning components are embedded in the traditional iterative algorithms to accelerate the optimization process. Although the complexities of the proposed algorithms have been reduced significantly, they still rely on iterative procedures to obtain optimal solutions, which still could be time-consuming and resource-consuming in low-power computing field-based applications. To further reduce the computational complexity, the authors of [79] formulated the MG dispatch as a dynamic programming problem and employed DNN to approximate the optimal value function in a one step process. Though the efficiency of the proposed algorithm is promising for real time implementation, the authors assume the generators are always kept online, which may not always be economical in practice as significant savings can be derived by decommitting generators in low net load (i.e., load less renewable generation) scenarios [80]. Reference [81] developed neural network-based method for accelerating the convergence of existing distributed demand response algorithm in MG. A reinforcement learning approach was proposed in [82] for establishing computationally efficient dispatching rules for battery energy storage systems in MG. However, the authors of [81] and [82] also did not consider the ON/OFF statuses of generators in their models.

Papers [83] and [84] used DT to approximate their MG dispatch algorithm. The dispatch decisions are predicted by directly passing the inputs through the DT, which is quite economical in terms of computation. However, the method in [83] relies on day-ahead scheduling results to satisfy the operating constraints, which may not always be available in practice. The authors of [84] used a single DT to approximate their MG dispatch algorithm. However, the direct output of the single DT is not guaranteed to be feasible in light of the original optimization problem constraints [73]. Thus, none of the existing works has provided a rigorous learning framework for implementing an optimization algorithm in a low-power computing MG controller either because the feasibility of the solution cannot be guaranteed, ON/OFF statuses of generators are not considered, or the algorithm relies on complex iterative procedures.

Inspired by previous works, in this chapter, the power of ML and flexibility-based operational planning paradigms are combined to design a computational efficient and high

quality solution for real time MG dispatch. Specifically, our work is based on DT. There are two reasons why DT are adopted in the context of this work: 1) DT can be efficiently programmed in a PLC-based MG controller. Although previous studies [79], [85] show the effectiveness of DNN in designing real time MG dispatch algorithms, such ML techniques are difficult to implement in a practical industrial controller; 2) Though the approximation abilities of DT are not on par with more advanced ML techniques, like DNN, a DT approach leads to highly interpretable dispatch rules [77], which is essential in mission-critical industries like power and energy.

Compared to the learning based approach for simplifying and accelerating optimization algorithms in existing literature, our DT-based framework has the following advantages.

- It is a rigorous framework where the operating constraints of MG, uncertainties in net load, the commitment statuses (ON/ OFF) of dispatchable generators, and the energy storage system are considered systematically.
- The pre-existing knowledge regarding the original MIP problem is well exploited to guarantee the feasibility and the optimality of the predicted dispatch decisions. The first improvement comes through a systematic partitioning of the feature space into different non-overlapping regions according to the flexibility limits of energy storage asset. Different DTs are built for different regions to guarantee the strict enforcement of the power balance. Second, the outputs of DT are post-processed using the operating constraints of individual units. Third, a metric for pruning DT is designed according to the objective function of the MIP instead of conventional error metics, which contributes to improve the optimality of the predicted dispatch decisions.

3.2 MG Economic Dispatch

A MG typically contains dispatchable generators, energy storage, renewable generation (wind and solar generators) and loads. In this chapter, we make the following simplifying assumptions.

- The network constraints of the MG are ignored;
- As is the case with [71], the interaction between the MG and the distribution network

is not considered in this chapter. Thus, the power exchange $p_{poi}(t)$ is taken as a parameter;

- We assume that the MG has a maximum of one dispatchable generator. If there were multiple generators, they would be started up and shut down simultaneously.

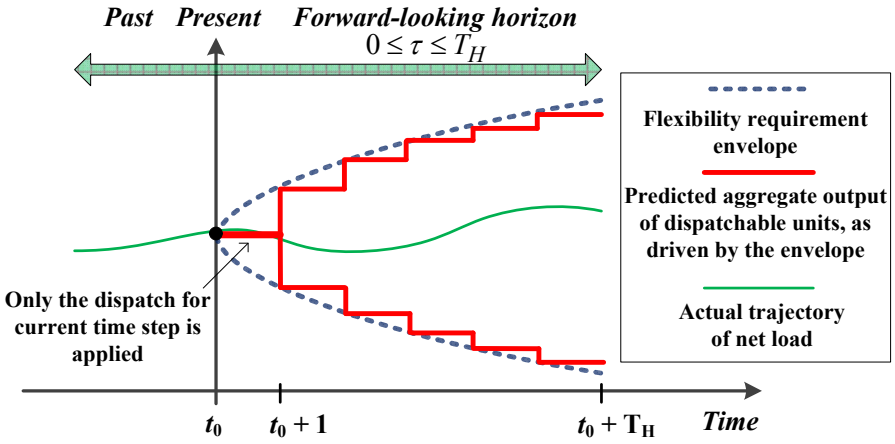

Figure 3.1: Illustration of the receding horizon operation using flexibility requirement envelopes. The forward-looking horizon is shifted at each time step.

The MG dispatch problem is formulated with the concept of flexibility requirement envelopes. First, the flexibility requirement envelope of the net load $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ is constructed using time series of $\{d(t)\}_{t=1}^{T}$ and $\{\delta(t)\}_{t=1}^{T}$, where T is the length of the time series. In this chapter, we assume that the envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ is a function of τ only (i.e., it is time invariant and thus assumes that renewable generation variability looking τ units of time in the future is essentially the same at all time t). We refer readers to [6] for further details about quantifying the envelope [1]. The envelope has an up-going ($\uparrow$) and a down-going ($\downarrow$) branch, as shown in Figure 3.1. The MIP of MG dispatch (with one dispatchable generator only) is presented in (C.1) – (C.9) in Appendix C.

We can see that the MIP is solved repeatedly at each time step t, differing only in two input parameters. They are the residual net load (i.e., $d(t) - \delta(t) - p_{poi}(t)$) and the energy

[1] Please note that the envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ used in this chapter can also be constructed using the approach we proposed in Chapter 2. However, it is more convenient to construct the envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ directly using the method in [6] because this flexibility requirement envelope is assumed to be time invariant in this chapter.

level of the storage asset at the end of last time step (i.e., $s_e(t-1)$).

At the current time, t_0, the energy scheduling problem (C.1) – (C.9) is solved, which requires dispatch decisions to consider a forward-looking "envelope" containing potential trajectories for net load for all $\tau \in \Xi_H$. However, only the scheduling decisions for the current time step t_0 (i.e., $\tau = 0$) are applied. The scheduled trajectories of dispatchable units are assumed to remain constant between each time step $t_0 \rightarrow t_0 + 1$. At the beginning of the next time step $t_0 + 1$, the above process is repeated. This procedure, known as the receding horizon operation, is shown in Figure 3.1.

The computational burden of the MIP (C.1) – (C.9) is quite high given that it needs to optimize over the forward-looking horizon and multiple branches of the envelope. Direct implementation of the MIP in a MG controller requires running online optimization algorithms to solve the problem at each time step. Traditional optimization algorithms rely on iterative procedures to obtain optimal solution [73]. Solving the MIP using such algorithms within the sampling time requires intensive computational power, which is difficult for implementation in practical MG where controllers usually have limited processing power. Thus, in this chapter we will construct an approximate DT-based solver capable of replacing the traditional computationally expensive optimization algorithms for real time MG dispatch.

3.3 Systematic Mapping of MIP-Based Dispatch onto Decision Trees

The overall workflow of the DT-based approach is shown in Figure 3.2, which is divided into the training phase and the testing phase. The details of each phase are presented below. We note that both the training and testing phases require offline simulation runs of the MIP-based MG dispatch problem (C.1) – (C.9) based on realistic field data (renewables and load). The collection of input data/simulated output dispatch instructions constitute the raw materials of the training and testing steps.

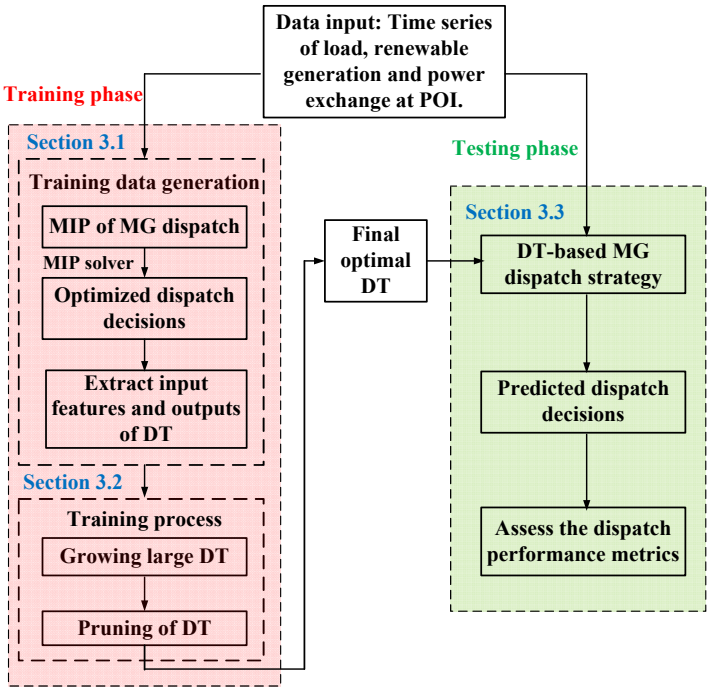

Figure 3.2: Overview of the DT-based optimization approach.

3.3.1 Training Data Generation

Conventionally, DT can perform both classification and regression tasks. In classification, the training data set is organized as a collection of T instances $\{(\boldsymbol{x}_t, y_t)\}_{t=1}^{T}$. The tth instance is described by a vector of N_x input features $\boldsymbol{x}_t = (x_t^1, \ldots, x_t^{N_x})$ and a discrete output value y_t. In regression, the data is organized as $\{(\boldsymbol{x}_t, z_t)\}_{t=1}^{T}$, where the output z_t is continuous.

In order to build a training data set for creating DT, we first define a Training Scenario (TAS). In the context of this chapter, where we seek to build a DT which maps MG states to MIP dispatch decisions, a TAS contains the following information:

1. The length of the offline dispatch simulation T;
2. Time step length T_Δ;
3. The parameters of dispatchable generation;
4. The parameters of storage;

5. Time series of $\{d(t)\}_{t=1}^{T}$, $\{\delta(t)\}_{t=1}^{T}$ and $\{p_{poi}(t)\}_{t=1}^{T}$.

The optimized dispatch decisions at time t (denoted with an asterisk) $p_g^*(t)$, $p_e^*(t)$, $d_c^*(t)$, and $\delta_c^*(t)$ $(t = 1, 2, \cdots, T)$ are obtained by solving (C.1) – (C.9) with a MIP solver. These MIP outputs are tabulated along the residual net load $d(t) - \delta(t) - p_{poi}(t)$ at time t and the previous time step energy storage state of charge $s_e^*(t-1)$ as mentioned previously.

We note that DT are approximations of an input/output relationship. Thus, the input features of the DT need to be transformed from the two original inputs of the MIP as follows $(t = 1, 2, \cdots, T)$. Here we perform feature scaling [86] to make the input features have similar scales.

$$x_t^1 = (d(t) - \delta(t) - p_{poi}(t))/\overline{p_e}.$$

$$x_t^2 = (s_e^*(t-1) - \underline{s_e})/(\overline{s_e} - \underline{s_e}).$$

The outputs of the DTs need to be transformed from the outputs of the MIP, i.e., the optimized dispatch decisions $p_g^*(t)$, $p_e^*(t)$, $d_c^*(t)$, and $\delta_c^*(t)$. Specifically, we take advantage of the dependency induced by the power balance equation (3.1) and define the dispatch options shown in Table 3.1

$$p_g^*(t) + p_e^*(t) = d(t) - \delta(t) - p_{poi}(t) \tag{3.1}$$

In doing so, we reduce the number of decision variables to be predicted. Note that the storage can be either charged or discharged in options Opt_2 and Opt_3. Options Opt_4 and Opt_5 are used as in last resort to maintain the power balance. The output y_t (discrete) is defined as follows $(t = 1, 2, \cdots, T)$.

y_t Depending on the values of $p_g^*(t)$, $p_e^*(t)$, $d_c^*(t)$ and $\delta_c^*(t)$, y_t maps to one of the dispatch options in Table 3.1.

The free variables in options Opt_1 and Opt_2 can be further determined using (3.1). However, it becomes further complicated with option Opt_3 when both generation and storage are dispatched. The power balance (3.1) does not suffice. Therefore, we introduce another set of DT, whose output z_t (continuous) is defined below, to completely determine the output power of generator and storage for option Opt_3.

$$z_t = p_e^*(t)/\overline{p_e} \ (t = 1, 2, \cdots, T).$$

Table 3.1: Dispatch options of the MG

Index	Description
Opt_1	Generator only ($p_g^*(t) > 0$ and $p_e^*(t) = 0$)
Opt_2	Storage only ($p_g^*(t) = 0$ and $p_e^*(t) \neq 0$)
Opt_3	Generator + Storage ($p_g^*(t) > 0$ and $p_e^*(t) \neq 0$)
Opt_4	Curtail excess power ($\delta_c^*(t) > 0$)
Opt_5	Load shedding ($d_c^*(t) > 0$)

3.3.2 Method for Building DT

From the data generated in the previous subsection, we first grow a large decision tree, and then prune it back in order to obtain a subtree, whose size is more reasonable for a practical implementation.

Stage 1: Growing Large DT

We define CT as a large classification tree (to represent y_t) and RT as a large regression tree (to represent z_t). They are grown as follows:

Step 1. *Preparation of Training Data Set:* First, a TAS for the MG is defined. The training data sets $\{(\boldsymbol{x}_t, y_t)\}_{t=1}^T$ and $\{(\boldsymbol{x}_t, z_t)\}_{t=1}^T$ are built as aforementioned.

Step 2. *Partitioning of the Space of Features:* The space of x_t^1 and x_t^2 $(t = 1, 2, \cdots, T)$ is divided into the four non-overlapping regions which are described in Table 3.2.

In Table 3.2, the space of features is partitioned according to the flexibility limits of the storage in terms of power capacity and energy storage level. In Reg_1 the residual net load stays within the power and energy limits of the storage. The generator can be kept off in this region. In Reg_2 the residual net load goes beyond the power capacity of the storage, whereas the energy holding of the storage is insufficient in Reg_3. The generator has to be dispatched in the two regions. There is surplus renewable generation in Reg_4.

Step 3. *Building Large DT:* Different large DT are built for different regions in Table 3.2.

Step 3.1: For each region Reg_r $(r = 1, 2, 3)$, a large classification tree CT_r is created using $\{(\boldsymbol{x}_t, y_t)\}_{t \in Reg_r}$ as the training data, which is the subset of the whole training data set corresponding to region Reg_r. Next, a large regression tree RT_r is grown using $\{(\boldsymbol{x}_t, z_t)\}_{t \in Reg_r \cap \{t | y_t = Opt_3\}}$ as the training data. A brief introduction about how to grow

Table 3.2: Different regions of the space of x_t^1 and x_t^2

Index	Description
Reg_1	$\{t \mid 0 < x_t^1 \leq 1 \text{ and } x_t^2/x_t^1 \geq \overline{p_e} T_\Delta/(\overline{s_e} - \underline{s_e})\}$
Reg_2	$\{t \mid x_t^1 > 1\}$
Reg_3	$\{t \mid 0 < x_t^1 \leq 1 \text{ and } x_t^2/x_t^1 < \overline{p_e} T_\Delta/(\overline{s_e} - \underline{s_e})\}$
Reg_4	$\{t \mid x_t^1 \leq 0\}$

large classification tree and large regression tree from the training data is provided in Appendix D

Step 3.2: The dispatch in region Reg_4 is very simple. It represents the case where the MIP (C.1) – (C.9) simply stores the excess electricity in storage (option Opt_2). After the storage is full, the excess electricity is curtailed (option Opt_4). Therefore, the dispatch in region Reg_4 requires no distinct decision tree.

Stage 2: Pruning of DT

The best substrees of CT_r and RT_r ($r = 1, 2, 3$) will be determined. However, evaluating every possible subtree would be cumbersome. Thus, the cost complexity pruning method [87] is used. We define nonnegative tuning parameters α_r and β_r for each CT_r and RT_r, respectively. The pruned DT are denoted as $CT_r^{\alpha_r}$ and $RT_r^{\beta_r}$, which are functions of α_r and β_r, respectively. The larger the tuning parameter is, the smaller the pruned DT will be. The cost complexity pruning method is introduced briefly in Appendix D.

Step 4. *Determining the Best Tuning Parameters:* First, for each large tree CT_r (or RT_r, $r = 1, 2, 3$), a K-fold cross-validation [87] is performed. The tuning parameter α_r^0 (or β_r^0, $r = 1, 2, 3$) that minimizes the cross-validation error is found.

Next, we explore all the combinations of $\{\alpha_r, \beta_r\}_{r=1}^3$ for $\alpha_r^0 \leq \alpha_r \leq \alpha_r^{max}$ and $\beta_r^0 \leq \beta_r \leq \beta_r^{max}$, where α_r^{max} and β_r^{max} are the maximum values for the tuning parameters. The total number of combinations of tuning parameters explored is $\prod_{r=1}^3 (\alpha_r^{\max} - \alpha_r^0 + 1)(\beta_r^{\max} - \beta_r^0 + 1)$.

For each combination, the corresponding subtrees are applied to MG dispatch following Algorithm 1, where the parameters and input time series of the MG are set according to the TAS defined in *Step 1.* Note that the variables $\overline{es}$ and $\underline{es}$ represent respectively the

maximum power that can be discharged from and charged into storage at the current time step. They take into account both the power and the energy limits of the storage. The Total Cost (TC) is calculated based on the corresponding dispatch results for the full offline dispatch simulation T, as a function of the combination $\{\alpha_r, \beta_r\}_{r=1}^{3}$

$$TC(\{\alpha_r, \beta_r\}_{r=1}^{3}) = \sum_{t=1}^{T} \left(a_g p_g(t)^2 + b_g p_g(t) + c_g u_g(t) + \pi^d d_c(t) + \pi^\delta \delta_c(t) \right) \tag{3.2}$$

Then the tuning parameters that have the minimal TC are the best tuning parameters, which are denoted as $\{\alpha_r^*, \beta_r^*\}_{r=1}^{3}$. Please note that all the operating constraints of MG have been respected in Algorithm 1 when we optimize the tuning parameters of DT.

The final *optimal* DT are denoted by $CT_r^{\alpha_r^*}$ and $RT_r^{\beta_r^*}$ $(r = 1, 2, 3)$. Algorithm 1 along with the corresponding optimal DT constitute the DT-based MG dispatch strategy which can be ported to a practical general-purpose industrial controller. It is worth noting that the loop in line 1 of Algorithm 1 applies only when we are tuning α_r and β_r. When in operation, the algorithm is done in open-loop.

Discussion

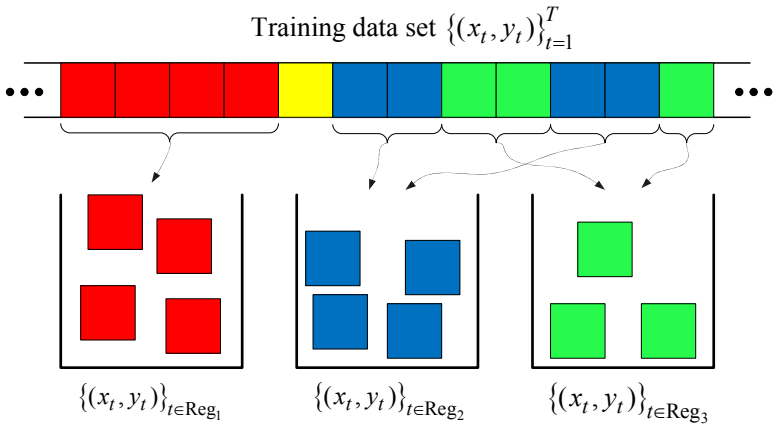

Figure 3.3: All the training samples $\{(\boldsymbol{x}_t, y_t)\}_{t=1}^{T}$ are sorted by region (Table 3.2). Each square block represents an instance $(\boldsymbol{x}_t, y_t)$. The red, blue, green and yellow blocks represent instances in region Reg_1, Reg_2, Reg_3 and Reg_4, respectively.

To guarantee that the dispatch decisions made by the DT-based MG dispatch strategy

Algorithm 1 DT-Based MG Dispatch Algorithm

Input: $CT_r^{\alpha_r}$ and $RT_r^{\beta_r}$ $(r = 1, 2, 3)$

Output: Dispatch decisions $\{p_g(t), p_e(t), d_c(t), \delta_c(t)\}$.

1: **for** $t = 1$ **to** T **do**
2: $x_1' \leftarrow (d(t) - \delta(t) - p_{poi}(t))/\overline{p_e}$
3: $x_2' \leftarrow (s_e(t-1) - \underline{s_e})/(\overline{s_e} - \underline{s_e})$
4: **if** $x_1' \leq 0$ (region Reg_4) **then**
5: $p_g(t) \leftarrow 0$, $p_e(t) \leftarrow d(t) - \delta(t) - p_{poi}(t)$
6: **else**
7: x_1' and x_2' are in region Reg_r $(r = 1, 2, 3)$.
8: Predict the dispatch option y' using $CT_r^{\alpha_r}$ with x_1' and x_2' as inputs.
9: **if** $y' = Opt1$ **then**
10: $p_g(t) \leftarrow d(t) - \delta(t) - p_{poi}(t)$, $p_e(t) \leftarrow 0$
11: **else if** $y' = Opt3$ **then**
12: Predict z' using $RT_r^{\beta_r}$ with x_1' and x_2' as inputs
13: $p_e(t) \leftarrow z'\overline{p_e}$
14: $p_g(t) \leftarrow d(t) - \delta(t) - p_{poi}(t) - p_e(t)$
15: **else**
16: $p_g(t) \leftarrow 0$, $p_e(t) \leftarrow d(t) - \delta(t) - p_{poi}(t)$
17: **end if**
18: **end if**
19: **if** $p_g(t) > 0$ **then**
20: $p_g(t) \leftarrow \min\{\max\{p_g(t), \underline{p_g}\}, \overline{p_g}\}$
21: **end if**
22: $\overline{es} \leftarrow \min\{\overline{p_e}, (s_e(t) - \underline{s_e})\eta_d/T_\Delta\}$
23: $\underline{es} \leftarrow \max\{\underline{p_e}, (s_e(t) - \overline{s_e})/(\eta_c T_\Delta)\}$
24: $p_e(t) \leftarrow \min\{\max\{d(t) - \delta(t) - p_{poi}(t) - p_g(t), \underline{es}\}, \overline{es}\}$
25: Calculate $s_e(t)$
26: **end for**

can always satisfy the operating constraints, we use the following methods.

- Add a post-processing stage in Algorithm 1 (lines 19 – 24) to satisfy the operating constraints of dispatchable generation and storage.
- As is discussed in *Step 2*, the different regions in Table 2 have different feasible dispatch options. It is for this reason that in *Step 3.1* all the training samples $\{(\boldsymbol{x}_t, y_t)\}_{t=1}^T$ are first sorted by region, as shown in Figure 3.3. The dispatch options in the sorted

instances $\{(\boldsymbol{x}_t, y_t)\}_{t \in Reg_r}$ for regions Reg_r $(r = 1, 2, 3)$ are guaranteed to be feasible in their respective regions. Therefore, the resulting classification tree for region Reg_r $(r = 1, 2, 3)$ is trained by using only strictly feasible instances, and the dispatch decisions predicted by Algorithm 1 are guaranteed to satisfy the power balance constraint.

The tuning parameters $\{\alpha_r, \beta_r\}_{r=1}^{3}$ address the optimality of the model for dispatch. In *Step 4*, the best tuning parameters are determined based on the metric TC instead of conventional error metrics used in the ML literature (e.g., mean square error). The TC is designed according to (C.1), which directly measures the optimality of the dispatch results including the generation costs and the penalties of load shedding and wind curtailment. Later in our case study, we will show that conventional error metrics used to train DT are not necessarily related to their performance in dispatch. A similar idea was also proposed in [73]. A possible reason of this phenomena is that the optimal solution to a problem is not always unique.

Nonetheless, in *Step 4* conventional error metrics used for DT are still used to calculate α_r^0 and β_r^0. The purposes of determining α_r^0 and β_r^0 are to: 1) reduce the risk of DT overfitting, which unnecessarily increases their sizes; 2) reduce the search space in *Step 4* when we determine the best tuning parameters. For example, if we consider one year's dispatch with a 5-minute time step when creating the training data set, then T is approximately 10^5 and the total number of possible combinations of tuning parameters in the search space can be up to 2×10^{10} if there is no reduction, while the total number of combinations explored is only approximately 3600 when the searching space is reduced. Thus, reducing the search space is clearly essential.

We can see from Algorithm 1 that the DT-based dispatch predicts dispatch decisions in a open-loop process. Compared with traditional iterative algorithms for solving the MIP, the DT-based method is much more computationally efficient and promising to enable real-time implementation in a industrial controller with limited computational power.

In Section 3.2 we ignore the network constraints of the MG. This is because the physical network constraints of a MG or distribution system are mainly concerned with node voltage constraints [88]. Voltage management in a MG or a distribution system is mostly concerned with reactive power management and the control of voltage compensation devices [89] (e.g., capacitors, reactors, static VAR compensators, etc), which all go beyond the scope of this chapter.

If the power exchange with the utility grid $p_{poi}(t)$ is allowed to deviate from the pre-determined transaction schedules, we would need to add an input feature in the DT, which would be the electricity price associated with $p_{poi}(t)$. The first feature of the DT should be modified to become $x_t^1 = (d(t) - \delta(t) - \overline{p_{poi}})/\overline{p_e}$, where $\overline{p_{poi}}$ is the upper limit of power exchange. Then, in each region of Table 3.2 (including region Reg_4), we would build a classification tree and regression tree following the same procedure as in *Step 3* and *Step 4*. Moreover, for each region we would need to construct another regression tree whose output is the grid power exchange $p_{poi}(t)$. As a result, there would be 12 DT in total instead of the original six. Therefore, in this case the complexity of the proposed DT-based dispatch would still not increase much.

Nonetheless, please note that in practice distribution systems usually require MG to "maintain synchronization at the POI with constant power exchange, following pre-arranged transaction schedules with minimum deviation and minimum accumulated inadvertent energy exchange" [90]. The MG controller is responsible for managing the intermittency of its internal distributed energy resources and respond quickly to the internal load and generation changes or disturbances. This way, there are minimum impacts on the distribution grid at the POI.

3.3.3 Testing of the DT-Based MG Dispatch

In order to test the proposed DT-based MG dispatch strategy, we first define a Testing Scenario (TES) which could have different time series of demand and renewable generation, or different dispatchable asset limitations and parameters compared to the TAS. Then, the proposed DT-based MG dispatch strategy is tested under the TES to evaluate its dispatch performance against unseen conditions.

3.4 Case Study

3.4.1 Test System Description

The approach proposed in this chapter is tested on a MG (shown in Figure 3.4) which includes one critical load, one priority load [2], one wind farm, one diesel generator and storage.

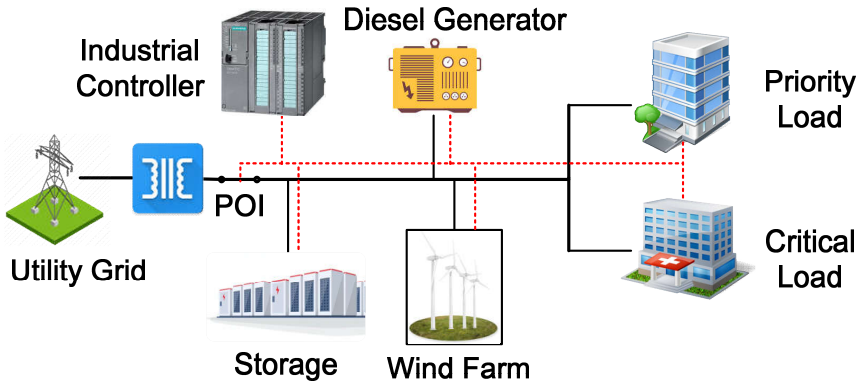

Figure 3.4: MG system structure. The controls in MG are typically implemented by PLC or other industrial controllers with limited computational power.

Scenarios Studied

We construct 10 scenarios for training and testing of the DT-based dispatch approach. In simulation, T_Δ is set to be 5 minutes and T is one year. Our overall objective here is to demonstrate how well the DT-based approach is able to generalize over a very wide array of MG operating conditions (i.e., not require re-training).

S_1 (Base Scenario): For storage: $\overline{p_e} = 100$ kW, $\underline{p_e} = -100$ kW, $\overline{s_e} = 90$ kWh, $\underline{s_e} = 10$ kWh and $\eta_c = \eta_d = 0.963$. The state-of-charge (SOC) of storage is initialized at 50%. For the diesel generator: $\overline{p_g} = 360$ kW, $\underline{p_g} = 45$ kW, $a_g = 1.98 \times 10^{-5}$ \$/kW^{2}h, $b_g = 0.02842$ \$/kWh and $c_g = 54.1$ \$/h. The operating cost of diesel generation is adapted from the data found in [42]. The data of aggregate load $\{d(t)\}_{t=1}^{T}$ is obtained by adding the critical load and the priority load together. The critical load is kept constant at 50 kW. The maximum

[2] According to [71], critical loads are the loads that must be served in all normal operating modes of the microgrid. They typically cannot be curtailed. Priority loads are the loads that may be curtailed if necessary, but should be satisfied if possible.

and minimum values of the priority load are 350 kW and 50 kW, respectively. The data for the priority load comes from an isolated village in the Canadian province of Quebec. The capacity of the wind farm is 100 kW, and the temporal wind generation data $\{\delta(t)\}_{t=1}^{T}$ is taken from NREL's database [66]. The MG is grid-connected; the contractual value of the power exchange at POI $p_{poi}(t)$ is kept constant for each day of the year, which is set to be the minimal net load of any given day. Penalties of load shedding and wind curtailment are set to be $\pi^d = \pi^\delta = 1000$ \$/kWh.

The remaining scenarios $S_2 - S_{10}$ represent cases where there might have been original MIP dispatch model calibration errors (e.g., $S_2 - S_4$), possible asset performance degradation (e.g., $S_5 - S_8$) or different wind power capacity and different utility energy exchange scenarios (S_9, S_{10}).

S_2: The maximum value of each priority load is 450 kW.

S_3: Next year's time series of priority load is used.

S_4: Next year's time series of wind power is used.

S_5: $\overline{p_e} = 150$ kW, $\underline{p_e} = -150$ kW.

S_6: $\overline{p_e} = 75$ kW, $\underline{p_e} = -75$ kW.

S_7: $\overline{s_e} = 140$ kWh.

S_8: $\overline{s_e} = 65$ kWh.

S_9: The capacity of each wind farm is 150 kW. $p_{poi}(t)$ is kept constant at 40 kW for the whole year.

S_{10}: The capacity of each wind farm is 150 kW. $p_{poi}(t)$ is kept at 0 kW in the whole year (islanded).

Other parameters and time series in Scenarios S_2 – S_{10} are same as those in the Base Scenario, S_1.

MG Dispatch Strategies

Three MG dispatch strategies are calculated for comparison purposes.

- *DT-MGDS:* This is the proposed DT-based MG dispatch strategy.
- *DT^{cmp1}-MGDS:* This is the single DT-based MG dispatch strategy proposed previously in [84]. In DT$^{\text{cmp1}}$-MGDS, the space of features is not divided. A single DT is built using the training data set and is then applied directly to perform MG dispatch.

- *Rule-MGDS:* This is a traditional human-engineered rule-based MG dispatch strategy, as proposed in [85].[3] The Rule-MGDS uses a simple priority list to perform MG dispatch. The distributed energy resources are dispatched in the following order: renewable generation, storage and diesel generation.

Performance Metrics

The dispatch strategies are mainly assessed by comparing their respective Total Generation Cost (TGC) when facing the same operating scenarios. We also consider Energy Not Served (ENS) and Energy Curtailed (EC) as further performance metrics.

$$TGC = \sum_{t=1}^{T} a_g p_g(t)^2 + b_g p_g(t) + c_g u_g(t) \tag{3.3}$$

$$ENS = T_\Delta \sum_{t=1}^{T} d_c(t) \tag{3.4}$$

$$EC = T_\Delta \sum_{t=1}^{T} \delta_c(t) \tag{3.5}$$

Analysis Tools

The model (C.1) – (C.9) is solved using the commercial MIP solver CPLEX. The proposed approach is carried out with ML software MATLAB [91] for validation.

3.4.2 Testing of the Proposed DT-Based MG Dispatch Strategy

First, the performance of the DT-MGDS is compared with the Rule-MGDS. The DT-MGDS algorithm is set up under TAS S_1 and its performance in MG dispatch is tested under TES S_m ($m = 1, 2, \cdots, 10$). The resulting TGC is compared with the TGC of Rule-MGDS calculated under S_m. The corresponding percentages of TGC saved by DT-MGDS are presented in Figure 3.5 (a). We can see that DT-MGDS saves an average of 30% of TGC compared with Rule-MGDS over the 10 test scenarios. Also, ENS is nil for both dispatch strategies, and differences in EC are under 1%. In fact, the Rule-MGDS uses a simple priority list in MG

[3] Though this dispatch strategy was called the "decision tree approach-based dynamic energy management system" in [85], it is not based on ML techniques. It is a traditional rule-based MG dispatch strategy in essence. In addition, please note that [85] proposed two sets of MG dispatch algorithms. The first one is the simple rule-based method mentioned above, while the second one is based on DNN. This second method is not used here.

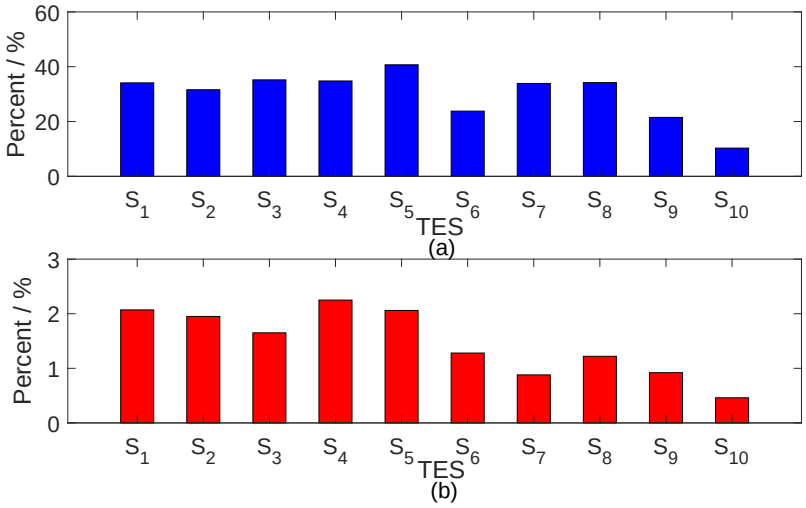

Figure 3.5: (a) Percentage of TGC saved by DT-MGDS compared with Rule-MGDS, (b) TGC optimality loss for DT-MGDS.

dispatch. The distributed energy resources are dispatched in the following order: renewable generation, storage and diesel generator. Although such a simple priority listing method can respect the technical constraints, it is not capable to use optimally the renewable generation and other MG assets, as evidenced by its poor TGC values.

Next, we compare DT-MGDS with dispatch results which would have been obtained by solving the full MIP (C.1) – (C.9). As before, DT are created with the data set of TAS S_1 and tested in MG dispatch under TES S_m ($m = 1, 2, \cdots, 10$). The percentage of optimality loss (taken with respect to the TGC calculated by the MIP solution under S_m) is shown in Figure 3.5 (b). The approximate average percentage optimality loss of DT-MGDS is only 1.5% over the 10 test scenarios. Therefore, DT-MGDS sacrifices little optimality in terms of TGC, yet its online dispatch calculations are significantly simpler. Both the DT-MGDS and the dispatch decisions calculated by MIP have no ENS at all times, and their differences in EC are negligible. As can be seen from Figure 3.5 (b), DT-MGDS still loses little optimality in TGC when its TES is different from its TAS (i.e., when the TES is S_2 – S_{10}). Thus, DT-MGDS is capable of generalizing beyond its TAS and achieving good dispatch performance in a MG with different parameters, load profiles, wind data or other operating conditions.

3.4.3 Comparison of Different DT-Based Dispatch Strategies

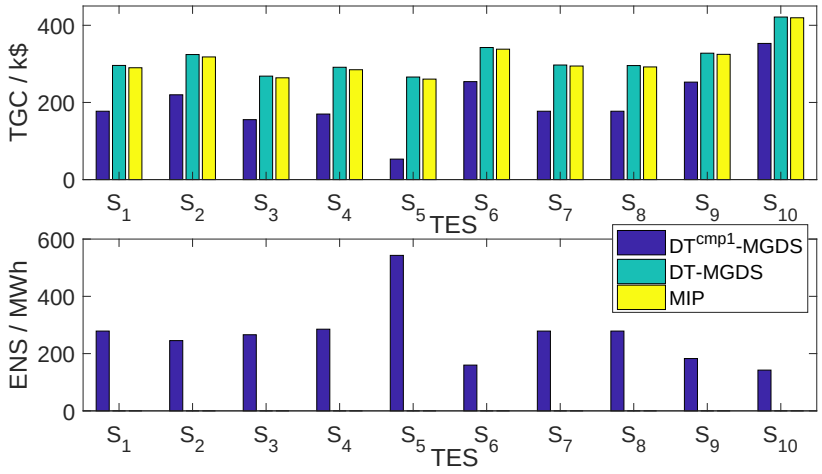

Figure 3.6: TGC and ENS of DT^{cmp1}-MGDS and DT-MGDS tested under all 10 test scenarios. The dispatch results calculated by MIP is also provided (MIP and DT-MGDS results have no ENS).

Next, the proposed DT-MGDS is compared against DT^{cmp1}-MGDS. Here, DT in DT^{cmp1}-MGDS are created with the data set of TAS S_1 and its performance in MG dispatch is tested under all the 10 scenarios.

The results are presented in Figure 3.6. We see that the TGC of DT^{cmp1}-MGDS are even lower than the TGC calculated by MIP. However, DT^{cmp1}-MGDS incurs significant ENS, whereas the results by MIP and DT-MGDS have no load curtailment at all time. A daily evolution of the dispatch results of DT^{cmp1}-MGDS on January 1st can be seen in Figure 3.7 (a). Here, DT^{cmp1}-MGDS was tested under TES S_1. The system started shedding load at 7:00 AM and 8:20 PM because DT^{cmp1}-MGDS still tries to discharge the storage (option Opt_2, Storage only) when the energy holding of storage reaches its minimum (see Figure 3.7 (b)). This indicates that DT^{cmp1}-MGDS fails to sense the operating constraints of the storage and maintain power balance in the system.

In general, the ML techniques cannot guarantee the feasibility of its output (i.e., satisfying constraints) when it is used to output solutions to an optimization problem [73]. Figure 3.7 (c) shows the dispatch options obtained by solving (C.1)–(C.9) in the different operating condition regions (Table 3.2). The statistics are calculated according to the corresponding

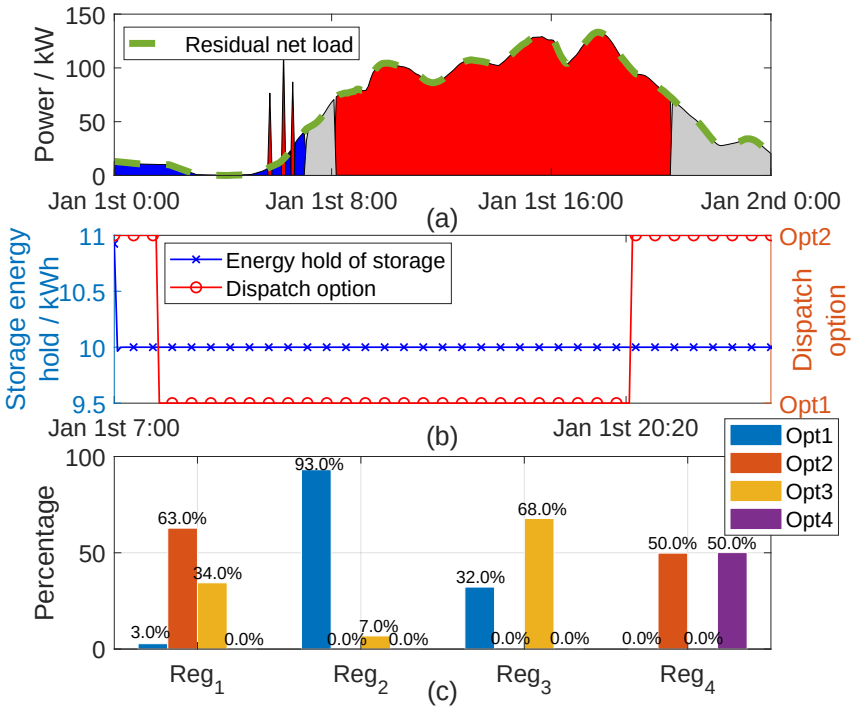

Figure 3.7: (a) Outputs of diesel generation and storage calculated by DT$^{\text{cmp1}}$-MGDS on Jan 1st. Blue and red regions indicate that the storage and the diesel generator supply power to the load, respectively. The energy in grey is curtailed. (b) Energy hold in storage and dispatch options chosen by DT$^{\text{cmp1}}$-MGDS. (c) The dispatch options (Table 3.1) chosen by MIP in the different operating condition regions (Table 3.2).

dispatch results under all of the 10 scenarios. It can be found that the operating condition regions in Table 3.2 have a clear impact on the feasibility of different dispatch options. The problem with DT$^{\text{cmp1}}$-MGDS is that it uses a single DT to represent vastly different operating conditions. Therefore, this makes it much harder to satisfy all operating constraints.

On the other hand, the proposed DT-MGDS prevents ENS much better by building different DTs for the different operating condition regions. The dispatch options in the training data corresponding to each region are prefiltered (as shown in Figure 3.7 (c)), which are guaranteed to be feasible in that region. Also, it adjusts the dispatch decisions made by DT to make sure that diesel generation and storage stay inside their technical limits. As a result, the power balance and the operating constraints of individual units are always

satisfied and no load is curtailed.

3.4.4 Comparison of Different Methods of DT Pruning

The way DT pruning is executed can have a significant effect on their performance. We now consider another DT-based MG dispatch strategy denoted as DT$^{\text{cmp2}}$-MGDS. DT$^{\text{cmp2}}$-MGDS has the same six large DT as DT-MGDS, but the final DT in DT$^{\text{cmp2}}$-MGDS are obtained by pruning the six large DT with α_r^0 and β_r^0 ($r = 1, 2, 3$). DT$^{\text{cmp2}}$-MGDS is set up by inputting $CT_r^{\alpha_r^0}$ and $RT_r^{\beta_r^0}$ into Algorithm 1. Note that α_r^0 and β_r^0 are obtained by minimizing the cross-validation error, which is the traditional method recommended for tuning hyper-parameters of a ML model [86].

First, we measure the Classification Error (CE) or the Regression Error (RE) of the pruned DT from DT-MGDS and DT$^{\text{cmp2}}$-MGDS. The CE and RE are defined as follows [86].

$$CE = \frac{1}{\kappa} \sum_{l=1}^{\kappa} I\{\hat{y}_l \neq y_l\} \tag{3.6}$$

$$RE = \frac{1}{\kappa} \sum_{l=1}^{\kappa} (\hat{z}_l - z_l)^2 \tag{3.7}$$

where κ is the number of test samples and $I\{\cdot\}$ is the indicator function. $\hat{y}_l$ is the class label predicted by the classification tree, and y_l is the desired class label. $\hat{z}_l$ is the value predicted by the regression tree, and z_l is the desired output value. DT-MGDS and DT$^{\text{cmp2}}$-MGDS are generated with the same TAS S_1 in MATLAB.

All the training data of Scenarios S_2 – S_{10} are combined together and used as test samples. We test for CE or RE of each existing DT in the two dispatch strategies using these test samples. Figure 3.8 (a) displays the results. Note that CE and RE calculated here estimate the generalization error of the DT as they measure the error rates on unseen cases. We can see that the DT of DT$^{\text{cmp2}}$-MGDS all have lower error rates. Next, we compare the TGC of DT$^{\text{cmp2}}$-MGDS with Rule-MGDS. DT$^{\text{cmp2}}$-MGDS is created under TAS S_1 and tested for MG dispatch under all 10 scenarios. Figure 3.8 (b) shows the corresponding percentage savings compared with Rule-MGDS. We can find that DT$^{\text{cmp2}}$-MGDS has much higher TGC than DT-MGDS. DT$^{\text{cmp2}}$-MGDS saves only 9.8% on average in terms of TGC across the 10 scenarios, whereas the proposed DT-MGDS saves 30% on average.

It can be seen that the approximation ability (e.g. classification or regression errors)

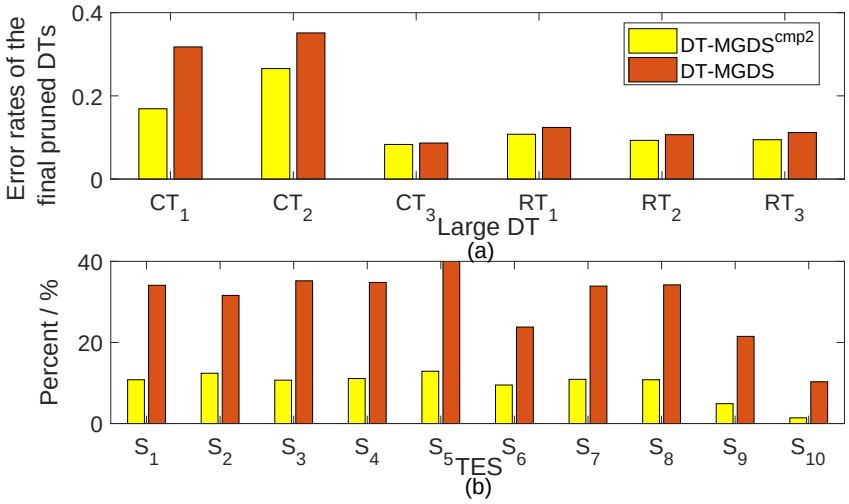

Figure 3.8: (a) Error rates of all the final pruned DT in DT^{cmp2}-MGDS and DT-MGDS. (b) Percentage of TGC saved by DT^{cmp2}-MGDS and DT-MGDS compared with Rule-MGDS.

of a ML technique does not necessarily have a direct effect on its dispatch performance. Therefore, the best tuning parameters of a large DT in DT-MGDS are determined based on TC, where TC, as calculated in (3.2), measures the quality of dispatch results. Unlike in classical DT training algorithms, here the focus is on dispatch performance, not classification or regression accuracy.

3.4.5 Scalability Test

In order to evaluate the effects of the system size on the performance of the DT-MGDS, we test it on a modified CIGRE LV benchmark MG in Figure 3.9. The original benchmark MG can be found in [92].

The DT-MGDS is set up under TAS S_1. The parameters of each diesel generator, storage system, priority load, critical load and wind farm are same as those defined in Figure 3.4. All the three diesel generators will be started up and shut down simultaneously. The performance of DT-MGDS is tested under TES S_m ($m = 1, \cdots, 10$). The percentages of TGC saved by DT-MGDS compared with the Rule-MGDS are presented in Figure 3.10 (a) and the percentages of optimality loss (compared with the TGC of full MIP solution) are shown in Figure 3.10 (b). The DT-MGDS saves 25% of TGC on average compared with Rule-MGDS.

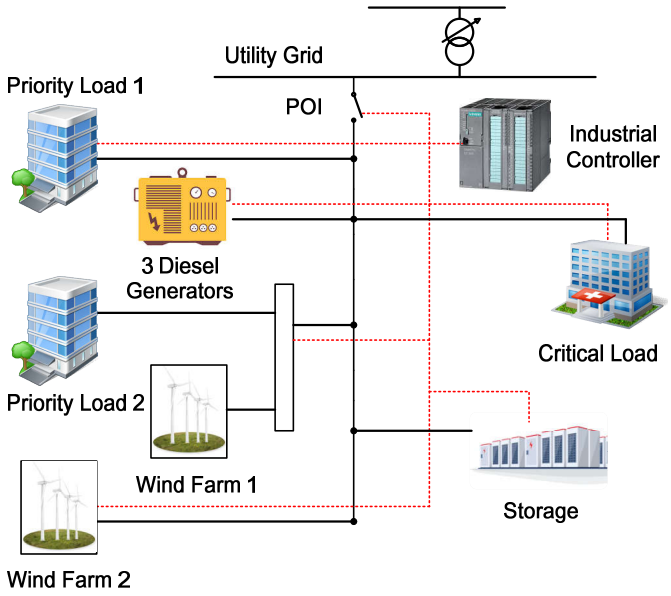

Figure 3.9: Modified CIGRE LV benchmark MG.

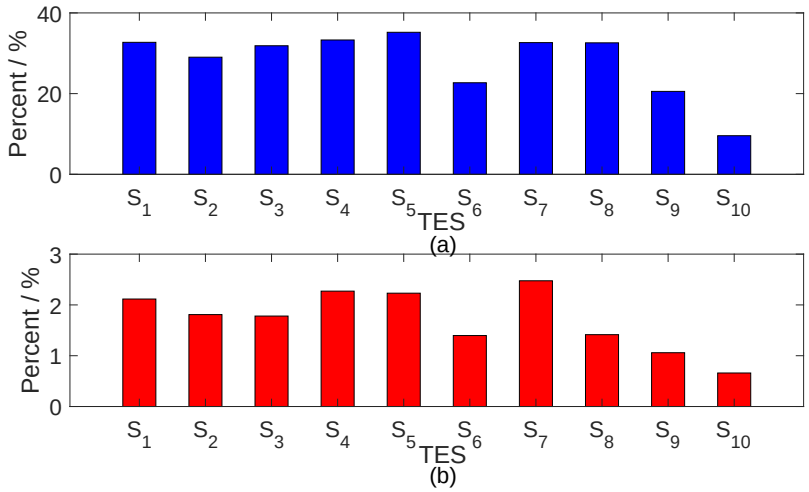

Figure 3.10: (a) Percentage of TGC saved by DT-MGDS compared with Rule-MGDS, (b) TGC optimality loss for DT-MGDS.

At the same time, the optimality loss of DT-MGDS is 2% on average, which means the DT-MGDS still sacrifices little optimality in terms of TGC on a more complicated MG.

3.4.6 Hardware-in-the-Loop (HIL) Validation Results

As is suggested in [93], it is highly relevant to validate the proposed DT-MGDS with real time HIL testing. The real time HIL setup is comprised of one real time simulator, which emulates the MG under study in Figure 3.4 and provides an analog interface to a FPGA-based controller (Xilinx Zynq-7020 platform), in which we implement the DT-MGDS. The analog ports of the simulator output the real time system states of the microgrid. At the other end, the FPGA controller receives the data and sends dispatch decisions back.

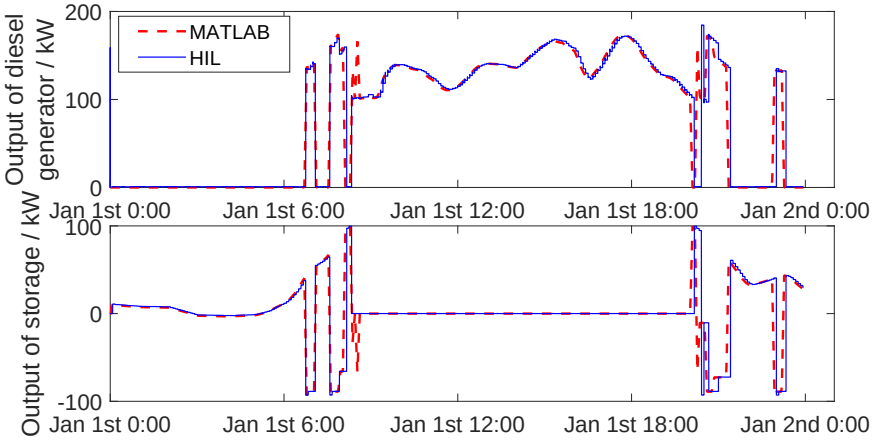

Figure 3.11: Dispatch results achieved by real time HIL simulation and MATLAB simulation on January 1st (TES S_9).

The DT in DT-MGDS are generated with the data set of TAS S_1 in MATLAB. It is tested in real time MG dispatch under TES S_9 and S_{10} from January 1st, 0:00 AM to January 2nd, 0:00 AM. Figure 3.11 and Figure 3.12 illustrate the power profiles of diesel generation and storage achieved by real time HIL and MATLAB simulation. As it can be seen, the power profiles of HIL simulation follow the evolution obtained in MATLAB simulation under both of the scenarios, which confirms the correct behavior of the proposed DT-MEMS. A comparison of the TGC computed for both the real time HIL and MATLAB simulation is listed in Table 3.3. We can see that the TGC obtained from the real time HIL and MATLAB simulation are in accordance, both of which lose little optimality compared with the results obtained by solving the full MIP. Thus, the proposed method can work near optimality in real time operation.

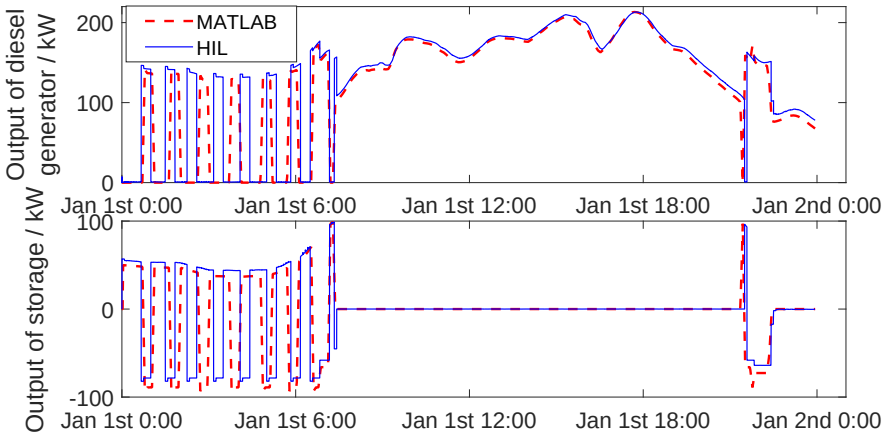

Figure 3.12: Dispatch results achieved by real time HIL simulation and MATLAB simulation on January 1st (TES S_{10}).

Table 3.3: TGC ($) in different simulations on Jan 1st.

TES	HIL	MATLAB	MIP
S_9	817.1	812.3	807.5
S_{10}	1143.7	1121.7	1115.2

The average computational time of each dispatch step of DT-MGDS measured on the FPGA platform is 0.32 second in both test scenarios, which is far below the length of a typical dispatch time step of five minutes. This is because the dispatch decisions are predicted by DT-MGDS in a open-loop process with no iterations, which is quite computationally efficient. Thus, the successful real time implementation highlights the computational efficiency of the proposed DT-MGDS, as it can be programmed into a FPGA platform with limited computing power and memory. The dispatch decisions are generated in a reasonable time to enable eventual online implementation.

Though we have demonstrated that the DT-MGDS can generalize beyond its TAS and achieve high level dispatch performance in a MG with different operating conditions (e.g., dispatchable asset limitations, load profile or renewable generation profile), it is still recommended to monitor the performance of DT-MGDS online when the operating conditions have changed significantly. If the optimality loss exceeds the desired range, it is time to re-train the DT.

3.5 Summary

In this chapter we proposed a decision tree-based optimization approach for implementing flexibility-based operational planning paradigm in real time MG dispatch. The MG in this chapter has a maximum of one dispatchable generator. The verified results have shown that recognizing and partitioning the operating space of the MG dispatch control is essential for providing feasible dispatch decisions. The proposed approach effectively reduces the total generation cost in MG compared with a traditional rule-based dispatch strategy. It also loses little optimality in terms of generation cost compared with the dispatch results obtained by solving the MIP with a state-of-the-art optimization solver. At the same time, the DT-based approach is capable of generalizing beyond its training scenario and achieving high level dispatch performance in a MG with different parameters, load profiles, renewable generation profiles, or other operating conditions. Moreover, the proposed approach is tested on a FPGA platform with limited computing power and memory in the laboratory real-time HIL simulation, which validates its computational efficiency and that it is capable of enabling eventual online real time implementation.

In the next chapter, we consider the general case where the MG has multiple dispatchable generators. A rigorous approach for implementing flexibility-based UC on multiple dispatchable generators in a MG controller with limited computational power is developed.

Chapter 4

Integrating Learning and EMPC for UC in Microgrids

In this chapter, we consider the general case where the MG has multiple dispatchable generators. We apply our flexibility-based operational planning method to microgrid unit commitment. The problem is formulated based on MPC paradigm. As is discussed in Chapter 1, such paradigm requires the online solution of a mixed-integer optimization problem, which faces difficulties in practical field implementations in a low-power computing MG controller. The EMPC can address this this problem of MPC by computing the control laws of MPC in an explicit form offline to enable fast online computation. However, the complexity of the explicit control laws usually grows exponentially with the dimension of the problem, which hinders the application of EMPC in larger systems.

In this chapter, we integrate learning and EMPC to develop a computationally efficient and rigorous approach for implementing flexibility-based UC paradigms in a microgrid controller with limited computational power. The computational complexity of the proposed approach can be adjusted to meet the hardware limitation of any given microgrid controller, while preserving as much as possible the optimality of the full-fidelity EMPC. This overcomes the drawbacks of traditional EMPC. Moreover, compared to the existing learning-based methods for accelerating optimization algorithms, the proposed approach is able to handle the variables and constraints of the original unit commitment problem systematically, which guarantees the feasibility of its output unit commitment schedules. We conduct case studies to demonstrate the effectiveness of the proposed approach.

4.1 Introduction

The deepening penetration of renewable generation brings new challenges to the energy management of MG [55]. Traditionally, a simple priority list is used for making UC schedules in MG [94], [95]. Although such kind of method can achieve fast computing and respect the operating constraints, it is becoming less capable of tackling the volatility brought by the increasing levels of renewable generation and guaranteeing the optimality of its generation schedules. In practice, UC algorithms are usually implemented in an industrial controller with limited computational power. At the same time, it must respond within specified time constraints for real time operation [71]. To overcome these challenges, alternative UC approaches capable of reaching a satisfactory trade-off between the computational efficiency and the optimality of solutions need to be developed.

For years, the concept of flexibility is acknowledged as a key resource for dealing with power fluctuations in microgrid UC. In Chapter 1, we reviewed the state of the art on the flexibility management in MGs. To summarize, the flexibility-based UC algorithms are usually formulated based on a MPC scheme (or receding horizon scheme), which requires the online solution of a mixed integer programming problem at each time step. Although recent literature has shown that flexibility-based UC paradigms can ensure the optimality of the UC schedules without violating operating constraints, running online optimization algorithms requires intensive online computational power and thus prevents the direct implementation of such MPC based UC algorithms in a practical MG where controllers usually have limited computational capabilities.

Recently, a large research effort has been devoted to simplify and accelerate the traditional optimization algorithms for MG energy management using ML techniques. By learning an effective and fast approximation of some underlying heavy computations, the resulting ML models, once well-trained, are computationally amenable to MG controllers with limited computational power. In Chapter 3, we reviewed the state of the art on the application of ML in accelerating MG energy management algorithms. To summarize, though the efficiencies of the proposed ML based algorithms in the existing literature are all promising, none of these has provided a rigorous solution for implementing flexibility-based UC algorithms in a MG controller either because the feasibility of the solution cannot be guaranteed, the commitment statuses of generators are not considered, or the algorithm still relies on complex iterative

procedures.

Another approach to extend the application of MPC-based algorithms to low-power computing applications reported in literature is usually called EMPC [96]. By exploring multi-parametric programming techniques, EMPC partitions the system's state space into different critical regions (CRs). For each CR, the optimal MPC control law is computed in an explicit form offline [97], [98]. In this process, all the variables and constraints of the original optimization problem are handled in a systematic manner, and the feasibility and optimality of the explicit control law are all guaranteed. The online operation of EMPC reduces to finding the CR where the system state is located (this problem is referred to as the *point location problem*) instead of solving optimizations, which significantly reduces the online computational burden in theory. However, in practice the total number of CRs can still grow exponentially with the number of constraints and with the length of the forward-looking horizon [99]. The inherent growth in the complexity of the point location problem and of the memory footprint limits the application of EMPC to small toy systems with short forward-looking horizons only (typically up to three or four time steps [96]). Such a property is a significant limitation in UC problems, where the number of variables is large and the forward-looking horizons are long (can be several hours with a 5-minute time step). A more detailed description of EMPC is presented in Section 4.2.2 of this chapter.

Table 4.1: Comparison of EMPC and ML

Property	**EMPC**	**ML**
Paradigm	model-driven	data-driven
Handling of constraints and variables of the optimization problem	mature ✓	immature ✗
Online computational complexity in large systems	high ✗	low ✓
Insights into the structure of the optimal control laws	yes ✓	no ✗
Guaranteeing the feasibility of solution	mature ✓	immature ✗

As the above discussion shows, the properties of EMPC and ML are clearly complementary (see Table 4.1). Therefore, in this chapter we integrate ML and EMPC to develop a computationally efficient and rigorous solution for implementing flexibility-based UC paradigms in a MG controller with limited computational power. We aim to fully

leverage the benefits of both techniques, while overcoming their respective drawbacks.

We first formulate the UC problem with the concept of flexibility requirement envelopes [6], which is in the form of a MILP and operates in a MPC fashion. The corresponding EMPC model is also formulated. Next, a two-stage learning based approach is developed to approximate the EMPC law of the original UC problem with tunable computational complexity, by accepting a certain level of suboptimality. Specifically, the first stage of the approach uses k-means clustering to group all the CRs in the state space into K clusters. In the second stage, we partition the state space into K disjoint regions, where each region consists of the CRs associated with one of the clusters obtained in the first stage. We train a DNN to identify the K regions in the state space, which is modeled as a classification problem. Finally, we build K ranking lists of the UC options for the K regions in the state space. A UC algorithm is developed using the trained DNN and the K ranking lists, which can be ported to the MG controller to make UC schedules online. The main benefit of our work proposed in this chapter is summarized as follows.

- The proposed approach allows the MG operator to adjust the computational complexity of the final online UC algorithm by accepting a certain level of suboptimality with respect to the original UC problem. It is achieved by tuning the hyper-parameter K, which is the tuning knob to trade-off between the optimality and the computational complexity of the final online UC algorithm. When K is increased, the UC algorithm has higher degree of optimality at the price of higher computational complexity. When the proposed UC algorithm is implemented, the parameter K can be tuned to meet the hardware limitation of the given MG controller, while preserving as much as possible the optimality of the full-fidelity EMPC.
- Compared to the existing ML based methods for accelerating optimization algorithms, the proposed approach is able to leverage the fundamental knowledge of the UC problem, including its variables and constraints, to guarantee the feasibility of the output UC schedules.

4.2 Modelling of Microgrid Unit Commitment

4.2.1 Flexibility Requirement Envelope Based Unit Commitment

We consider a MG with multiple dispatchable generators, energy storage systems, loads and renewable generation. In this chapter, the UC is formulated based on the concept of flexibility requirement envelope which requires UC decisions to consider a forward-looking "envelope" of potential trajectories for net load. Its overall objective is to achieve UC decisions which are cognizant of typical upcoming variability of net load.

First, we construct the flexibility requirement envelope of the net load $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ using the time series of renewable generation and load demand which contain their historical data. T_H is the length of the forward-looking horizon. Further details about calculating the envelope can be found in [6] [1]. The envelope has an up-going ($\uparrow$) and a down-going ($\downarrow$) branch, as shown in Figure 4.1 (a).

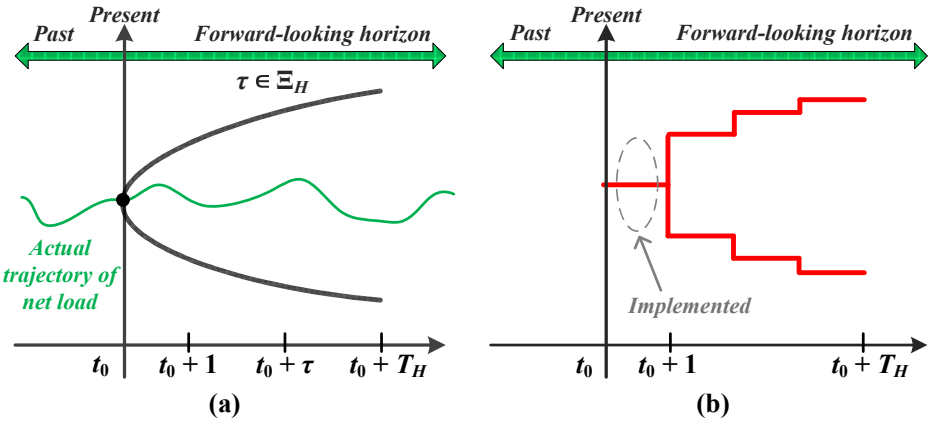

Figure 4.1: (a) Flexibility requirement envelope encompassing the actual net load trajectory. (b) Predicted optimal trajectory of the aggregate output of generators and storage, as driven by the envelope. Only the scheduling decisions for current time step t_0 are implemented.

To formulate the UC, we make the following simplifying assumptions:

[1] Again, please note that the envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ used in this chapter can also be constructed using the approach we proposed in Chapter 2. However, it is more convenient to construct the envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ directly using the method in [6] because this flexibility requirement envelope is assumed to be time invariant in this chapter.

- We do not consider the network constraints of the MG;
- The cost functions of generators are assumed to be piecewise linear;
- There is no operating cost associated with storage systems;
- As is the case with [71], the interaction between the MG and the distribution system is not considered and $p_{poi}(t)$ is treated as a parameter.

We also assume that the envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ is time invariant (i.e., it is a function of the look-ahead time parameter τ only) and thus assume that the variability of net load looking τ units of time in the future is essentially the same at all time t. The resulting MILP of UC (with multiple dispatchable generators) is presented in (E.1) – (E.12) in Appendix E.

At the current time t_0, the UC problem (E.1) – (E.12) is solved. However, only the scheduling decisions for current time step t_0 (i.e., $\tau = 0$) are implemented, as shown in Figure 4.1 (b). The scheduled trajectories of dispatchable generators and storage are assumed to remain constant between $t_0 \rightarrow t_0{+}1$. At the beginning of next time step, $t_0{+}1$, the procedure is repeated over a shifted forward-looking horizon with the updated operating conditions. In this way, the power fluctuations in MG can be handled effectively, thanks to the foresight and self-correcting capabilities of the flexibility-based method.

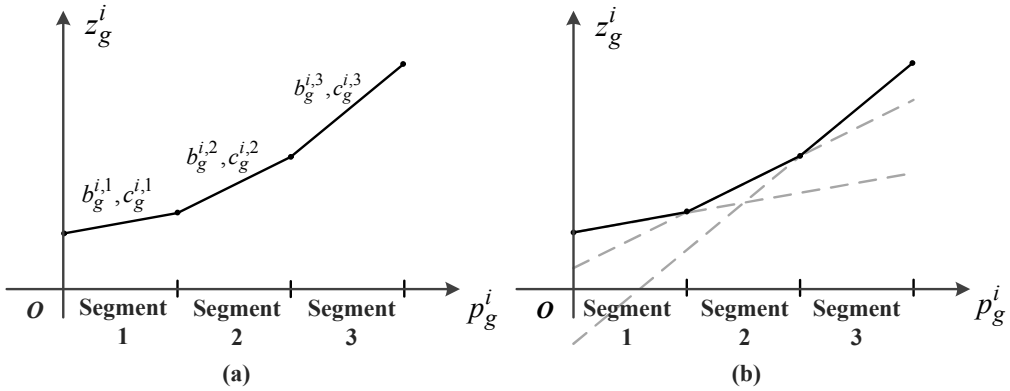

Figure 4.2: (a) Three-segment piecewise linear cost of generator $i \in \Xi_g$. (b) Piecewise linear cost of generator i expressed as the maximum of three linear functions.

The cost function of generator $i \in \Xi_g$ is represented using a piecewise linear function with W_i segments. An example is shown in Figure 4.2 (a) where there are $W_i = 3$ segments. $b_g^{i,w}$ and $c_g^{i,w}$ $(w = 1, 2, \cdots, W_i)$ are the cost parameters of the wth segment associated with generator i. It is worth noting that usually the cost function of a generator is convex [80].

Thus, it can be expressed as the maximum of W_i linear functions, as shown in (E.9) and Figure 4.2 (b). Ahmadi and others [100] provides general discussions about how to choose the best W_i and determine the length of each segment. However, please note that the cost function of a diesel generator, which is widely used in practical MG, can usually be approximated quite well using a straight line (i.e., $W_i = 1$) [101], [102].

For the sake of generality, the upward, downward, start-up and shut-down ramping limits of generators are described in (E.7) and (E.8). However, in practice diesel generators usually have high levels of flexibility in terms of ramping. They can start and supply the load within 10 seconds [103] and ramp up to full capacity within 1 minute [104]. Therefore, the inequality constraints in (E.7) and (E.8) are usually inactive in MG unit commitment. [2]

The network constraints of the MG are ignored in this chapter. This is because in a MG or distribution system the physical network constraints are mainly concerned with node voltage constraints [105]. After the optimal active power outputs of generators and storage systems are obtained by solving (E.1) – (E.12), the controller of the MG will check if there are voltage violations and perform voltage management. The voltage violations can be eliminated by dispatching voltage compensation devices [89] (e.g., capacitors, reactors, static VAR compensators, etc) and correcting the set-points of generators and storage systems (as a last resort) [105], [106]. This process is mostly concerned with reactive power management which goes beyond the scope of this chapter.

4.2.2 Explicit Model Predictive Control Based Formulation

The flexibility-based UC presented above works in a MPC fashion and needs the online solution of the MILP (E.1) – (E.12) at each time step. The significant computational cost of online optimization prevents the implementation of such method in a MG controller with limited computational power.

Alternatively, EMPC can help alleviate this problem. The key observation is that the UC model (E.1) – (E.12) solved repeatedly is fully known except for the upcoming operating

[2] In addition, please note that the minimum uptime and minimum downtime of diesel generators are usually less than 5 minutes [104]. The time step length used in this chapter is 5 minutes. Therefore, the constraints of minimum uptime and minimum downtime associated with diesel generators are not considered in this chapter.

conditions, which are updated at every time step t

$$\boldsymbol{\theta}_t = \left(\hat{d}(t) - p_{poi}(t), \boldsymbol{s}_e(t-1)^\intercal\right)^\intercal \tag{4.1}$$

The state vector $\boldsymbol{\theta}_t$ consists of the residual net load $\hat{d}(t) - p_{poi}(t)$ and the vector $\boldsymbol{s}_e(t-1)$ that contains the energy level $s_e^j(t-1)$ of each storage system $j \in \Xi_e$. $\hat{d}(t) = d(t) - \delta(t)$ is the net load in the system. Please note that the envelope $\{e_A^\uparrow(\tau), e_A^\downarrow(\tau)\}_{\tau\in\Xi_H}$ is not included because it is time invariant. The space of $\boldsymbol{\theta}_t$ is denoted as $\boldsymbol{\Theta}$. The MILP (E.1) – (E.12) can be expressed by the following compact matrix form

$$v_t^* = \min \quad \boldsymbol{c}_c^\intercal \boldsymbol{x}_c + \boldsymbol{c}_d^\intercal \boldsymbol{x}_d \tag{4.2}$$

$$s.t. \quad \boldsymbol{A}_c \boldsymbol{x}_c + \boldsymbol{A}_d \boldsymbol{x}_d \leq \boldsymbol{b} + \boldsymbol{B}\boldsymbol{\theta}_t \tag{4.3}$$

where $\boldsymbol{x}_c$ is the vector consisting of the continuous variables $p_g^i(\tau, q)$ and $p_e^j(\tau, q)$, while $\boldsymbol{x}_d$ is the vector containing the discrete variables $u_g^i(\tau, q)$ ($i \in \Xi_g$, $j \in \Xi_e$, $\tau \in \Xi_H$ and $q \in \{\uparrow, \downarrow\}$). The coefficient matrices and vectors in the above matrix formulation are constant and can be obtained from the parameters in (E.1) – (E.12). To save space, we omit them here.

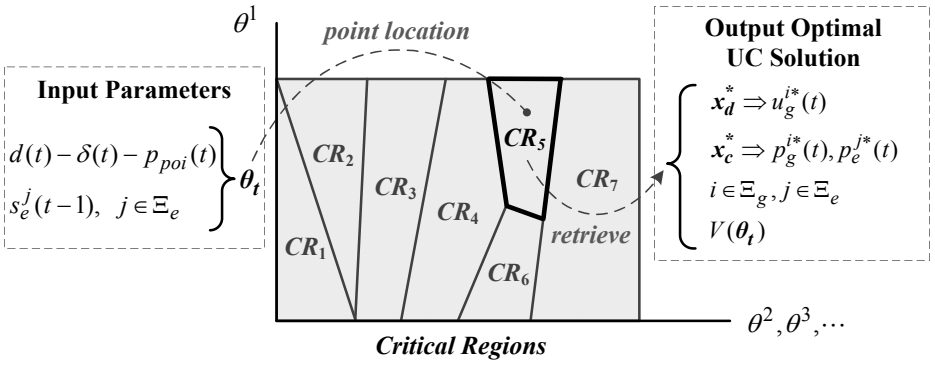

Figure 4.3: Schematic illustration of the idea of EMPC method. For illustrative purpose, the state space consists of seven CRs only. However, please note that the number of CRs in the state space for a UC problem is usually quite large.

Figure 4.3 illustrates the idea of EMPC. The EMPC treats $\boldsymbol{\theta}_t$ as a vector of input parameters, and the MILP (4.2) – (4.3) becomes a multi-parametric mixed-integer linear programming (mp-MILP) problem [96]. Through mp-MILP, one can obtain the variables of

the original MILP problem (4.2) – (4.3) as an explicit function of the vector $\boldsymbol{\theta}_t$ [107]. This is achieved by partitioning the state space $\boldsymbol{\Theta}$ into different CRs [96]. A CR is the region in the state space $\boldsymbol{\Theta}$ where the optimal integer variables $\boldsymbol{x}_d^*$ and the active constraints of (4.3) (for determining the optimal continuous variables $\boldsymbol{x}_c^*$) are the same. CRs can be determined using a variety of approaches. For example, the authors of [97] decomposed the mp-MILP problem into two subproblems (a MILP problem and a multi-parametric linear programming problem) and then iterated between them to obtain the final CRs analytically. An alternative approach was taken in [108], [109] and [110], where the authors employed ML approaches (e.g., DNN or support vector machine) to identify the CRs in the state space. Specifically, the identification of CRs was formulated as a classification problem, where a DNN or support vector machine was trained to learn the mapping between the state vectors and the CRs. Please note that the calculations of CRs are all performed offline in the literature we discussed above.

In addition, $V(\boldsymbol{\theta}_t)$ is the value function (VF) [96] which associates with every state vector $\boldsymbol{\theta}_t$ the corresponding optimal value v_t^* of (4.2). According to the theory of mp-MILP, $V(\boldsymbol{\theta}_t)$ is piecewise polyhedral [107, 111, 112]. Each CR maps to a distinct hyperplane, as is shown in Figure 4.4.

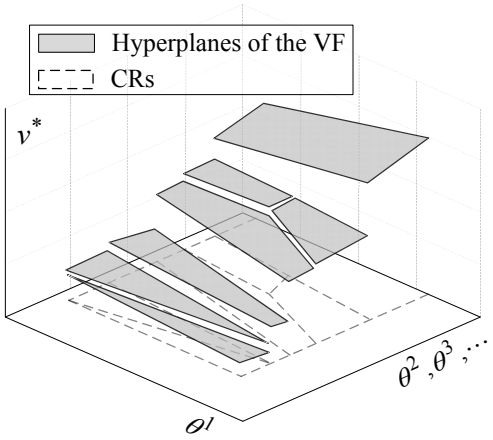

Figure 4.4: Illustration of the piecewise polyhedral VF $v_t^* = V(\boldsymbol{\theta}_t)$.

The EMPC stores each CR along with its corresponding optimal integer variables, active constraints and VF precomputed offline in a lookup table. Consequently, the online operations of EMPC are reduced to a simple location problem of the state $\boldsymbol{\theta}_t$ in the lookup

table at each time step (see Figure 4.3) instead of solving the full MILP, which greatly reduces the online computational burden in theory. However, in practice, the exponential growth in the number of CRs and the possible explosion in the need for memory and CPU requirements still hinder the application of EMPC method in UC problems.

4.3 Methodology

To address the limitations of the traditional EMPC method, in this section we integrate ML techniques with it to design a high quality approximate UC solver that is implementable in practical settings. The overall workflow of the proposed method is shown in Figure 4.5, which is divided into the training phase and the testing phase. Both the training and testing phases require offline simulation runs of the UC problem (4.2) – (4.3) based on realistic renewable generation and load data. The collections of input data and simulated UC results are the raw materials of the training and testing phases.

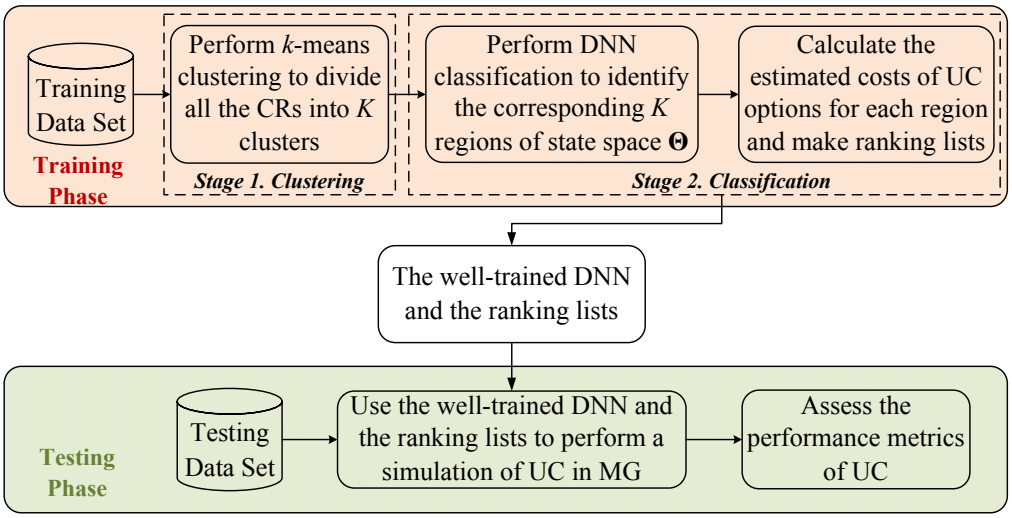

Figure 4.5: Overall workflow of the proposed learning based method.

4.3.1 Data Set Preparation

For the application of ML techniques we first create a training data set of pre-solved UC problems. In order to do this, we need to define a Training Scenario (TAS) as follows.

(1) The length of the offline simulation T and the length of each time step T_Δ;
(2) All the parameters of each dispatchable generator $i \in \Xi_g$ and energy storage system $j \in \Xi_e$;
(3) Time series of $\{d(t)\}_{t=1}^T$, $\{\delta(t)\}_{t=1}^T$ and $\{p_{poi}(t)\}_{t=1}^T$.

The TAS contains the historical data and MG parameters that will be fed into a state-of-the-art MILP solver to generate the corresponding optimal UC solutions $\{u_g^{i*}(t)\}_{t=1}^T$, $\{p_g^{i*}(t)\}_{t=1}^T$, $\{p_e^{j*}(t)\}_{t=1}^T$ and $\{s_e^{j*}(t)\}_{t=1}^T$ ($i \in \Xi_g$, $j \in \Xi_e$). Here we only record the first step of all the scheduling decisions obtained by solving (4.2) – (4.3) for each time t according to the principle of MPC. The time series of $\{\boldsymbol{\theta}_t\}_{t=1}^T$ and $\{v_t^*\}_{t=1}^T$ are also obtained.

Table 4.2: Dispatch Options in the MG

Index	Description	Index	Description
1	$p_g^{M*}(t) = \overline{p_g^M}$	2	$p_g^{M*}(t) = \underline{p_g^M}$
3	$\sum_{j\in\Xi_e} p_e^{j*}(t) = \sum_{j\in\Xi_e} \overline{p_e^j}$	4	$\sum_{j\in\Xi_e} p_e^{j*}(t) = \sum_{j\in\Xi_e} \underline{p_e^j}$
5	$\sum_{j\in\Xi_e} s_e^{j*}(t) = \sum_{j\in\Xi_e} \overline{s_e^j}$	6	$\sum_{j\in\Xi_e} s_e^{j*}(t) = \sum_{j\in\Xi_e} \underline{s_e^j}$

Instead of storing the optimal UC solutions directly in the training data set, we define the following *UC option* a_t for each time step t [3].

$$a_t = \{\boldsymbol{u}_g^*(t), y_t\} \tag{4.4}$$

It consists of the vector $\boldsymbol{u}_g^*(t)$ that contains the optimal commitment status $u_g^{i*}(t)$ of each generator $i \in \Xi_g$ and a *dispatch option* y_t. The dispatch option is used to characterize the optimal power outputs of all the generators and storage. Depending on the patterns of the active constraints in (4.3), y_t maps to one of the six options in Table 4.2. Note that we only focus on the marginal generator and the total output of all the storage systems. This

[3]The UC option is only used to characterize the first control step of all the scheduling decisions obtained by solving (4.2) – (4.3). Thus, a UC option is insufficient for characterizing a CR. Because we need the scheduling decisions of the entire forward-looking horizon for defining a CR.

is because there is only one marginal generator (a marginal generator is the generator that serves the next unit incremental load [113], its output is denoted as p_g^M) in the MG given the piecewise linear operating cost functions. Other committed generators will simply operate at their maximum or minimum generation limits. At the same time, none of the storage systems have operating costs, and thus we only need to know the total power output of the storage systems. The power output of each storage system can be further determined using a pre-determined power sharing method (e.g. [114]). Therefore, the power outputs of all the committed generators and storage can be fully determined using one of the dispatch options in Table 4.2 and the power balance equation (E.2) [4]. The UC option a_t thus constitutes the minimal information required to recover the corresponding full optimal UC solution, which reduces the number of variables to be handled in our learning task by exploiting the pre-existing knowledge of the UC problem.

In addition, we fix the integer variables $\boldsymbol{x}_d = \boldsymbol{x}_d^*$ after solving (4.2) – (4.3) and the MILP becomes a linear programming (LP) problem. We also record the vector of shadow prices $\boldsymbol{\lambda}_t$ associated with (4.3) for every time step t, which can be obtained by performing LP sensitivity analysis.

4.3.2 The Proposed Learning Based Approach

In this section, we develop a two-stage learning based approach whose framework is shown in Figure 4.5, which exploits the combined use of clustering and classification techniques. Details of each stage are introduced below.

Stage 1. Clustering

In this stage, we divide all the CRs in the state space into K clusters. The CRs will be clustered based on the similarities of their corresponding hyperplanes of the VF.

We first identify the parameters of the hyperplanes of the piecewise polyhedral VF from the time series $\{\boldsymbol{\theta}_t\}_{t=1}^T$ and $\{v_t^*\}_{t=1}^T$. Each data point $(\boldsymbol{\theta}_t, v_t^*)$ is located on one of the hyperplanes of the VF, as shown in Figure 4.6 (a). Therefore, the optimal objective

[4] Although the term 'dispatch option' is used both in Table 4.2 and Table 3.1, it has different meanings. The dispatch options in Table 4.2 consist of the active constraints that are used to determine the optimal power outputs of all generators and storage (i.e., continuous variables), whereas the dispatch options in Table 3.1 are mainly used to characterize the ON/OFF status of the dispatchable generator (i.e., integer variable).

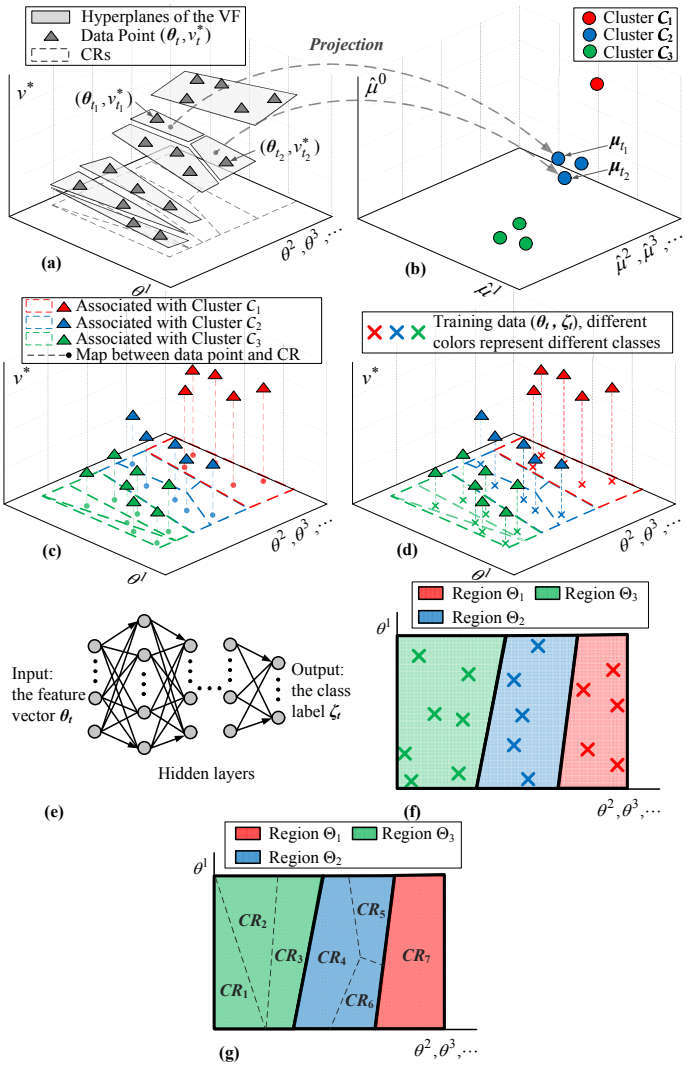

Figure 4.6: (a) Hyperplanes of the piecewise polyhedral VF and the corresponding CRs. (b) Projections of the hyperplanes of the VF, which are grouped into $K = 3$ clusters. (c) All the data points are also grouped into 3 clusters. The CRs are also divided into 3 clusters implicitly. (d) Maps among the data point $(\boldsymbol{\theta}_t, v_t^*)$, the feature vector $\boldsymbol{\theta}_t$, and the cluster ζ_t. (e) General structure of the DNN classifier. (f) Illustration of the classification process. (g) The three regions in the state space consist of the CRs associated with the three clusters in (c), respectively.

function value v_t^* can be expressed in the following form

$$v_t^* = \hat{\boldsymbol{\mu}}_t^\intercal \boldsymbol{\theta}_t + \hat{\mu}_t^0 \tag{4.5}$$

$\hat{\boldsymbol{\mu}}_t$ is the parameter vector associated with $\boldsymbol{\theta}_t$, which can be calculated using the corresponding vector of shadow prices $\boldsymbol{\lambda}_t$.

$$\hat{\boldsymbol{\mu}}_t = \boldsymbol{B}^\intercal \boldsymbol{\lambda}_t \tag{4.6}$$

where $\boldsymbol{B}$ is the constant parameter matrix in (4.3). $\hat{\mu}_t^0$ is a scalar which can be determined using (4.5) after $\hat{\boldsymbol{\mu}}_t$ is calculated. It is worth noting that (4.6) assumes the optimal integer variables do not change in a small neighbourhood of $\boldsymbol{\theta}_t$.

To illustrate our clustering procedure efficiently, we define

$$\boldsymbol{\mu}_t = \left(\hat{\mu}_t^0, \hat{\boldsymbol{\mu}}_t^\intercal\right)^\intercal \tag{4.7}$$

$\boldsymbol{\mu}_t$ is the vector of parameters that fully characterizes the hyperplane of the VF containing the data point $(\boldsymbol{\theta}_t, v_t^*)$. Then, all the data points on the same hyperplane of the VF are projected onto the same point in the space of $\boldsymbol{\mu}_t$, as shown in Figure 4.6 (b). To effectively perform clustering, we apply min-max scaling [86] to each component of the vector in the time series $\{\boldsymbol{\mu}_t\}_{t=1}^T$ so that they end up ranging from 0 to 1. The min-max scaling is done by subtracting the minimal value and dividing by the maximal value minus the minimal value.

In this chapter, the similarity between any two hyperplanes of the VF is measured by the L_1-distance between their projections in the space of $\boldsymbol{\mu}_t$. For example, the hyperplanes containing the data points $(\boldsymbol{\theta}_{t_1}, v_{t_1}^*)$ and $(\boldsymbol{\theta}_{t_2}, v_{t_2}^*)$ in Figure 4.6 (a) are projected onto two different points $\boldsymbol{\mu}_{t_1}$ and $\boldsymbol{\mu}_{t_2}$ in Figure 4.6 (b). The similarity between the two hyperplanes of the VF is measured by

$$\begin{aligned} D_1(\boldsymbol{\mu}_{t_1}, \boldsymbol{\mu}_{t_2}) &= \textstyle\sum_l \left|\boldsymbol{\mu}_{t_2}^l - \boldsymbol{\mu}_{t_1}^l\right| \\ &= \left|\hat{\mu}_{t_2}^0 - \hat{\mu}_{t_1}^0\right| + \textstyle\sum_l \left|\hat{\boldsymbol{\mu}}_{t_2}^l - \hat{\boldsymbol{\mu}}_{t_1}^l\right| \end{aligned} \tag{4.8}$$

where $D_1(\cdot,\cdot)$ is the L_1-distance and $(\cdot)^l$ is the lth component of the vector.

Next, we perform k-means clustering on the data $\{\boldsymbol{\mu}_t\}_{t=1}^T$. k-means clustering is selected

because of its efficiency and simplicity in dealing with large data sets [87]. To perform k-means clustering, we need to first specify the desired number of clusters K. Then the T observations $\{\boldsymbol{\mu}_t\}_{t=1}^{T}$ will be grouped into K clusters $\{\mathcal{C}_r\}_{r=1}^{K}$ using the distance metric in (4.8). An example is shown in Figure 4.6 (b), where there are $K = 3$ clusters. The detailed procedure of k-means clustering can be found in [87].

At the end of this stage, each data point $(\boldsymbol{\theta}_t, v_t^*)$ is associated to a cluster, which is denoted as ζ_t ($\zeta_t \in \{\mathcal{C}_r\}_{r=1}^{K}$), see Figure 4.6 (c). At the same time, all the CRs in the state space $\boldsymbol{\Theta}$ are also grouped into K clusters implicitly, because of the maps between the data points and the CRs, as is shown in Figure 4.6 (c).

Stage 2. Classification

In this stage, we partition the state space $\boldsymbol{\Theta}$ into K disjoint regions $\{\Theta_r\}_{r=1}^{K}$, where Θ_r consists of the CRs associated with cluster $\mathcal{C}_r$ ($r = 1, 2, \cdots, K$).

The identification of $\{\Theta_r\}_{r=1}^{K}$ in the state space underlies a classification problem, where the K clusters $\{\mathcal{C}_r\}_{r=1}^{K}$ can be regarded as K classes. $\boldsymbol{\theta}_t$ is the feature vector and each feature vector is associated to a class label ζ_t ($\zeta_t \in \{\mathcal{C}_r\}_{r=1}^{K}$), as is shown in Figure 4.6 (d). Again, we first apply min-max scaling to each component of the feature vector in time series $\{\boldsymbol{\theta}_t\}_{t=1}^{T}$.

We use a fully connected feedforward DNN for our classification task. DNN has strong approximation abilities and scales well with massive data sets [5]. The DNN has an input layer, multiple hidden layers and an output layer. Figure 4.6 (e) shows the general structure of the DNN classifier. In this study, the rectifier linear units (ReLU) function is used as the activation function [115]. The number of hidden layers and the number of neurons in each hidden layer are hyper-parameters which should be selected according to the number of regions in the state space (i.e., K). Details about tuning the hyper-parameters and training the DNN can be found in [115]. [6]

The well-trained DNN can identify the regions $\{\Theta_r\}_{r=1}^{K}$ of the state space $\boldsymbol{\Theta}$ by learning the separating boundaries between different classes from the training data $\{(\boldsymbol{\theta}_t, \zeta_t)\}_{t=1}^{T}$. An

[5] In this chapter, decision trees are not used because the state space partition generated by a decision tree is always orthogonal. However, the boundaries of CRs can be general. Moreover, for a given architecture of DNN, we can compute the lower bound on the maximal number of regions in the state space that this architecture can represent (details will be described later in this chapter).

[6] Please note that the DNN is trained using batch learning [86] (i.e., the DNN is trained using the entire data set, which is done offline). First the DNN is trained, and then the well-trained DNN is launched into production and runs without learning anymore.

example is shown in Figure 4.6 (f) where the state space $\boldsymbol{\Theta}$ is divided into $K = 3$ regions (classes). Moreover, the three regions in the state space consist of the CRs associated with the three clusters in Figure 4.6 (c), respectively, as shown in Figure 4.6 (g). Note that the seven CRs in Figure 4.6 (g) are the same as those in Figure 4.3.

After all the K regions $\{\Theta_r\}_{r=1}^{K}$ are identified, we calculate the estimated cost of any UC option $\hat{a}$ in each region Θ_r $(r = 1, 2, \cdots, K)$, $\overline{C}(\hat{a}, \Theta_r)$, as follows

$$\overline{C}(\hat{a}, \Theta_r) = \frac{1}{N(\hat{a}, \Theta_r)} \sum_{t \in \{t | a_t = \hat{a} \text{ and } \theta_t \in \Theta_r\}} v_t^* \tag{4.9}$$

where $N(\hat{a}, \Theta_r)$ is the number of times that $\hat{a}$ is chosen in region Θ_r in the training data set. $\overline{C}(\hat{a}, \Theta_r)$ represents the sample mean of the total operating cost for the forward-looking horizon after choosing UC option $\hat{a}$ at current time step in region Θ_r. Thus, it measures the direct operating cost associated with $\hat{a}$, while taking into account the "potential" impact of $\hat{a}$ on the operating cost of future time steps.

For each region Θ_r $(r = 1, 2, \cdots, K)$, we make a ranking list RL_r by sorting each UC option $\hat{a}$ in ascending order of its corresponding estimated cost $\overline{C}(\hat{a}, \Theta_r)$. Thus, the first option in the list is the one with the lowest estimated cost. Note that if a UC option has never been chosen in region Θ_r in the training data set, it will not be included in RL_r.

4.3.3 Testing of the Learning Based Approach

In order to test the proposed learning based approach for UC, we first define a Testing Scenario (TES). A TES could have different time series of demand, renewable generation, or power exchange at the POI compared to the TAS. The final output of the learning based approach (i.e., the DNN and the K ranking lists $\{RL_r\}_{r=1}^{K}$) will be used to perform a simulation of UC under the TES. Figure 4.7 (a) shows the flowchart of the proposed online UC algorithm using the DNN and the ranking lists. At each time step t, the system updates the feature vector $\boldsymbol{\theta}_t$ according to the updated operating conditions in the MG. It then classifies the updated $\boldsymbol{\theta}_t$ with the DNN. It determines in which region Θ_r $(r = 1, 2, \cdots, K)$ that $\boldsymbol{\theta}_t$ belongs to. The corresponding ranking list RL_r is retrieved afterwards.

Then the first UC option in RL_r is analyzed to see if it is a feasible UC option for the MG at time t. In this chapter, the following method is used to ensure the feasibility.

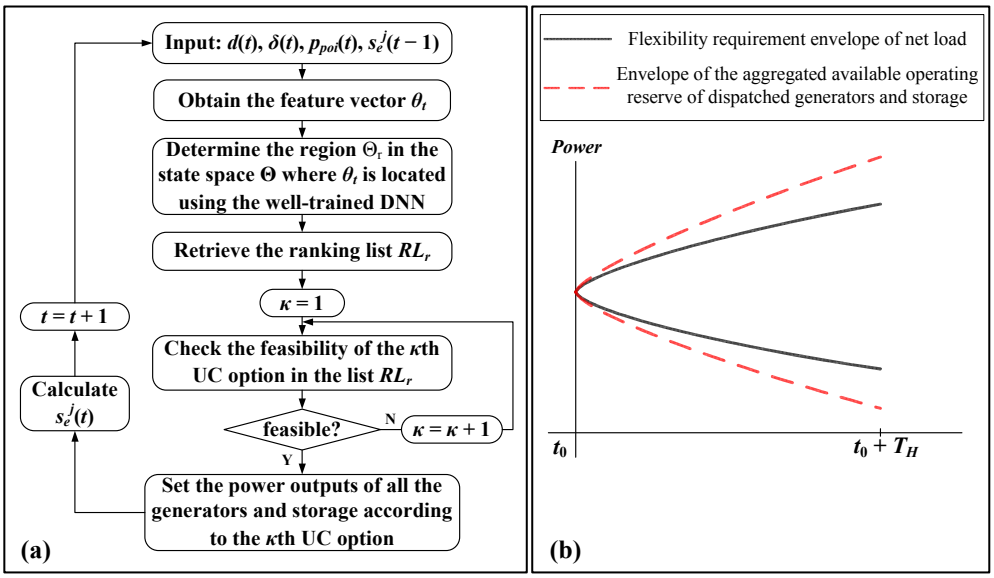

Figure 4.7: (a) Flowchart of the proposed online UC algorithm for MG. (b) Proper enclosure of the flexibility requirement envelope.

We first calculate the envelope of the aggregated available operating reserve of dispatched generators and storage using the method in [29]. Then we check if the envelope of the aggregated available operating reserve is able to enclose the flexibility requirement envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$ of net load constructed using the data in the TES, as illustrated in Figure 4.7 (b). The proper enclosure of the flexibility requirement envelope guarantees that the power balance (E.2) and the operating constraints of individual units (E.3) – (E.6) can all be satisfied in the forward-looking horizon. If the current UC option cannot ensure the enclosure of the flexibility requirement envelope, we then go to the next UC option in the list RL_r until the feasibility of the UC option is guaranteed [7]. Finally, the performance of the proposed online UC algorithm in Figure 4.7 (a) will be assessed based on its UC schedules produced in simulation.

[7] If all the UC options in the ranking list are exhausted, load shedding or generation curtailment can be used as the last resort.

4.3.4 Discussion

As mentioned before, the main drawback of traditional EMPC is its exponentially increasing number of CRs and the possible explosion in the need for memory and CPU requirements, which makes the implementation intractable for a MG controller with limited computational power.

To tackle this problem, our proposed learning-based approach first clusters the CRs based on the similarities of their corresponding hyperplanes of the VF. The CRs associated with the same cluster are merged, so that the state space is partitioned into a pre-specified number (i.e., the number of clusters K, see Figure 4.6 (g)) of regions. Note that usually K is far less than the total number of CRs in the state space. Thus, our approach is able to eliminate the problem of excessive partitioning in traditional EMPC.

However, reducing the number of partitions of the state space may introduce suboptimality to the UC schedules produced by the algorithm in Figure 4.7 (a). An example is shown in Figure 4.8, where we select two hyperplanes of the VF and two data points on them in Figure 4.6 (a) for illustration.

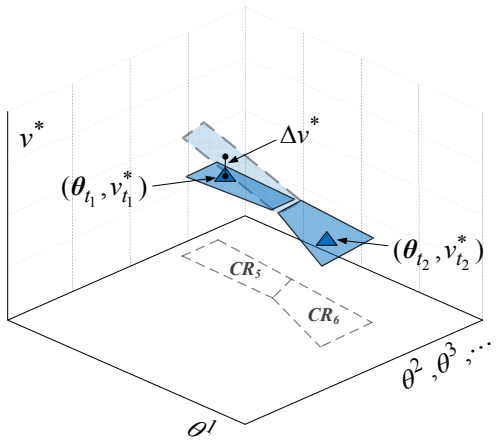

Figure 4.8: The optimality loss of the UC schedule after merging different CRs.

We can see from Figure 4.6 (g) that CR_5 and CR_6 are merged into region Θ_2. If the estimated of a_{t_2} is lower than that of a_{t_1} (i.e., $\overline{C}(a_{t_2}, \Theta_2) < \overline{C}(a_{t_1}, \Theta_2)$), a_{t_2} will have a higher priority than a_{t_1} in the ranking list of region Θ_2.

Then, the algorithm in Figure 4.7 (a) may choose UC option a_{t_2} under state $\boldsymbol{\theta}_{t_1}$ (if it is still feasible) since a_{t_2} has higher priority in region Θ_2. However, a_{t_2} is suboptimal under

$\boldsymbol{\theta}_{t_1}$, as can be seen from the blue plane in dotted frame in Figure 4.8 [8]. The optimality loss of choosing a_{t_2} under state $\boldsymbol{\theta}_{t_1}$ is denoted by Δv^*, which has the following property.

$$\begin{aligned}
\Delta v^* &= \left|(\hat{\mu}^0_{t_2} - \hat{\mu}^0_{t_1}) + \textstyle\sum_l \boldsymbol{\theta}^l_{t_1}(\hat{\boldsymbol{\mu}}^l_{t_2} - \hat{\boldsymbol{\mu}}^l_{t_1})\right| \\
&\leq \left|\hat{\mu}^0_{t_2} - \hat{\mu}^0_{t_1}\right| + \textstyle\sum_l \left|\boldsymbol{\theta}^l_{t_1}(\hat{\boldsymbol{\mu}}^l_{t_2} - \hat{\boldsymbol{\mu}}^l_{t_1})\right| \\
&= \left|\hat{\mu}^0_{t_2} - \hat{\mu}^0_{t_1}\right| + \textstyle\sum_l \left(\left|\boldsymbol{\theta}^l_{t_1}\right| \cdot \left|\hat{\boldsymbol{\mu}}^l_{t_2} - \hat{\boldsymbol{\mu}}^l_{t_1}\right|\right) \\
&\leq \left|\hat{\mu}^0_{t_2} - \hat{\mu}^0_{t_1}\right| + \textstyle\sum_l \left|\hat{\boldsymbol{\mu}}^l_{t_2} - \hat{\boldsymbol{\mu}}^l_{t_1}\right| \\
&= D_1(\boldsymbol{\mu}_{t_1}, \boldsymbol{\mu}_{t_2})
\end{aligned} \tag{4.10}$$

Note that the value of each component of $\boldsymbol{\theta}_{t_1}$ is between 0 and 1 because of the min-max scaling applied. We can see that the optimality loss Δv^* is bounded by the L_1-distance $D_1(\boldsymbol{\mu}_{t_1}, \boldsymbol{\mu}_{t_2})$ between the two hyperplanes of VF in Figure 4.8. [9]

When the number of clusters K becomes larger, the maximum pairwise L_1-distance between the hyperplanes of VF in each cluster $\mathcal{C}_r$ $(r = 1, 2, \cdots, K)$ will be smaller. The upper bound of the optimality loss of the UC schedules produced by the proposed UC algorithm in Figure 4.7 (a) is thus expected to be smaller. However, in such cases, the state space is divided into more regions and there are more ranking lists. Then the computational complexity of the UC algorithm is expected to be higher. Thus, K is a tuning knob to trade-off between the optimality and the computational complexity of the proposed UC algorithm.

The proposed UC algorithm in Figure 4.7 (a) is promising to enable real-time implementation in a computationally limited MG controller because its computational complexity is tunable. For any given MG controller, K can be pre-specified, which is equal to the value that the memory or any other online computational bound is met. This value of K also allows the proposed UC algorithm to preserve the optimality of the full-fidelity EMPC to the maximum possible extent.

[8] The blue plane in dotted frame is obtained by extending the hyperplane of VF corresponding to CR_6.

[9] Please note that this is just an illustrative example showing how our proposed algorithm can introduce suboptimality to its output UC schedules. If $\overline{C}(a_{t_2}, \Theta_2) > \overline{C}(a_{t_1}, \Theta_2)$ or if a_{t_2} is not feasible under $\boldsymbol{\theta}_{t_1}$, then the proposed algorithm will still choose a_{t_1} at time t_1 and no suboptimality will be introduced in this example. However, the suboptimality can still be introduced by other data points and hyperplanes in Figure 4.6 (a) in a similar way.

4.4 Case Study

4.4.1 Test System Description

The proposed learning based approach is tested on a modified CIGRE LV benchmark MG. The diagram of the original benchmark MG can be found in [92]. The modified MG has priority loads, one wind farm, three diesel generators, one centralized storage system and one critical load.

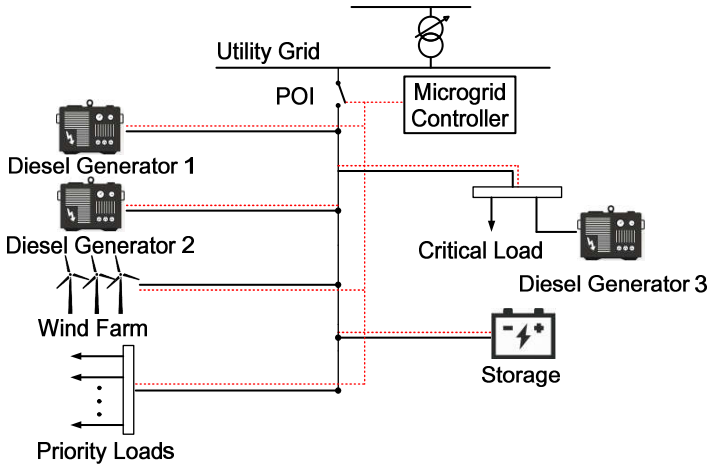

Figure 4.9: Modified CIGRE LV benchmark MG under study.

Scenarios Studied

We construct the following four scenarios S_1 – S_4 for training and testing of the proposed learning based approach [10]. In simulation, T_Δ is set to be 5 minutes and T is one year (i.e., 105 120 5-minute time intervals). The length of the forward-looking horizon T_H is one hour. Our overall objective here is to demonstrate how well the proposed learning based approach is able to generalize over various testing scenarios (i.e., not require re-training when faced with new operating conditions).

S_1: For the three diesel generators: $\overline{p_g^1} = 400$ kW, $\overline{p_g^2} = 280$ kW and $\overline{p_g^3} = 200$ kW. The minimum power output of each generator is set to be 30% of their corresponding maximum

[10] Please note that each scenario requires a new data set. Moreover, when the configuration of MG has changed (e.g., Section 4.4.5), a new data set is also required.

power limit. The cost parameters of each generator can be found in [116]. The cost functions of the diesel generators are represented using straight lines (i.e., $W_i = 1, i = 1, 2, 3$). The cost of diesel fuel is set to \$1.30/l [42]. The upward, downward, start-up and shut-down ramping limits of each generator are all set to be 100% of its capacity in 5 minutes [11]. For storage: $\overline{p_e} = 100$ kW, $\underline{p_e} = -100$ kW, $\overline{s_e} = 90$ kWh, $\underline{s_e} = 10$ kWh and $\eta_c = \eta_d = 0.963$. The state-of-charge (SOC) of storage is initialized at 50% of maximum capacity. The maximum and minimum values of the aggregate priority load are 750 kW and 150 kW, respectively. The data for the priority loads are taken from a village in the Canadian province of Quebec. The critical load is kept constant at 50 kW. The capacity of wind farm is set to be 150 kW, and the data of wind generation come from NREL's database [66]. The MG is in grid-connected mode. The contractual value of the power exchange at POI $p_{poi}(t)$ is kept constant for each day of the year, which is set to be the minimal net load of any given day.

S_2: Next year's time series of priority loads are used.

S_3: Next year's time series of wind generation is used.

S_4: The MG is in islanded mode for the whole year.

Other parameters and time series in Scenarios S_2 – S_4 are same as those in Scenario S_1.

UC Strategies

Three microgrid UC strategies are simulated for comparison purposes.

- *LB-MGUC:* This is our proposed learning based microgrid UC strategy in Figure 4.7 (a).
- *PL-MGUC:* This is a traditional simple priority list based microgrid UC strategy previously proposed in [95].
- *AP-CLUC:* This is the adaptive contextual learning based microgrid UC strategy previously proposed in [61]. A brief introduction of this UC strategy is provided in Appendix F.

Performance Metrics

The UC strategies are mainly evaluated by comparing their respective Total Generation Cost (TGC) when facing the same operating scenarios. Energy Not Served (ENS) and Energy

[11] Remember that diesel generators can start and supply the load within 10 seconds and ramp up to full capacity within 1 minute [103], [104]. The length of each time step here is 5 minutes.

Curtailed (EC) are also considered as further performance metrics.

$$TGC = \sum_{t=1}^{T} \sum_{i \in \Xi_g} b_g^i p_g^i(t) + c_g^i u_g^i(t) \tag{4.11}$$

$$ENS = T_\Delta \sum_{t=1}^{T} d_c(t) \tag{4.12}$$

$$EC = T_\Delta \sum_{t=1}^{T} \delta_c(t) \tag{4.13}$$

Analysis Tools

The model (E.1) – (E.12) is solved using the commercial MILP solver CPLEX. The proposed learning based approach is validated using the ML software from MATLAB [91]. The computer used is equipped with an Intel Core i5 2.70 GHz processor and 8 GB RAM.

4.4.2 Testing of the Proposed Learning Based UC Strategy

First, the proposed LB-MGUC is compared against the PL-MGUC. Here, the LB-MGUC is set up and trained under TAS S_1. The number of regions in the state space K is set to be six. We use a DNN which has three hidden layers and each hidden layer has six neurons [12]. Afterwards, the performance of the LB-MGUC is tested under TES S_m $(m = 1, 2, 3, 4)$. We first compare the TGC of LB-MGUC with the TGC of PL-MGUC calculated under TES S_m. The percentages of TGC saved by LB-MGUC are shown in Figure 4.10 (a). We see that the LB-MGUC saves 23% on average in terms of TGC compared with PL-MGUC over the four TES. Then, the AP-CLUC is set up under TAS S_1 [13] and its TGC calculated under TES S_m $(m = 1, 2, 3, 4)$ is compared with LB-MGUC. The results are presented in Figure 4.10 (b). The proposed LB-MGUC saves an average of 3.3% of TGC compared with AP-CLUC across the four TES. Next, we compare the LB-MGUC with the UC results which would

[12] According to [115], a lower bound on the maximal number of regions that a DNN (with ReLUs as activation functions) can represent is given by the equation $\left(\prod_{\kappa=1}^{N_L-1} \left\lfloor \frac{N_M}{N_x} \right\rfloor^{N_x}\right) \sum_{\kappa=0}^{N_x} \binom{N_L}{\kappa}$, where N_L is the number of hidden layers, N_x is the number of inputs and N_M is the number of neurons in each hidden layer. Thus, this DNN configuration can represent at least 567 regions. Note that the state space is divided into 256 regions at maximum in all the numerical examples of this chapter. Therefore, this DNN configuration will be used throughout the rest of this chapter.

[13] We tune the parameter of AP-CLUC so that it has the same number of regions in the state space as LB-MGUC.

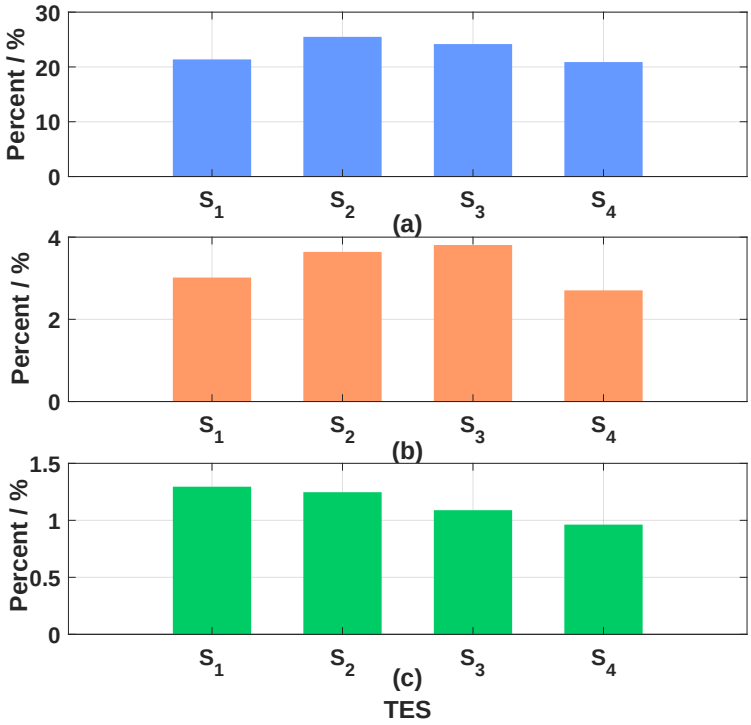

Figure 4.10: (a) Percentages of TGC saved by LB-MGUC compared with PL-MGUC, (b) Percentages of TGC saved by LB-MGUC compared with AP-CLUC, (c) TGC optimality loss for LB-MGUC.

have been obtained by solving the full MILP (E.1) – (E.12) using CPLEX [14]. As before, the LB-MGUC is set up with TAS S_1 and tested under TES S_m ($m = 1, 2, 3, 4$). The percentage of optimality loss (compared with the TGC obtained by solving the full MILP under S_m) is shown in Figure 4.10 (c). We can see that the average percentage optimality loss of the LB-MGUC is only 1.1% across the four TES. Therefore, the proposed LB-MGUC sacrifices little optimality in terms of TGC.

In addition, the proposed LB-MGUC has no ENS or EC under the four TES, which means it can guarantee the feasibility of its outputs (i.e., satisfying the operating constraints of MG).

In fact, the PL-MGUC uses a fixed ranking list of generators in UC for the whole state

[14]The UC solutions obtained by solving the full MILP (implicit MPC) and the UC solutions produced by the full-fidelity explicit MPC are the same.

space, which is usually made according to the average costs of generators at certain fixed production points. However, the optimal production points of generators can change in different regions of the state space and the ranking order of these generators will change accordingly. This explains why the PL-MGUC gives UC schedules with much higher operating costs. In contrast, our proposed LB-MGUC builds different ranking lists for different regions of the state space, which helps mitigate the possible bias of a single ranking list and reduce the resulting operating costs effectively.

In AP-CLUC, the state space is partitioned into different square regions simply based on the number of feature vectors in the training data set they contain. Consequently, CRs where the corresponding hyperplanes of the VF have very different parameters could be merged, which may increase the optimality loss of the resulting UC schedules (see the analysis we provided in Section 4.3.4). In contrast, our proposed LB-MGUC identifies and merges the CRs where the hyperplanes of VF have similar parameters by leveraging the knowledge derived from the EMPC model of the UC problem, which contributes to a further reduction of the operating costs.

Moreover, we can see from Figure 4.10 (b) that the proposed LB-MGUC still loses little optimality in terms of TGC when its TES is S_2 – S_4 (i.e., when the TES is different from its TAS). Hence, our proposed LB-MGUC is capable of generalizing beyond its TAS and achieving good performance in UC in a MG with different load profile, renewable generation profile, or utility energy exchange schedules.

4.4.3 Impact of the Number of Regions in the State Space

We study the impact of the number of regions in the state space (i.e., the parameter K in Section 4.3.2) on the operating costs and computational complexity of the proposed LB-MGUC. Here, the LB-MGUC is created with the data set of TAS S_1 and its performance in UC is also tested under S_1.

Figure 4.11 (a) shows how the percentage optimality loss of LB-MGUC (taken with respect to the TGC calculated by the full MILP solution under S_1) varies with K. We observe that the percentage optimality loss decreases with the increase of K. Moreover, the percentage optimality loss decreases more sharply when K is less than six. In Figure 4.11 (b), we can see that the computation time of each dispatch step associated with LB-MGUC increases with K. At the same time, we also record the memory footprint associated with

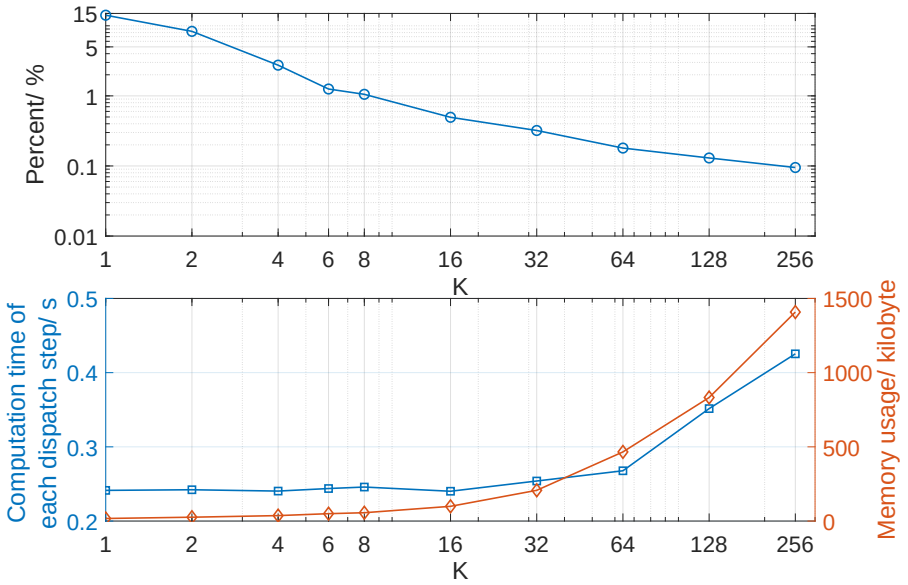

Figure 4.11: (a) Effect of the hyper-parameter K on TGC optimality loss of LB-MGUC, (b) Effect of the hyper-parameter K on the computational time and memory footprint associated with LB-MGUC.

LB-MGUC (i.e., the amount of memory that LB-MGUC uses while running) when K has different values. It can be seen that the memory footprint increases exponentially with K.

If the full-fidelity EMPC is applied to solve the UC problem in Figure 4.11, the total number of regions in the state space can be up to 2^{72}. This astronomical number makes it difficult to measure the computational complexity of the full-fidelity EMPC directly. However, please note that the full-fidelity EMPC is nothing but the limiting case of LB-MGUC as K tends to the total number of regions that form the full-fidelity EMPC. Given that the memory footprint of LB-MGUC grows exponentially with K in Figure 4.11 (b), the memory footprint of the full-fidelity EMPC can easily violate the memory limit of any hardware controller.

Moreover, we also measure the computation time and memory usage when traditional MPC is used to solve the UC problem. Here, the optimal solution is obtained using a branch-and-bound algorithm (see [117] for further details) implemented in MATLAB. The corresponding computation time of each dispatch step is 160.3 s and the memory footprint is 89.8 kilobyte. We can see that the computation time of traditional MPC is much longer than

that of LB-MGUC. This is because the optimal solution is obtained by complex iterative procedures in traditional MPC, which could take a long time to converge. Furthermore, in practice the online optimization algorithm is implemented in a computationally limited industrial controller, which will make it even more intractable to solve the optimization problem within the specified time period.

Therefore, both traditional MPC and EMPC face difficulties in implementations in a low-power computing microgrid controller. In contrast, our proposed LB-MGUC is promising to enable real-time implementation in a MG controller with limited computational power. This is because the computational complexity of the proposed LB-MGUC is tunable by accepting a certain level of suboptimality, as is shown in Figure 4.11. When the LB-MGUC is implemented in a MG controller, the maximum possible value of K can be pre-specified, which is equal to the value that the memory, the computational time, or any other online computational bound of the MG controller is met [15]. Since the percentage optimality loss decreases with the increase of K, this maximum K also allows the LB-MGUC to preserve the optimality of the full-fidelity EMPC to the maximum possible extent.

In Section 4.4.2, the hyper-parameter K was set to be six. We can see from Figure 4.11 (b) that the computational cost of LB-MGUC is very low at this time. Nonetheless, the corresponding TGC optimality loss is only approximately 1%. Thus, the proposed LB-MGUC is able to have excellent performance in optimality approximation with very low computational cost.

4.4.4 Sensitivity of the Total Generation Cost to the Length of the Forward-looking Horizon

Figure 4.12 shows how the length of the forward-looking horizon T_H influences the TGC of the UC solutions obtained by solving the full MILP (E.1) – (E.12). The parameters of the MG was set up in Section 4.4.1. We can see that the TGC decreases with the growth of T_H. However, the TGC changes very little when T_H is greater than one hour. If the T_H is set to be 20 – 30 minutes, we can see from Figure 4.12 that the corresponding TGC obtained by solving the full MILP is approximately 10% higher than the TGC when T_H is 1 hour.

[15] At this time, the computation time and memory footprint in Figure 4.11 (b) can be obtained based on simulation using the given MG controller.

Thus, in Section 4.4.1 the length of the forward-looking horizon T_H is set to be 1 hour. This is because a smaller T_H will degrade the optimality of the UC solutions in the training data set, which will also degrade the performance of our proposed LB-MGUC.

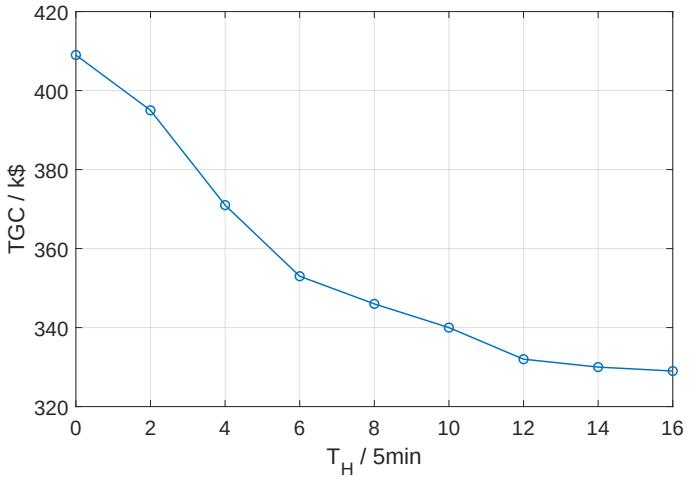

Figure 4.12: Sensitivity of the TGC associated with the UC solutions obtained by solving the full MILP to the forward-looking horizon length T_H.

In fact, the choice of T_H is system dependent, which is mainly affected by the energy-to-power ratio of the storage (i.e., the ratio of the energy capacity to the power rating of the storage) in MG. When the MG controller is designed, the MG operator should perform a sensitivity analysis to determine the proper length of the forward-looking horizon T_H.

4.4.5 Scalability Test

We consider that the MG has more loads, wind turbines and diesel generators compared with the MG studied previously. In Scenario S_1, there are now eight diesel generators, where $\overline{p_g^1} = \overline{p_g^2} = 640$ kW, $\overline{p_g^3} = \overline{p_g^4} = 400$ kW, $\overline{p_g^5} = \overline{p_g^6} = 280$ kW, and $\overline{p_g^7} = \overline{p_g^8} = 200$ kW. The power limits of storage are $\overline{p_e} = 300$ kW and $\underline{p_e} = -300$ kW. The maximum and minimum values of the aggregate priority load are 2450 kW and 650 kW. The critical load is kept constant at 150 kW. The capacity of wind farm is 500 kW. Other parameters are same as those in Section 4.4.2.

As before, the two learning based UC strategies LB-MGUC and AP-CLUC are set up

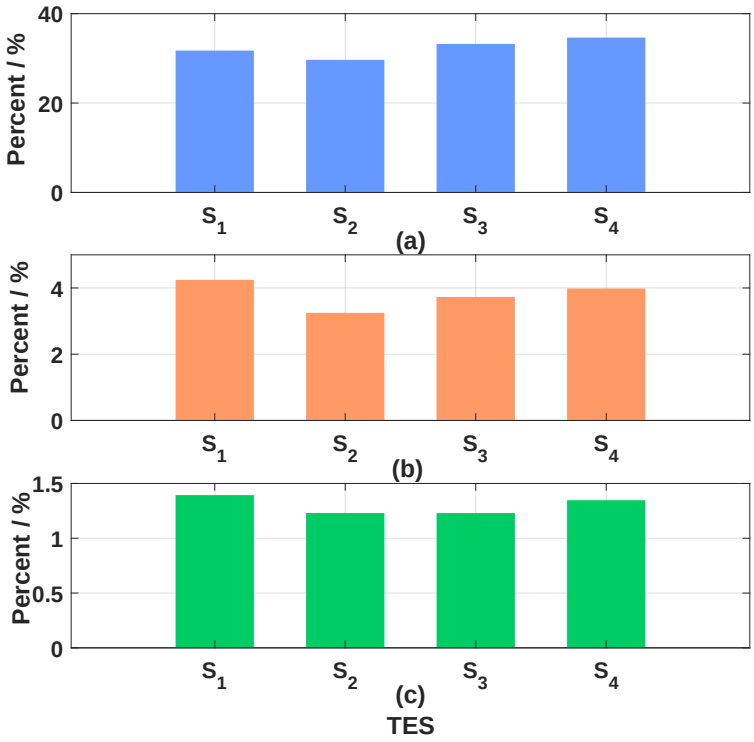

Figure 4.13: (a) Percentages of TGC saved by LB-MGUC compared with PL-MGUC, (b) Percentages of TGC saved by LB-MGUC compared with AP-CLUC, (c) TGC optimality loss for LB-MGUC.

under TAS S_1, and the performances of all the three UC strategies are tested under TES S_m ($m = 1, 2, 3, 4$). Here the parameter K of LB-MGUC is set to be 15. The percentages of TGC saved by LB-MGUC compared with the PL-MGUC and AP-CLUC are presented in Figure 4.13 (a) and (b), respectively. Figure 4.13 (c) shows the percentages of optimality loss (compared with the TGC of the full MILP solution). The LB-MGUC saves 32.3% and 3.8% on average in terms of TGC compared with PL-MGUC and AP-CLUC, respectively. At the same time, the optimality loss of LB-MGUC is 1.3% on average, which shows that the proposed LB-MGUC is still applicable to a more complicated MG.

When the LB-MGUC is tested under TES S_1, the computation time of each dispatch step and the memory footprint ($K = 15$) are shown in Table 4.3. Remember that the computation time is 0.24 s and the memory footprint is 99.2 kilobyte when there are three generators and the state space has a similar number of regions. We can see that the computation time

of each step is almost unchanged when there are more generators. However, the memory footprint increased significantly. This is because the possible combinations of the generator statuses increase exponentially with the number of generators in the MG. As a result, the ranking list in each region of the state space will have many more UC options. Therefore, a larger MG will put higher requirements on the memory size of its MG controller. We also measure the computation time and memory usage when traditional MPC is used to solve the UC problem in the larger MG, which is also presented in Table 4.3. We can see that the computation time of each dispatch step associated with traditional MPC is even longer than the length of each time step (5 minutes), which indicates that the online optimization algorithm failed to converge within the specified time period. This again highlights the difficulty of applying traditional MPC in low-power and real-time computing applications.

Table 4.3: Computation time and memory footprint associated with LB-MGUC and traditional MPC

Method	**Computation time of each dispatch step (s)**	**Memory usage (kilobyte)**
LB-MGUC ($K = 15$)	0.27	450
MPC	372.4	420

In addition, although we have demonstrated that the LB-MGUC can generalize beyond its TAS and achieve high level performance in UC in a MG with different load profiles, renewable generation profiles, or utility energy exchange schedules, it is still recommended to monitor the performance of LB-MGUC online when the operating conditions have changed significantly. If the optimality loss exceeds the desired range, it is time to re-train the DNN and update the ranking lists.

4.5 Summary

In this chapter we integrated ML and EMPC to design a rigorous and computationally efficient solution for implementing flexibility-based UC paradigms in a MG controller with limited computational power. A two-stage learning based approach has been developed to approximate the EMPC control laws of the original UC problem with tunable complexity by accepting a certain level of suboptimality. When the proposed approach is implemented in a

MG controller, its computational complexity can be tuned to meet the hardware limitation of the given MG controller, while preserving as much as possible the optimality of the full-fidelity EMPC. This entirely eliminates the excessive partitioning problem of traditional EMPC and enables eventual online real time implementation of the proposed approach.

The verified results have shown that the proposed approach effectively reduces the total generation cost in MG compared with a simple priority list based UC strategy. By leveraging the knowledge derived from the EMPC model of the UC problem, this approach provides feasible UC decisions and outperforms another learning based UC strategy. When the percentage optimality loss of the proposed approach compared with the full-fidelity EMPC is approximately 1% in terms of generation cost, its computational burden is still very low. The proposed approach is also capable of generalizing beyond its training scenario and achieving high level performance in UC in a MG with different load profiles, renewable generation profiles, or utility energy exchange schedules. Finally, the proposed approach also scales easily well with the size of the MG.

Chapter 5

Conclusion

5.1 Thesis Overview

The increasing renewable generation in power systems has posed critical challenges to its energy management. The reason is that renewable generation is dependent upon variable weather conditions, which introduces increased variability and uncertainty in power balancing and raises the flexibility requirements in the system. In this environment, there is a need for appropriate approaches capable of quantifying a power system's flexibility requirements in an accurate manner. Moreover, there is an increasing need for approaches capable of simplifying and accelerating traditional flexibility-based operational planning paradigms so that they can be implemented in practical controllers in power system with limited computational power, especially in microgrid.

Chapter 2 first proposed the modeling approach called "spatio-temporal flexibility requirement envelope". It seeks to model the flexibility requirements in a power system in a more accurate manner by comprehensively capturing and representing the temporal trends and spatial correlation of multisite renewable generation and load demand using historic generation and demand data. A robust optimization based framework is also developed for applying this envelope to power system unit commitment and dynamic dispatch through projections of the spatio-temporal envelopes. The case study provided illustrated the effectiveness of the spatio-temporal envelopes both in microgrid and in transmission system. It is revealed that spatio-temporal envelope can effectively capture the temporal trends and spatial correlation of RES and uncertain loads by adjusting its shape and size,

which can effectively reduce the over-conservatism of the energy scheduling results. Compared to other modeling methods for flexibility requirements (i.e., ellipsoidal, polyhedral and box shaped methods), we demonstrated that the proposed approach leads to a decrease in total operating cost in the long run, while not incurring more load shedding or renewable generation curtailment.

In Chapter 3 we proposed a rigorous DT-based learning framework for real-time implementation of flexibility-based MG dispatch paradigm in a MG controller with limited computational power. Here, we assume that the MG has a maximum of one dispatchable generator. The proposed DT-based framework is able to consider the operating constraints of MG, uncertainties in net load, the commitment status (ON/OFF) of a dispatchable generator, and the energy storage system systematically. It is shown through illustrative examples that the proposed DT-based approach can provide feasible dispatch decisions. It also has satisfactory performance in optimality approximation. At the same time, the DT-based approach is capable of generalizing beyond its training scenario and achieving high level dispatch performance in a MG with different parameters, power profiles, or other operating conditions. Moreover, HIL simulation shows that the computational efficiency of the DT-based approach is very high and that it is capable of enabling eventual online real time implementation.

Moreover, in Chapter 4 we considered the general case where the MG has multiple dispatchable generators. We developed a rigorous approach which combines k-means clustering, DNN classification and EMPC for implementing flexibility-based UC paradigms in a MG controller with limited computational power. The proposed learning based approach is able to handle the variables and constraints of the original UC problem systematically. At the same time, it introduces a tuning parameter that allows the MG operator to adjust the computational complexity of the approach to meet the hardware limitation of the given MG controller, while preserving as much as possible the optimality of the full-fidelity EMPC. This entirely eliminates the excessive partitioning problem of traditional EMPC and enables eventual online real time implementation of the proposed approach. The effectiveness of the proposed approach was demonstrated through several illustrative examples. It is shown that the proposed approach can provide feasible UC schedules. It is able to have excellent performance in optimality approximation with very low computational costs. The proposed approach is also capable of generalizing beyond its

training scenario and achieving high level performance in UC in a MG with different load profile, renewable generation profile, or utility energy exchange schedules.

5.2 Recommendations for Future Work

We summarize several key open questions and recommendation for future work here, which could contribute to advancing the state of the art further.

1. Develop online envelope updating approaches. The parameters of the spatio-temporal flexibility requirement envelope in Chapter 2 can be adjusted online according to the system state and the operation history, so that the approach can be less conservative further and the decisions made can be more cost-efficient. The best parameter can be obtained by optimization. Machine learning can be used here to accelerate the optimization process. Also the parameter in system forecast (interval forecast) can be adjusted according to operation history.

2. In Chapter 4, we assumed that the flexibility requirement envelope used in MG unit commitment is time invariant. However, this envelope could be time varying (e.g., when there is solar generation). Then the flexibility requirement envelope used should be generated by the projection of the spatio-temporal flexibility requirement envelope in Chapter 2 instead of the time invariant envelope in [6]. At a result, the state vector $\boldsymbol{\theta}_t$ should include the envelope and it will have many more dimensions. Dimension reduction methods such as PCA can be used.

3. Incorporate the learning-based model we developed in Chapter 3 and Chapter 4 into the long-term investment planning of MG. Usually long-term planning is computationally time-consuming because the optimal operation of the MG needs to be calculated for a large number of samples. Our proposed model can provide a fast and high quality approximation of the MG dispatch or unit commitment, which can accelerate the calculation of the long-term investment planning.

4. Incorporate the network constraints into the learning-based framework we developed in Chapter 4. Then the nodal injection and line flows will have to be considered.

5. If a market existed, the interaction between the MG and the utility grid can be taken into consideration in the optimization formulation in Chapter 2.

6. Incorporate demand response into the spatio-temporal flexibility requirement envelope in Chapter 2.

7. Another important area for future research is the application of the spatio-temporal flexibility requirement envelope for short-term operational planning or long-term investment planning studies in a power system with high penetration of electric vehicles.

Appendix A

Time Series Gaussianization and PCA

The detrended time series are transformed into corresponding Gaussian time series. We use $y'_n(t)$ to represent $\omega'_n(t)$ for wind speed/load time series or $\delta^{s\prime}_n(t)$ for solar power time series at bus n. The transform is shown as follows for $t = t_0, t_0 - 1, \ldots, t_0 - T_S + 1$ [62], [63]

$$y''_n(t) = \Phi^{-1}\left[\hat{F}^y_n\left(y'_n(t)\right)\right] \tag{A.1}$$

where $\hat{F}^y_n$ is the estimated cumulative distribution function (CDF) of time series $y'_n(t)$ ($t = t_0, t_0 - 1, \ldots, t_0 - T_S + 1$), while $\Phi(\cdot)$ is the CDF of the standard normal distribution. The Gaussianized time series are assembled into the $R \times T_S$ matrix $\mathbf{Y}''$, whose rows are time series $y''_n(t)$ and R is the number of separate location-specific time series. It is further approximated that all the rows of $\mathbf{Y}''$ are also jointly Gaussian distributed as seen in [63] and [118]. We define $\mathbf{\Sigma}$, the $R \times R$ covariance matrix of $\mathbf{Y}''$. Letting $\rho_1 \geq \rho_2 \geq \cdots \geq \rho_R$ be the eigenvalues of $\mathbf{\Sigma}$ and their corresponding eigenvectors $U_1, U_2, \cdots, U_R$, the rows of $\mathbf{Y}''$ can be decorrelated using the following PCA transform [63], [119]:

$$\mathbf{Z}=[U_1\ U_2\ \cdots\ U_R]^T\mathbf{Y}'' \tag{A.2}$$

The rth ($1 \leq r \leq R$) row of $\mathbf{Z}$ is the rth PC of $\mathbf{Y}''$. All the PCs are uncorrelated and Gaussian distributed. Thus, they are independent of each other. Moreover, as the PC index r increases, the relevance of the components decreases in capturing the variability of the time series (as seen with the decreasing values of the corresponding ρ_r). The PCA transformation

in (A.2) can be inverted back through:

$$\mathbf{Y}'' = [U_1\ U_2\ \cdots\ U_R]\mathbf{Z} \tag{A.3}$$

Appendix B

Flexibility Requirement Envelopes for PCs

The method in [6] based on persistence forecast is applied to construct the flexibility requirement envelope for each PC. For the time series of the rth ($1 \leq r \leq R$) PC, $Z_r(t)$ ($t = t_0, t_0 - 1, \ldots, t_0 - T_S + 1$), the intra-hourly deviations are modeled as step changes, which are defined as $\Delta Z_r(t, \tau) = Z_r(t + \tau) - Z_r(t)$, where τ is the look ahead time. For example, if the length of each time step is five minutes and the length of the forecast horizon is set to be one hour, then $\tau \in \Xi_H = \{0, 5, 10, 15, ..., 55, 60\}$ minutes.

Note that $\Delta Z_r(t, \tau)$ is a stationary time series indexed by t, whose probability distribution function (PDF) is a function of τ only. The PDF of $\Delta Z_r(t, \tau)$ can be computed empirically for every τ using a relative frequency plot. The interval $[\varepsilon_r^{\downarrow}(\tau), \varepsilon_r^{\uparrow}(\tau)]$ enclosing $\nu\%$ of the probability of the empirical PDF of $\Delta Z_r(t, \tau)$ is defined as follows

$$\int_{\varepsilon_r^{\uparrow}(\tau)}^{+\infty} \varphi_r(\tau, z)dz = \int_{-\infty}^{\varepsilon_r^{\downarrow}(\tau)} \varphi_r(\tau, z)dz = \frac{1 - \nu\%}{2} \tag{B.1}$$

where $\varphi_r(\tau, z)$ is the empirical PDF of $\Delta Z_r(t, \tau)$. Connecting the $\varepsilon_r^{\downarrow}(\tau)$ and $\varepsilon_r^{\uparrow}(\tau)$ at all τ and we obtain the flexibility requirement envelope $\{\varepsilon_r^{\downarrow}(\tau), \varepsilon_r^{\uparrow}(\tau)\}_{\tau \in \Xi_H}$ for the rth PC $Z_r(t)$.

Appendix C

MIP for MG Dispatch

This Appendix presents the MIP formulation for the MG dispatch (with one dispatchable generator only) based on the flexibility requirement envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau\in\Xi_H}$. The variables $(\cdot)(\tau, q)$ ($\tau \in \Xi_H$, $q \in \Xi_q = \{\uparrow, \downarrow\}$) are forward-looking decisions, as driven by the envelope.

1) Objective function:

$$\min \sum_{\tau\in\Xi_H, q\in\{\uparrow,\downarrow\}} \Big(a_g p_g(\tau, q)^2 + b_g p_g(\tau, q) + c_g u_g(\tau, q)\Big) + \pi^d d_c(\tau, q) + \pi^\delta \delta_c(\tau, q) \tag{C.1}$$

Equation (C.1) minimizes the cost of dispatchable generation at current time t_0 ($\tau = 0$), while considering the forward-looking horizon $t_0 + \tau$ for each $\tau \in \Xi_H$. Parameters a_g, b_g and c_g are the cost parameters of generator, while π^d and π^δ are penalties associated to load shedding and renewable generation curtailment.

2) Power balance for $\tau \in \Xi_H$ and $q \in \{\uparrow, \downarrow\}$:

$$d(t_0) - \delta(t_0) - p_{poi}(t_0) = p_g(\tau, q) + p_e(\tau, q) + d_c(\tau, q) - \delta_c(\tau, q) - e_A^q(\tau) \tag{C.2}$$

3) Capacity limits of generator for $\tau \in \Xi_H$ and $q \in \{\uparrow, \downarrow\}$:

$$u_g(\tau, q)\underline{p_g} \le p_g(\tau, q) \le u_g(\tau, q)\overline{p_g} \tag{C.3}$$

$$u_g(\tau, q) \in \{0, 1\} \tag{C.4}$$

4) Capacity limits of storage for $\tau \in \Xi_H$ *and* $q \in \{\uparrow, \downarrow\}$*:*

$$\underline{s_e} \leq s_e(t_0 - 1) - T_\Delta \sum_{t=0}^{\tau} p_e(t, q)\eta \leq \overline{s_e} \tag{C.5}$$

$$\underline{p_e} \leq p_e(\tau, q) \leq \overline{p_e} \tag{C.6}$$

η is the efficiency of storage. $\eta = \eta_c$ when the storage is charging and $\eta = 1/\eta_d$ when it is discharging.

5) At current time t_0 *(*$\tau = 0$*), we also require the consistency of dispatch decisions as follows:*

$$p_g(t_0) = p_g(0, \uparrow) = p_g(0, \downarrow) \tag{C.7}$$

$$u_g(t_0) = u_g(0, \uparrow) = u_g(0, \downarrow) \tag{C.8}$$

$$p_e(t_0) = p_e(0, \uparrow) = p_e(0, \downarrow) \tag{C.9}$$

Appendix D

Decision-Tree Construction

The large classification trees CT_r $(r = 1, 2, 3)$ and the large regression trees RT_r are built using the CART (Classification And Regression Tree) algorithm [87]. The CART algorithm first uses recursive binary splitting to grow the large classification tree or the large regression tree based on the corresponding training data.

For classification, the Gini index or cross-entropy [87] are used as criteria for making binary splits over features. For regression, the sum squared error can be used as the splitting criterion [87]. The tree stops growing only when the number of observations in each leaf node reaches a pre-specified minimum value.

The large trees are pruned using the cost complexity pruning method. For the large classification trees CT_r $(r = 1, 2, 3)$, each value of the tuning parameter α_r corresponds to a subtree $CT_r^{\alpha_r}$ such that the following cost complexity measure is minimized

$$CCM(CT_r^{\alpha_r}) = CE + \alpha_r \left|CT_r^{\alpha_r}\right| \tag{D.1}$$

where $CCM(CT_r^{\alpha_r})$ is the cost complexity measure of subtree $CT_r^{\alpha_r}$. The classification error CE is associated with $CT_r^{\alpha_r}$ defined in (3.6). The term $|CT_r^{\alpha_r}|$ indicates the number of terminal nodes of the subtree $CT_r^{\alpha_r}$. When $\alpha_r = 0$, then the subtree is simply equal the large classification tree CT_r. As α_r increases, the branches are pruned from the tree in a nested and predictable fashion. The cost complexity pruning for large regression trees is similar to that of the classification trees.

Further details about the CART algorithm and cost complexity pruning can be found

in [87].

Appendix E

MILP of Microgrid Unit Commitment

This Appendix presents the MILP for the UC (with multiple dispatchable generators) at current time t_0. The variables $(\cdot)(\tau, q)$ ($\tau \in \Xi_H$, $q \in \Xi_q = \{\uparrow, \downarrow\}$) are forward-looking decisions, as driven by the flexibility requirement envelope $\{e_A^{\uparrow}(\tau), e_A^{\downarrow}(\tau)\}_{\tau \in \Xi_H}$. Here, $\Xi_H = \{1, 2, \cdots, T_H\}$.

1) Objective function: The objective function minimizes the total cost of using dispatchable generators at current time t_0 (i.e., $\tau = 0$), while considering the forward-looking horizon $t_0 + \tau$ ($\tau \in \Xi_H$) and each branch ($q \in \{\uparrow, \downarrow\}$) of the envelope. z_g^i represents the operating cost of generator i, which will be introduced in detail later in this section. π^d and π^δ are penalties associated to load shedding and renewable generation curtailment.

$$v_{t_0}^* = \min \sum_{\tau \in \Xi_H, q \in \{\uparrow,\downarrow\}, i \in \Xi_g} z_g^i(\tau, q) + \sum_{\tau \in \Xi_H, q \in \{\uparrow,\downarrow\}} \Big(\pi^d d_c(\tau, q) + \pi^\delta \delta_c(\tau, q)\Big) \tag{E.1}$$

The optimization is subject to:

2) Power balance constraint for $q \in \{\uparrow, \downarrow\}$ and $\tau \in \Xi_H$:

$$\hat{d}(t_0) - p_{poi}(t_0) = \sum_{i \in \Xi_g} p_g^i(\tau, q) + \sum_{j \in \Xi_e} p_e^j(\tau, q) + d_c(\tau, q) - \delta_c(\tau, q) - e_A^q(\tau) \tag{E.2}$$

3) Technical limits of storage $j \in \Xi_e$ for $q \in \{\uparrow, \downarrow\}$ and $\tau \in \Xi_H$: η^j represents the efficiency of storage j. $\eta^j = \eta_c^j$ when the storage is charging and $\eta^j = 1/\eta_d^j$ when it is

discharging. η_c^j and η_d^j are the charging and discharging efficiency of storage j.

$$\underline{p_e^j} \leq p_e^j(\tau, q) \leq \overline{p_e^j} \tag{E.3}$$

$$\underline{s_e^j} \leq s_e^j(t_0 - 1) - T_\Delta \sum_{t=0}^{\tau} p_e^j(t, q)\eta^j \leq \overline{s_e^j} \tag{E.4}$$

4) Technical limits of generator $i \in \Xi_g$ for $q \in \{\uparrow, \downarrow\}$ and $\tau \in \Xi_H$:

$$u_g^i(\tau, q)\underline{p_g^i} \leq p_g^i(\tau, q) \leq u_g^i(\tau, q)\overline{p_g^i} \tag{E.5}$$

$$u_g^i(\tau, q) \in \{0, 1\} \tag{E.6}$$

$$p_g^i(\tau - 1, q) - p_g^i(\tau, q) \leq r_{g\mathrm{dn}}^i u_g^i(\tau, q) + r_{g\mathrm{sd}}^i\Big(u_g^i(\tau - 1, q) - u_g^i(\tau, q)\Big) + \overline{p_g^i}\Big(1 - u_g^i(\tau - 1, q)\Big) \tag{E.7}$$

$$p_g^i(\tau, q) - p_g^i(\tau - 1, q) \leq r_{g\mathrm{up}}^i u_g^i(\tau - 1, q) + r_{g\mathrm{su}}^i\Big(u_g^i(\tau, q) - u_g^i(\tau - 1, q)\Big) + \overline{p_g^i}\Big(1 - u_g^i(\tau, q)\Big) \tag{E.8}$$

5) Auxiliary constraint of generator $i \in \Xi_g$ for $q \in \{\uparrow, \downarrow\}$, $\tau \in \Xi_H$ and $w = 1, 2, \cdots, W_i$: W_i represents the total number of segments of the piecewise linear cost function of generator i. w is the index of the segment.

$$z_g^i(\tau, q) \geq b_g^{i,w} p_g^i(\tau, q) + c_g^{i,w} u_g^i(\tau, q) \tag{E.9}$$

6) The consistency of UC decisions for generator $i \in \Xi_g$ and storage $j \in \Xi_e$ at current time t_0 ($\tau = 0$):

$$p_g^i(t_0) = p_g^i(0, \uparrow) = p_g^i(0, \downarrow) \tag{E.10}$$

$$u_g^i(t_0) = u_g^i(0, \uparrow) = u_g^i(0, \downarrow) \tag{E.11}$$

$$p_e^j(t_0) = p_e^j(0, \uparrow) = p_e^j(0, \downarrow) \tag{E.12}$$

Appendix F

A Brief Introduction of AP-CLUC

We give a brief introduction of the adaptive contextual learning based microgrid UC strategy (AP-CLUC) previously proposed in [61].

The AP-CLUC first partitions the state space into different square regions. It then builds a priority list for different generators for each region, which will be used for making UC decisions online (please note that storage systems are not considered in [61]). The partitioning of the state space in AP-CLUC is briefly introduced as follows.

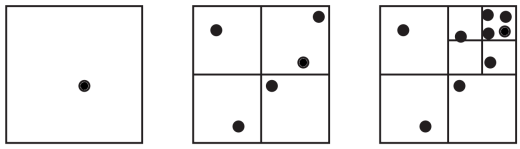

Figure F.1: Illustration of the partitioning process of AP-CLUC, which is adapted from [61].

The algorithm first sets a threshold for the number of state vectors $\boldsymbol{\theta}_t$. If the total number of state vectors in a square region exceeds this threshold, then this square region will be divided into four smaller square regions with the same size. This process repeats until the number of state vectors in each square region of the state space is lower than the pre-determined threshold.

Figure F.1 illustrates the partitioning process of AP-CLUC. The threshold of the number of state vectors in this figure is four. When the state space has only one state vector, it is not partitioned. When the state space has five state vectors, it is partitioned into four smaller

square regions, so that the number of state vectors in each square region is less than the threshold (four). Finally, if one of the smaller square regions still has more than four state vectors, it is further divided into four smaller square regions until the number of state vectors in each square region is less than the threshold (four).

References

[1] IEA, "Renewables 2020, IEA, Paris." `https://www.iea.org/reports/renewables-2020`, 2020. [Online; accessed 23-December-2020].

[2] A. Akrami, M. Doostizadeh, and F. Aminifar, "Power system flexibility: an overview of emergence to evolution," *Journal of Modern Power Systems and Clean Energy*, vol. 7, no. 5, pp. 987–1007, 2019.

[3] B. Mohandes, M. S. E. Moursi, N. Hatziargyriou, and S. E. Khatib, "A review of power system flexibility with high penetration of renewables," *IEEE Transactions on Power Systems*, vol. 34, no. 4, pp. 3140–3155, 2019.

[4] T. Heggarty, J.-Y. Bourmaud, R. Girard, and G. Kariniotakis, "Quantifying power system flexibility provision," *Applied Energy*, vol. 279, p. 115852, 2020.

[5] S. Khan, W. Gawlik, and P. Palensky, "Reserve capability assessment considering correlated uncertainty in microgrid," *IEEE Transactions on Sustainable Energy*, vol. 7, no. 2, pp. 637–646, 2016.

[6] H. Nosair and F. Bouffard, "Flexibility envelopes for power system operational planning," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 3, pp. 800–809, 2015.

[7] M. Bucher, *On operational flexibility in transmission constrained electric power systems*. PhD thesis, ETH Zurich, 2016.

[8] Council of European Energy Regulators (CEER), "Flexibility use at distribution level - a CEER conclusions paper. Technical report, Council of European Energy Regulators," 2018.

[9] International Renewable Energy Agency (IRENA), "Power system flexibility for the energy transition. Technical report, International Renewable Energy Agency," 2018.

[10] H. Chandler, "Harnessing variable renewables: A guide to the balancing challenge," *Paris, France: International Energy Agency*, 2011.

[11] J. Ma, V. Silva, R. Belhomme, D. S. Kirschen, and L. F. Ochoa, "Evaluating and planning flexibility in sustainable power systems," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 1, pp. 200–209, 2013.

[12] IEA, "Status of power system transformation 2018." `https://www.oecd-ilibrary.org/content/publication/9789264302006-en`, 2018.

[13] H. Nosair, *Flexibility envelopes for power system operational planning*. PhD thesis, McGill University, 2015.

[14] P. D. Lund, J. Lindgren, J. Mikkola, and J. Salpakari, "Review of energy system flexibility measures to enable high levels of variable renewable electricity," *Renewable and Sustainable Energy Reviews*, vol. 45, pp. 785 – 807, 2015.

[15] A. Kalantari, J. F. Restrepo, and F. D. Galiana, "Security-constrained unit commitment with uncertain wind generation: The loadability set approach," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1787–1796, 2013.

[16] A. A. Jahromi and F. Bouffard, "On the loadability sets of power systems—part i: Characterization," *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 137–145, 2017.

[17] A. Abiri-Jahromi and F. Bouffard, "On the loadability sets of power systems—part ii: Minimal representations," *IEEE Transactions on Power Systems*, vol. 32, no. 1, pp. 146–156, 2017.

[18] F. Bouffard, F. Galiana, and A. Conejo, "Market-clearing with stochastic security-part i: formulation," *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1818–1826, 2005.

[19] F. Bouffard, F. D. Galiana, and A. J. Conejo, "Market-clearing with stochastic security-part ii: case studies," *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1827–1835, 2005.

[20] J. Arroyo and F. Galiana, "Energy and reserve pricing in security and network-constrained electricity markets," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 634–643, 2005.

[21] N. W. Miller, K. Clark, and M. Shao, "Frequency responsive wind plant controls: Impacts on grid performance," in *2011 IEEE Power and Energy Society General Meeting*, pp. 1–8, 2011.

[22] E. Lannoye, D. Flynn, and M. O'Malley, "Evaluation of power system flexibility," *IEEE Transactions on Power Systems*, vol. 27, no. 2, pp. 922–931, 2012.

[23] E. Lannoye, D. Flynn, and M. O'Malley, "Power system flexibility assessment — state of the art," in *2012 IEEE Power and Energy Society General Meeting*, pp. 1–6, 2012.

[24] C. De Jonghe, B. F. Hobbs, and R. Belmans, "Optimal generation mix with short-term demand response and wind penetration," *IEEE Transactions on Power Systems*, vol. 27, no. 2, pp. 830–839, 2012.

[25] N. K. Dhaliwal, F. Bouffard, and M. J. O'Malley, "A fast flexibility-driven generation portfolio planning method for sustainable power systems," *IEEE Transactions on Sustainable Energy*, vol. 12, no. 1, pp. 368–377, 2021.

[26] H. Nosair and F. Bouffard, "Economic dispatch under uncertainty: The probabilistic envelopes approach," *IEEE Transactions on Power Systems*, vol. 32, no. 3, pp. 1701–1710, 2017.

[27] C. Shao, X. Wang, M. Shahidehpour, X. Wang, and B. Wang, "Security-constrained unit commitment with flexible uncertainty set for variable wind power," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 3, pp. 1237–1246, 2017.

[28] Y. Dvorkin, M. Lubin, S. Backhaus, and M. Chertkov, "Uncertainty sets for wind power generation," *IEEE Transactions on Power Systems*, vol. 31, no. 4, pp. 3326–3327, 2016.

[29] H. Nosair and F. Bouffard, "Energy-centric flexibility management in power systems," *IEEE Transactions on Power Systems*, vol. 31, no. 6, pp. 5071–5081, 2016.

[30] Y. V. Makarov, C. Loutan, J. Ma, and P. de Mello, "Operational impacts of wind generation on california power systems," *IEEE Transactions on Power Systems*, vol. 24, no. 2, pp. 1039–1050, 2009.

[31] Y. Dvorkin, D. S. Kirschen, and M. A. Ortega-Vazquez, "Assessing flexibility requirements in power systems," *IET Generation, Transmission Distribution*, vol. 8, no. 11, pp. 1820–1830, 2014.

[32] F. Golestaneh, P. Pinson, R. Azizipanah-Abarghooee, and H. B. Gooi, "Ellipsoidal prediction regions for multivariate uncertainty characterization," *IEEE Transactions on Power Systems*, vol. 33, no. 4, pp. 4519–4530, 2018.

[33] P. Li, X. Guan, J. Wu, and X. Zhou, "Modeling dynamic spatial correlations of geographically distributed wind farms and constructing ellipsoidal uncertainty sets for optimization-based generation scheduling," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 4, pp. 1594–1605, 2015.

[34] X. Jiang, Y. C. Chen, and A. D. Domínguez-García, "A set-theoretic framework to assess the impact of variable generation on the power flow," *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 855–867, 2013.

[35] F. Golestaneh, P. Pinson, and H. B. Gooi, "Polyhedral predictive regions for power system applications," *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 693–704, 2019.

[36] R. J. Bessa, "From marginal to simultaneous prediction intervals of wind power," in *2015 18th International Conference on Intelligent System Application to Power Systems (ISAP)*, pp. 1–6, 2015.

[37] M. A. Bucher, S. Chatzivasileiadis, and G. Andersson, "Managing flexibility in multi-area power systems," *IEEE Transactions on Power Systems*, vol. 31, no. 2, pp. 1218–1226, 2016.

[38] Y. Wu, W. Tan, S. Huang, Y. Chiang, C. Chiu, and C. Su, "Impact of generation flexibility on the operating costs of the taiwan power system under a high penetration of renewable power," *IEEE Transactions on Industry Applications*, vol. 56, no. 3, pp. 2348–2359, 2020.

[39] J. Zhao, T. Zheng, and E. Litvinov, "A unified framework for defining and measuring flexibility in power system," *IEEE Transactions on Power Systems*, vol. 31, no. 1, pp. 339–347, 2016.

[40] E. Du, N. Zhang, B. Hodge, Q. Wang, C. Kang, B. Kroposki, and Q. Xia, "The role of concentrating solar power toward high renewable energy penetrated power systems," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6630–6641, 2018.

[41] Y. Wang, H. Bayem, M. Giralt-Devant, V. Silva, X. Guillaud, and B. Francois, "Methods for assessing available wind primary power reserve," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 1, pp. 272–280, 2015.

[42] M. Ross, C. Abbey, F. Bouffard, and G. Jos, "Multiobjective optimization dispatch for microgrids with a high penetration of renewable generation," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 4, pp. 1306–1314, 2015.

[43] J. Jimeno, J. Anduaga, J. Oyarzabal, and A. G. de Muro, "Architecture of a microgrid energy management system," *European Transactions on Electrical Power*, vol. 21, no. 2, pp. 1142–1158, 2011.

[44] N. Holjevac, T. Capuder, N. Zhang, I. Kuzle, and C. Kang, "Corrective receding horizon scheduling of flexible distributed multi-energy microgrids," *Applied Energy*, vol. 207, pp. 176 – 194, 2017.

[45] M. Costley, M. J. Feizollahi, S. Ahmed, and S. Grijalva, "A rolling-horizon unit commitment framework with flexible periodicity," *International Journal of Electrical Power & Energy Systems*, vol. 90, pp. 280–291, 2017.

[46] J. Silvente, G. M. Kopanos, E. N. Pistikopoulos, and A. Espuña, "A rolling horizon optimization framework for the simultaneous energy supply and demand planning in microgrids," *Applied Energy*, vol. 155, pp. 485–501, 2015.

[47] M. Legry, F. Colas, C. Saudemont, J. Y. Dieulot, and O. Ducarme, "Mixed integer quadratic programming receding horizon microgrid supervisor," in *2019 IEEE Milan PowerTech*, pp. 1–6, 2019.

[48] D. Romero-Quete and J. R. Garcia, "An affine arithmetic-model predictive control approach for optimal economic dispatch of combined heat and power microgrids," *Applied Energy*, vol. 242, pp. 1436–1447, 2019.

[49] Y. Du, W. Pei, N. Chen, X. Ge, and H. Xiao, "Real-time microgrid economic dispatch based on model predictive control strategy," *Journal of Modern Power Systems and Clean Energy*, vol. 5, no. 5, pp. 787–796, 2017.

[50] J. R. Nelson and N. G. Johnson, "Model predictive control of microgrids for real-time ancillary service market participation," *Applied Energy*, vol. 269, p. 114963, 2020.

[51] T. Morstyn, B. Hredzak, and V. G. Agelidis, "Dynamic optimal power flow for dc microgrids with distributed battery energy storage systems," in *2016 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 1–6, 2016.

[52] C. Wu, S. Gao, Y. Liu, T. E. Song, and H. Han, "A model predictive control approach in microgrid considering multi-uncertainty of electric vehicles," *Renewable Energy*, vol. 163, pp. 1385–1396, 2021.

[53] I. Novickij and G. Joós, "Model predictive control based approach for microgrid energy management," in *2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, pp. 1–4, 2019.

[54] I. Aldaouab, M. Daniels, and R. Ordóñez, "Model predictive control energy dispatch to optimize renewable penetration for a microgrid with battery and thermal storage," in *2018 IEEE Texas Power and Energy Conference (TPEC)*, pp. 1–6, 2018.

[55] M. Quashie, F. Bouffard, and G. Joós, "Business cases for isolated and grid connected microgrids: Methodology and applications," *Applied Energy*, vol. 205, pp. 105 – 115, 2017.

[56] M. Ross, C. Abbey, F. Bouffard, and G. Joós, "Microgrid economic dispatch with energy storage systems," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3039–3047, 2018.

[57] Y. Guan and J. Wang, "Uncertainty sets for robust unit commitment," *IEEE Transactions on Power Systems*, vol. 29, no. 3, pp. 1439–1440, 2014.

[58] Wikipedia contributors, "Programmable logic controller – Wikipedia, the free encyclopedia." `https://en.wikipedia.org/w/index.php?title=Programmable_logic_controller&oldid=996430466`, 2020. [Online; accessed 31-December-2020].

[59] E. Adrover, *Introduction to PLCs: A Beginner's Guide to Programmable Logic Controllers*. Elvin Perez Adrover, 2012.

[60] Y. Du and F. Li, "Intelligent multi-microgrid energy management based on deep neural network and model-free reinforcement learning," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1066–1076, 2020.

[61] H. Lee, C. Tekin, M. van der Schaar, and J. Lee, "Adaptive contextual learning for unit commitment in microgrids with renewable energy sources," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 4, pp. 688–702, 2018.

[62] J. Ekström, M. Koivisto, I. Mellin, R. J. Millar, and M. Lehtonen, "A statistical model for hourly large-scale wind and photovoltaic generation in new locations," *IEEE Transactions on Sustainable Energy*, vol. 8, no. 4, pp. 1383–1393, 2017.

[63] D. D. Le, G. Gross, and A. Berizzi, "Probabilistic modeling of multisite wind farm production for scenario-based applications," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 3, pp. 748–758, 2015.

[64] J. Heckenbergerova, P. Musilek, and J. Marek, "Analysis of wind speed and power time series preceding wind ramp events," in *Proceedings of the 2014 15th International Scientific Conference on Electric Power Engineering (EPE)*, pp. 279–283, 2014.

[65] R. Rosenthal, "Gams – a user's guide," *Internal Documents*, 2008.

[66] National Renewable Energy Laboratory, "Renewable Resource Data." `https://www.nrel.gov/grid/data-tools.html`, 2019.

[67] Iowa Environmental Mesonet, "ASOS 1 minute interval data." `https://mesonet.agron.iastate.edu/request/asos/1min.phtml`, 2019.

[68] Natural Resources Canada, "High-Resolution Solar Radiation Datasets." `https://www.nrcan.gc.ca/energy/renewable-electricity/solar-photovoltaic/18409`, 2019.

[69] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "Matpower: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, 2011.

[70] EirGrid Group, "System and Renewable Data." `http://www.eirgridgroup.com/how-the-grid-works/renewables/`, 2019.

[71] "IEEE standard for the specification of microgrid controllers," *IEEE Std 2030.7-2017*, pp. 1–43, 2018.

[72] G. Joos, J. Reilly, W. Bower, and R. Neal, "The need for standardization: The benefits to the core functions of the microgrid control system," *IEEE Power and Energy Magazine*, vol. 15, no. 4, pp. 32–40, 2017.

[73] Y. Bengio, A. Lodi, and A. Prouvost, "Machine learning for combinatorial optimization: A methodological tour d'horizon," *European Journal of Operational Research*, 2020.

[74] S. Misra, L. Roald, and Y. Ng, "Learning for constrained optimization: Identifying optimal active constraint sets," 2018.

[75] Q. Zhang, K. Dehghanpour, Z. Wang, and Q. Huang, "A learning-based power management method for networked microgrids under incomplete information," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1193–1204, 2020.

[76] A. S. Xavier, F. Qiu, and S. Ahmed, "Learning to solve large-scale security-constrained unit commitment problems," 2019.

[77] B. Cornelusse, G. Vignal, B. Defourny, and L. Wehenkel, "Supervised learning of intra-daily recourse strategies for generation management under uncertainties," in *2009 IEEE Bucharest PowerTech*, pp. 1–8, IEEE, 2009.

[78] A. Jahanbani Ardakani and F. Bouffard, "Prediction of umbrella constraints," in *2018 Power Systems Computation Conference (PSCC)*, pp. 1–7, 2018.

[79] P. Zeng, H. Li, H. He, and S. Li, "Dynamic energy management of a microgrid using approximate dynamic programming and deep recurrent neural network learning," *IEEE Transactions on Smart Grid*, vol. 10, no. 4, pp. 4435–4445, 2019.

[80] A. J. Wood, B. F. Wollenberg, and G. B. Sheblé, *Power generation, operation, and control*. John Wiley & Sons, 2013.

[81] G. Ruan, H. Zhong, J. Wang, Q. Xia, and C. Kang, "Neural-network-based lagrange multiplier selection for distributed demand response in smart grid," *Applied Energy*, vol. 264, p. 114636, 2020.

[82] Y. Shang, W. Wu, J. Guo, Z. Ma, W. Sheng, Z. Lv, and C. Fu, "Stochastic dispatch of energy storage in microgrids: An augmented reinforcement learning approach," *Applied Energy*, vol. 261, p. 114423, 2020.

[83] P. Moutis, S. Skarvelis-Kazakos, and M. Brucoli, "Decision tree aided planning and energy balancing of planned community microgrids," *Applied Energy*, vol. 161, pp. 197 – 205, 2016.

[84] T. Barbier, M. F. Anjos, and G. Savard, "Optimization of diesel, wind and battery hybrid power systems," tech. rep., Les Cahiers du GERAD, 2014.

[85] G. K. Venayagamoorthy, R. K. Sharma, P. K. Gautam, and A. Ahmadi, "Dynamic energy management system for a smart microgrid," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 8, pp. 1643–1656, 2016.

[86] A. Geron, *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems*. Sebastopol, CA: O'Reilly Media, 2017.

[87] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*, vol. 112. Springer, 2013.

[88] F. Katiraei, R. Iravani, N. Hatziargyriou, and A. Dimeas, "Microgrids management," *IEEE Power and Energy Magazine*, vol. 6, no. 3, pp. 54–65, 2008.

[89] X. Wang, C. Wang, T. Xu, H. Meng, P. Li, and L. Yu, "Distributed voltage control for active distribution networks based on distribution phasor measurement units," *Applied Energy*, vol. 229, pp. 804–813, 2018.

[90] N. Kang, J. Wang, R. Singh, and X. Lu, "Interconnection, integration, and interactive impact analysis of microgrids and distribution systems," tech. rep., Argonne National Lab.(ANL), Argonne, IL (United States), 2017.

[91] M. Paluszek and S. Thomas, *MATLAB Machine Learning*. Apress, 2017.

[92] V. Mohan, R. Suresh, J. G. Singh, W. Ongsakul, and N. Madhu, "Microgrid energy management combining sensitivities, interval and probabilistic uncertainties of renewable generation and loads," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 7, no. 2, pp. 262–270, 2017.

[93] "IEEE standard for the testing of microgrid controllers," *IEEE Std 2030.8-2018*, pp. 1–42, 2018.

[94] H. Farhangi, *Smart microgrids: lessons from campus microgrid design and implementation*. CRC Press, 2016.

[95] C. Sun, S. Q. Ali, G. Joos, and F. Bouffard, "A modular generic microgrid controller adaptive to different compositions," in *2020 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 2472–2479, 2020.

[96] A. Alessio and A. Bemporad, "A survey on explicit model predictive control," in *Nonlinear model predictive control*, pp. 345–369, Springer, 2009.

[97] V. Dua and E. N. Pistikopoulos, "An algorithm for the solution of multiparametric mixed integer linear programming problems," *Annals of operations research*, vol. 99, no. 1, pp. 123–139, 2000.

[98] J. Acevedo and E. N. Pistikopoulos, "A multiparametric programming approach for linear process engineering problems under uncertainty," *Industrial & engineering chemistry research*, vol. 36, no. 3, pp. 717–728, 1997.

[99] Y. Qiu, J. Lin, F. Liu, and Y. Song, "Explicit mpc based on the galerkin method for agc considering volatile generations," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 462–473, 2020.

[100] H. Ahmadi, J. R. Martí, and A. Moshref, "Piecewise linear approximation of generators cost functions using max-affine functions," in *2013 IEEE Power Energy Society General Meeting*, pp. 1–5, 2013.

[101] A. M. Eltamaly and M. A. Mohamed, "8 - optimal sizing and designing of hybrid renewable energy systems in smart grid applications," in *Advances in Renewable Energies and Power Technologies* (I. Yahyaoui, ed.), pp. 231–313, Elsevier, 2018.

[102] HOMER Pro 3.14, "Fuel curve." `https://www.homerenergy.com/products/pro/docs/latest/fuel_curve.html`, 2021.

[103] M. Sechilariu and F. Locment, "Chapter 3 - backup power resources for microgrid," in *Urban DC Microgrid* (M. Sechilariu and F. Locment, eds.), pp. 93–132, Butterworth-Heinemann, 2016.

[104] International Renewable Energy Agency (IRENA), "Innovation landscape brief: Flexibility in conventional power plants," 2019.

[105] H. S. Bidgoli and T. Van Cutsem, "Combined local and centralized voltage control in active distribution networks," *IEEE Transactions on Power Systems*, vol. 33, no. 2, pp. 1374–1384, 2018.

[106] O. Samuelsson, S. Repo, R. Jessler, J. Aho, M. Kärenlampi, and A. Malmquist, "Active distribution network — demonstration project adine," in *2010 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT Europe)*, pp. 1–8, 2010.

[107] E. N. Pistikopoulos, N. A. Diangelakis, and R. Oberdieck, *Multi-parametric Optimization and Control*. John Wiley & Sons, 2020.

[108] X. Geng and L. Xie, "Learning the lmp-load coupling from data: A support vector machine based approach," *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 1127–1138, 2017.

[109] D. Deka and S. Misra, "Learning for dc-opf: Classifying active sets using neural nets," in *2019 IEEE Milan PowerTech*, pp. 1–6, 2019.

[110] J. Katz, I. Pappas, S. Avraamidou, and E. N. Pistikopoulos, "Integrating deep learning models and multiparametric programming," *Computers & Chemical Engineering*, vol. 136, p. 106801, 2020.

[111] T. K. Ralphs and A. Hassanzadeh, "On the value function of a mixed integer linear optimization problem and an algorithm for its construction," *COR@ L Technical Report 14T–004*, 2014.

[112] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.

[113] T. Hu, C. Wang, and C. Miller, "Identification of marginal generation units based on publicly available information," *Applied Energy*, vol. 281, p. 116073, 2021.

[114] J. Choi, I. Choi, G. Ahn, and D. Won, "Advanced power sharing method to improve the energy efficiency of multiple battery energy storages system," *IEEE Transactions on Smart Grid*, vol. 9, no. 2, pp. 1292–1300, 2018.

[115] B. Karg and S. Lucia, "Efficient representation and approximation of model predictive control laws via deep learning," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3866–3878, 2020.

[116] FW POWER, "Diesel Generator Fuel Consumption." `https://fwpower.co.uk/knowledge-centre/diesel-generator-fuel-consumption-chart-in-litres/`, 2020.

[117] L. A. Wolsey, *Integer programming*. John Wiley & Sons, 2020.

[118] B. Klockl, "Multivariate time series models applied to the assessment of energy storage in power systems," in *Proceedings of the 10th International Conference on Probablistic Methods Applied to Power Systems*, pp. 1–8, 2008.

[119] D. J. Burke and M. J. O'Malley, "A study of principal component analysis applied to spatially distributed wind power," *IEEE Transactions on Power Systems*, vol. 26, no. 4, pp. 2084–2092, 2011.