POWER SYSTEMS OPTIMISATION AND MANAGEMENT BY DISTRIBUTED METHODS

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE FACULTY OF ENGINEERING AND PHYSICAL SCIENCES

2021

Yi Dong
Department of Electrical and Electronic Engineering

Contents

List of Figures

Notations

$\mathbb{R}$	set of real numbers
$\mathbb{R}^{++}$	positive real numbers
$\mathcal{A}$	set of action variables
$\mathcal{M}$	set of Markovian decision processes
$\mathcal{P}$	set of transfer probabilities
$\mathcal{R}$	set of reward variables
$\mathcal{S}$	set of state variables
Ω	set of projection area
$charge$	subscript of BESS charge
$discha$	subscript of BESS discharge
$down$	superscript of regulation-down market
e	superscript of energy market
I	identity matrix
i, j	index of units

reg	superscript of regulation market
S_B	set of BESS units
S_C	set of controllable load units
S_D	set of demand units
S_G	set of generator units
S_R	set of renewable generator units
up	superscript of regulation-up market
η_c, η_d	charging and discharging efficiency
λ_i, ξ_i	control variables of consensus algorithm
ϕ_i	communicated information of ith power unit between its neighbours
Profit	profit of a BESS
θ_i	the percentage of ith unit that knows the total power mismatch
a	decision action variable
b_p	bidding price
b_c	regulation capacity bids
C^W	penalty function of operation constraints
C^{total}	total cost of a BESS owner
C_i	the cost function of ith power unit
C_{ag}	ageing cost function of BESS

C_{loss}	bidding price
$C_{M\&O}$	maintenance and operation cost function
d	depth of discharge
E	energy state of BESS
$f_{G,i}^{E}$	emission penalty function of ith generator
$f_{G,i}^{O\&M}$	the operation and mainteinance penalty function of ith generator
$f_{B,i}$	the penalty function of ith BESS
k	numbers of power generator units
m	numbers of power demand units
n	numbers of battery energy storage system units
N_{d}^{fail}	maximum number of charging/discharging cycles at d DoD
P	power output of a BESS
p	the electricity price
p^{e}	energy market clearing price
Q	value function related to state-action pair
r	immediate reward signal from environment
s	observation variable from environment
soc	state of charge
V	value function related to state variable

v^{-1}	clearing price of the previous day
$W_{B,i}$	the welfare function of ith BESS
$W_{D,i}$	the welfare function of ith demand unit
$W_{G,i}$	the welfare function of ith generator
α, β	learning rate of reinforcement learning
ΔT	regulation period
γ	discount factor of reinforcement learning
$\kappa_f, \kappa_c, \iota$	the fixed operation cost, the charging/discharging cost and the charging efficiency of ith BESS
$\mathcal{E}$	probability of exploration
μ	component replacement cost of BESS
τ	time index of regulation step
a_i, b_i, c_i	cost parameters
C_a	annual cost of BESS
C_E	weight of energy market
C_F	weight of energy market
C_P	weight of energy market
C_{inv}	investment cost of BESS
E_{max}	maximum energy capacity of BESS

h	operation period
K_p	parameter for different types of BESS
N_μ	number of maximum episodes
N_f	number of features
N_t	number of maximum steps
P_D	active power mismatch between supply and demand in a smart grid
P_i^{max}	maximum active power output of ith power unit
P_i^{min}	minimum active power output of ith power unit
$P_{B,i}$	active power output of ith BESS
$P_{C,i}$	active power output of ith controllable demand unit
$P_{D,i}$	consumed active power of ith demand unit
$P_{G,i}$	active power output of ith generator unit
P_{max}	maximum power output of BESS
$P_{R,i}$	active power output of ith renewable generator
t	hourly time index
w_e	weight of energy market
w_{reg}	weight of regulation market

Abbreviations

ACE Area Control Error.

AGC Automation Generation Control.

ANN Artificial Neural Network.

ARMA AutoRegressive Moving Average.

BESS Battery Energy Storage Systems.

DBN Deep Belief Networks.

DCNN Deep Convolution Neural Network.

DDBN Distributed Deep Belief Networks.

DER Distributed Energy Resources.

DG Distributed Generator.

DL Deep Learning.

DoD Depth of Discharge.

DSM Demand Side Management.

ELM Extreme Learning Machine.

EV Electric Vehicle.

FARL Function Approximation based Reinforcement Learning.

GEFCom Global Energy Forecasting Competition.

GOA Grasshopper Optimisation Algorithm.

KKT Karush-Kuhn-Tucker.

LSTM Long Short Term Memory.

MAE Mean Absolute Error.

MAPE Mean Absolute Percent Error.

MCP Market Clearing Price.

MDP Markov Decision Process.

MSE Mean Squared Error.

O&M Operation and Maintenance.

PBR Performance-Based Regulation.

PDRNN Pooling Deep Recurrent Neural Network.

PEV Plug-in Electric Vehicles.

PI Proportional-Integral.

PV Photovoltaic.

RBM Restricted Boltzmann Machine.

RegD Regulation D: resources.

RL Reinforcement Learning.

RMCCP Regulation Market Capacity Clearing Price.

RMPCP Regulation Market Performance Clearing Price.

RNN Recurrent Neural Network.

SO System Operater.

SoC State of Charge.

STLF Short Term Load Forecasting.

SVM Support Vector Machine.

SVR Support Vector Regression.

ToU Time-of-Use.

Abstract

The research in this thesis mainly focuses on the power system management and optimisation by distributed methods. The traditional power system management and optimisation approaches are based on centralised fashions. However, it is difficult or impossible to establish a single centralised control centre in modern distributed power systems. As the amount of data in the power system continues to increase, centralised algorithms are limited by insufficient computing power. To solve the problems mentioned above, we will look into the distributed optimisation problem in power system.

Firstly, to deal with the initialisation error and local physical constraints, a distributed demand side management strategy is studied by maximising the total welfare from demand side to supply side. To achieve the objectives of demand side management, controllable power units generate their optimal power reference based on the proposed distributed algorithm by coordinating information with the neighbours. Thus, it is a completely distributed algorithm, and the analysis denotes that the proposed algorithm can solve the optimal economic dispatch problem in an initialisation-free approach and adapt to the plug-and-play operation.

Secondly, a uncertain power market environment is considered in this thesis. To maximise the benefit of a battery energy storage system in power markets, a novel reinforcement learning based optimal bidding strategy is investigated. The objective is to extend the battery life and maximise the benefit for battery energy storage system owners. Using this strategy, the battery energy storage system can make different decisions based on various environmental states. Meanwhile, the proposed bidding strategy overcomes the discrete limits by the function approximation approach.

Finally, from the above researches, it is noticed that the data volume and customer privacy may influence the reliability of the power systems. To avoid the computing burden

and consider the data privacy. A novel Markovian switching based distributed deep belief networks for short term load forecasting is introduced. The proposed algorithm trains the short-term load forecasting model with local datasets and update the model parameters by communicating with connected neighbours, which does not need to transfer any information about the load data. In addition, it can simultaneously achieve fast training speed and superior robustness to cyberattacks due to the Markovian switching structure and the unsupervised training process of the deep belief networks.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright Statement

i. The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the "Copyright") and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.

ii. Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made **only** in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.

iii. The ownership of certain Copyright, patents, designs, trade marks and other intellectual property (the "Intellectual Property") and any reproductions of copyright works in the thesis, for example graphs and tables ("Reproductions"), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.

iv. Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=487), in any relevant Thesis restriction declarations deposited in the University Library, The University Library's regulations (see http://www.manchester.ac.uk/library/aboutus/regulations) and in The University's Policy on Presentation of Theses.

List of Publications

1). **Y. Dong**, T. Zhao and Z. Ding, "Demand-side management using a distributed initialisation-free optimisation in a smart grid", *IET Renewable Power Generation*, vol. 13, no. 9, pp.1533-1543, 2019.

2). **Y. Dong**, T. Zhao and Z. Ding, "Short Term Load Forecasting Based on Markovian Switching Distributed Deep Belief Networks", *International Journal of Electrical Power and Energy Systems*, vol. 130, pp. 106942, 2021.

3). **Y. Dong**, T. Zhao and Z. Ding, "A Strategic Day-ahead Bidding Strategy and Operation for Battery Energy Storage System by Reinforcement Learning", *Electric Power Systems Research*. (Second Revision)

4). **Y. Dong**, T. Zhao and Z. Ding, "An Optimal Day-ahead Bidding Strategy and Operation for Battery Energy Storage System by Reinforcement Learning", *The 21st IFAC World Congress*, Berlin, JUL, 2020. (Accepted)

The following publications are not included in this thesis.

1). Z. Dong, Z. Li, **Y. Dong**, S. Jiang, and Z. Ding, "Fully-Distributed Deloading Operation of DFIG-based Wind Farm for Load Sharing", *IEEE Transactions on Sustainable Energy*, vol. 12, no. 1, pp. 430-440, 2020.

2). Z. Dong, Z. Li, **Y. Dong**, Z. Liang and Z. Ding, "A Completely-Autonomous Distributed Control of Multiple DFIG-Based Wind Farms for Active Power Regulation", *Applied Energy*. (Under Review)

3). Z. Dong, **Y. Dong** and Z. Ding, "Novel Cooperative Control of DFIG-Based Grid-Connected Wind Farm for Active Power Dispatch Considering Wake Effect", *The 21st IFAC World Congress*, Berlin, JUL, 2020, (Accepted).

Acknowledgements

I would like to thank all the people who have helped me to successfully complete my PhD research program.

First and foremost, I would like to offer my deepest and sincere gratitude to Prof. Zhengtao Ding for his supervision through this four-year research period. He pointed out the new research field to form my PhD project. Without his valuable suggestions and support, the research of this thesis would not have been possible.

I would like to thank all my colleagues in Control System Centre, The University of Manchester. They helped me to solve the difficulties during my research and gave me an unforgettable memory.

Finally, I owe many thanks to my parents, Mr. Yunsheng Dong and Mrs. Cuilan Chen, who have provided me with unconditional love, support, and understanding for my research and life. Last but not least, my special thanks would go to my beloved wife, Mrs. Yang Chen, for her loving considerations and great confidence in me all through these years.

To My Parents and Wife

Chapter 1

Introduction

1.1 Background of Modern Power Systems

With the increasing depletion of traditional energy sources (such as coal, oil, natural gas.), the world is facing an energy crisis and global warming issues. The power system relies on these traditional energy sources, and the massive use of energy has attracted great attention from the scientific and industrial researchers. The UK has put forward a UK Future Energy Scenarios and clearly pointed out that the power system is a key area of energy-saving [1]. Especially with the development of smart grid, high requirements have been placed on the power system in terms of energy, emissions, and the environment. Under the pressure of global warming, renewable energy plays an important role in the power system, and strengthening the integration of renewable energy and the power system becomes the main challenge. Obviously, its purpose is to reduce conventional energy consumption and greenhouse gas emissions. Therefore, the rational use of various resources of the power system to achieve the best operation of the power system is of practical significance for energy conservation, including conventional energy and

renewable energy.

Based on the vigorous development of modern communication technology, distributed control protocol becomes possible. Along with the revolution of renewable energy, the power system would be further developed in an intelligent, data-based and distributed way. The development of modern power systems has brought new challenges to distributed and scalable architectures for traditional System Operater (SO), in terms of the management and control of modern smart grids. The smart grid will integrate more and more new power elements in the future, such as Plug-in Electric Vehicles (PEV), distributed energy storage systems, distributed generators. Distributed algorithms can provide a potential solution to meet these challenges [2–5]. Designing distributed control algorithms and frameworks in the concept of power systems, power units are considered in a given environment, and modelled as intelligent agents. They can communicate with their adjacent neighbours and have the ability of local computing and data processing. According to different tasks, the distributed algorithms can be applied to any level of partitioning. For example, an individual agent could be a Distributed Energy Resources (DER) in resources allocation problem, a Battery Energy Storage Systems (BESS) company in power market bidding tasks, or even a large region of the network in a short-term load forecasting mission.

The optimisation and control methods of traditional power systems are mostly centralised and model-based. However, these traditional methods are gradually unable to meet the requirements of power system development, for example, the dynamic optimisation of grid operations, distributed energy resources and generation, distributed Demand Side Management (DSM), uncertain market environment, lack of bidding rivals' information and massive power data processing.

In order to meet the above requirements, the power systems need to introduce advanced

optimisation and management methods, which can solve the problem locally. At the same time, the information from the neighbours of power units should be utilised to optimise and integrate the whole power system. In addition, the large amount of historical power data and neighbourhood data should be collected to achieve accurate regulation and management of power system.

1.2 Motivation

The high penetration of renewable energy resources might cause a more complicated design of power applications in power systems [6–8]. It is difficult to manage and predict the power output for power markets due to the intermittent nature of renewable energy sources. In addition, improper operation and management could bring about an increase in peak demand that could threat the stability of the power systems.

In this project, the management and operation problems in power systems are considered. With the development of smart grid and the attention paid to renewable energy resources, the management and operation methods with distributed and data-driven are in good graces. As a result, the thesis is divided into three major parts:

Firstly, this thesis proposes a distributed initialisation-free optimisation strategy to handle the optimal demand management under the uncertain renewable generation. To deal with the physical constraints and the initialisation problem, we combine the Proportional-Integral (PI) consensus dynamics [9] and the projection algorithm [10], which can solve the constraints problem by local power units. This strategy considers the costs of demand units, BESS and Distributed Generator (DG). Different from the existing results, the emission costs and the battery degradation costs are also considered in our DSM model. In addition, the proposed algorithm can handle DSM problem with any initial

errors so that it can adapt the plug-and-play operation. In order to achieve the control objectives of DSM, each power unit generates an optimal power reference through the proposed algorithm by coordinating information with its neighbours. At the meantime, each power unit will meet the power reference through a local controller. The effectiveness of the proposed distributed algorithm is validated through simulation studies in an IEEE 14-bus system and a complex power grid with 40 generators, 15 BESS and 200 loads.

Then, this thesis proposes a novel Markovian based bidding model that decides the optimised bidding strategy of the BESS in day-ahead energy and regulation markets, considering the charging/discharging losses and the ageing cost of the BESS. Additionally, the Function Approximation based Reinforcement Learning (FARL) algorithm is applied to the proposed model to solve the multiple rival bidding problem. The function approximation approach is introduced in this thesis to address the redundancy caused by massive data, and therefore prevent the dimension curse. Based on the proposed model, the BESS could obtain a more accurate and profitable bidding strategy.

Finally, a novel STLF method is proposed, which can be trained under a distributed framework. To protect the data security and deal with the mass dataset, we introduce the Markovian switching consensus algorithm to decentralise the DBN and the load dataset. It can solve the STLF model by local computing agents (CA) and update the model parameters by communicating with connected neighbours, which does not need to transfer any information about the load data. Different from the existing results, the proposed DDBN model will pre-train their local model by unsupervised training to obtain better initialisation of the model coefficients. Such a pre-trained initialisation process generates more suitable local models, which consequently yields fast convergence to the global model with less consensus steps. In the meantime, the DDBN model will also amend their local parameters based on the information from their connected neighbours

to avoid the over-fitting problem.

1.3 Novel contribution to knowledge

The research investigates the power system optimisation and management by distributed methods. The novel contributions to monitor in the thesis are summarised as follows:

- A distributed initialisation-free optimisation strategy is proposed to handle the optimal demand management under the uncertain renewable generation:
    1. The power outputs of all units including controllable loads, BESS and DG are optimised according to their welfare functions while maintaining the supply-demand balance in various conditions, i.e., different prices, communication failures and time-varying active power mismatches.
    2. The proposed strategy is distributed based on a distributed average estimator. Thus, the proposed strategy is scalable and could be applied to large-scale systems. Additionally, it only requires the exchange of one information variable that does not contain the information of the welfare function. The customer privacy could be protected during the information transmission process.
    3. The practical implementation of a distributed algorithm would have initial mismatches. To this end, the proposed algorithm is initialisation-free, which has the advantage of solving the initial errors. Meanwhile, this feature enables the plug-and-play functions in DSM since it does not require any initialisation process.
- A novel model that decides the optimal bidding strategy of the BESS in day-ahead

energy and regulation markets has been proposed:

1. The BESS bidding problem is modelled as a Markov Decision Process (MDP) framework for learning the optimised bidding policy to increase the welfare of BESS in energy and regulation markets. The model is delicately designed, especially on the losses during the power transfer and the ageing cost of the BESS.
2. Since RL involves discrete-state transition, a function approximation approach is introduced to transfer the uncertain and continuous bidding environment into a set of discrete states, such that the memory and computational complexities can be reduced. This makes the state transition tractable, and avoids the curse of dimensionality.
3. The proposed bidding strategy of BESS owners considers both energy market and regulation arbitrage, which shows flexibility to the uncertain bidding environments, such as prior knowledge of other rivals and any dynamics of the system operator. As an individually profit maximisation bidding strategy, it can help the BESS owner to obtain its bidding strategy for individual market participants.

- A novel Distributed Deep Belief Networks (DDBN) with Markovian switching consensus algorithm for Short Term Load Forecasting (STLF) has been proposed. The major contributions are summarised as follows:

  1. The proposed DDBN algorithm updates each local model parameters by self-training and limited information exchange with neighbours during STLF. As a result, implementing the proposed algorithm can reduce training time due to the parallel computations with local data collection. Further, the overfitting problem in the local model can be significantly reduced by mutual correction.

2. A Markovian-based switching topology is designed to deal with uncertain cyberattacks during neighbour communication. With the help of the stochastic mechanism, it is much more flexible as well as robust, compared with the fixed communication network. Whilst its stability and the convergence to the optimal value are not compromised and strictly guaranteed by a delicately designed controller gain.

1.4 Thesis Outline

The organisation of each chapter is described in detail at the beginning of each chapter. To understand the whole thesis structure, a general overview is summarised in this section.

Chapter 1 Introduction. Brief overviews of the modern power system have been introduced. Meanwhile, the motivations and contributions of this thesis are discussed.

Chapter 2 Literature Review. The results are reviewed in this chapter, including the distributed optimisation algorithms for demand side management, reinforcement learning methods for BESS optimal bidding in power market and distributed Deep Learning (DL) for STLF.

Chapter 3 Preliminaries. Some notations and related preliminaries are introduced in this chapter, such as graph theory, Reinforcement Learning (RL) and Deep Belief Networks (DBN). The strict mathematical description of novel concept and associated analysis of this new concept are given as well.

Chapter 4 Demand-side Management. This chapter studies the demand side management problem for smart grid to maximise the total welfare, while respecting each

power unit's constraints. An objective function is formulated to establish the total welfare of smart grid. Then, a distributed cooperative scheme of demand side management is proposed to solve the economic dispatch problem in an initialisation-free manner. Moreover, the stability of the designed algorithm is analysed and the effectiveness is evaluated on a simulation example.

Chapter 5 Battery Energy Storage Systems Optimal Bidding. In this chapter, a bidding model for BESS owner is presented to maximise the benefit and a function approximation reinforcement learning algorithm is designed for BESS optimal bidding environment. The objective function for BESS owner is formulated to maximise the daily bidding revenue in an integrated energy and regulation market. Without any rivals' information, the proposed algorithm can train its network parameters and optimise the bidding strategy under an uncertainty environment. Moreover, the function approximation approach is used in this chapter to handle the discrete limits of reinforcement learning algorithms.

Chapter 6 Short-term Load Forecasting. By considering the data privacy, a novel fully distributed deep learning algorithm for STLF is proposed to achieve accurate short term load forecasting on power load. This new algorithm is designed based on DBN and Markovian switching consensus algorithm, the convergence analysis and the illustrative case studies are given respectively. For STLF problem, the studied model is fully data-driven and Markovian switched, which could adapted to further time-varying load and environments. In other words, this proposed algorithm is more robust for STLF problems.

Chapter 7 Conclusions. This chapter summarises the key research finds from this thesis and provides the suggestions for future research. The possible extensions and interesting topics are also provided for further studies.

1.5 Summary

This chapter firstly presents the background of a modern power system and the essential elements of smart grids. The motivation of this thesis has been outlined. Furthermore, the contributions are also summarised. Finally, the structure of each chapter has been introduced.

Chapter 2

Literature Review

This chapter includes four sections. In Section 2.1, a literature review of typical demand side management for power systems and the corresponding key issues are discussed. Then, Section 2.2 provides a review of existing results on power market and BESS. Finally, a review of STLFs and DDBN is given in Section 2.3. The summary of this chapter is provided in Section 2.4.

2.1 Demand Side Management and Economic Dispatch

Due to the increasing energy demand, economic consideration and the emission concerns, the power grid is facing the challenges and opportunities of transforming the traditional power grid into a smart grid. With the integration of renewable energy resource, battery energy storage systems, and controllable loads, the power grid becomes distributed and complex [11]. The operation situation of the power grid may change frequently, and therefore the reasonable energy management strategy of the power grid

is an important aspect in the smart grid research which is designed to meet the demand requirements at different time intervals and to realise the efficient operation of the smart grid [12]. Due to increasing power demand, economic consideration and emission concern, the smart grid is facing the challenges and opportunities of integrating renewable energy [11]. The percentage of energy derived from renewable sources has risen from 6.7% in 2009 to 29.4% in the UK, 2017 [13]. Because of the intermittency and unpredictability of renewable energy, the future smart grid inevitably integrates more dynamic elements. Meanwhile, the smart grid must be able to maintain the balance between the supply and demand [14, 15].

DSM refers to all those strategies aiming at varying controllable load profiles to optimise the entire power system from the supply side to the demand side, optimising power allocation to obtain efficient and eco-friendly usage of electricity [16]. DSM can be implemented by additional equipment to reduce and shift consumption, such as BESS and smart control of Electric Vehicle (EV). Unbalanced conditions resulting from the uncertain load changes and the renewable power generation affect the power quality, and may even damage customer equipment [17]. Therefore, it is crucial for DSM to have an effective and optimal strategy [18].

Typically, DSM focuses on centralised algorithms. For example, the authors in [19] proposed a hierarchical control structure to maximise the economic benefits through a central controller, whereas it does not consider the inequality constraints, such as the maximum and the minimum power generation of DG and the limitations of the charging/discharging power in BESS. To keep the power units working in a feasible mode, the authors in [20] divided an inequality constraint into five intervals, which greatly increases the complexity of the algorithm. The penalty function is applied to eliminate the inequality constraints in [21], whereas the studied algorithm may still exceed the constraint area in some cases. In [22], a centralised second-order economic power dispatch

system is utilised to minimise the consumption costs among the supply and demand profiles. The authors in [23] solved an optimal power and heat scheduling problem, which minimises the operation cost of the smart grid with the uncertain power market prices. In [24], a centralised optimisation strategy is applied in a Photovoltaic (PV) solar farm, which optimises the investment cost, operation and maintenance cost, fuel cost, emission cost and network losses cost. However, this centralised computation process is very complex, and it is time-consuming to calculate the optimal results. The centralised algorithms require a powerful control centre and a large data server centre to collect the global information and process the massive data [25], which are not conducive to the development and the upgrade of smart grids. Besides, the complexity of the centralised demand management system grows exponentially with the increasing number of power units [26].

Because the conventional centralised algorithm puts all the computing equipment in one place, the calculation burden of the control centre is higher, and the scalability of the centralized algorithm is poor due to geographical restrictions. In the distributed algorithms, each unit have independent computing centres and only process locally collected data, which will greatly reduce the computational burden and enable the algorithm to quickly respond to power changes, and therefore most distributed algorithms are plug-and-play [21, 27]. As a result, the flexible distributed control strategies are studied in a significant number of papers, e.g., [28–38]. In [28], the authors worked on a social maximal welfare problem in a smart grid in a distributed manner. Hug *et al.* [29] investigated a consensus based distributed power management algorithm to solve the demand of a smart grid. In [30], the authors introduce a cooperative distributed demand management system based on Karush-Kuhn-Tucker (KKT) conditions. Deng *et al.* in [31] presented a distributed demand response problem and define sub-problems by decomposing the main optimisation problem and solving each sub-problem locally. In [32], a

three-layer strategy was used to control a power grid, which includes supervision, optimisation and execution process. Meanwhile, the authors in [34] employed a cooperative control strategy of multiple BESS to maintain the active power balance and minimise the total active power loss associated with BESS's charging/discharging inefficiency. In [36], the authors proposed an optimal distributed solution for economic dispatch to minimise the operation cost. However, most of them are sharing the information of the output power or the incremental cost, which violate privacy principles. Furthermore, the optimal solutions of the algorithms in [32, 34, 35, 39] can be only obtained when certain initial conditions, i.e., the sum of initial power allocations should equal to the system active power mismatch, are satisfied. In other words, the network resource constraint can be ensured only if it satisfies the initial conditions. Hence, it is not compatible with the plug-and-play function.

2.2 Battery Energy Storage System in Power Markets

Battery Energy Storage System (BESS) gets the opportunity to play an important role in the future smart grid. With the rapid development of battery technology, the BESS can bring more benefits for the owners and the cost of BESS construction is gradually reduced [6, 40, 41]. There will be more companies focusing on the development and construction of the BESS. As the BESS capacity increases, the BESS will participate in different markets and benefit from multiple services [42, 43]. Additionally, the frequency regulation market demands rapid response and offers high returns [44, 45], so that the BESS owners will put more attention on the regulation market with their BESS, which will lead to competition in the future smart grid. Therefore, how to allocate the capacity of BESS and make bidding decisions has become an important issue.

One major application for the BESS is frequency regulation services in the AGC market.

BESS has the characteristics of easy storage, high reliability and fast response, which is more suitable than pumped-storage plant and heat storage plant for the frequency regulation market. Moreover, the AGC market offers 3 times mileages for Regulation D: resources (RegD) service, which will bring high revenue for the BESS owners. As a result, more BESS owners are expected to compete in the AGC market and some researchers have been paying more attention to the AGC market [46–51]. In [46], a control strategy for the BESS in frequency regulation was provided, considering the ageing cost while keeping the State of Charge (SoC) of the BESS. In [48], a coordinated control strategy of BESS was proposed to ensure the wind power plants' commitment to frequency ancillary services, focusing on reducing the BESS's size and extending the lifetime of the BESS. However, mentioned literature only consider the application of the BESS in one market. With the emergence of large-capacity BESS, some articles study the operation strategies of the BESS in multiple markets, so as to maximise the overall profit of the BESS by controlling the placement proportion of the BESS in different markets. For example, He, et al. [50] integrated the energy storage system and solar power plant and proposed an optimal strategy for Concentrating Solar Power (CSP) plant, which considered the energy, reserve and regulation market. He also proposed a Performance-Based Regulation (PBR) based optimal bidding model in [51]. It not only addressed the optimal strategy for the BESS in different markets but also considered the battery life.

Another problem missed by these literature is that the bidding strategies only solve the allocation problem of the single BESS, in which their bidding rivals are neglected. With the entry of the rivals, the bidding market of the BESS presents some challenges [52]. During the process of bidding, the bidder does not know the rivals' bidding price and bidding quantity, which is hard to solve by traditional optimisation algorithms. Furthermore, since bidding is a highly random and uncertain process, the bidders cannot know the specific revenue model during bidding. They only know the offer results from

the SO in the smart grid. Considering incomplete information of stochastic demand from the market and unknown bids from rivals, some individual based approaches have been widely applied for bidding strategies in electric market, where the individual agent learns to maximise its own profit [53–55]. For example, Kebriaei, et al. [56] combined state estimation and fuzzy Q-learning to learn the optimal decision of the generators. Li, et al. [57] applied the model-free reinforcement learning algorithm to solve the optimal carbon capture in the wholesale market bidding problem. Nanduri, et al. [58] formulated a stochastic game model for the energy market and proposed a reinforcement learning based solution methodology. Lakic, et al. [59] simulated the market as a stochastic environment and proposed a novel agent based SA-Q-learning for demand-side system reserve provision. However, there is very little understanding of the potential benefits of BESS in the wider power system or micro-grids [60].

2.3 Short Term Load Forecasting

Electricity load forecasting is not only an essential basis for power scheduling of the power grid but also an indispensable part of the energy management system [61]. For different applications in the power systems, electricity load forecasting is mainly divided into very short-term, short-term, medium-term and long-term predictions. Each type of load forecasting applies different algorithms to meet the specific goals of the applications. STLF helps utilities and energy suppliers cope with an increasingly complex electricity market, such as the integration of renewable energies, and development of electricity markets with sophisticated pricing strategies [62]. Additionally, electric load usually presents a non-linear and non-stationary change in time series [63], which challenges and degrades the accuracy of forecast. Therefore, the importance of STLF and the complexity of power load have motivated numerous research works in this field

in recent years.

The earlier methods are mainly based on statistical algorithms, such as AutoRegressive Moving Average (ARMA), Kalman filtering and quantile regression [63]. Then, the artificial intelligence algorithms have been widely applied because they can well address some non-linear and non-stationary characteristics in electrical load [64]. Ahmad *et al.* used an improved Artificial Neural Network (ANN) to accelerate the training speed and improve forecast accuracy [65]. Zhang *et al.* designed an ensemble model of Extreme Learning Machine (ELM) and apply it in STLF of Australian National Electricity Market [66]. Barman *et al.* proposed a STLF model utilising Support Vector Machine (SVM) with Grasshopper Optimisation Algorithm (GOA) [67]. With the development of artificial intelligence and the increase of load complexity, several deep learning methods have been applied to the field of STLF in recent years. Kong *et al.* applied the Long Short Term Memory (LSTM) Recurrent Neural Network (RNN) in STLF, especially for individual residential households [68]. Shi *et al.* proposed a novel approach of Pooling Deep Recurrent Neural Network (PDRNN), which improved the forecasting accuracy by adding more hidden layers [69]. Khan *et al.* analysed the electricity load by a Deep Convolution Neural Network (DCNN) and compare the forecasting results with ELM, RNN and auto regressive methods [70]. Among these deep learning algorithms, DBN had been gradually applied into STLF in recent years due to its high efficiency in seeking optimal parameters [71–73]. Dedinec *et at.* incorporated DBN model into a feed forward NN, and applied it to STLF [72]. Fu introduced and properly designed a DBN model to forecast the cooling load data, which mitigated the impact of uncertainties on forecasting accuracy [73].

The aforementioned methods use centralised algorithms to build STLF models. However, with the integration of renewable energies and the rapid development of smart

meters, these centralised algorithms face the following three issues. Firstly, for large-scale power systems, it is difficult to collect all load variation with a single centralised computing centre [74]. Secondly, a large number of electricity meters will bring a huge amount of load data. Processing all load data and training with a single computing centre will increase the load and the losses of the computing centre [75]. Finally, centralised collection and processing of load data raise security issues [27]. For instance, once a single point failure occurs, it will cause incalculable damage to the power systems. It became more challenging to deal with such massive high-dimensional load data in terms of the timely requirements in STLF, and therefore many scholars have studied distributed frameworks to accelerate STLF process. Li *et al.* separated the STLF model into 24 ELM to forecast the load of the next 24 hours [76]. Wang *et al.* employed the cloud computing technology on the ELM algorithm [77]. The proposed method not only improves the accuracy of STLF, but also overcomes the challenge of insufficient computing resources of a single machine under the massive high-dimensional data due to the intellectualisation of smart grid. Liu *et al.* separated the STLF into several local forecasting models and propose a distribute STLF model based on local weather information, which can enhance the accuracy of system-level load forecasting model [74].

As for STLF, load data security is an essential issue for decision maker in electric market and directly related to economic benefits [78]. The distributed structure will inevitably lead to data security considerations and the research on cyberattacks in power systems has been going on for a while. Cui *et al.* detected the cyberattacks on load forecasting by a supervised machine learning method, and estimate the influence of cyberattacks on the STLF parameters [79]. Gu *et al.* applied time forward kriging method on load forecast to overcome the challenge of communication failure [80]. For a multi-agent based distributed structure, the security margin of the system network was determined by the fixed communication protocol [81]. For example, the actual load cannot be collected

timely and accurately due to communication failures, which will reduce the forecast accuracy of STLF, further compromising power systems reliability. Therefore, it became a very pragmatic issue to design a distributed algorithm, which can improve the security of distributed STLF and meanwhile maintain its model accuracy comparable to that of centralised models. This motivates our current work.

2.4 Summary

In this chapter, the literature review surrounding distributed optimisation is given firstly, and then the reinforcement learning based bidding strategy about energy and regulation market is briefly introduced. In addition, the background of distributed deep belief networks based short-term load forecasting is also presented. Through comparison with the traditional control and optimisation algorithms, the advantages of these distributed methods are illustrated as well.

Chapter 3

Preliminaries

In this chapter, we recall some preliminaries related to this thesis, including graph theory, convex analysis and the projection, reinforcement learning and restricted Boltzmann machine.

3.1 Graph Theory

Following [82], an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ can be used to describe the communication topology among the power units, where $\mathcal{V} = \{\nu_1, \cdots, \nu_N\}$ is the vertex set and $\mathcal{E} \in \mathcal{V} \times \mathcal{V}$ is the edge set. The adjacency matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$ of $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is an $N \times N$ matrix, such that $a_{ij} = 1$ if $(\nu_j, \nu_i) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. Define the degree matrix $\mathcal{D} = \text{diag}\{\sum_{j=1}^{N} a_{1j}, \sum_{j=1}^{N} a_{2j}, \cdots, \sum_{j=1}^{N} a_{Nj}\}$. A graph is connected if and only if every pair of vertices can be connected by a path, namely, a sequence of edges. In this thesis, we assume that the graph is connected and undirected. The

Laplacian matrix related to $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{A})$ is defined as $\mathcal{L} = \mathcal{D} - \mathcal{A}$, i.e.,

$$\mathcal{L} = \begin{cases} l_{ij} = -a_{ij}, i \neq j \\ l_{ii} = \sum_{i \neq j} a_{ij}. \end{cases} \tag{3.1}$$

When $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is a connected undirected graph, 0 is an eigenvalue of Laplacian $\mathcal{L}$ with the eigenvector $\mathbf{1}_N$ and all the other eigenvalues are positive. Then,

$$\mathcal{L}1_N = 0_N, \;\; 1_N^T \mathcal{L} = 0_N^T. \tag{3.2}$$

3.2 Convex Analysis and Projection

From [83], a set $\Omega \subseteq \mathbb{R}^n$ is convex if $\theta x_1 + (1-\theta)x_2 \in \Omega$ for any $x_1, x_2 \in \Omega$ and $0 \leqslant \theta \leqslant 1$. For a closed convex Ω, the projection map $P_\Omega : \mathbb{R}^n \to \Omega$ is defined as

$$P_\Omega(x) = \arg\min_{y \in \Omega} \|x - y\|. \tag{3.3}$$

Then the following inequalities hold, $\exists x \in \mathbb{R}^n, y \in \Omega$

$$\begin{aligned} &(x - P_\Omega(x))^T (P_\Omega(x) - y) \geqslant 0, \\ &\|x - P_\Omega(x)\|_2^2 + \|P_\Omega(x) - y\|_2^2 \leqslant \|x - y\|_2^2, \end{aligned} \tag{3.4}$$

For $x \in \Omega$, the normal cone to Ω is

$$\mathcal{N}_\Omega(x) \triangleq \{v \in \mathbb{R}^n | v^T(y - x) \leqslant 0, \text{for all } \; y \in \Omega\}. \tag{3.5}$$

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if $f(\theta x_1 + (1-\theta)x_2) \leqslant \theta f(x_1) + (1-\theta) f(x_2)$ for any $x_1, x_2 \in \Omega$ and $0 \leqslant \theta \leqslant 1$.

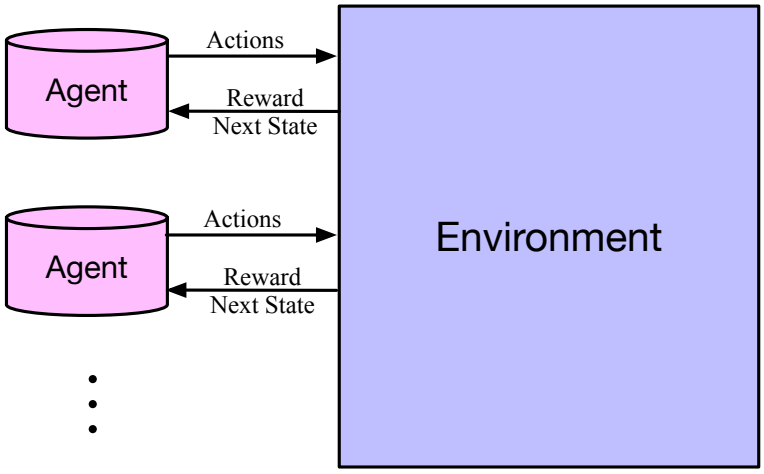

Figure 3.1: The framework of reinforcement learning.

3.3 Reinforcement Learning

In this section, we recall some necessary concepts related to the reinforcement learning algorithm. The basic reinforcement learning elements are agent, environment, actions, reward and states, which are shown in Fig. 3.1.

The reinforcement learning problems can be viewed as MDP, which is the stochastically changing system. It is composed of state $\mathcal{S}$, action $\mathcal{A}$, transition probability function $\mathcal{P}$, reward function, $\mathcal{R}$ and discount factor γ. Therefore, MDP is defined as a five-element tuple in this thesis:

$$\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma\} \tag{3.6}$$

At each time slot t, the intelligent agent has its observation of the environment, namely state s_t. Then, the agent will choose its action followed by a policy function $\pi(s_t) : \mathcal{S} \rightarrow \mathcal{A}$, which denotes a distribution over actions for each state. In the reinforcement learning, there is a transition function $\mathcal{P}(s_t, a_t, s_{t+1}) : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$. It maps state s_t to s_{t+1} by action a_t, which means the dynamics of the environment. The transition function is unknown and has part of stochastic factors. Thus, the agent needs to learn

it through different $\{s_t, a_t, s_{t+1}\}$ sets during the training process. Specific to each state transition between adjacent time slots, the environment will provide a reward signal $r_t \in \mathcal{R}$ to the agent. Then the trajectory $\{s_0, a_0, s_1, a_1, \cdots, s_T, a_T\}$ can be derived with the discounted trajectory return $\sum_{t=0}^{T} \gamma^t R(s_t, a_t)$. For any policy π, the value function of state s can be defined as the expected total discounted reward:

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t R(s_t, a_t) | s_t = s\right], \forall s \in \mathcal{S} \tag{3.7}$$

Then the corresponding state-action value function Q^{π} is defined as:

$$Q^{\pi}(s, a) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t R(s_t, a_t) | s_t = s, a_t = a\right], \forall s \in \mathcal{S}, \forall a \in \mathcal{A} \tag{3.8}$$

According to the Bellman equation [84], the value function Q^{π} can be represented in a recursive format:

$$Q^{\pi}(s_t, a_t) = \mathbb{E}\left[R(s_t, a_t, s_{t+1}) + \gamma Q^{\pi}(s_{t+1}, \pi(s_{t+1}))\right] \tag{3.9}$$

where $R(s_t, a_t, s_{t+1})$ is the observed reward after taking action a_t at state s_t and resulting in state s_{t+1}. The equation (3.9) indicates that the Q function can be improved by using current value of the Q^{π} estimation. To reduce computational complexity, TD learning is one of the most famous update methods instead of Monte-Carlo and Dynamic Programming. It only requires current state s_t, current action a_t, reward $R(s_t, a_t, s_{t+1})$ and next state s_{t+1}:

$$Q^{\pi}(s_t, s_t) = Q^{\pi}(s_t, a_t) + \alpha \delta_t \tag{3.10}$$

$$\delta_t = r_t + \gamma \max_{a_{t+1}} Q^{\pi}(s_{t+1}, a_{t+1}) - Q^{\pi}(s_t, a_t) \tag{3.11}$$

where $0 \leq \alpha \leq 1$ is the learning rate. δ_t is the TD error at the time slot t, which implies the correction between the estimation and target value of Q function. When time goes to infinity, $Q^{\pi}(s, a)$ will converge to its optimal value $Q^{*}(s, a)$ for all state-action pairs.

Here, the optimal value function $Q^*(s,a) = \sup_\pi Q^\pi(s,a)$ is defined for all state action pairs $(s,a) \in \mathcal{S} \times \mathcal{A}$. With the optimal Q^* function, the optimal policy π^* can be obtained by greedy algorithm:

$$\pi^*(a|s) = 1, \quad \text{if} \quad Q^\pi(s,a) = \max_{a'} Q^*(s,a') \tag{3.12}$$

where a' is any possible action associate with state s.

3.4 Restricted Boltzmann Machine

The Restricted Boltzmann Machine (RBM) is the underlying architecture of a DBN. It combines the traditional neural networks with the energy model and the probabilistic model. For the STLF, the inputs are different features related to the load data. To get more accurate results, many previous works have been conducted on the feature selection and parameter initialisation, such as deleting the unusual features or tuning the parameter weights. However, STLF dataset is usually vast and complex, and tuning the initial parameters using the entire dataset requires significant computation resources within a short time. In the proposed DDBN model, the RBM is applied to handle the parameter initialisation problem, which is an unsupervised learning process and trained by minimising the energy function, and thus it can keep all features in the model.

An RBM aims to learn a probability distribution over the input sets. It has one hidden layer and one visible layer, shown as the Fig. 3.2. The original probability of distribution through an energy function is described as:

$$P(\boldsymbol{v}) = \frac{1}{Z}\exp[-\mathcal{F}(\boldsymbol{v})], \tag{3.13}$$

$$Z = \sum \exp[-\mathcal{F}(\boldsymbol{v})], \tag{3.14}$$

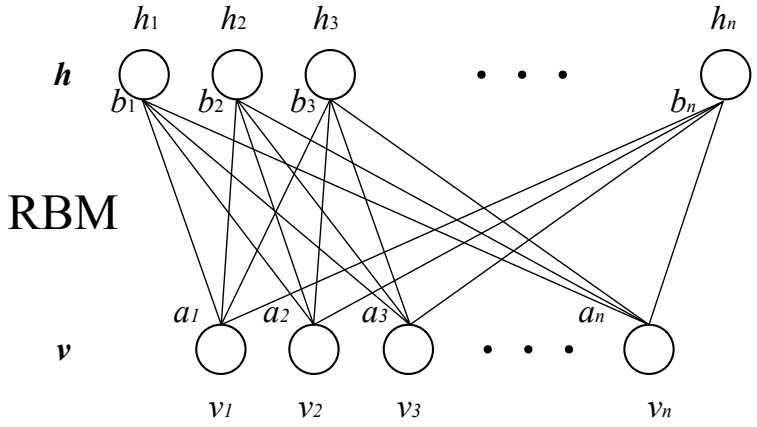

Figure 3.2: The framework of RBM.

where Z is the normalising factor, $\mathcal{F}(\cdot)$ represents the energy function and $\boldsymbol{v}$ denotes the visible layer of the RBM. A lower energy value means that the network has more probability $P(\boldsymbol{v})$. Since the RBM has the hidden layer parameters $\boldsymbol{h}$, the probability function can be rewritten as:

$$P(\boldsymbol{v}, \boldsymbol{h}) = \frac{1}{Z} \exp[-\mathcal{F}(\boldsymbol{v}, \boldsymbol{h})]. \tag{3.15}$$

In RBM, the energy function is formulated as a second-order polynomial, which is described as:

$$\mathcal{F}(\boldsymbol{v}, \boldsymbol{h}) = -\sum_{i=1}^{n_v} b_i v_i - \sum_{j=1}^{n_h} c_j h_j - \sum_{i=1}^{n_v} \sum_{j=1}^{n_h} v_i h_j W_{ij}, \tag{3.16}$$

where a_i, b_j are the biases and W_{ij} is the weight between the hidden layer and the visible layer, v_i, h_j are the binary states of unit i and unit j (values 0 or 1), n_v and n_h are the number of visible and hidden units.

3.5 Summary

In this chapter, we have recalled some basic concepts and preliminaries related to the thesis including graph theory, convex analysis, projection, RBM and reinforcement learning.

Chapter 4

Demand Side Management Using a Distributed Initialisation-free Optimisation in a Smart Grid

Due to the integration of the renewable generation and the distributed load that inherently uncertain and unpredictable, developing an efficient distributed management structure of such a complex system remains a challenging issue. Most of the existing works on the demand side management concentrate on the centralised methods or need a proper initialisation process; This chapter proposed a demand side management strategy that can solve the optimisation problem in a distributed manner without initialisation. The objective of the designed demand management system is to maximise the social welfare of a smart grid by controlling the active power economically. The proposed optimisation strategy generates the optimal power references uses the neighbouring information while considering the local feasible constraints by using a projection operation.

Furthermore, the optimisation algorithm is initialisation-free, which avoids any initialisation process when plugging-in new customers or plugging-out power units, such as demand loads, battery energy storage systems and distributed generators. Our strategy only uses the neighbouring information, so that the proposed approach is scalable and potentially applicable to large-scale smart grids. The effectiveness and scalability of the proposed algorithm are established and verified through case studies.

4.1 Problem formulation

In this section, we formulate the welfare functions of the power units involved in a smart grid, consisting of demand units, BESS, and DG. The objective of the proposed model is to maximise the total welfare associated with the demand units, BESS and DG, including operating costs [11, 15]. Also, we consider emission cost in our model. Therefore, we have

$$\max_{P_i} \left\{ \sum_{i \in S_D}^{m} W_{D,i}(P_i) + \sum_{i \in S_B}^{n} W_{B,i}(P_i) + \sum_{i \in S_G}^{k} W_{G,i}(P_i) \right\}, \tag{4.1}$$

To ensure that all power units work in the normal mode, the formulated problem should be subject to the power balance constraint and local power constraints that will be discussed later. Note that the power output of the battery can be positive or negative, depending on discharging and charging states, respectively.

4.1.1 Controllable Power Units

1. Welfare on Controllable Demand Units

The welfare function of the ith load is formulated as the level of consumer satisfaction, which is related to the power consumption of applications. For the demand customers, consuming more power will bring more satisfaction. Therefore, similar to [15], the welfare function is defined as

$$W_{D,i}(P_{D,i}) := \begin{cases} -\alpha_i P_{D,i}^2 + \beta_i P_{D,i}, & P_i^{\min} \leqslant P_{D,i} \leqslant \beta_i/2\alpha_i \\ \beta_i^2/4\alpha_i, & \beta_i/2\alpha_i \leqslant P_{D,i} \leqslant P_i^{\max} \end{cases} \tag{4.2}$$
$$i = 1, 2, \cdots, m$$

where the power utility satisfies $P_i^{\min} \leq P_{D,i} \leq P_i^{\max}$. Here, the satisfaction level of customer increases with the consumption of electrical power and will eventually get saturated.

2. Welfare on BESS

To save electricity and balance the uncertain power generation, some BESS are also installed in the smart grid. Referring to [85], we formulate the welfare function of BESS as

$$W_{B,i}(P_{B,i}) := pP_{B,i} - f_{B,i}(P_{B,i}). \tag{4.3}$$

It is noticed that the BESS can be counted as both demand side and supply side. The power output of BESS can be positive or negative, depending on its discharging or charging states, respectively. If the electricity price is cheap and no dramatically power mismatch, then the BESS would be charged to storage power energy, and vice versa. Because of the cost varies with the characteristics of BESS, following the approximation in [85], both of the charging and discharging process will increase the Depth of Discharge (DoD) cost of BESS. Hence, the cost function can be uniformly expressed as

$$\begin{aligned} f_{B,i}(P_{B,i}) &:= \kappa_f + \kappa_c(1+\iota)|P_{B,i}|. \\ i &= 1, 2, \cdots, n. \end{aligned} \tag{4.4}$$

Furthermore, the power output satisfies $P_{B,i}^{\min} \leq P_{B,i} \leq P_i^{\max}$ with the minimum and maximum power output $P_i^{\min}, P_i^{\max}$. Here we assume that BESS are eco-friendly [86], and therefore the emission cost of BESS is zero.

3. Welfare on Generators

The welfare function for the power generators is usually formulated by the income minus the costs [11, 15]. The costs of DG mainly include the Operation and Maintenance (O&M) costs [24,87] which can be expressed as a quadratic function and a linear function of active power respectively. Generally, the O&M cost of DG is expressed as:

$$\begin{aligned} f_{G,i}^{O\&M}(P_{G,i}) &:= a_i P_{G,i}^2 + b_i P_{G,i} + c_i, \\ & i = 1, 2, \cdots, k, \end{aligned} \tag{4.5}$$

where the power output satisfies $P_i^{\min} \leq P_{G,i} \leq P_i^{\max}$ with $P_i^{\min}, P_i^{\max} \in \mathbb{R}^{++}$.

For comparison purposes, the total emission cost of various pollutants is generally expressed as [88–90]

$$\begin{aligned} f_{G,i}^{E}(P_{G,i}) &:= \alpha_i P_{G,i}^2 + \beta_i P_{G,i} + \gamma_i + \eta_i \exp(\varphi_i P_{G,i}), \\ & i = 1, 2, \cdots, k, \end{aligned} \tag{4.6}$$

where $f_{G,i}^{E}(P_i)$ is the total pollution emission cost for the ith power generator.

Therefore, the welfare function is expressed as

$$W_{G,i}(P_{G,i}) := pP_{G,i} - f_{G,i}^{O\&M}(P_{G,i}) - f_{G,i}^{E}(P_{G,i}). \tag{4.7}$$

Basically, the welfare for the generation unit represents the benefit of selling power minus the cost of operation and emission. The first term in (4.7) means the income by selling energy and the other terms denote the cost caused by maintenance and pollution.

4.1.2 Uncontrollable Power Units

From [12,24], renewable generators and users' loads are uncontrollable power units and their power output and consumption are related to the light intensity, illumination time, wind speed, and customers' habits, etc. Therefore, these uncontrollable power units generate power by different conditions. These uncontrollable power units are considered as undispatchable in this chapter.

In the smart grid, transmission losses are inevitable, accounting for around 5-7$\%$ of the total power load [91], which can be modelled by multiplying the load with this percentage. Overall, to maintain system stability, the active power balance between the supply and the demand side is described as

$$\sum_{i \in S_G} P_{G,i} + \sum_{i \in S_R} P_{R,i} = \sum_{i \in S_D} P_{D,i} + \sum_{i \in S_B} P_{B,i}. \tag{4.8}$$

Since the renewable source is undispatchable and the load is related to the customers' habits which are partial controllable, we rewrite the above constraint as

$$P_D = \sum_{i \in S_G}^{k} P_{G,i} + \sum_{i \in S_B}^{n} P_{B,i} + \sum_{i \in S_C}^{m} P_{C,i}. \tag{4.9}$$

where P_D represents the power mismatch of the systems, $P_{G,i}$ is the power output of generators. $P_{C,i}$ denotes the power output of controllable loads and S_C is the set of controllable loads. Here, $P_{C,i}$ are the negative numbers since the power is consumed. Here, $P_{B,i}$ can be negative or positive.

4.1.3 Problem Reformulation

In this chapter, our objective is to design a reliable demand side management system that can maximise the social welfare while maintaining active power balance under various conditions. To this end, an objective function is formulated by integrating the above welfare functions and subjecting to physical constraints. For notation convenience, we denote the power vector as $P = [P_1, P_2, \cdots, P_m, P_{m+1}, P_{m+2}, \cdots, P_{m+n}, P_{m+n+1}, P_{m+n+2}, \cdots, P_{m+n+k}]^T \in \mathbb{R}^N$, where m, n and k denote the numbers of demand units, BESS and DG with $N = m + n + k$.

$$\begin{aligned} &\min \sum C_i(P_i), \;\; i = 1, 2, \cdots, N, \\ &\text{s.t.} \;\; \sum_i P_i = P_D, \\ &\qquad P_i^{\min} \leq P_i \leq P_i^{\max}, \end{aligned} \tag{4.10}$$

where $C_i(P_i) \triangleq -\sum_{i \in S} W_i(P_i)$, $S = S_D \cup S_B \cup S_G$ and $W_i(P_i)$ denotes the welfare of the ith unit. Notice that the cost function $C_i(P_i)$ of each unit is strictly convex and continuously differentiable.

Traditionally, the constrained optimisation problem can be solved using centralised methods, but those algorithms require a powerful control centre to collect data from the subsystems and distribute control instruction to the units after calculation. In the following section, we design a distributed algorithm, where each subsystem is allocated with a low-price processor, by which the collection and calculation can be performed locally.

4.2 Distributed Solution

In this section, a projection-based gradient decent algorithm is developed, by which the inequality constraints can be tackled accordingly. We consider the Lagrangian function for each unit with the affine equality constraint, written as

$$L = \sum_{i=1}^{N} C_i(P_i) - \bar{\lambda}[\sum_{i=1}^{N} P_i - P_D], \tag{4.11}$$

where $\bar{\lambda}$ is the Lagrangian multiplier which is used to ensure the equality constraints are met during the optimisation process. The inequality constraints are not considered in the Lagrangian function since the inequality constraint can be solved by local units with projection algorithm.

The optimal solution can be obtained by using a well-known centralised saddle-point dynamics as

$$\begin{aligned} \frac{\partial L}{\partial P_i} &= \nabla C_i(P_i) - \bar{\lambda} = 0 \\ \frac{\partial L}{\partial \bar{\lambda}} &= \sum_{i=1}^{N} P_i - P_D = 0. \end{aligned} \tag{4.12}$$

Its equilibrium points (4.11) satisfy the KKT conditions (see, e.g., [83]). However, it collects the global information about the Lagrangian multiplier ($\bar{\lambda}$) of all the power units. In the modern smart grid, most loads are distributed, so that it is desired to design a distributed algorithm for DSM that solves the optimisation problem locally. To facilitate our design, a local copy $\bar{\lambda}_i$ of the global variable is used to estimate the global ($\bar{\lambda}$). As a result, the problem is solved when the local copy variables converge to the $\bar{\lambda}^\star$. According to the KKT conditions, the following lemma is proposed.

Lemma 4.2.1. *The optimisation problem has an optimal solution $P^\star$ if and only if there exists a vector of Lagrangian multiplier $\lambda^\star \triangleq \bar{\lambda}^\star \mathbf{1}_N \in \mathbb{R}^N$ such that*

$$\nabla C(P) = \bar{\lambda}^\star \mathbf{1}_N, \tag{4.13}$$

where $C(P) \triangleq [C_1(P_1), C_2(P_2), \cdots, C_N(P_N)]^T$ *denotes cost function vector, and* $\nabla C(P)$ *represents its gradient.*

4.2.1 Algorithm Design

Inspired by [9, 10], the objective of the distributed algorithm is developed in order to achieve a consensus of the Lagrangian multiplier and reach the optimal power output. Let $\Omega_i \triangleq [P_i^{\min}, P_i^{\max}]$ denote the feasible domain of the ith unit's power output, which is clearly a compact and convex set. The problem (4.10) can be solved by the following distributed algorithm, $\forall i = 1...N$:

$$\dot{P}_i = P_{\Omega_i}[P_i - \nabla C_i(P_i) + \phi_i] - P_i, \tag{4.14a}$$

$$\dot{\phi}_i = \sum_{j=1}^{N} a_{ij}(\phi_j - \phi_i) - \xi_i + (\theta_i P_D - P_i) + \lambda_i - \phi_i, \tag{4.14b}$$

$$\dot{\lambda}_i = -\lambda_i + \phi_i, \tag{4.14c}$$

$$\dot{\xi}_i = \sum_{j=1}^{N} a_{ij}(\phi_i - \phi_j), \sum_{i=1}^{N} \xi_i(0) = 0, \tag{4.14d}$$

where $P_i \in \mathbb{R}$ is the power output of the ith power unit and $\phi_i, \xi_i \in \mathbb{R}$ are two auxiliary variables of the ith unit. The convergence analysis of (4.14) is detailed in section 4.2.2.

Note that the proposed algorithm can ensure the local feasible constraints during the process with projection operations. Furthermore, the algorithm (4.14) does not require any initialisation process which is proved in next section, and therefore, it can ensure the network constraint asymptotically without concern whether it is satisfied at the initial points. Due to free of any control centre and initialisation process, the proposed algorithm can work in a "plug-and-play" manner for the power system with plugging-in or plugging-off of power units.

The algorithm (4.14) is distributed, in this sense that the ith unit only needs the local data and the information which is shared with the neighbouring units. Therefore, (4.14) does not require any centre to process the data or coordinate the units. Since the local data do not need to be uploaded and downloaded from the centre, each unit can respond to local data changes rapidly which can quickly adapt the local decisions. The algorithm can be understood concerning singular perturbation, where the third dynamic is on a faster scale than the second one. Hence, λ_i goes to ϕ_i, as t goes to infinity, and substituting this to the algorithm yields a saddle-point seeking algorithm. We will show the detailed proof in the next section. Different from the algorithm in [10], the proposed algorithm does not require each unit to know the load information, $\sum_{i\in S} P_i$ is not always equal to P_D. Furthermore, the algorithm (4.14) introduced the estimation ability $\theta_i \in [0, 1]$ to detect the total mismatch, which can be selected according to [27] and $\sum_i^N \theta_i = 1$. Therefore, this algorithm could be applied to a large grid.

Notably, the proposed algorithm does not require the initialisation process so that it is more compatible with the operation of a smart grid. The initialisation for the network resource constraint is quite restrictive for a sizeable dynamical grid because it is related to the global coordination and has to be performed whenever the network data/configuration changes. However, the communication network may change frequently and rapidly in a smart grid, such as when plugging in an electric vehicle and routine maintenance. Furthermore, it is not trivial to achieve the initialisation coordination with both the local feasibility and the network resource constraints.

In addition, the proposed algorithm has some limitation on the communication networks. The communication networks should be connected and no time delay. Also, the proposed algorithm can asymptotically converge to its global optimal location, but cannot converge in a fixed time.

4.2.2 Convergence Analysis

The convergence analysis follows similar steps to [10]. To analysis the convergence of our algorithm, the algorithm (4.14) is rewritten in a compact form as

$$\begin{aligned}
\dot{P} &= P_{\Omega}(P - \nabla C(P) + \Phi) - P, \\
\dot{\Phi} &= -\mathcal{L}\Phi - \Xi + \Theta P_D - P + \Lambda - \Phi, \\
\dot{\Lambda} &= \Phi - \Lambda, \\
\dot{\Xi} &= \mathcal{L}\Phi,
\end{aligned} \tag{4.15}$$

where $P = [P_1, P_2, \cdots, P_N]^T$, $\Omega = \Omega_1 \times \Omega_2 \times \cdots \times \Omega_N$, $\Lambda = [\lambda_1, \lambda_2, \cdots, \lambda_N]^T$, Θ=$[\theta_1, \theta_2, \cdots, \theta_N]^T$, $\Phi = [\phi_1, \phi_2, \cdots, \phi_N]^T$ and $\Xi = [\xi_1, \xi_2, \cdots, \xi_N]^T$. Firstly, we consider the equilibrium point $\bar{P}, \bar{\Lambda}, \bar{\Xi}$ and $\bar{\Phi}$, which can be obtained by

$$\begin{aligned}
0 &= P_{\Omega}(\bar{P} - \nabla C(\bar{P}) + \bar{\Phi}) - \bar{P}, \\
0 &= -\mathcal{L}\bar{\Phi} - \bar{\Xi} + \Theta P_D - \bar{P} + \bar{\Lambda} - \bar{\Phi}, \\
0 &= \bar{\Phi} - \bar{\Lambda}, \\
0 &= \mathcal{L}\bar{\Phi}.
\end{aligned} \tag{4.16}$$

Left multiplying the second equation in (4.16) by 1_N^T yields

$$1_N^T \Xi = 1_N^T(\Theta P_D - P). \tag{4.17}$$

With the algorithm (4.14), Λ will converge to Φ as $t \to \infty$ and ϕ_i will achieve consensus by the last equation. Furthermore, we have the first equation to make sure that $\Delta C(P)$ converges to Φ, which is equal to $\lambda^\star$ at $t \to \infty$. Therefore, with the Lemma 4.2.1, the equilibrium point satisfies

$$\bar{P} = P^\star, \bar{\Phi} = \Phi^\star, \bar{\Lambda} = \Lambda^\star, \bar{\Xi} = \Xi^\star, \tag{4.18}$$

where $P^\star, \Lambda^\star, \Xi^\star$ and $\Phi^\star$ are the optimal solution of the allocation problem in (4.10).

We define the new variables for equilibrium point of (4.15),

$$\begin{aligned} &\hat{P} = P - P^\star, \;\; \hat{\Lambda} = \Lambda - \Lambda^\star, \;\; \hat{\Phi} = \Phi - \Phi^\star, \\ &\hat{\Xi} = \Xi - \Xi^\star, \;\; S = [r \; R]^T \hat{\Phi}, \;\; U = [r \; R]^T \hat{\Xi}, \end{aligned} \tag{4.19}$$

where $[r \; R]^T$ is an orthonormal matrix, here we let $r \in \mathbb{R}^N, R \in \mathbb{R}^{N\times(N-1)}$, then we have

$$\begin{aligned} &[r \; R]^T [r \; R] = [r \; R][r \; R]^T = I_N, \\ &r^T R = 0_{N-1}^T, R^T R = I_{N-1}, \\ &RR^T = I_N - rr^T, \end{aligned} \tag{4.20}$$

and we divide the new variables as $U = (u_1, U_{2:N})$ and $S = (s_1, S_{2:N})$

$$\begin{aligned} &\dot{u}_1 = 0, \\ &\dot{U}_{2:N} = R^T \mathcal{L} R S_{2:N}, \\ &\dot{s}_1 = -s_1 + r^T(\hat{\Lambda} - \hat{P}), \\ &\dot{S}_{2:N} = -R^T \mathcal{L} R S_{2:N} - U_{2:N} - S_{2:N} + R^T(\hat{\Lambda} - \hat{P}), \\ &\dot{\hat{\Lambda}} = -\hat{\Lambda} + [r \; R] S, \\ &\dot{\hat{P}} = P_\Omega(\hat{P} + P^\star - \nabla C(\hat{P} + P^\star) + \nabla C(P^\star) + [r \; R] S) \\ &\quad - (\hat{P} + P^\star). \end{aligned} \tag{4.21}$$

Let $Q(\hat{P}, S) = \nabla C(\hat{P} + P^\star) - [r \; R] S - \nabla C(P^\star)$ and define the Lyapunov function as

$$\begin{aligned} V = &- Q(\hat{P}, S)^T (P_\Omega(\hat{P} + P^\star - Q(\hat{P}, S)) - (\hat{P} + P^\star)) \\ &- \frac{1}{2} \| P_\Omega(\hat{P} + P^\star - Q(\hat{P}, S)) - (\hat{P} + P^\star) \|_2^2 \\ &+ \frac{1}{2} \|\hat{P}\|_2^2 + \frac{1}{2} \|\hat{\Lambda}\|_2^2 + \frac{1}{2} \|S\|_2^2 + \frac{1}{2} U_{2:N}^T (R^T \mathcal{L} R)^{-1} U_{2:N}. \end{aligned} \tag{4.22}$$

With (3.4) and if $(\hat{P}(0)+P^\star)\in\Omega$ then the parameter $(\hat{P}(t)+P^\star)\in\Omega$. We have

$$\begin{aligned}V =&\frac{1}{2}\|Q(\hat{P},S)\|_2^2\\&-\frac{1}{2}\|Q(\hat{P},S)+P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-(\hat{P}+P^\star)\|_2^2\\&+\frac{1}{2}\|\hat{P}\|_2^2+\frac{1}{2}\|\hat{\Lambda}\|_2^2+\frac{1}{2}\|S\|_2^2+\frac{1}{2}U_{2:N}^T(R^T\mathcal{L}R)^{-1}U_{2:N}\\ \geqslant&\frac{1}{2}\|P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-(\hat{P}+P^\star)\|_2^2+\frac{1}{2}\|\hat{P}\|_2^2\\&+\frac{1}{2}\|\hat{\Lambda}\|_2^2+\frac{1}{2}\|S\|_2^2+\frac{1}{2}U_{2:N}^T(R^T\mathcal{L}R)^{-1}U_{2:N}.\end{aligned}\tag{4.23}$$

Therefore, $V\geqslant 0$ and $V=0$ if and only if $\hat{P}=0$. With the Theorem 3.2 in [92], differentiating V yields

$$\begin{aligned}\dot{V} =&-(P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-(\hat{P}+P^\star))^T\nabla_{\hat{P}}Q(\hat{P},S)\dot{P}\\&+\|P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-(\hat{P}+P^\star)\|_2^2\\&+Q(\hat{P},S)^T(P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-(\hat{P}+P^\star))\\&+\hat{P}^T(P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-(\hat{P}+P^\star))\\&+\hat{\Lambda}\dot{\Lambda}+\hat{S}\dot{S}+S_{2:N}^TU_{2:N}.\end{aligned}\tag{4.24}$$

From (3.4), we have $(\hat{P}+P^\star-Q(\hat{P},S)-P_\Omega(\hat{P}+P^\star-Q(\hat{P},S)))^T(P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-P^\star)\geqslant 0$. Hence,

$$\begin{aligned}&(P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-(\hat{P}+P^\star))^T(\hat{P}+Q(\hat{P},S))\\\leqslant&-\|P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-(\hat{P}+P^\star)\|_2^2-Q(\hat{P},S)^T\hat{P}.\end{aligned}\tag{4.25}$$

Since $\nabla_{\hat{P}}Q(\hat{P},S)=\nabla^2C(\hat{P}+P^\star)\geqslant 0$, and therefore $-(P_\Omega(\hat{P}+P^\star-Q(\hat{P},S))-$

$(\hat{P}+P^{\star}))^T\nabla_{\hat{P}}Q(\hat{P},S)\dot{P}\geqslant 0$. Consequently

$$\begin{aligned}\dot{V}\leqslant & -Q(\hat{P},S)^T\hat{P}\\ & +\dot{\hat{\Lambda}}^T\hat{\Lambda}+\dot{s}_1s_1+\dot{S}_{2:N}^TS_{2:N}+S_{2:N}U_{2:N}\\ = & -\hat{P}^T(\nabla C(\hat{P}+P^{\star})-[r\ R]S-\nabla C(P^{\star}))\\ & -\hat{\Lambda}^T\hat{\Lambda}+\hat{\Lambda}^T[r\ R]S-s_1^2+S^T[r\ R]^T(\hat{\Lambda}-\hat{P})\\ & -S_{2:N}^TR^T\mathcal{L}RS_{2:N}-U_{2:N}^TS_{2:N}+S_{2:N}^TU_{2:N}\\ \leqslant & -\hat{P}^T(\nabla C(\hat{P}+P^{\star})-\nabla C(P^{\star}))-S_{2:N}^TR^T\mathcal{L}RS_{2:N}\\ \leqslant & -\hat{P}^T(\nabla C(\hat{P}+P^{\star})-\nabla C(P^{\star})).\end{aligned}\tag{4.26}$$

Because the Hessian matrix of $C(P)$ is positive definite, we have

$$\nabla C(\hat{P}+P^{\star})=\nabla C(P^{\star})+\int_0^1\nabla^2C(1-\tau\hat{P})^T\hat{P}d\tau.\tag{4.27}$$

Therefore, $\dot{V}\leqslant 0$. Based on the LaSalle invariance principle and the Lyapunov stability theory. The system (4.15) converges to its equilibrium point if and only if $\dot{V}=0$, and therefore $\hat{P}$ goes to 0, then $P=P^{\star}$, which means the optimisation problem is solved.

The algorithm described in this chapter uses projection operations to map the variables into their specified domains, and completes the optimisation of power dispatch in the smart grid.

4.2.3 Algorithm Implementation

The proposed distributed algorithm adopts a distributed average consensus estimator to measure the global mismatch information locally. The step-by-step algorithm for all power units is shown in Algorithm 1. To illustrate more clearly, a flow chart for solving distributed demand side management by proposed algorithm is shown in Fig.

4.2. As in [21], the communication structure can be designed independently of the power system in a cost-efficient way based on the location and the convenience of the smart grid. For example, a communication topology is designed as Fig. 4.1. DG, BESS and controllable loads are connected through a communication network as the power units in a smart grid. To obtain the optimal power outputs, each power unit only interacts with its neighbouring units to exchange the information through the communication network at the top level, and then each power unit performs the proposed algorithm accordingly.

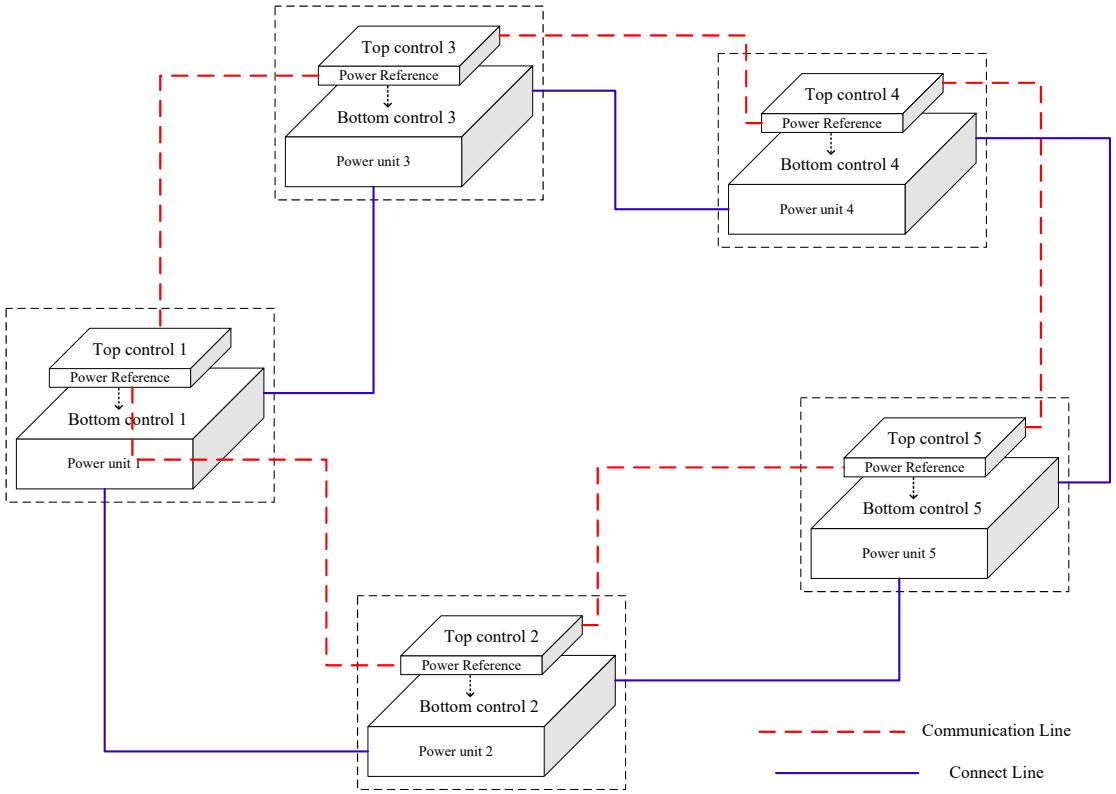

Figure 4.1: Two-level structure of optimisation algorithm.

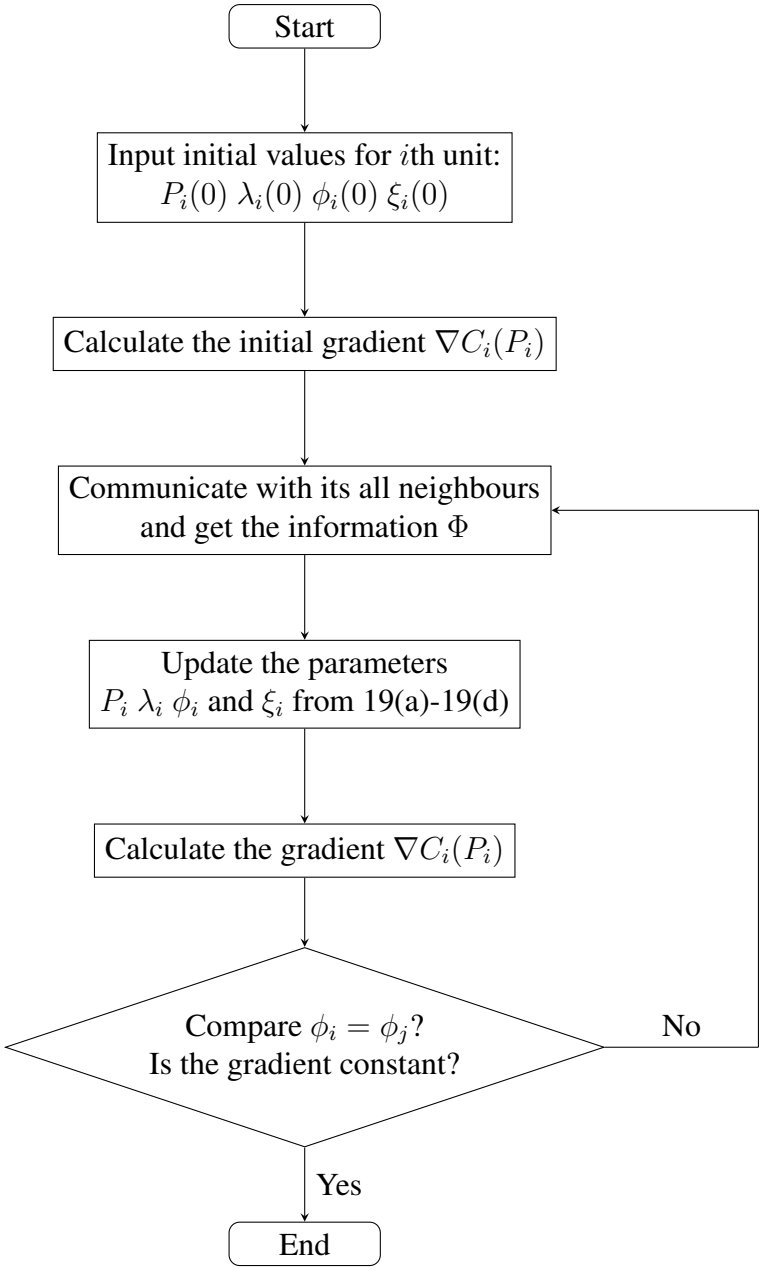

Figure 4.2: Flowchart for ith power unit management.

Algorithm 1 Distributed Optimal Demand Side Management

Input: Power mismatch $\theta_i P_D$ for ith power unit.
Output: Optimal power reference for each power unit.
Initialisation:
For i=1,2,$\cdots$,N
$P_i = P_i(0), \lambda_i = \lambda_i(0), \phi_i = \phi_i(0), \xi_i = \xi_i(0)$
Consensus Algorithm:
For ith power unit:
Communicate and get the parameter ϕ_j from its neighbour j;
Calculate and updates its own parameters ϕ_i, ξ_i, λ_i and P_i according to 19(a)-19(d)
End if: Each power unit achieves the optimal values.

4.3 Case Study

In this section, four cases are employed to validate the effectiveness and applicability of the proposed algorithm. At the beginning, two sub-cases in the first case study are used to test the algorithm in a modified IEEE-14 bus system. First sub-case considers a constant power mismatch as 20MW, and the results obtained using the proposed algorithm are compared with the results based on previous work [39]. In the second sub-case, the optimisation algorithm is studied under different time-of-use prices. Next, Case 2 is studied to test the proposed algorithm under communication failures, which assumes that all communication links of the DG_4 fail to exchange information with neighbours. The plug-and-play adaptability of our algorithm has been tested in Case 3. Lastly, a large-scale power system is adopted to test the scalability of the proposed algorithm in Case 4. The parameters are chosen from [93]. Results of the four cases are discussed in the following section.

4.3.1 Case 1: Base Case

In this study, we test the algorithm on an IEEE 14-bus system which consists of 4 DG, 2 BESS and 10 loads in Fig. 4.3, whose coefficients are shown in Table 4.1.

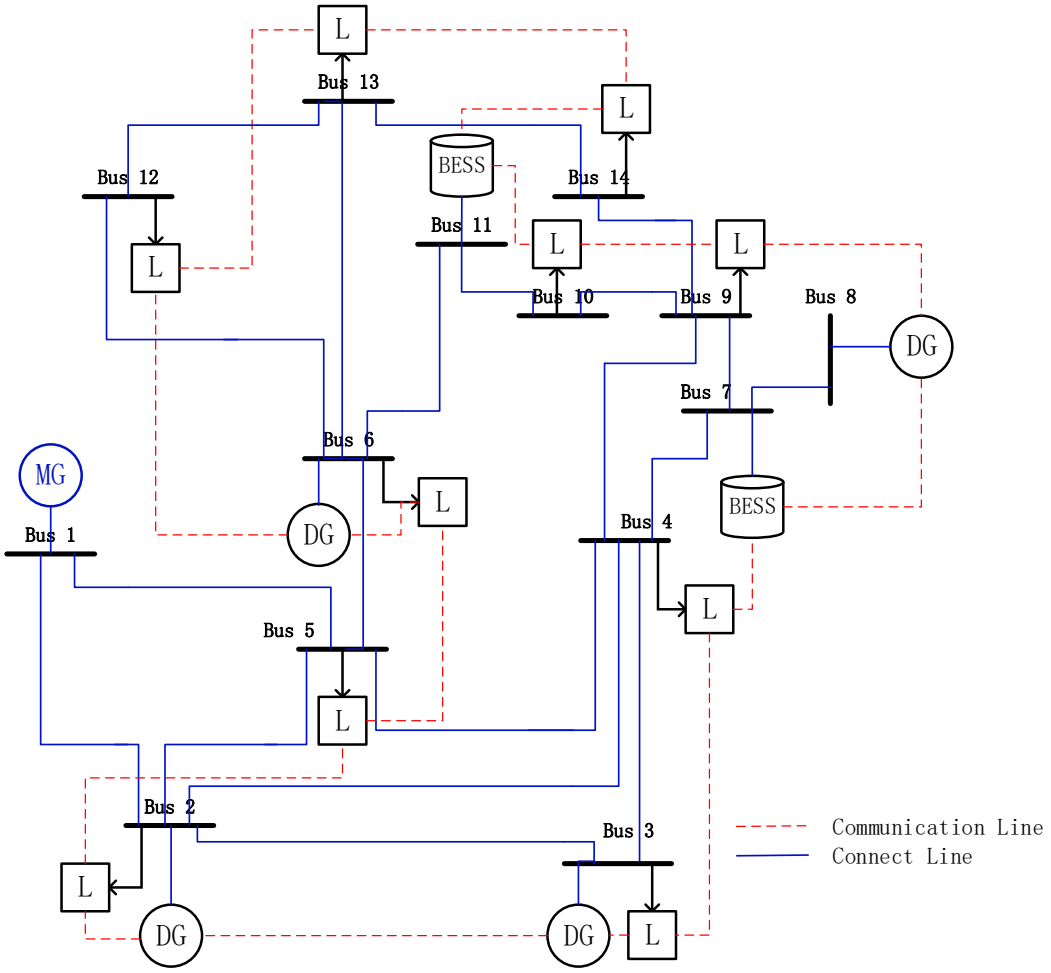

Figure 4.3: Modified IEEE 14-bus system.

Constant Demand and Price

As a baseline test, we first consider the situation with a constant power mismatch and electricity price, where all the power units are connected. To reveal the effectiveness of the proposed strategy, our algorithm is first compared with another algorithm in [39]. Here, the electrical price is set to 40 £/MWh, which is chosen from the UK electric price report [94], and the BESSs are distributed among the communication network. The electricity price and the supply-demand mismatch are assumed to be constant, and the system parameters are set to be the same to make the comparison study more convincible.

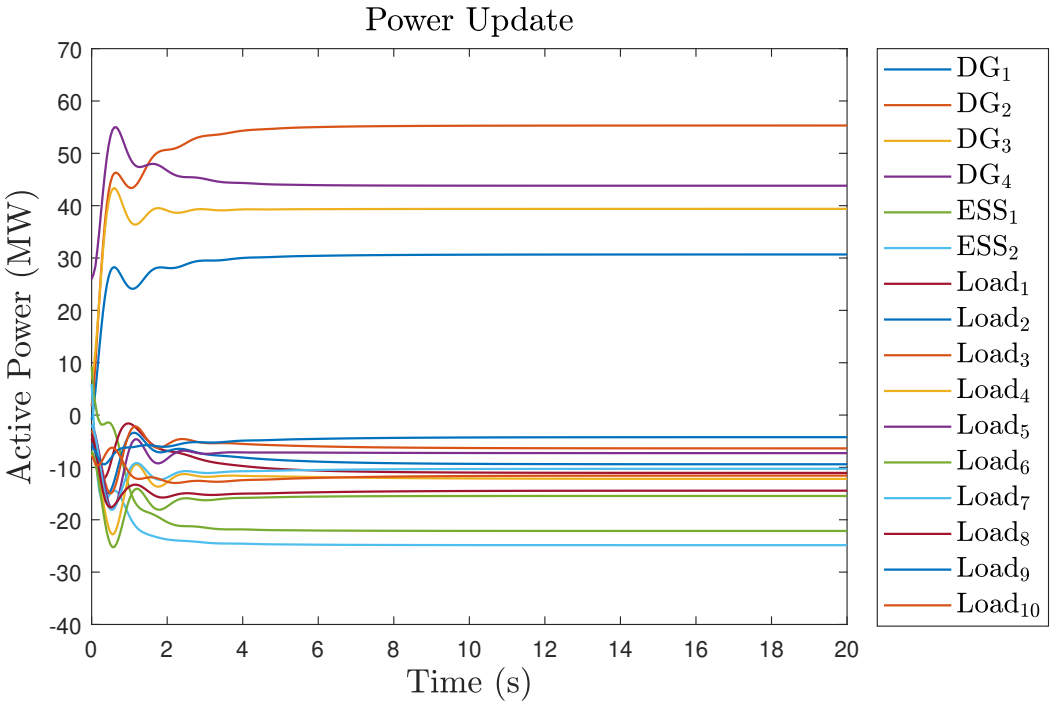

Figure 4.4: The update of power outputs with the proposed algorithm.

As shown in Fig 4.4, the power outputs of each power unit can converge to its optimal value with the proposed algorithm. Here, the negative/positive power value means the

consumed/generated power. The BESS can be charged or discharged depending on the electricity price and the power mismatch. Fig. 4.5 shows the power mismatch during the optimisation process. It indicates that the power is not balanced at the beginning and the power mismatch converges to zero within 5 s, which reflects that our algorithm is capable of maintaining the power balance. Thus, the proposed strategy can address the DSM problem from any initial error.

The proposed algorithm is designed in a fully distributed fashion, so that the local operator can only receive the information from its neighbours. We can see that there are some undershoot at the beginning. However, the proposed algorithm is based on the project function, which can guarantee the operation is limited in its physical reachable domain. Thus, the proposed algorithm will not influence the local constraints. In terms of the power mismatch, the proposed algorithm would asymptotically control the local power units to reach their optimal location after communicated with its neighbours.

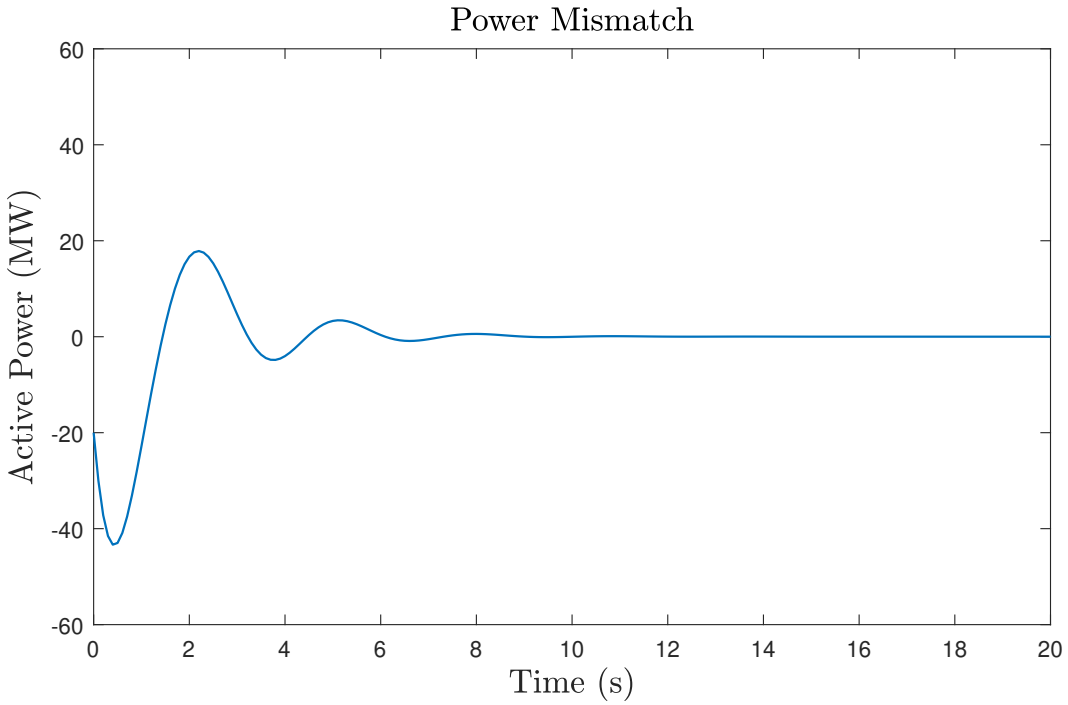

Figure 4.5: The power mismatch with the proposed algorithm.

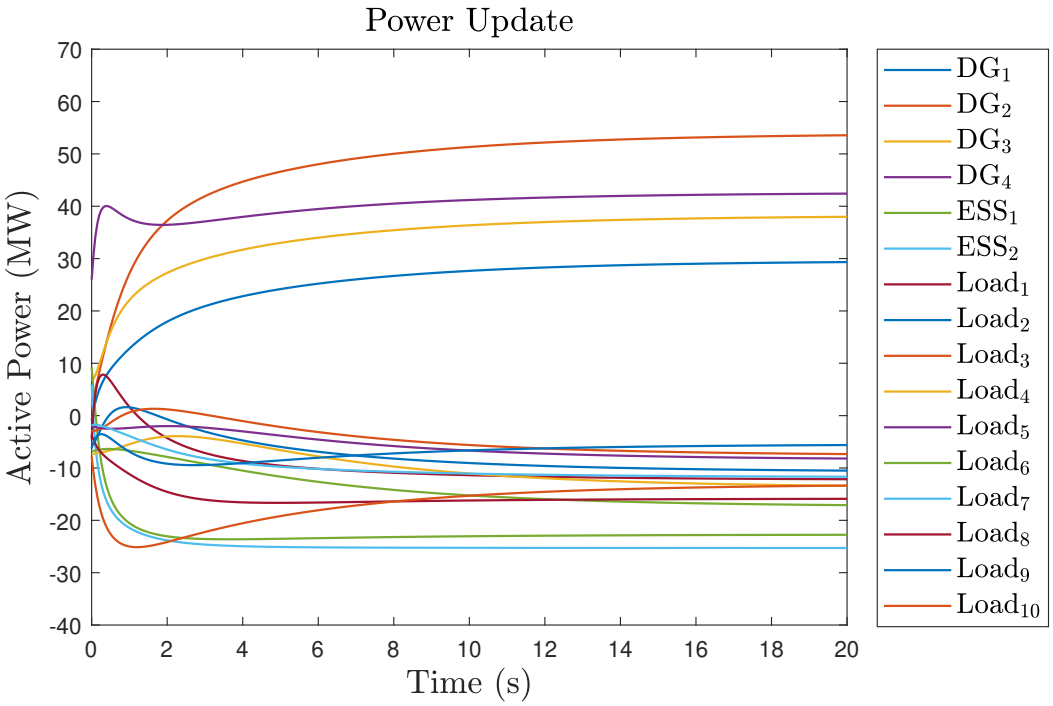

Figure 4.6: The update of power outputs with the algorithm in [36].

In order to show the advantage of initialisation-free feature, we firstly compare the proposed algorithm with an existing work in [39]. In the comparison study, we assumed both algorithms are initialised randomly and do not satisfy the initialisation conditions in [39]. The results are shown in Fig. 4.6. The power outputs using these algorithms can converge to stable values, but the results of the algorithm in [39] are not optimal since it needs the sum of all initial power output is equal to the power mismatch during the initialisation process. Then, we further compare the results with a discrete-time algorithm in [38]. To make the comparison more clearly, the ideal results are obtained through a centralised solver in Matlab. It shows that the proposed algorithm can reach the optimal active power outputs under initial errors compared with the algorithms in [38] and [39].

Remark 1. *Note that the proposed distributed algorithm generates the optimal power reference for each unit in a communication network. Its convergence speed will be*

related to the structure, namely the second smallest eigenvalue of the Laplacian matrix, of the communication network. Meanwhile, by tuning the algorithm parameter, the convergence speed can be adjusted according to different applications.

Different Prices

In the smart grid, the electricity price is changing over time, and different electricity prices bring different economic influence to customers, BESS and DG. Therefore, in the second sub-case, we test the algorithm with 24-hour Time-of-Use (ToU) prices shown as Fig. 4.7, which comes from the UK annual report [94].

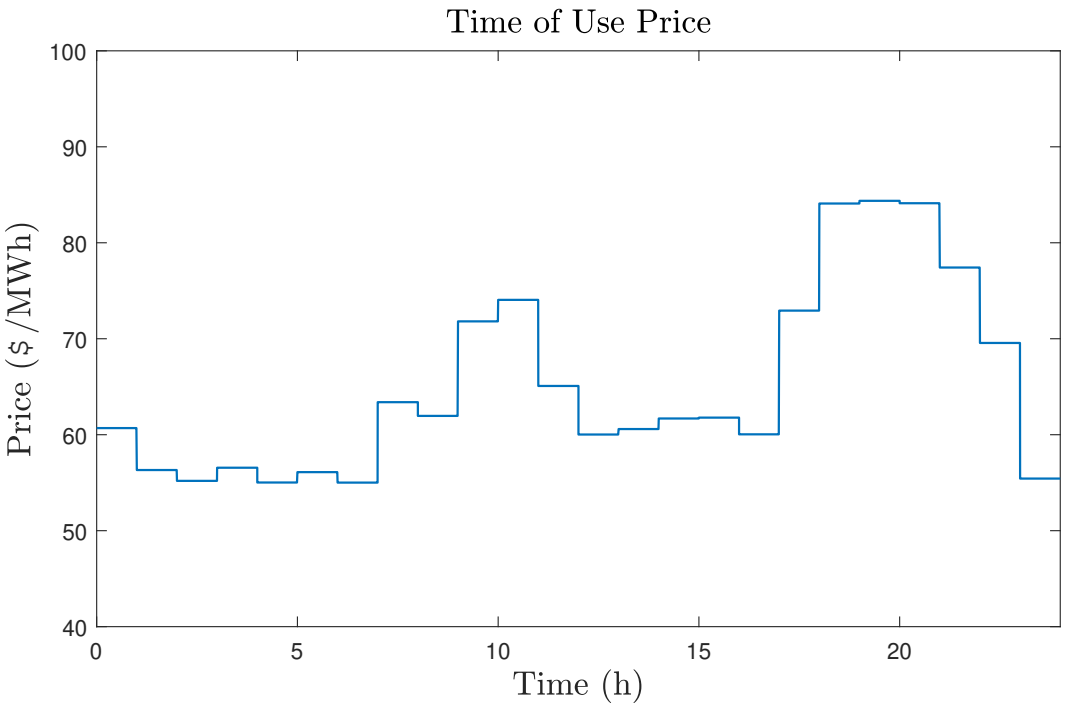

Figure 4.7: Time-of-use prices.

The results are presented in Fig. 4.8. Note that the power outputs of each power unit are

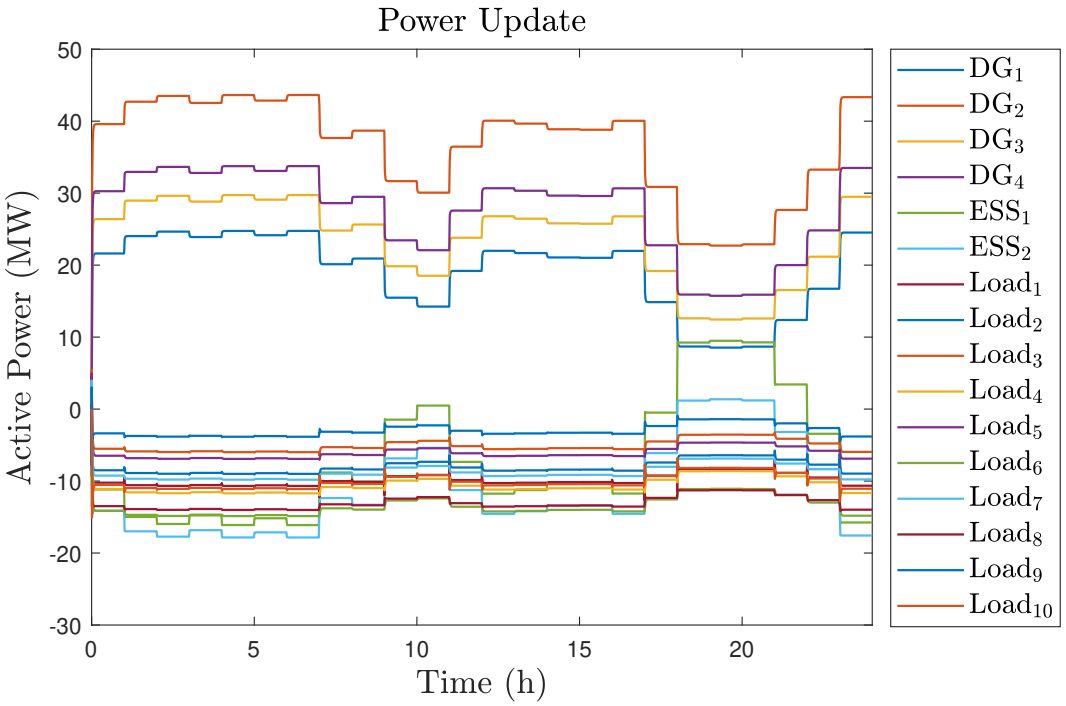

Figure 4.8: The update of power outputs with the proposed algorithm.

varying with ToU prices to maximise the social welfare. At low electricity prices periods, customers will consume more power and the battery will be charged to make sure it has enough energy to sell at high prices. When the electricity price rises, the algorithm will control the usage of these controllable loads, reduce their electricity consumption to save the electricity bills, such as reducing the load of air-conditioner and changing the EV charging time. At the same time, the battery will be discharged to increase the income. Therefore, the proposed algorithm could maximise the social welfare according to the different prices, which could be a potential solution for DSM in a future smart grid.

4.3.2 Case 2: Adaptability of Single-point Failure

In the real power grid, the communication network may change by many causes. For instance, a generator is disconnected from the smart grid for maintenance and overhaul. In this case, it is assumed that there is a power generator DG_4 which losses its all communication links at 50 s, and the links are repaired so that the DG_4 is reconnected at 150s. From (4.14), if the communication links are connected, then its neighbours have the information about the communication variable ϕ. It is assumed that the neighbour of DG_4 knows the last information ϕ when the communication links breaks. To better verify the effectiveness of the proposed algorithm under the communication failure, we further assume that the total active power mismatch P_D is increased from 20 MW to 40 MW at time t = 100 s.

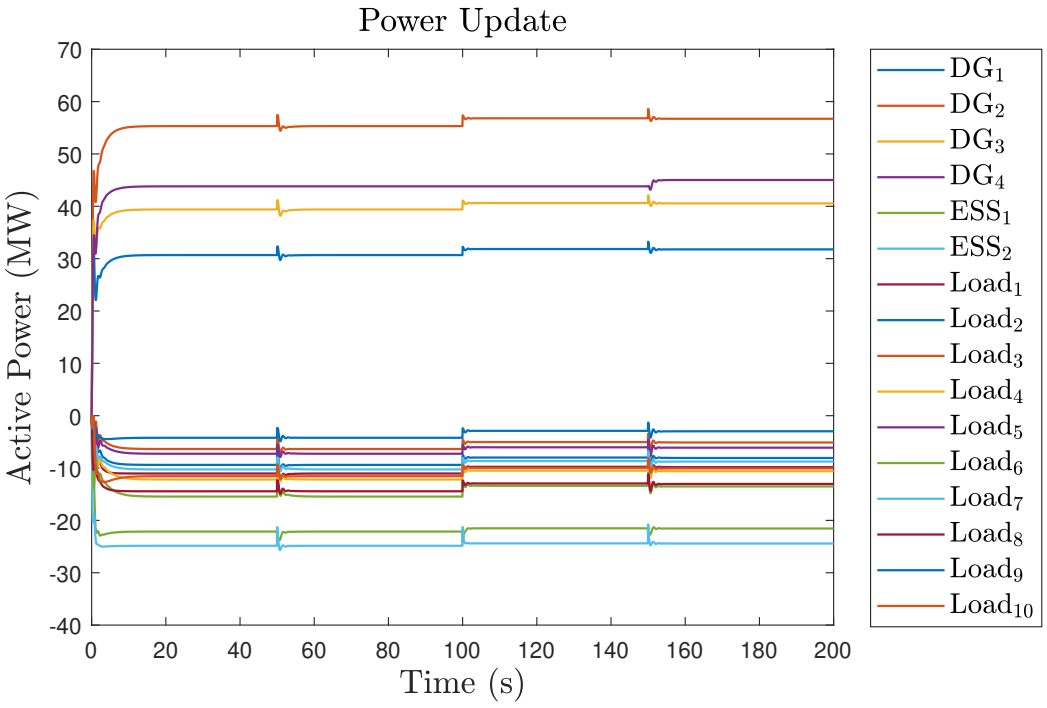

Figure 4.9: All communication links of DG_4 are failed.

It can be seen in Fig. 4.9 that the rest of power units still work at their optimal power

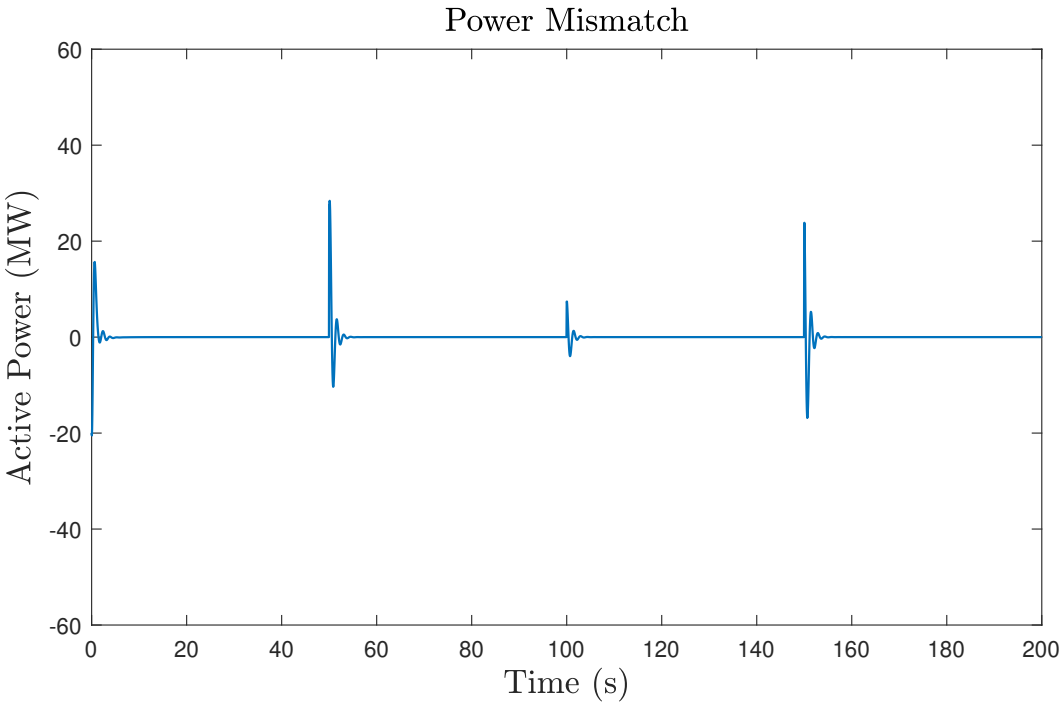

Figure 4.10: The power mismatch with failed communication network.

states to hold the power balance when the communication network of DG_4 is failed. If the power mismatch changes during the broken period, the DG_4 will not response to the power mismatch changing and cannot be optimised because it loses all communication links. However, the rest of power units will converge to new optimal values according to the current mismatch condition using the proposed algorithm. After the links are repaired, the DG_4 will be reconnected into the system so that all power units can share and update their information to reach the new optimal values. As shown in Fig. 4.10, the total power demand and total active power generation can be balanced after a very short period. Therefore, the proposed algorithm can keep the active power balance even if some communication links are failed.

4.3.3 Case 3: Plug-and-play Adaptability

To verify that the algorithm can adapt to the changes in the topology, the plug-and-play adaptability of our strategy is studied in this case.

Constant Power Mismatch

The plug-and-play adaptability of BESS under a constant power mismatch is studied in this sub-case. For example, the $BESS_2$ is disconnected at 50s for daily maintenance, and then after 50s, the BESS is connected back to the smart grid. The total power mismatch at the beginning is assumed as 20MW. The results show that the proposed algorithm can handle the plug-in and plug-out operations.

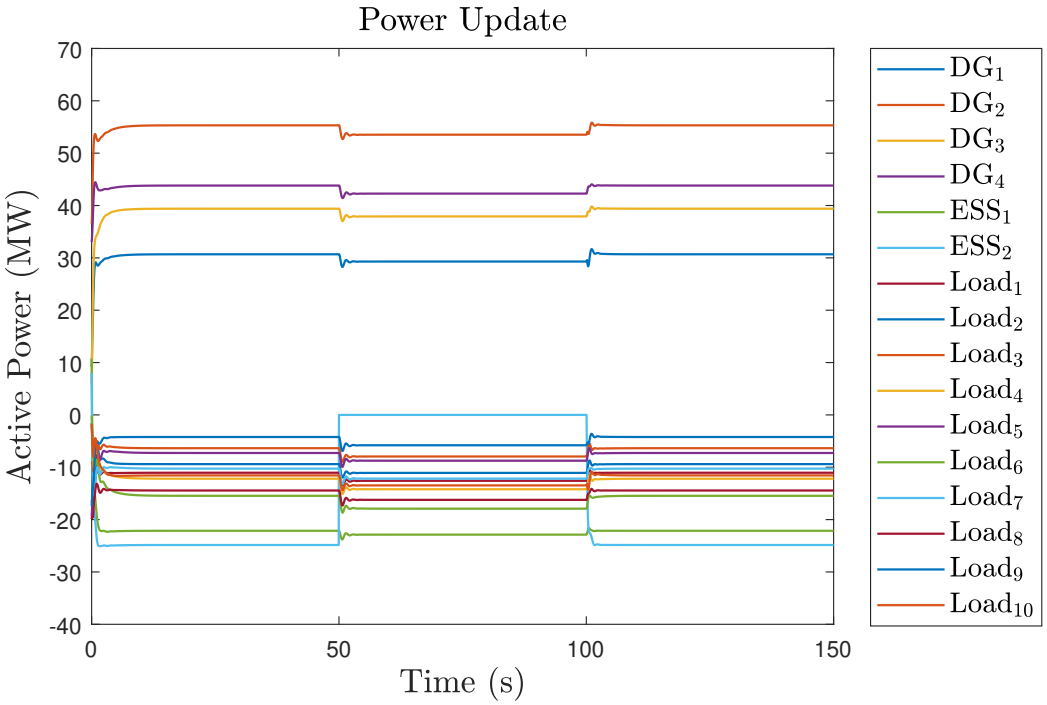

Figure 4.11: The update of power outputs during plug-and-play.

As shown in Figs. 4.11 - 4.12, the power outputs of all power units quickly converge to

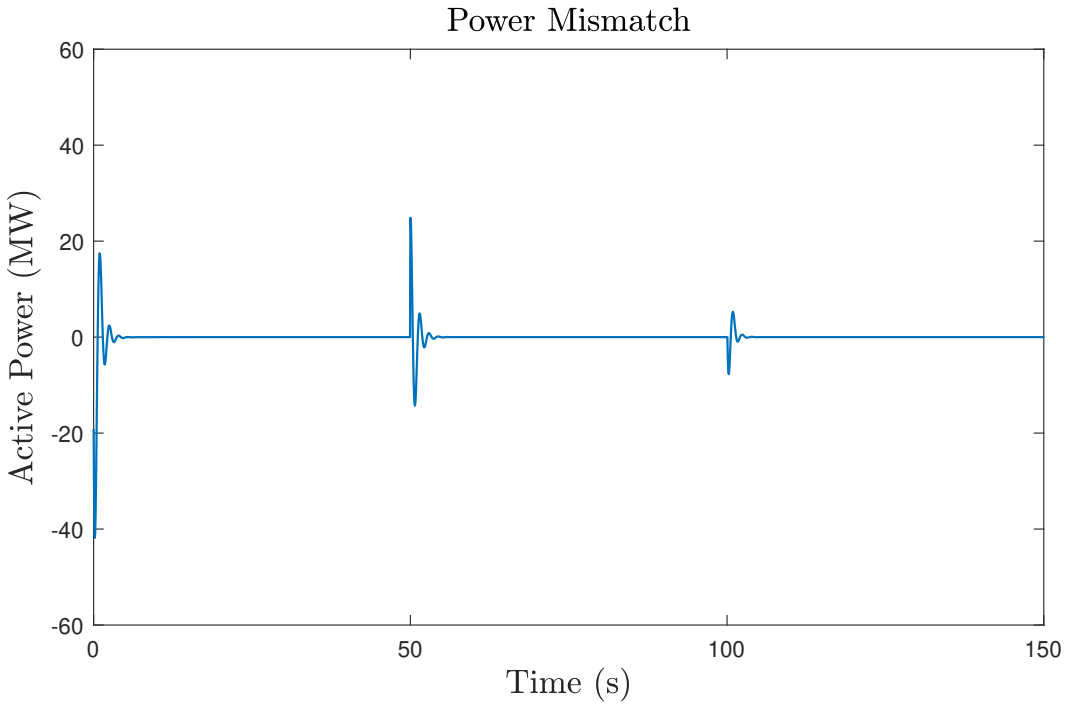

Figure 4.12: The power mismatch during plug-and-play.

their optimal values, and the total power mismatch converges to zero. When the BESS_2 is plugged out from the system at 50s, the rest of power units will fast converge to the new optimal values and keep the power balance at the same time. With plugging in a new battery at 100s, all power units converge back to the optimal values which are same as the previous optimal values.

Time-varying Power Mismatch

In fact, due to the intermittent of renewable energy sources and customer behaviours, the smart grid may face problems with uncertain power mismatch. To this end, we test the proposed strategy under a time-varying supply-demand condition. It is assumed that there is a time-varying uncontrollable power mismatch given by $P_{\text{total}} = 20 + 50\sin(0.3t)$ (MW). The communication network and the other parameters are same

as Case 1. To reflect the effectiveness of the proposed algorithm, we assume that a distributed generator is disconnected from the smart grid for daily maintenance at 50s and connected back to the smart grid at 100s.

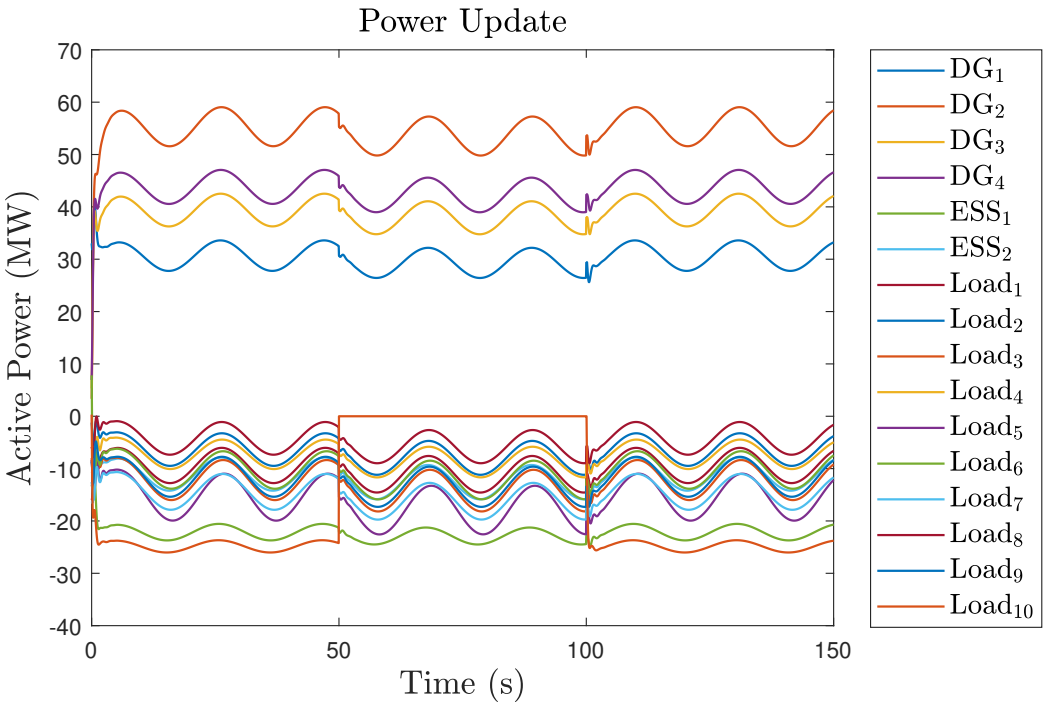

Figure 4.13: The power mismatch with time-varying demand.

As shown in Fig. 4.13, the optimal power references change with the time-varying power mismatch. However, the power outputs are allocated to their optimal value under the time-varying power mismatch condition, and while the algorithm keeps the power balance under the time-varying power mismatch. Besides, the proposed algorithm can keep working when plugging in/out the DG_4 at 50/100s, respectively. Therefore, the plug-and-play adaptability is also guaranteed by our algorithm even under the time-varying supply-demand condition.

4.3.4 Case 4: Scalability

In this case, to verify the scalability of the proposed algorithm, the algorithm is applied in a complex scale system, consisting of 40 DG, 15 BESS and 200 controllable loads, which is introduced in [93]. Each power unit is connected to its adjacent 10 neighbours in the communication network.

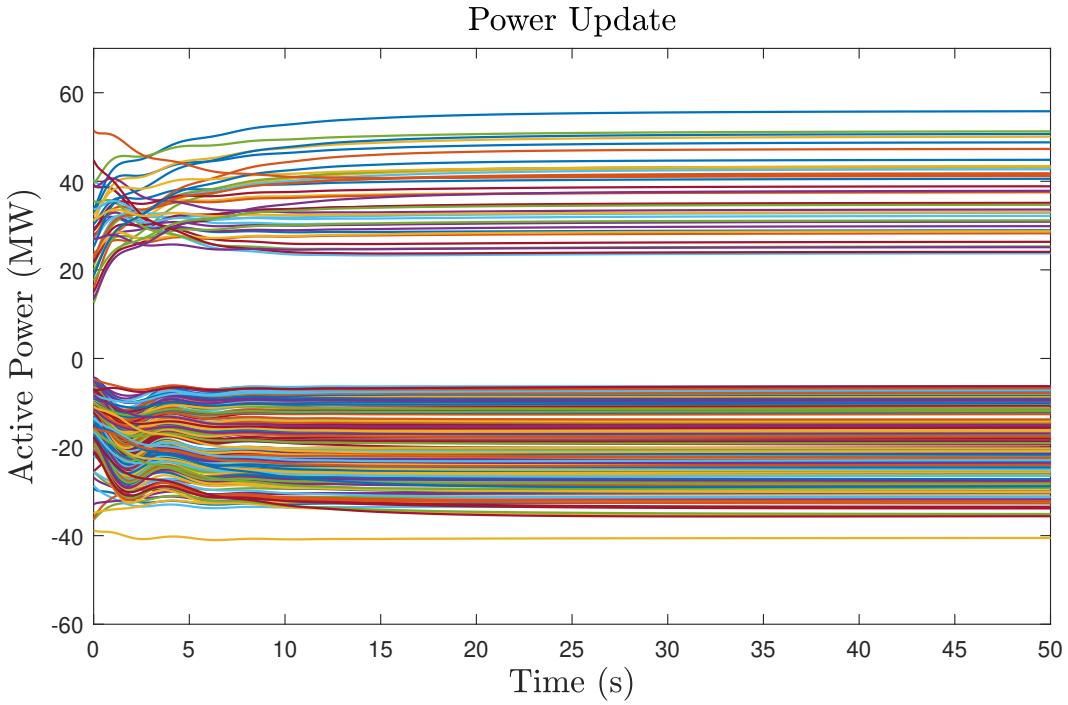

Figure 4.14: The update of power outputs with the proposed algorithm.

As shown in Fig. 4.14, the studied algorithm can guarantee the allocated power outputs to converge to their optimal values within 10s. Also, it shows that the fluctuation of our strategy is much less than the algorithm in [11]. Therefore, the proposed optimisation strategy can be applied in a large and complicated smart grid.

4.4 Summary

In this chapter, a distributed algorithm for DSM has been proposed in the context of smart grids. It maximises the social welfare of participants, i.e., customers, BESS and DG, according to their different objectives while subjecting to the system active power constraint and the local physical constraints. The proposed algorithm could solve DSM in a distributed fashion to handle the problems caused by the centralised method, such as communication interruptions, computing burdens, and single-point failures. Additionally, without the requirement of certain initialisation processes, the proposed algorithm can be executed in an initialisation-free manner that can deal with initial errors and enables the plug-and-play functions in DSM. The effectiveness and the scalability of the proposed algorithm are demonstrated by several case studies.

Table 4.1: Cost coefficients for simulation studies.

	a_i	b_i	α_i	β_i	η_i	φ_i	κ_f	κ_c	ι	$P^{\min}$(MW)	$P^{\max}$(MW)
DG_1	0.008	33.83	2.3e-3	-1.5e-3	2.0e-4	2.857	-	-	-	0	70
DG_2	0.0062	34.03	2.1e-3	-1.82e-3	5.0e-4	3.333	-	-	-	0	65
DG_3	0.0075	33.93	2.2e-3	-1.249e-3	1.0e-6	8.0	-	-	-	0	70
DG_4	0.0072	33.97	2.3e-3	-1.355e-3	2.0e-3	2.0	-	-	-	0	65
$BESS_1$	-	-	-	-	-	-	0.025	3.41	95%	-40	40
$BESS_2$	-	-	-	-	-	-	0.025	3.41	95%	-30	30
L_1	-	-	0.072	8.25	-	-	-	-	-	0	50
L_2	-	-	0.066	7.90	-	-	-	-	-	0	60
L_3	-	-	0.070	7.55	-	-	-	-	-	0	30
L_4	-	-	0.055	8.00	-	-	-	-	-	0	40
L_5	-	-	0.075	7.75	-	-	-	-	-	0	40
L_6	-	-	0.045	8.05	-	-	-	-	-	0	60
L_7	-	-	0.058	7.85	-	-	-	-	-	0	60
L_8	-	-	0.062	8.45	-	-	-	-	-	0	40
L_9	-	-	0.070	7.25	-	-	-	-	-	0	50
L_{10}	-	-	0.058	8.00	-	-	-	-	-	0	50

Table 4.2: Comparison results.

	Algorithm (4.14)	Algorithm in [39]	Algorithm in [38]	Matlab (fmincon)
DG_1	30.6819	22.9948	26.7769	30.6820
DG_2	55.3153	43.9412	50.2758	55.3155
DG_3	39.3940	30.2443	35.2270	39.3941
DG_4	43.8133	35.7782	39.4714	43.8134
$BESS_1$	-22.1363	-24.7803	-24.2210	-22.1362
$BESS_2$	-24.8522	-25.8211	-26.4163	-24.8522
L_1	-11.0477	-10.8696	-15.3938	-11.0477
L_2	-9.4004	-6.4048	-14.1400	-9.4006
L_3	-6.3632	-1.6057	-10.8304	-6.3634
L_4	-12.1895	-4.7591	-17.8732	-12.1898
L_5	-7.2724	-1.9037	-11.4392	-7.2725
L_6	-15.4540	-7.9295	-22.3970	-15.4542
L_7	-10.2661	-7.0456	-15.6520	-10.2661
L_8	-14.4426	-14.7112	-19.4805	-14.4425
L_9	-4.2206	-7.6875	-8.6827	-4.2205
L_{10}	-11.5595	-19.4404	-16.9449	-11.5593

Chapter 5

An Optimal Day-ahead Bidding Strategy and Operation for Battery Energy Storage System by Reinforcement Learning

The BESS plays an essential role in the smart grid, and the ancillary market offers a high revenue. It is important for BESS owners to maximise their profit by deciding how to balance between the different offers and bidding with the rivals. Therefore, this chapter first formulates the BESS bidding problem as a MDP to maximise the total profit from the AGC market and the energy market, considering the factors such as charging/discharging losses and the lifetime of the BESS. In the proposed algorithm, function approximation technology is introduced to handle the continuous massive bidding scales and avoid the dimension curse. As a model-free approach, the proposed algorithm can learn from the stochastic and dynamic environment of a power market, so

as to help the BESS owners to decide their bidding and operational schedules profitably. Several case studies illustrate the effectiveness and validity of the proposed algorithm.

5.1 Market Design

This section studies the bidding mechanism of battery energy storage system in different power markets. In this chapter, we assume that the BESS can offer more than one service in different markets. The BESS owner has to provide the day-ahead hourly bids to the system operator, including bidding capacities and bidding prices. The system operator determines the requirements of different services according to short-term load forecasting, renewable energy prediction and reliability constraints. On the basis of these, the Market Clearing Price (MCP) and offers of different markets are derived related to the different quotations and capacity bids. During the bidding process, the participants cannot know the bidding data of their rivals, but the MCP and offers from the system operator are public.

With the development of battery technology, the capacity of BESS is increasing rapidly. According to the importance of batteries in AGC market service, we assume that the BESSs have the market power to influence AGC market [41]. Since the main services and revenues of BESS come from the AGC market, according to [43], supplying sufficient power and energy capacity for the AGC market has the highest priority among all the services from the perspective of the system operator. In this chapter, based on the prediction of energy market and AGC market, the winning bids of BESS are determined considering the AGC market conditions.

5.1.1 Automatic Generation Control (AGC) Market

In the AGC market, the operation of smart grid must be subjected to keep the supply and demand balance. During the frequency control, the supply-demand balance of the whole network is met by adjusting the output of frequency modulation units, such as BESS, capacitive energy storage system, superconducting magnetic energy storage system, thermal energy storage system and Flywheel energy storage system [95]. In the frequency adjustment, there are various components involved, shown in Fig. 5.1.

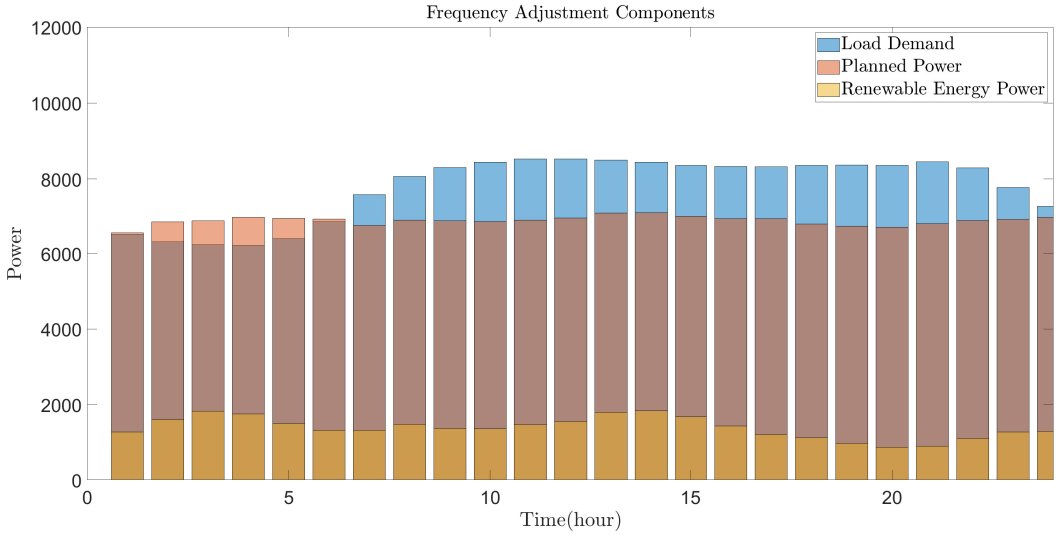

Figure 5.1: Frequency adjustment components.

Therefore, the AGC market should take the whole grid demand as the benchmark, and obtain the mismatched power by calculating the capacity demand caused by load change, the renewable energy output power, and the planned output power.

$$\Delta P_{td} = P_{load} - P_{energy} - P_{plan} \tag{5.1}$$

where ΔP_{td} is the power mismatch; P_{load}, P_{energy} and P_{plan} are the load demand, renewable power output and the planned power output, respectively.

In power grid dispatching, Area Control Error (ACE) are usually sent to AGC with a period of 2-4 seconds.

$$P_{ACE} = \Delta P_{td} + \beta_f \cdot \Delta f \tag{5.2}$$

where β_f is the coefficient of frequency deviation, Δf is the frequency deviation and P_{ACE} is the ACE signal.

From 2017, the conditional neutrality controller has been applied to control the regulation resources in PJM market [96]. It is a hybrid PID controller which includes a RegD integral feedback loop to ensure the energy of RegD is neutral. If system conditions allow, RegA will be utilised to balance the neutrality of RegD. For example, if the RegA resources are fully utilised to control ACE, then it will not be able to assist RegD. PJM area control error (ACE) signal is fed to high-pass/low-pass filters and a PID regulation controller to generate two regulation signals: a fast responding dynamic regulation signal D (RegD) and a slow responding traditional regulation signal A (RegA) [96, 97].

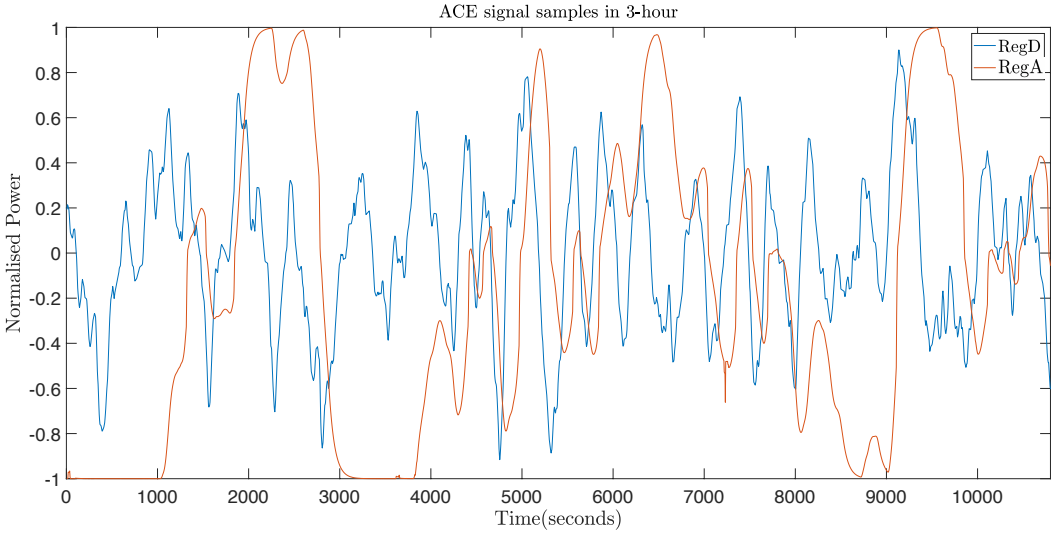

Figure 5.2: Real-time RegD and RegA data.

Frequency regulation mileage refers to the sum of the absolute changes in output power

within a period of time, and it is usually measured in megawatts (MW). As shown in Fig. 5.2, the RegA signal moves much slower than RegD signal. The mileage of RegD is about 7.17 times that of RegA in PJM market in 2019 [98]. Therefore, RegD offers more opportunity and higher performance compensation to exploit the potential of fast response energy storage systems. The RegD signal changes every 2 to 4 seconds, and the response time of BESS is usually on the time scale of seconds or milliseconds. Nevertheless, PJM market requires the mean value of RegD signal to be zero, which is suitable for energy limited power units like BESS.

5.1.2 Energy Market

In our model, the revenue of energy market is mainly from the planned output power. Compared with traditional generating units, a BESS only supplies or consumes small portion of electricity, the BESSs are supposed to be the price-takers, who will not affect the energy price in the energy market. The BESS will submit the day-ahead bids to the energy market system operator, and then the system operator will allocate the electric energy according to different requirements. Since BESS has the characteristics of low cost, good power quality and fast response, we assume that the battery will win the bids in the energy market. Therefore, the revenue in the energy market can be described as:

$$R_{e,t} = p_t^e \cdot b_{e,t} \tag{5.3}$$

where p_t^e is the electricity price in energy market, $b_{e,t}$ is the energy bidding quantity of the BESS and $R_{e,t}$ is the revenue of BESS in energy market at time slot t.

5.1.3 Model of BESS

The BESS unit should provide AGC services frequently in long term running. Therefore, two types of BESS costs are considered in this chapter, i.e., charging/discharging loss cost and the BESS ageing cost.

Loss Cost of BESS

According to [99], charging efficiency and discharge efficiency are different, and the charging/discharging efficiency can be formulated as η_c and η_d, respectively. We assume that the energy price is p_t^e. The charging/discharging losses then represented as

$$C_{chaloss} = p_t^e \cdot P_{charge}(1 - \eta_c) \cdot \Delta T \tag{5.4}$$

$$C_{disloss} = p_t^e \cdot P_{discha}(\frac{1}{\eta_d} - 1) \cdot \Delta T \tag{5.5}$$

where ΔT is the control period of regulation service and it is set as 4 seconds.

Ageing Cost of BESS

Ageing cost is an important expenditure when BESSs provide the power system service, and the BESS may not meet the requirements of the system after excessive ageing. Therefore, the ageing cost model needs to be considered when calculating the revenue of the BESS. Based on [100, 101], the maximum energy capacity of BESS will be reduced by the increase of charging/discharging cycles. In addition, the smaller the depth of discharge is, the more cycles there will be [102]. For a given depth of discharge of a lead-acid battery, the number of cycles before failure can be seen in Fig. 3 below.

For different types of battery, N_d^{fail} is a function of DoD(%), which can be calculated

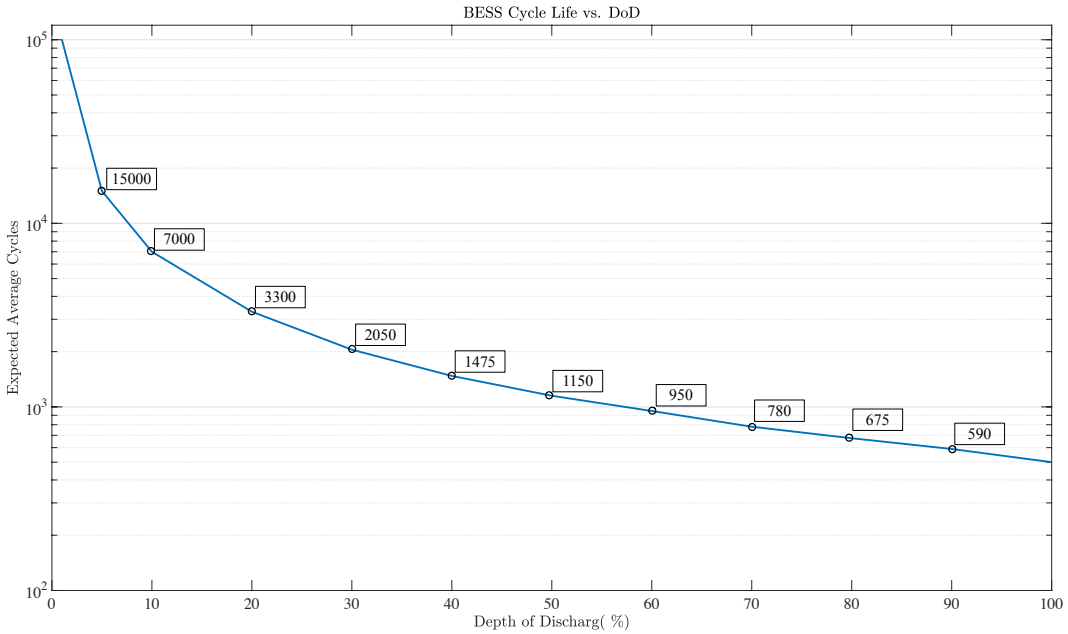

Figure 5.3: The relationship between DoD and cycle life of a BESS.

as

$$N_d^{fail} = N_{100}^{fail} \cdot d^{(-k_P)} \tag{5.6}$$

where N_d^{fail} is the maximum number of charging/discharging cycles at d DoD, d is the depth of discharge (DoD), and k_P is a constant parameter for different types of batteries ranging from 1.1 to 2.2 [103]. In the reinforcement learning algorithm, the time interval between s_t and s_{t+1} is one hour. Depth of Discharge (DoD) is the fraction or percentage of the capacity which has been removed from the BESS, which can only be calculated after a charging or discharging event. However, the time interval of the regulation market publishing regulation signal is 2-4 seconds, which means that there will be one charging or discharging event in each 2-4 seconds interval. The ageing cost between s_t and s_{t+1} should be the sum of every charging/discharging event's ageing cost. The DoD of BESS is changed in each half cycle, so that d will response to each RegD signal in each time period $(\tau - \Delta T) \rightarrow \tau$. Meanwhile, the BESS can only accept one charging mission or one discharging mission at a time. Thus, we can formulate the

ageing cost for one event, named half cycle, as $C_{ag,\tau}^{half}$, which can be calculated as:

$$C_{ag}^{half} = \frac{(d_\tau)^{k_P}}{2 \cdot N_{100}^{fail}} \cdot C_{inv} \tag{5.7}$$

where C_{inv} is the average daily investment cost of the battery energy storage system, which can be calculated by

$$C_{inv} = (1 + \xi) \cdot (C_P \cdot P_{max} + C_E \cdot E_{max} + C_F) \tag{5.8}$$

where ξ is the component replacement cost. $C_E \cdot E_{max}$ is the cost of the storage unit, where E_{max} is the energy capacity of the BESS. $C_P \cdot P_{max}$ is the costs of power conversion system, P_{max} is the power capacity of BESS. C_P, C_E and C_F are the unit costs of power conversion system, energy storage and facility infrastructure costs, respectively.

In the bidding market, the BESS company cannot predict the positive and negative power command signals to be given by the system operator, so that the BESS owner will provide one bid for charging and one bid for discharging at each time slot [50]. Therefore, we could get the equivalent one cycle ageing cost as:

$$C_{ag,\tau} = \frac{|(d_{\tau+1})^{k_P} - (d_\tau)^{k_P}|}{2 \cdot N_{100}^{fail}} \cdot C_{inv} \tag{5.9}$$

$$d_{\tau+1} = d_\tau + \frac{P \cdot \Delta T}{E} \tag{5.10}$$

Combining (5.6)-(5.10), we can obtain the equivalent ageing cost for one hour as:

$$C_{ag} = \sum_{\tau=1}^{3600/\Delta T} \frac{|(d_{\tau+1})^{k_P} - (d_\tau)^{k_P}|}{2 \cdot N_{100}^{fail}} \cdot C_{inv} \tag{5.11}$$

where ΔT is the time step of frequency regulation signal, and there are $3600/\Delta T$ charging/discharging cycles within one hour.

5.2 Model Formulation

The proposed model of BESS bidding in the pool based electricity market is described in detail. The decision variables are the capacity bids in energy market $b_{e,t}$, the capacity bids in AGC market $b_{c,t}^{up}$ and $b_{c,t}^{down}$ and the price bids in AGC market $b_{p,t}$ of the BESS for each hour in the next day.

5.2.1 Objective Function

The bidding model is to maximise the total profit of a BESS owner, which is described as follows

$$\max \text{Profit} = \sum_{t \in T} (\text{Profit}_t^{\text{e}} + \text{Profit}_t^{\text{reg}} - \text{Cost}_t^{\text{total}}) \tag{5.12}$$

where $\text{Profit}_t^{\text{e}}$ and $\text{Profit}_t^{\text{reg}}$ are the hourly revenue from energy market and the regulation market, respectively. $\text{Cost}_t^{\text{total}}$ is the hourly cost, which includes operation and maintenance cost, charging/discharging cost and the ageing cost. t is the hour index and Profit_t is the 24-hour total profit.

In the electricity market, there is a system operator between the supply companies and the retailers. The suppliers are bidding in the power pools, and the system operator makes the decision of market price and power generation offers. Since the BESSs are the price-takers in the energy market, the total revenue of a BESS in energy market $\text{Profit}_t^{\text{e}}$ can be calculated by [104, 105].

$$\text{Profit}_t^{\text{e}} = p_t^e \cdot b_{e,t} \cdot h_e \tag{5.13}$$

$$P_{e,t} = \begin{cases} b_{e,t} \cdot \dfrac{1}{\eta_d}, & \text{if} \quad b_{e,t} > 0 \\ b_{e,t} \cdot \eta_c, & \text{if} \quad b_{e,t} < 0 \end{cases} \tag{5.14}$$

where $b_{e,t}$ is the winning power offer of the BESS at time slot t, termed as the capacity bidding quantity. h_e is the normally operation period in energy market, typically 1 hour or 15 mins. $P_{e,t}$ is the charged and discharged power in the BESS. Note that $b_{e,t}$ can be positive or negative, which is related to charging and discharging requirement. A power supplier can only generate power if its offers are accepted. Otherwise, the extra penalties should be paid. The subscript "t" is the index of the hours in each day, since the bidding strategy is day-ahead with hourly bids in the wholesale electricity market.

In (5.12), $\text{Profit}_t^{\text{reg}}$ is the revenue of the regulation markets, which can be described as

$$\text{Profit}_t^{\text{reg}} = \text{Profit}_t^{cap} + \text{Profit}_t^{perf} \tag{5.15}$$

where Profit_t^{cap} is the revenue of the regulation capability, which can be described as

$$\text{Profit}_t^{cap} = (P_{cap,t}^{up} + P_{cap,t}^{down}) \cdot p_t^{cap} \cdot h_{reg} \tag{5.16}$$

where $b_{c,t}^{up}$ and $b_{c,t}^{down}$ are the capacity bids; p_t^{cap} is the Regulation Market Capacity Clearing Price (RMCCP), which is influenced by the bidding prices. h_{reg} is the normally operation period in regulation market, and it is typically 1 hour and 15 mins. Different energy storage systems provide different regulation capacity bids. Then the system operator will make the decision and send the regulation signal to the frequency modulation units. If the regulation bid of the BESS is accepted by the system operator, the regulation capability compensation and the regulation performance based profit can be formulated as

$$\text{Profit}_t^{perf} = (P_{cap,t}^{up} + P_{cap,t}^{down}) \cdot p_t^{perf} \cdot s_c \cdot \Delta T \tag{5.17}$$

where p_t^{perf} is the Regulation Market Performance Clearing Price (RMPCP). s_c is the performance score, which related to the accuracy, delay and precision [45]. ΔT is the regulation period, typically from 2s - 4s. According to the report [45], the performance revenue is not related to the bidding capacity of the BESS, but the real-time regulation signal and the clearing price. Since each time slot, the regulation signal will only

have one sign, we separate the regulation signal into regulation up signal $P^{up}_{cap,\tau}$ and the regulation down signal $P^{down}_{cap,\tau}$, where τ is the time index of regulation step.

The total cost is calculated in (5.18).

$$\text{Cost}^{\text{total}}_t = C_{O\&M,t} + C_{loss,t} + C_{ag,t} \tag{5.18}$$

where $C_{O\&M,t}, C_{loss,t}$ and $C_{ag,t}$ are the operation and maintenance cost, charging/discharging cost and the ageing cost, respectively. The operation and maintenance cost of BESS is usually a variable cost proportional to the size of BESS, which can be calculated as $C_{O\&M,t} = C_a \times E_{max}$, where C_a is the annual maintenance cost of BESS [106].

The charging/discharging cost is the sum of the charging part and the discharging part. In this model, the charging power P_{charge} is equal to the regulation down signal $P^{down}_{cap,\tau}$ and
$P_{discha} = P^{up}_{cap,\tau}$. Therefore, the charging/discharging cost for each hour is formulated as

$$C_{loss,t} = \sum_{\tau=1}^{3600/\Delta T} p^e_t \cdot (P^{down}_{cap,\tau}(1-\eta_c) + P^{up}_{cap,\tau}(\frac{1}{\eta_d} - 1)) \cdot \Delta T \tag{5.19}$$

The last part of total cost is the ageing cost, which can be calculated by (18).

5.2.2 Constraints

Power Constraints

In this part, the capacity limits of the BESS are considered and formulated in (5.20)-(5.22) regarding market requirements, physical constraints and regulation constraints.

The sum of the BESS bids must be kept within the maximum power of the BESS.

$$P_{e,t} + P_{c,t}^{up} \leq P_{max} \tag{5.20}$$

$$P_{e,t} - P_{c,t}^{down} \geq -P_{max} \tag{5.21}$$

where P_{max} is the maximum output power of the BESS. It is related to the type of the BESS.

Furthermore, the maximum regulation capacity has to be limited in a reasonable range, described in (5.22).

$$0 \leq P_{c,t}^{up}, P_{c,t}^{down} \leq \rho^{reg} \cdot P_{max} \tag{5.22}$$

where ρ^{reg} is the maximum ratio of regulation capacity to the high sustained limit.

To meet the transmission constraints in power system, the BESS is required to hold enough energy to response the system operator for dispatch or reserves [107]. Therefore, we consider that the BESS must maintain the output power level for at least h_e for energy market and h_{reg} for regulation market [51].

$$E_{max} \cdot soc_t \geq (b_{e,t} \cdot h_e + b_{c,t}^{up} \cdot h_{reg})/\eta_d \tag{5.23}$$

$$E_{max} \cdot soc_t \leq E_{max} + (b_{e,t} \cdot h_e - b_{c,t}^{down} \cdot h_{reg})\eta_c \tag{5.24}$$

Charging/Discharging Constraints

This part models the energy balance model of the BESS based on the physical constraints and the market requirement.

We assume that there is no energy loss during the charging/discharging process. The SoC of the BESS can be calculated as:

$$soc_t = soc_{t-1} + \Delta soc_t \tag{5.25}$$

where Δsoc_t indicates the amount of energy change between time $t-1$ and t, which is usually expressed in percentage (%). According to the energy selling and buying, the value of Δsoc_t can be positive and negative. For different types of BESS, the charging efficiency are different. Therefore, the charging/ discharging rate of the BESS (Δsoc_t) is expressed as

$$\Delta soc_t = (\Delta E_t^e + \Delta E_t^c)/E_{max} \tag{5.26}$$

$$\Delta E_t^e = P_{e,t} \cdot h_e \tag{5.27}$$

$$\Delta E_t^c = (\eta_c \cdot P_{cap,t}^{down} - \frac{1}{\eta_d} \cdot P_{cap,t}^{up}) \cdot h_{reg} \tag{5.28}$$

where $\Delta E_t^e, \Delta E_t^c$ represent the amount of energy change in energy market and frequency regulation market, respectively. Note that the energy loss formulated here will not influence the reward calculation. soc_t is used to calculate the next state of the reinforcement learning algorithm, which is the actual state of BESS.

The BESS must keep its SoC within its energy capacity limits. According to [108], the BESS performs its best working characteristics between 20% - 80%. To get the best performance of the BESS, in this chapter, the capacity limits is set as

$$\rho_{min} \cdot E_m \leq soc_t \cdot E_{max} \leq \rho_{max} \cdot E_m \forall t \in T \tag{5.29}$$

where ρ_{min} and ρ_{min} are the minimum and maximum efficiency operation rate. E_m is the rated energy capacity of the battery storage.

SoC Constraints

The initial and final SoC usually are set to be same during the optimization period, as described below. t_0 and t_{24} represent the begin and end of the day.

$$soc_{t_0} = soc_{t_{24}} \tag{5.30}$$

5.3 Algorithm Design

5.3.1 Function Approximation based Reinforcement Learning

Reinforcement learning is a valid approach to solve the decision problem for an unknown and uncertain environment. In this chapter, we deploy reinforcement Q-learning to achieve the optimal bidding results. In the traditional Q learning, it needs to generate a complicated Q table, and the dimension of the Q table will fall into the dimension curse with the increase of actions and states. Function approximation is a valid method to solve the generalisation problem for the large dimension of state and action pairs in the reinforcement learning method. Therefore, to avoid such dimension curse in BESS bidding problem, we apply the function approximation method to approximate the Q value. Here, the linear approximator analysed in this chapter is:

$$Q(s_t, a_t) = \sum_{j=1}^{n} \phi_j(s_t, a_t)\theta_j = \phi_t^T(s_t, a_t)\theta_t \tag{5.40}$$

where θ_t is the approximation parameter vector with n elements, and $\phi_t(s_t, a_t)$ is the feature vector, which is given by

$$\phi_t(s_t, a_t) = \{\phi_1(s_t, a_t), \phi_2(s_t, a_t), \cdots, \phi_{N_f}(s_t, a_t)\} \tag{5.41}$$

where $\phi_j(s_t, a_t)$ are the basis functions, and N_f is the number of features.

According to the projected Bellman error J [109], the optimal updating law for the parameters θ^π can be obtained by evaluating the approximation performance.
$Q^\pi(s_t, a_t)$ is simplified to Q_θ in following equations. The mean-square project Bellman

error objective function can be formed as

$$J(\theta) = \|Q_\theta - \Pi T_\pi Q_\theta\|_D^2 \tag{5.42}$$

$$= (\Pi(T_\pi Q_\theta - Q_\theta))^T D(\Pi(T_\pi Q_\theta - Q_\theta)) \tag{5.43}$$

$$= (T_\pi Q_\theta - Q_\theta)^T \Pi^T D \Pi(T_\pi Q_\theta - Q_\theta) \tag{5.44}$$

where Π is a projection matrix which projects any action values to the linear space of approximate action values, and D is a diagonal matrix with $N \times N$ dimension, which is use to reflect the state-action pair frequency under current policy π. We have

$$\Pi = \Phi(\Phi^T D \Phi)^{-1} \Phi^T D \tag{5.45}$$

Further, we transfer the objective function to statistical expectation forms as

$$\begin{aligned} J(\theta) =& (T_\pi Q_\theta - Q_\theta)^T (\Phi(\Phi^T D \Phi)^{-1} \Phi^T D)^T D \\ & \Phi(\Phi^T D \Phi)^{-1} \Phi^T D (T_\pi Q_\theta - Q_\theta) \end{aligned} \tag{5.46}$$

$$= (\Phi^T D(T_\pi Q_\theta - Q_\theta))^T (\Phi^T D \Phi)^{-1} \Phi^T D (T_\pi Q_\theta - Q_\theta) \tag{5.47}$$

$$= \mathbb{E}[\delta\phi]^T \mathbb{E}[\phi\phi^T]^{-1} \mathbb{E}[\delta\phi] \tag{5.48}$$

where

$$\mathbb{E}[\delta\phi] = \sum_{s,a} D_{(s,a),(s,a)} \phi(s,a) \mathbb{E}[\delta_t] \tag{5.49}$$

$$= \Phi^T D(T_\pi Q_\theta - Q_\theta) \tag{5.50}$$

and

$$\mathbb{E}[\phi\phi^T] = \sum_{s,a} D_{(s,a),(s,a)} \phi(s,a) \phi^T(s,a) = \Phi^T D \Phi \tag{5.51}$$

Note that all statistical expectations are under current behaviour policy π. Also, δ is the temporal difference error, which is defined as

$$\delta_t = r_{t+1} + \gamma \hat{\phi}_t^T \theta_t - \phi_t^T \theta_t \tag{5.52}$$

In order to avoid the need for two independent samples, a modifiable parameter $w \in \mathcal{R}^n$, named quasi-stationary estimate, is introduced as follows

$$w \approx \mathbb{E}[\phi\phi^T]^{-1}\mathbb{E}[\delta\phi] \tag{5.53}$$

Then, the negative gradient of objective function can be calculated as

$$\begin{aligned} -\frac{1}{2}\nabla J_i =& \mathbb{E}[(\phi - \gamma\phi')\phi]^T\mathbb{E}[\phi\phi^T]^{-1}\mathbb{E}[\delta\phi] && (5.54)\\ =& (\mathbb{E}[\phi\phi^T] - \gamma\mathbb{E}[\phi'\phi^T])\mathbb{E}[\phi\phi^T]^{-1}\mathbb{E}[\delta\phi] && (5.55)\\ =& \mathbb{E}[\delta\phi] - \gamma\mathbb{E}[\phi'\phi^T]\mathbb{E}[\phi\phi^T]^{-1}\mathbb{E}[\delta\phi] && (5.56)\\ \approx& \mathbb{E}[\delta\phi] - \gamma\mathbb{E}[\phi'\phi^T]w && (5.57) \end{aligned}$$

Since the expectations in (5.54) are not know, it is generally using stochastic gradient-descent approach. To get the quicker convergence speed of the reinforcement learning algorithm, the correction term is applied to adjust the update law as follows

$$\theta_{t+1} = \theta_t + \alpha_t(\delta_t\phi_t - \gamma w_t^T\phi_t\hat{\phi}_t) \tag{5.58}$$

$$w_{t+1} = w_t + \beta_t(\delta_t - \phi_t^T w_t)\phi_t \tag{5.59}$$

where $\hat{\phi}_t$ is the approximation of $\max_{a'} Q^\pi(s_{t+1}, a_{t+1})$, which can be estimated as

$$\hat{\phi}_t \approx \arg\max_{\phi(s_{t+1},a_{t+1})} \phi^T(s_{t+1}, a_{t+1})\theta_t \tag{5.60}$$

5.3.2 Problem Reformulation

Aiming at the stochastic environment of power market, the optimal bidding problem in an stochastic environment is reformulated based on equation (3.6), which includes the state space $\mathcal{S}$, action space $\mathcal{A}$, transition probability function $\mathcal{P}$, reward function $\mathcal{R}$ and discount factor γ in detail.

At each time slot t, the BESS owner has its observation of the bidding market, namely state s_t. Considering the bidding quantity and bidding price of rivals are uncertain, the state of the BESS owner is set as:

$$s_t = (v_t^{-1}, a_{t-1}^T, SOC_t, t)^T \tag{5.42}$$

where $s_t \in \mathcal{S}$ presents the observable information and v_t^{-1} is the clearing price of the previous day at time slot t. a_{t-1} is the last decided bidding actions, including bidding quantities and bidding prices. In the wholesale electricity market, each BESS owner only knows its own bidding quantity and price. The bidding data of the other bidders must be estimated by the previous bidding history. In this chapter, the bidding quantities and prices of other rivals are presumed to be influenced by the market clearance price and the sold offer at time slot $t-1$. Some similar state-choosing methods are studied for electricity market in [105]. soc_t is the SoC of BESS at time slot t, which can be accurately estimated by battery management systems. Here, our objective is to maximise the BESS owner's profit within its bidding period, which is 24 hours of a day; therefore, time slot t is set as a part of state so that the decision maker can take different actions in different hours of the day-ahead bidding strategy.

The action $a_t \in \mathcal{A}$ consists of decision variables made by BESS owner. Since bidding environment are unpredictable, we only formulate the decision variables concerning the bids of own BESS units:

$$a_t = (b_{e,t}, b_{p,t}, b_{c,t}^{up}, b_{c,t}^{down})^T \tag{5.43}$$

where $(b_{e,t}, b_{p,t}, b_{c,t}^{up}, b_{c,t}^{down})$ are the capacity bids in energy market, price bids, up and down capacity bids in AGC market, respectively. Here, $\mathcal{A}$ is the discrete action set for all s_t. At each time slot t, the BESS owner will provide a bidding action a_t from action set $\mathcal{A}$.

In the reinforcement learning, there is a transition function $\mathcal{P}(s_t, a_t, s_{t+1}) : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow$

$[0, 1]$. It maps state s_t to s_{t+1} by action a_t, which means the dynamics of the environment. In the electric bidding market, the transition function is unknown and depends on some stochastic factors, such as real-time load mismatch and uncontrollable renewable power generation. Thus, our agent needs to learn it through different $\{s_t, a_t, s_{t+1}\}$ sets during the training process. After taking the action a_t, the state of BESS will automatically transfer to next state $s_{t+1} = (v_{t+1}^{-1}, a_t^T, soc_{t+1}, t+1)^T$ based on the transition function $\mathcal{P}$.

Specific to each state transition between adjacent time slots, the system operator will provide an offer to the BESS owner, which indicates the reward signal $r_t \in \mathcal{R}$. The algorithm can be trained by the reward information to select best policy to achieve maximum reward. In this chapter, the detailed reward definition is designed as follows:

$$r_t = \text{Profit}_t - C_t^W \tag{5.44}$$

where r_{t+1} is defined under the framework of the reinforcement learning. Generally, the reward at $t+1$ time slot is the BESS owner profit in terms of the state s_t and the action a_t at t time slot. C_t^W is set as a penalty term, which is related to the local constraints, including battery and generator constraints. For example, the SoC of a BESS should be kept between 20% to 80% to obtain the efficiency operation [110]. If the action leads to these inefficiency areas, the reward of this state action pair should be negative and get corresponding penalty. In this chapter, the BESS owner can get the finite-time horizon reward sequence as $\{s_t, a_t, r_t, s_{t+1}, \cdots, s_{t+N-1}, a_{t+N-1}, r_{t+N-1},$
$s_{t+N}\}$, which is an episode of bidding and operation. The parameter N is the trading period, which is set as 24 in this chapter.

The objective of the reinforcement learning for the BESS owner i is to obtain the best

24-hour reward given by

$$\mathbb{E}(\sum_{t=1}^{24}\gamma^t r_t|s_0) \tag{5.45}$$

where s_0 is the initial state; $r(t)$ is the reward based on state-action pair at time slot t; γ is the discount factor which is applied to reduce the effect of future reward. In this 24-hour bidding environment, $\gamma = 1$.

According to (3.8), a Q function can be defined as follows:

$$Q^{\pi}(s,a) = \mathbb{E}\left[\sum_{t=1}^{24}\gamma^t r(t)|s_t = s\right], \forall s \in \mathcal{S}, \forall a \in \mathcal{A} \tag{5.46}$$

Following the updating rules (3.10), $Q^{\pi}(s,a)$ will converge to the optimal Q value $Q^*(s,a)$. To ensure the proposed algorithm can find the optimal policy which covering maximum state values, we applied the $\mathcal{E}$-greedy policy to keep the exploration behaviour, so that all exploratory actions have probability to be chosen during the training period. The policy is settled by following equations:

$$\widetilde{a} = \arg\max_{a} Q^{\pi}(s,a), \forall s \in \mathcal{S}, \forall a \in \mathcal{A} \tag{5.47}$$

$$\pi(s,a) = \begin{cases} 1 - \mathcal{E} + \mathcal{E}/n_A, & \text{if } a = \widetilde{a} \\ \mathcal{E}/n_A, & \text{if } a \neq \widetilde{a} \end{cases} \tag{5.48}$$

where $\widetilde{a}$ is the greedy action which has the maximum Q value under current policy π, n_A is the number of possible actions in action set $\mathcal{A}$, and $0 \leq \mathcal{E} \leq 1$ is the probability of choosing any action in the action set $\mathcal{A}$. Therefore, we have a $(1 - \mathcal{E} + \mathcal{E}/n_A) \times 100\%$ chance of taking the greedy action $\widetilde{a}$, and a $(\mathcal{E}(n_A - 1)/n_A) \times 100\%$ chance of exploring new behaviours.

Since the states setting in the model is continuous and the dimension of the states is large, this chapter applies the function approximation to solve the reinforcement learning problem. An off-policy model-free algorithm is implemented so as to find the BESS bidding strategy, which helps the BESS to get a higher profit during the trading period.

5.3.3 Algorithm Implementation

Based on the equation 39, we randomly initialise the parameter θ_0 to calculate the the Q value of each state-action pair. To get quicker convergence speed of reinforcement learning algorithm, we apply a correction term w_0 to adjust the update law, which is designed in Section 5.3.1. Our algorithm is developed to update these parameters and get the optimal Q value Q^*.

In our algorithm, each hour is seen as a time step within N_t and each day is an episode within N_μ, which means the training process has $N_\mu \cdot N_t$ steps in total. At each episode, the algorithm will start from a random state s_0. Then, the agent chooses its action a_t according to policy π, then its state s_t will transfer to s_{t+1} and get the reward r_t. With all these obtained information and equations (5, 6, 65, 66), the algorithm can calculate and update the parameters θ_t, ω_t. After several explorations and training loops, our Q value Q^π is roughly equivalent to the optimal Q value Q^*. The details of the proposed algorithm for BESS optimal bidding are summarised in Algorithm I. It is started from a policy π, learning rate α, β and discount coefficient γ.

5.4 Case Study

In this section, consider an electricity market with 4 BESSs, and these four BESSs bid in the AGC market to get their rewards. The planning horizon is next day 24-hour bids. In each state, BESSs make their decision for next day bids and each bid has capacity bidding price and capacity bidding quantity. And during the bidding process, BESS1 does not know how other BESSs are going to bid. The BESS1 can get the history clearing price information of the electricity market.

Algorithm 2 Function Approximation based Reinforcement Learning Algorithm for Supply-side BESS Bidding

Input: Learning rate α, β, Policy π, Discount coefficient γ
Output: The bidding action $a_i(t)$ of BESS owner for next day's trading market
Initialisation: θ_0, ω_0
for every episode $\mu = 1$ **to** N_μ **do**
  Initialise s_0, choose a for state s_0 with the $\mathcal{E}$-greedy policy π
  for every time slot $t = 1$ **to** N_t **do**
    Calculate the feature vector ϕ_t of state s_0
    Take action a_t, obeying the policy π
    Get the reward r_t from the environment
    Calculate the TD error $\delta(t)$
    Estimate the next step feature vector $\hat{\phi}_t$
    Update the parameter θ_i, ω_i
    $t \leftarrow t + 1$
  end for
  $\mu \leftarrow \mu + 1$
end for
return 24-hour Action Sequence;

5.4.1 Datasets

In our simulation study, the real world datasets are applied to illustrate the effectiveness of our model. A 4-s based RegD & RegA signal is generated based on real RegD signal data by PJM's data set.

Table 5.1: Cost coefficients for simulation studies.

	μ	C_P $(£/kW)$	C_E $(£/kWh)$	C_F $(£)$	C_a $(£/kWh)$
BESS1	15%	2300	300	2.58e5	14.6
BESS2	15%	2250	450	2.52e5	15.8
BESS3	15%	2470	360	2.49e5	16.2
BESS4	15%	2320	280	2.63e5	15.4

5.4.2 Case Implementation

In the case studies, it is supposed that all of 4 BESSs can participate in the AGC market. In this market, BESS1 is assumed as our own BESS, which tends to maximise the profit of next 24-hour. The decision maker of the BESS1 will implement the function approximation based reinforcement learning algorithm, seeking the proper bidding price and bidding capacities. In the stochastic bidding market environment, the bidding strategies of all the other BESS and the bidding environment are unknown, which means that BESS1 only have its individual bidding data and MCP history data. Similar to [111], the rule of clearing price is simply the highest bid from all accepted regulation offers in this chapter. After all market participates submit their hourly bids to the SO, the SO needs to schedule the regulation offer and publish an MCP according to the real-time load demand, as shown in Fig. 5.4. Then, the BESS owners can calculate their rewards and costs based on the regulation offer and MCP. For the objective function shown in Eq. (5.12), the initial time slot is set as $t = 1$ and the end time slot is set as $t = 24$.

The round-trip efficiency of the BESS is derived to be 0.868 [50]. Since the function approximation algorithm is applied in the chapter, the state variables in Eq. (5.42) will not be aggregated into discrete levels. And the action variables in Eq. (5.43) are aggregated into discrete levels to get more accurate results. The action aggregate is achieved by follows.

The bidding prices are set in advance by the BESS and the AGC market, and the other three capacity bids are considered together, since when two of them are specific in a domain, the other one should be constrained in some specific values. For example, if the regulation up and down capacities are defined, then the capacity bids of energy market should be limited in a specific domain.

Based on the real-time price data from PJM. the bidding price is aggregated into 11

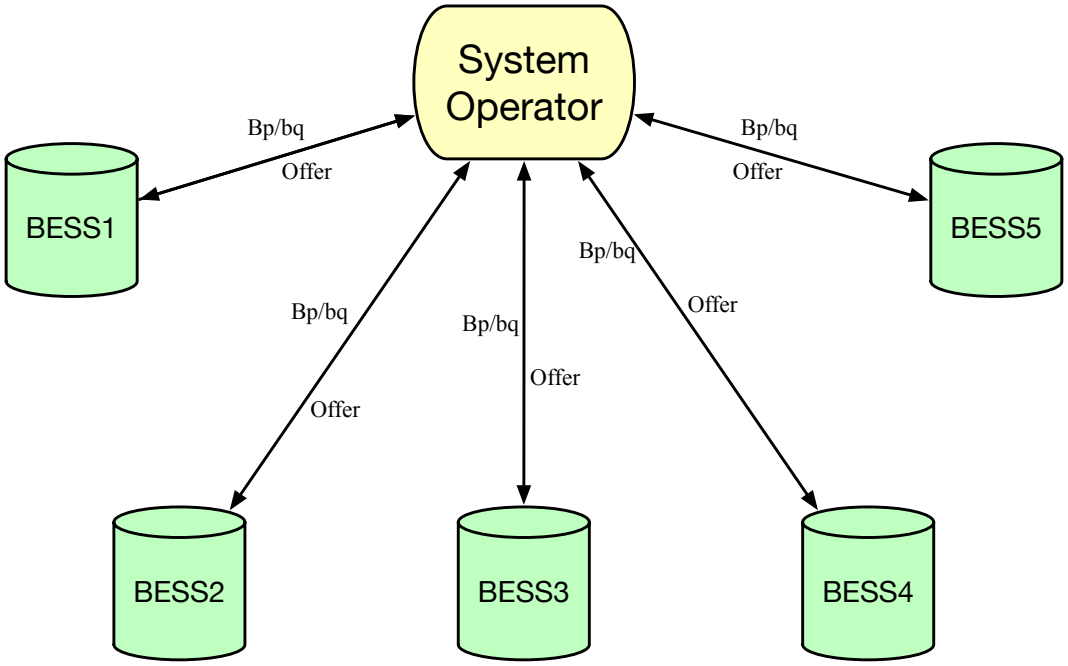

Figure 5.4: Framework of bidding environment.

levels, and the capacity bids are separated into different 11 levels. Thus the test model has 14,641 aggregated actions.

The relevant information and cost parameters of the BESSs are shown in Tabs. 1 - 2.

Table 5.2: Parameters of BESSs for simulation studies.

	P_{max}(MW)	E_{max}(MWh)	η_c	η_d	N_{100}^{fail}	k_p
BESS1	406	900	0.868	0.92	10,000	0.85
BESS2	207	1000	0.88	0.95	10,000	0.85
BESS3	250	625	0.86	0.88	10,000	0.85
BESS4	362	830	0.82	0.86	10,000	0.85

5.4.3 Training Performance

In this section, we analyse the convergence characteristic of the function approximation based reinforcement learning algorithm. Note that since the action variables have been aggregated into 14641 elements and the states have been approximated into 400 elements, we have 5856400 (14641$\times$ 400) state and action pairs. Here, the convergence curves of part of theta parameters are shown in Fig. 5.5.

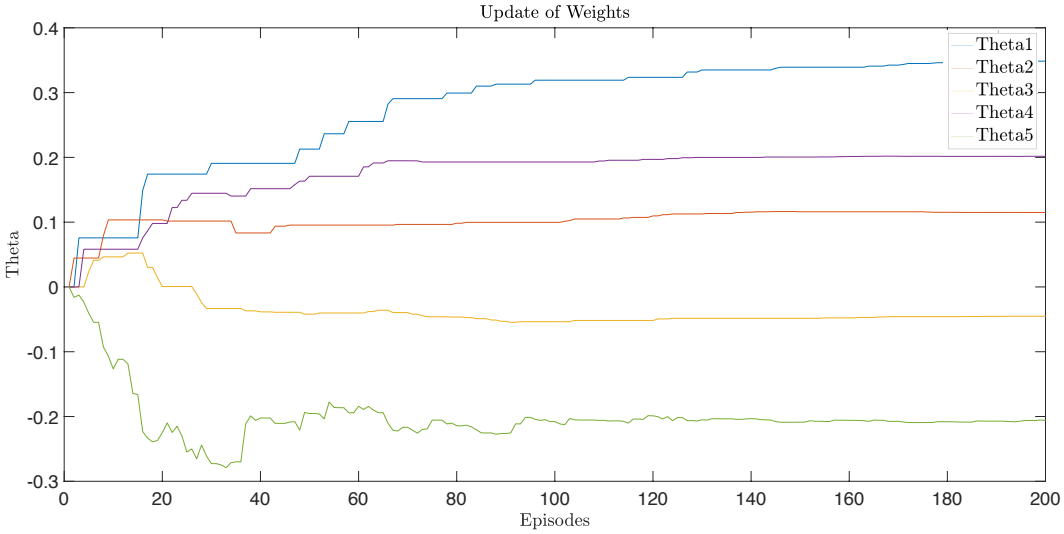

Figure 5.5: Update of theta elements.

Since the value of theta converges, the optimal Q-value of each state-action pair will automatic converge. Thus the optimal policy can be found by searching for the action with maximum action value for each state. After several training episodes, the profits line still has some fluctuations due to exploration behaviours of reinforcement learning algorithm. This exploration behaviour is to ensure the algorithm can reach the optimal results and find the solution for the bidding strategy as shown in Fig. 5.6.

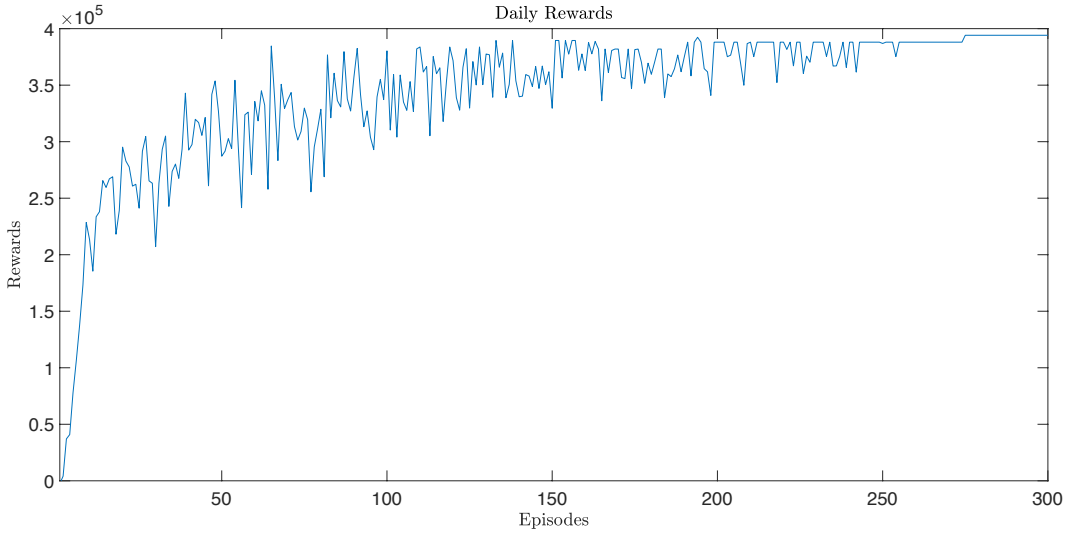

Figure 5.6: BESS daily rewards during the training period.

5.4.4 Results and Comparison

Figs. 5.7 - 5.8 show the optimal bidding strategies and bidding prices of the BESS in different time slots. In this case, regulation capacity dominates most of the day, since the compensation of the regulation services are high. Furthermore, we test bidding strategy of the BESS1 in regulation market and energy market with its rivals. To win the regulation services offer and earn high compensation profits, the bidding regulation price is trained to be less than the history clearing prices and the rivals' bids. When the regulation price are cheap, the BESS will not do much regulation mileages, so as to the BESS owner will purchase or sell the energy in the energy market to balance the energy loss and earn some revenue. During that period, the regulation bids are reduced because of the physical constraints of the charging/discharging rate.

To reveal the impact of the ageing losses and transmission losses, we magnify the loss coefficient ten times, and the simulation results are shown in Figs. 5.9 - 5.10. The impact of considering these losses on BESS can be observed by comparing Fig. 5.7

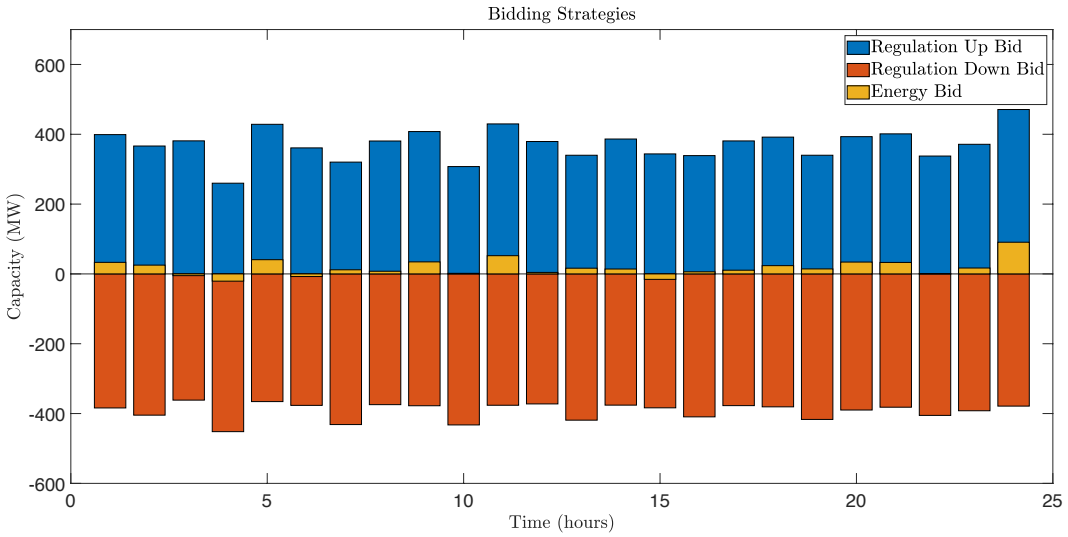

Figure 5.7: BESS bids.

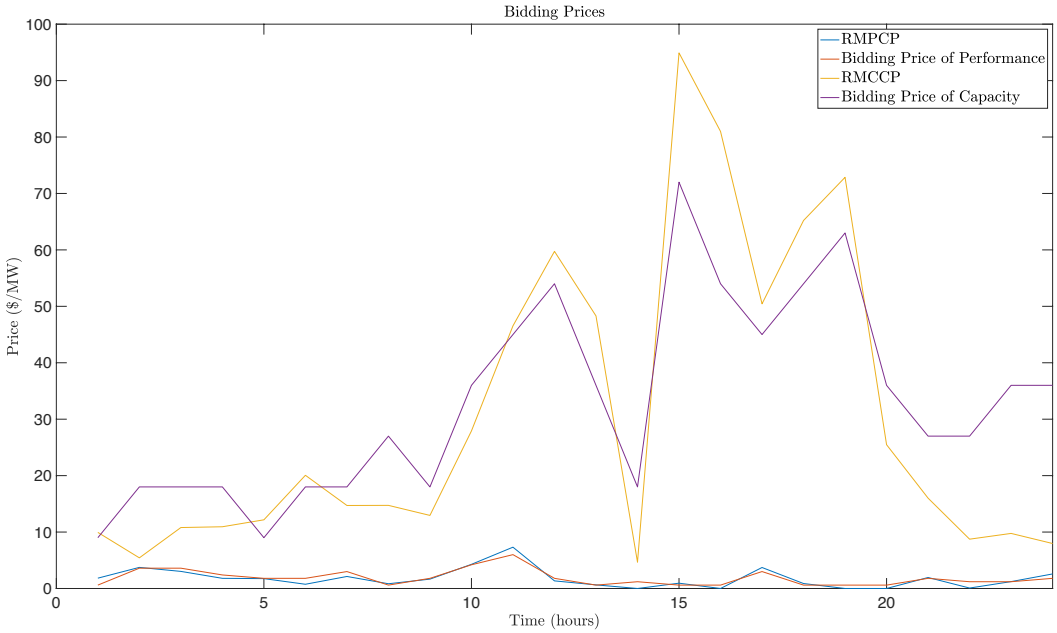

Figure 5.8: BESS bidding prices.

with Fig. 5.9. Due to the increase of ageing losses, the income from the regulation market is comparatively lower than that before, so that the regulation bids are decreased to extend the lifetime of BESS and reduce the transmission losses. Furthermore, with the increase of transmission losses, the Energy bids are increased to balance energy losses during operations.

In Fig. 5.10, the bidding prices are different from the base case. Because of the high cost of losses, it is not worth operating the BESS when the prices are low. In this case, the proposed algorithm will increase the bidding prices to save the cost of regulation market. The higher bidding price will lose more frequency offers, but reduce the transmission and ageing losses.

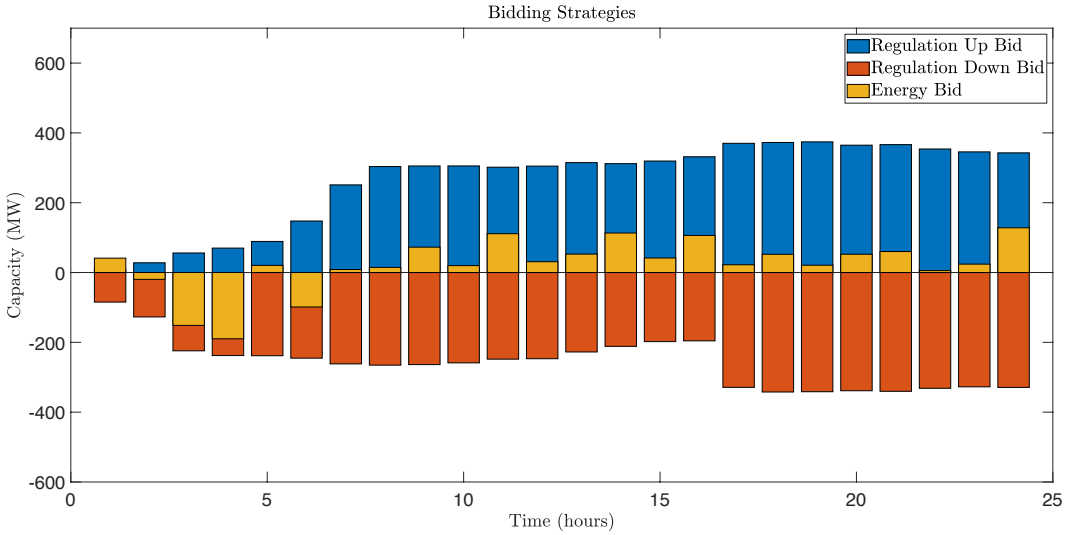

Figure 5.9: BESS bids with ten times losses penalty.

Table. 3 summarises the profit in different markets and costs separately. It can be seen that the benefit from regulation market is the major revenue of BESSs. For our bidding strategy in BESS 1, the BESS has to purchase electricity to balance the energy consumption and losses, so as to the reward from energy market is negative. It

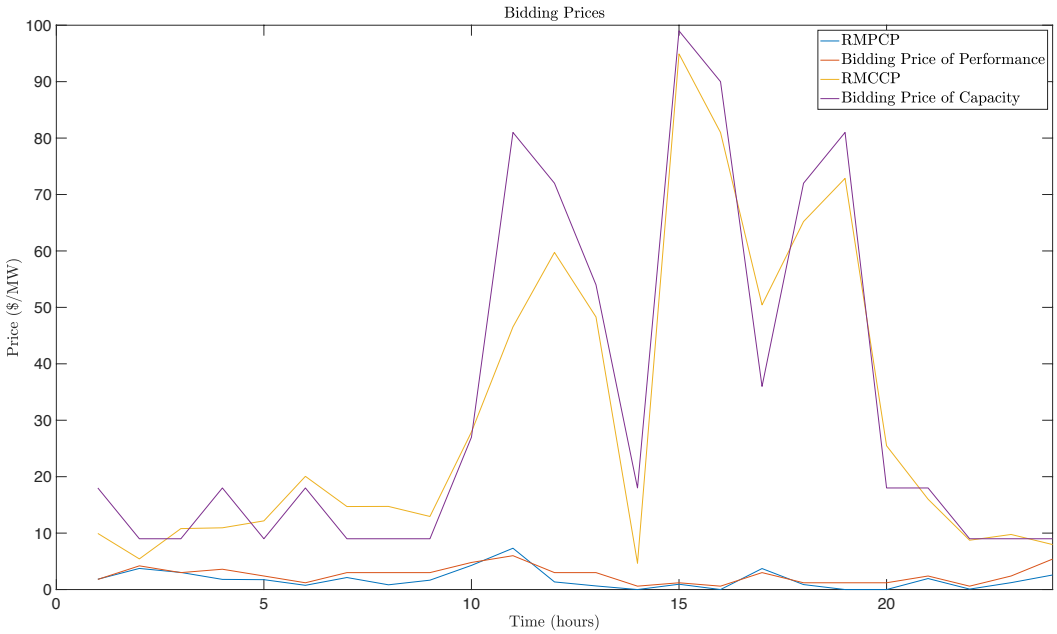

Figure 5.10: BESS bidding prices with ten times losses penalty.

means that the BESS would be deeply involved in regulation market to get high revenue. For other BESSs, BESS2 and BESS3 get lower rewards since their maximum charging/discharging rate limits. The BESS4 has similar parameters as the BESS1, but earns around $70,000 less than the BESS1. This is because the proposed bidding strategy of the BESS1 can receive and learn the reward/penalty signal from the system operator, which does not require any other prior knowledge and study of its rivals. The comparison results show that the proposed model considering the ageing and transmission losses presents a more effective bidding strategy for BESS owners in a bidding environment of multiple rivals, provides a more realistic and accurate cost-benefit result for investors as well.

In addition, to further verify that the proposed algorithm can obtain the maximum profit for BESS owners, the comparison cases with different algorithms are studied and listed

Table 5.3: Income and cost comparison.

	BESS1	BESS2	BESS3	BESS4
Profit^{e} ($)	-17426	-1270	-8147	-14058
Profit^{cap} ($)	346673	182785	208725	284561
Profit^{perf} ($)	37327	14140	18489	23415
Cost^{total} ($)	-10676	-4803	-6419	-8218
Daily Income ($)	355898	190852	212648	285700

in Table 4. Due to the uncertainty of the bidding environment and lack of rival's information, some traditional numerical optimisation approaches, such as game theoretic [112], are not suitable for this environment. Therefore, we compare our results with other learning and stochastic optimisation approaches, Q-learning [113], State-Action-Reward-State-Action (SARSA) [7] and PSO [114]. The proposed FARL algorithm, as shown in the second column in Table 4, successfully achieves highest revenue compared with other methods. The highest incomes for each hour are highlighted in Table 4. Although Q-learning, SARSA and PSO could have higher economic performance at some time slots, our algorithm could help BESS owner get the highest profit for the majority time periods. Judging from the total income of the day, the proposed FARL has the advantage by making around 2.5%, 13% and 6.2% improvement than Q-learning, SARSA and PSO.

5.5 Conclusion

This chapter studied the optimised bidding strategy of the BESS to maximise the profits under a multi-rivals environment. We firstly proposed a bidding model for the BESS in the AGC and energy market, then solved the bidding problem with the reinforcement learning, which using function approximation to avoid aggregated states and dimension curse. Simulation results verified that the proposed method not only get the higher

Table 5.4: Hourly income comparison.

Hour	**FARL**	Q-Learning [113]	SARSA [7]	PSO [114]
01:00	**6390**	5274	3875	5579
02:00	3707	3548	3638	**4419**
03:00	**8468**	2570	3268	2723
04:00	4526	**4633**	2855	2417
05:00	5883	6969	**8008**	4838
06:00	**11803**	8070	6901	5744
07:00	7635	16929	9934	**19091**
08:00	7622	7024	**9171**	7377
09:00	5795	4817	**13588**	6920
10:00	9162	6784	**14808**	13761
11:00	12519	**28564**	10401	13796
12:00	23876	**38742**	8322	8787
13:00	**24214**	2517	18872	16594
14:00	2149	**14809**	10586	5802
15:00	**43959**	40732	35518	42943
16:00	**42546**	21063	24420	26300
17:00	26039	27564	**32043**	23424
18:00	**34487**	18029	28299	27555
19:00	**36691**	27275	17270	32065
20:00	10878	**30830**	11964	23783
21:00	10381	10893	11153	**12171**
22:00	5289	7170	12820	**19624**
23:00	8165	8914	**10088**	6374
24:00	3714	3215	**7147**	3029
Total	**355898**	346935	314949	335116

revenue from the AGC market, but also extends the lifetime of the BESS and reduces the losses.

Chapter 6

Short Term Load Forecasting with Markovian Switching Distributed Deep Belief Networks

In modern power systems, centralised STLF methods raise concern on high communication requirements and reliability when a central controller undertakes the processing of massive load data solely. As an alternative, distributed methods avoid the problems mentioned above, whilst the possible issues of cyberattacks and uncertain forecasting accuracy still exist. To address the two issues, a novel DDBN with Markovian switching topology is proposed for an accurate STLF, based on a completely distributed framework. Without the central governor, STLF is modelled and trained locally, while obtaining the updates through communication with stochastic neighbours under a designed consensus procedure. The overall network reliability against cyberattacks is enhanced by continually switching communication topologies. In the meanwhile, to ensure that

the distributed structure is still stable under such a varying topology, the consensus controller gain is delicately designed, and the convergence of the proposed algorithm is theoretically analysed via the Lyapunov function. Besides, restricted Boltzmann machines (RBM) based unsupervised learning is employed for DDBN initialisation and thereby guaranteeing the success rate of STLF model training. Several case studies and comparison results validate the accuracy and effectiveness of the proposed method.

6.1 Problem Description

The purpose of STLF is to find a proper model that can predict load in an accurate and timely way. Complex and variable load data will affect the accuracy of the load forecasting model. To get better forecast accuracy, one of the most useful approaches is to increase the number of neurons and the hidden layers of the neural network, so that the model can better fit the data with non-linear characteristics. However, increasing the load data amount and model complexity will increase the training time of the model, which cannot meet the time requirements of some power applications. Moreover, training massive data with a single computing centre will decrease the efficiency and increase the cost [27].

In this chapter, we study a novel DBN based STLF method is proposed, which can be trained under a distributed framework. To protect the data security and deal with the mass dataset, we introduce the Markovian switching consensus algorithm to decentralise the DBN and the load dataset. It can solve the STLF model by local computing agents and update the model parameters by communicating with connected neighbours, which does not need to transfer any information about the load data. Different from the existing results, the proposed DDBN model will pre-train their local model by unsupervised training to obtain better initialisation of the model coefficients. Such a pre-trained

initialisation process generates more suitable local models, which consequently yields fast convergence to the global model with less consensus steps. In the meantime, the DDBN model will also amend their local parameters based on the information from their connected neighbours to avoid the over-fitting problem.

6.2 Short-term Load Forecasting Based on Markovian Switching Distributed Deep Belief Networks

6.2.1 Single Deep Belief Network

The DBN consists of RBM and one output layer to forecast the load. In this chapter, we apply two different types of RBM to get more accurate results for the STLF, which are binary RBM and the Gaussian RBM, respectively. The basic structure for STLF is shown in Fig. 6.1.

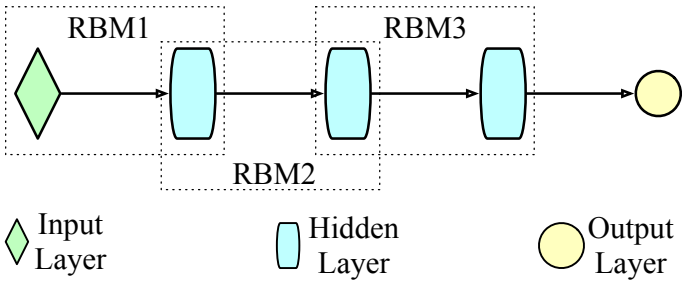

Figure 6.1: DBN based STLF model.

For binary RBM, the conditional probabilities $P(v_i = 1|\boldsymbol{h})$ and $P(h_j = 1|\boldsymbol{v})$ are represented by:

$$P(v_i = 1|\boldsymbol{h}) = \frac{e^{b_i + \sum_{j=1}^{n_h} W_{ij} * h_j}}{1 + e^{b_i + \sum_{j=1}^{n_h} W_{ij} * h_j}} = \boldsymbol{\sigma}(b_i + \sum_{j=1}^{n_h} W_{ij} * h_j), \tag{6.1}$$

$$P(h_i = 1|\boldsymbol{v}) = \frac{e^{c_i + \sum_{i=1}^{n_v} W_{ij} * v_i}}{1 + e^{c_i + \sum_{i=1}^{n_v} W_{ij} * v_i}} = \boldsymbol{\sigma}(c_i + \sum_{i=1}^{n_v} W_{ij} * v_i), \tag{6.2}$$

where $\boldsymbol{\sigma}(\boldsymbol{x}) = 1/[1 + \exp(-\boldsymbol{x})]$ is the sigmoid function.

In order to solve the continuously-valued inputs, the binary RBM can be extended to Gaussian RBM [115], which is more suitable for STLF since the electricity load is continuous and random:

$$\mathcal{F}(\boldsymbol{v}, \boldsymbol{h}) = \sum_{i=1}^{n_v} \frac{(v_i - \bar{v})^2}{2\sigma_i^2} - \sum_{j=1}^{n_h} c_j h_j - \sum_{i=1}^{n_v} \sum_{j=1}^{n_h} v_i h_j W_{ij}, \tag{6.3}$$

where $\bar{v}$ and σ_i are the mean and the standard deviation of the Gaussian distribution for the visible input v_i.

In our model, the input data has been normalised to zero mean and unit variance, and therefore, the Gaussian RBMs are simplified as:

$$\begin{aligned}\mathcal{F}(\boldsymbol{v}, \boldsymbol{h}) =& \frac{1}{2} \sum_{i=1}^{n_v} v_i^2 - \sum_{i=1}^{n_v} b_i v_i - \sum_{j=1}^{n_h} c_j h_j \\ & - \sum_{i=1}^{n_v} \sum_{j=1}^{n_h} v_i h_j W_{ij}.\end{aligned} \tag{6.4}$$

For the Gaussian RBM, the conditional probabilities $P(v_i = 1|h)$ and $P(h_j = 1|\boldsymbol{v})$ are

derived as:

$$P(v_i|\boldsymbol{h}) = \frac{1}{\sqrt{2\pi}}\exp\{-\frac{1}{2}(v_i - b_i - \sum_{j=1}^{n_h} W_{ij}h_j)^2\}$$
$$\sim N(b_i + \sum_{j=1}^{n_h} W_{ij}h_j, 1), \tag{6.5}$$
$$P(h_i =1|\boldsymbol{v}) = \frac{e^{c_i+\sum_{i=1}^{n_v} W_{ij}*v_i}}{1+e^{c_i+\sum_{i=1}^{n_v} W_{ij}*v_i}} = \boldsymbol{\sigma}(c_i + \sum_{i=1}^{n_v} W_{ij} * v_i). \tag{6.6}$$

Therefore, the $P(h_i = 1|\boldsymbol{v})$ obeys a Gaussian distribution with mean of $b_i+\sum_{j=1}^{n_h} W_{ij}h_j$ and unit variance.

In the RBM models, there are only one visible layer and one hidden layer. Therefore, the main task of the optimization is to get the maximize $P(\boldsymbol{v})$, which is the probability of the model simply on the training input data. In order to achieve the maximize $P(\boldsymbol{v})$, we would like to minimize the average negative log-likelihood (NLL):

$$\min_{\boldsymbol{\theta}} -\sum_{\boldsymbol{v}} logP(\boldsymbol{v}, \boldsymbol{\theta}) \tag{6.7}$$

where $\boldsymbol{\theta} = \{\boldsymbol{b}, \boldsymbol{c}, \boldsymbol{W}\}$. The gradients of the NLL for RBM is calculated as:

$$\frac{\partial - \log P(\boldsymbol{v})}{\partial \boldsymbol{\theta}} = \underbrace{\mathbb{E}_{\boldsymbol{h}}\left[\frac{\partial \mathcal{F}(\boldsymbol{v}, \boldsymbol{h})}{\partial \boldsymbol{\theta}}\middle| \boldsymbol{v}\right]}_{\text{positive phase}} - \underbrace{\mathbb{E}_{\boldsymbol{v},\boldsymbol{h}}\left[\frac{\partial \mathcal{F}(\boldsymbol{v}, \boldsymbol{h})}{\partial \boldsymbol{\theta}}\right]}_{\text{negative phase}} \tag{6.8}$$

In the positive phase, the visible variables $\boldsymbol{v}$ are given by the dataset. Therefore, it is easy to calculate the conditional expectation of $\partial\mathcal{F}(\boldsymbol{v}, \boldsymbol{h})/\partial\boldsymbol{\theta}$ based on the equations (6.1)-(6.6). However, the negative phase is hard to calculated, because it is constructed in current RBM, it cannot get the accurate results. As an effective solution, the contrastive divergence (CD) approximates the negative terms by means of sampling approaches [116].

For binary and Gaussian RBMs, the terms $\partial\mathcal{F}(\boldsymbol{v},\boldsymbol{h})/\partial\boldsymbol{\theta}$ can be calculated as:

$$\frac{\partial[\mathcal{F}(v_i,h_j)]}{\partial W_{ij}} = -h_j v_i \tag{6.9}$$

$$\frac{\partial[\mathcal{F}(v_i,h_j)]}{\partial b_i} = -v_i \tag{6.10}$$

$$\frac{\partial[\mathcal{F}(v_i,h_j)]}{\partial c_j} = -h_j \tag{6.11}$$

Representing as the matrix form:

$$\bigtriangledown_{\boldsymbol{W}}\boldsymbol{\mathcal{F}}(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{h}\boldsymbol{v}^{\boldsymbol{T}} \tag{6.12}$$

$$\bigtriangledown_{\boldsymbol{b}}\boldsymbol{\mathcal{F}}(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{v} \tag{6.13}$$

$$\bigtriangledown_{\boldsymbol{c}}\boldsymbol{\mathcal{F}}(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{h} \tag{6.14}$$

For the negative phase, there exist variables $\widetilde{\boldsymbol{v}}$ that can be used to calculate the model expectation. Hence,

$$\mathbb{E}_{\boldsymbol{v},\boldsymbol{h}}\left[\frac{\partial\mathcal{F}(\boldsymbol{v},\boldsymbol{h})}{\partial\boldsymbol{\theta}}\right] = \mathbb{E}_{\boldsymbol{h}}\left[\bigtriangledown_{\boldsymbol{W}}\boldsymbol{\mathcal{F}}(\widetilde{\boldsymbol{v}},\boldsymbol{h})\middle|\,\widetilde{\boldsymbol{v}}\right] \tag{6.15}$$

where the negative sample $\widetilde{\boldsymbol{v}}$ can be generated by the k steps of Gibbs sampling, starting at $\boldsymbol{v}$. This method is also called contrastive divergence (CD) learning, which is proposed in [116].

The two-stage Gibbs sampling approach is iterating the following steps:

$$\text{1. Sample } \boldsymbol{h}^{(\boldsymbol{k+1})} \text{ from } P(\boldsymbol{h}|\boldsymbol{v}^{(\boldsymbol{k})}) \tag{6.16}$$

$$\text{2. Sample } \boldsymbol{v}^{(\boldsymbol{k+1})} \text{ from } P(\boldsymbol{v}|\boldsymbol{h}^{(\boldsymbol{k+1})}) \tag{6.17}$$

Therefore, $\boldsymbol{h^0}, \boldsymbol{v^1}$ and $\boldsymbol{h^1}$ can be sampled from the one step Markov chain. Then, the gradient of the log-likelihood function of the binary states can be approximated as:

$$\boldsymbol{\Delta W} = \epsilon(\boldsymbol{v^0}\cdot\boldsymbol{h^0} - \boldsymbol{v^1}\cdot\boldsymbol{h^1}) \tag{6.18}$$

$$\boldsymbol{\Delta b} = \epsilon(\boldsymbol{v^0} - \boldsymbol{v^1}) \tag{6.19}$$

$$\boldsymbol{\Delta c} = \epsilon(\boldsymbol{h^0} - \boldsymbol{h^1}) \tag{6.20}$$

6.2.2 Distributed Deep Belief Networks

For DDBN, the gradient descent method is employed to optimise the parameters of local DBN. When the gradient descent method stops running, the consensus algorithm is used to optimise the parameters between connected DBN. To improve the data security during the transmission, the Markovian switching consensus is adopted, instead of the traditional consensus algorithm. The procedure code of DDBN is shown in Algorithm 3.

Algorithm 3 Distributed Deep Belief Networks.

procedure DDBN(DBN, Agents)
    DBN ← Unsupervised Learning RBMs
    Sites ← Array of Network pipes to local CA_i
    while *Sites* contains unused data **do**
        Gradients Reset
        //Initialise the last layer parameters of DBN
        for each $CA_i \in$ *Sites* **do**
            $CA_i \leftarrow$ *DBN*
        end for
        //Start local training for each local site
        for each $CA_i \in$ *Sites* **do**
            $CA_i \leftarrow$ Forward Propagation
            $CA_i \leftarrow$ Gradient Descent based on (6.21)
            $CA_i \leftarrow$ Back Propagation with (6.25)
        end for
        //Consensus the weight of local DBN
        while weights are different **do**
            Switching the topologies by transition probability γ
            Apply the typical Laplacian matrix
            for each $CA_i \in$ *Sites* **do**
                Consensus weight with (14)
            end for
        end while
        DDBN ← Consensus weight
    end while
end procedure

Let E be the cost function of the neural networks, written as:

$$E(\mathcal{D}, w) = \frac{1}{m}\sum_{k=1}^{m} e((x^k, y^k), w), \tag{6.21}$$

where w is composed of the weights and bias of the neural networks; m is the number of data samples and e denotes the cost of one sample pair. Separating the data set $\mathcal{D}$ into local agent $\mathcal{D}_i$, the cost function is reformulated as:

$$E(\mathcal{D}, w) = \frac{1}{m}\sum_{i=1}^{n}\sum_{k=1}^{m_i} e(x_i^k, y_i^k, w_i) \tag{6.22}$$

$$= \sum_{i=1}^{n}\frac{m_i}{m}\frac{1}{m_i}\sum_{k=1}^{m_i} e(x_i^k, y_i^k, w_i) \tag{6.23}$$

$$= \sum_{i=1}^{n}\frac{m_i}{m}E(\mathcal{D}_i, w_i). \tag{6.24}$$

Using the gradient-based method [117], the local update law can be designed as:

$$w_i^k(t+1) = w_i^k(t) - \eta\frac{\partial}{\partial w_i^k}e((x_i^k, y_i^k), w_i^k), \tag{6.25}$$

where η is the step size of the training process.

Here, we formulate the STLF model as a model of the DDBN gradient descent system:

$$w_i(t+1) = w_i(t) + v_i(t) + u_i(t), \quad i = 1, \cdots, n, \tag{6.26}$$

where w_i is the propagation weight of the ith data centre; $v_i = -\eta(\partial E(\mathcal{D}_i, w_i)/\partial w_i)$ is the local gradient descent, and u_i is the consensus control input. Based on the gradient descent algorithm and the actual training situation, $v_i \in \mathbb{R}^{\ltimes}$ (where $\ltimes$ is the number of parameters) and assume that the local gradients are bounded in the sense of probability $P\left\{v_i(t) < \bar{\delta}\right\} \equiv 1$.

Motivated by [118], the consensus controller is designed as following:

$$u_i(t) = \kappa\sum_{j=0}^{n} a_{ij}(t)(w_j(t) - w_i(t)), \tag{6.27}$$

where κ is the controller gain, and $a_{ij}(t)$ is time varying since the topological structures are Markovian switching. Let $L_\tau, (\tau \in Q)$ be the Laplacian matrix and we have:

$$\boldsymbol{w}(t) = (w_1^T(t), w_2^T(t), \cdots, w_n^T(t))^T, \tag{6.28}$$

$$\boldsymbol{v}(t) = (v_1^T(t), v_2^T(t), \cdots, v_n^T(t))^T, \tag{6.29}$$

$$\boldsymbol{u}(t) = (u_1^T(t), u_2^T(t), \cdots, u_n^T(t))^T. \tag{6.30}$$

Then the model (6.26) can be rewritten as the matrix form based on the controller (6.27):

$$\boldsymbol{w}(t+1) = (\boldsymbol{I} - \kappa \boldsymbol{L_\tau})\boldsymbol{w}(t) + \boldsymbol{v}(t). \tag{6.31}$$

The controller (6.27) can solve the consensus problem of system (6.26), if there exists a positive constant $A \in \mathbb{R}$ and the following inequality hold:

$$\lim_{t\to\infty} \mathbb{E}[|w_i(t) - w_j(t)|^2] < A, \;\; i \neq j, \;\; i, j = 1, \cdots, n. \tag{6.32}$$

6.2.3 Convergence Analysis

In this section, we will discuss the DDBN gradient update system with Markovian switching topological structure, and analyse the mean-square stability of this system.

Let $\xi_i(t) = w_i(t) - w_j(t)$ and $\zeta_i(t) = v_i(t) - v_j(t)$. Then the system (6.31) can be formulated as:

$$\boldsymbol{\xi}(t+1) = (\boldsymbol{I} - \kappa \boldsymbol{L_\tau})\boldsymbol{\xi}(t) + \boldsymbol{\zeta}(t), \tag{6.33}$$

where $\boldsymbol{\xi}(t) = (\xi_1^T(t), \xi_2^T(t), \cdots, \xi_n^T(t))^T$ and $\boldsymbol{\zeta}(t) = (\zeta_1^T(t), \zeta_2^T(t), \cdots, \zeta_n^T(t))^T$. Corresponding to this transformation, a new Laplacian matrix is obtained:

$$L_{i\tau} = \begin{bmatrix} l_{11} - l_{i1} & \cdots & l_{1n} - l_{in} \\ \vdots & \ddots & \vdots \\ l_{n1} - l_{i1} & \cdots & l_{nn} - l_{in} \end{bmatrix}. \tag{6.34}$$

Considering system (6.26), assume there is a local neural network $h(h = 1, 2, \cdots, n)$ under a topology $\tau, (\tau \in Q)$. After comparing this network w_h with any network w_i, the Laplacian matrix $L_{h\tau}$ can make the controller gain κ satisfy the following conditions:

$$0 < \kappa\lambda < 2, \tag{6.35}$$

where λ is the minimal positive eigenvalue of Laplacian matrix and $\lambda \in \bigcup_{\tau=1}^{Q} \{\lambda(L_{h\tau}), \tau \in Q\}$.

If the value of κ satisfies $\kappa\lambda \in (0, 2)$, and we consider that $\phi_{h\tau} = \boldsymbol{I} - \kappa \boldsymbol{L_\tau}$, then we have $\rho(\phi_{h\tau}) < 1$, which means $\|\phi_{h\tau}\| < 1$.

$P_{\tau \in Q}$ is a positive symmetric matrix set. Consider a Lyapunov function, we have:

$$V(\boldsymbol{\xi_h}(t+1)) = \boldsymbol{\xi_h}^T(t+1) P_\tau \boldsymbol{\xi_h}(t+1) \geqslant 0. \tag{6.36}$$

Let $\overline{\tau} = \lambda_{max} \{P_\tau\}$, $\underline{\tau} = \lambda_{min} \{P_\tau\}$, $\overline{\phi_{h\tau}} = \max_{\tau \in Q} \{\phi_{h\tau}\}$. Due to $\|\phi_{h\tau}\| < 1$, then we always have $\overline{\phi_{h\tau}} < 1$.

As gradient descent algorithm is used in each iteration, $v_i(t) < \bar{\delta}$ will be always true, i.e., $P\left\{v_i(t) < \bar{\delta}\right\} = 1$. Then we have:

$$\lim_{t \to \infty} \mathbb{E}[V(\boldsymbol{\xi_h}(t+1))] = \lim_{t \to \infty} \mathbb{E}[\boldsymbol{\xi_h}^T(t+1) P_\tau \boldsymbol{\xi_h}(t+1)]. \tag{6.37}$$

Since $\gamma_{ij} \leqslant 1$, we have:

$$\begin{aligned}
&\lim_{t\to\infty} \mathbb{E}[V(\boldsymbol{\xi_h}(t+1))] \\
\leqslant &\lim_{t\to\infty} \sum_{j=1}^{Q} \gamma_{ij} [\boldsymbol{\phi_\tau}\boldsymbol{\xi_h}(t) + \boldsymbol{\zeta_h}(t)]^T P_\tau [\boldsymbol{\phi_\tau}\boldsymbol{\xi_h}(t) + \boldsymbol{\zeta_h}(t)] && (6.38)\\
\leqslant &\lim_{t\to\infty} \sum_{j=1}^{Q} \gamma_{ij} \Bigg[\boldsymbol{\phi_\tau} \sum_{j=1}^{Q} \gamma_{ij} [\boldsymbol{\phi_\tau}\boldsymbol{\xi_h}(t-1) + \boldsymbol{\zeta_h}(t-1)] \\
&+ \boldsymbol{\zeta_h}(t) \Bigg]^T P_\tau \Bigg[\boldsymbol{\phi_\tau} \sum_{j=1}^{Q} \gamma_{ij} [\boldsymbol{\phi_\tau}\boldsymbol{\xi_h}(t-1) + \boldsymbol{\zeta_h}(t-1)] \\
&+ \boldsymbol{\zeta_h}(t) \Bigg] && (6.39)\\
\leqslant &\cdots && (6.40)\\
\leqslant &\lim_{t\to\infty} [\bar{\boldsymbol{\phi}}_{\boldsymbol{\tau}}\boldsymbol{\xi_h}(0) + \sum_{j=1}^{t+1} \bar{\boldsymbol{\phi}}_{\boldsymbol{\tau}}\boldsymbol{\zeta_h}(j)]^T P_\tau [\bar{\boldsymbol{\phi}}_{\boldsymbol{\tau}}\boldsymbol{\xi_h}(0) \\
&+ \sum_{j=1}^{t+1} \bar{\boldsymbol{\phi}}_{\boldsymbol{\tau}}\boldsymbol{\zeta_h}(j)] && (6.41)\\
\leqslant &\lim_{t\to\infty} (\frac{1-\|\bar{\phi}\|^t}{1-\|\bar{\phi}\|}\bar{\delta}) \bar{\tau} (\frac{1-\|\bar{\phi}\|^t}{1-\|\bar{\phi}\|}\bar{\delta}) && (6.42)\\
\leqslant &\frac{\bar{\tau}\bar{\delta}^2}{(1-\|\bar{\phi}\|)^2}. && (6.43)
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\lim_{t\to\infty} \mathbb{E}[|\boldsymbol{w_i}(t) - \boldsymbol{w_j}(t)|^2] \\
= &\lim_{t\to\infty} \mathbb{E}[|(\boldsymbol{w_h}(t) - \boldsymbol{w_j}(t)) - (\boldsymbol{w_h}(t) - \boldsymbol{w_i}(t))|^2] && (6.44)\\
\leqslant &2 * \lim_{t\to\infty} \mathbb{E}[|\boldsymbol{w_h}(t) - \boldsymbol{w_j}(t)|^2 - |\boldsymbol{w_h}(t) - \boldsymbol{w_i}(t)|^2] && (6.45)\\
\leqslant &2 * \lim_{t\to\infty} \mathbb{E}[\|\boldsymbol{\xi_h}(t)\|^2] && (6.46)\\
\leqslant &\frac{2\bar{\tau}\bar{\delta}^2}{\underline{\tau}(1-\|\bar{\phi}\|)^2} < +\infty. && (6.47)
\end{aligned}$$

For the proposed DDBN model, the mean-square consensus can be achieved with the designed controller (6.27). It means that all the errors of local DBN parameters $\lim_{t\to\infty} \mathbb{E}[|\boldsymbol{w_i}(t)-\boldsymbol{w_j}(t)|^2]$ will converge to zero, which further implies the weights of neural networks can achieve consensus to the optimal weights w^*. Referring to [119], the distributed learning framework proposed in this chapter can generate equivalent results as the centralised DBN.

Distinct from the aforementioned works, we employ the tools of the Mean Squared Error (MSE) to establish the convergence. Notice that the MSE can diffuse over the distributed network via the communication channels, and thus, we have considered the worst case scenario where the most significant error expectation of the agent is used, as demonstrated in (6.37) - (6.47). It is worth emphasising that [119] has used some restrictive assumptions, e.g., convexity and basin of attraction to establish the convergence, while the proposed algorithm releases them to the commonly-used assumption.

6.3 Simulation Results and Analysis

6.3.1 Experimental Environment and Data Resources

To test the effectiveness of the proposed DDBN model, fully distributed networks are built based on MATLAB R2019b and Python 3.6.1.

For the STLF model, the dataset used in case studies is provided by the Global Energy Forecasting Competition (GEFCom) 2017 competition [8] and ISO New England, including the historical load, daily average temperature and holiday type data for the period 2016-2019. The electricity load data and average temperature are recorded once per hour.

To evaluate the performance of the proposed model, the MSE, Mean Absolute Error (MAE) and Mean Absolute Percent Error (MAPE) are applied as the performance indicators. MAPE is estimated through the following equation:

$$\varepsilon_{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\%, \tag{6.48}$$

where y_i is the actual load at period i; N is the total length of forecasting periods; and $\hat{y}_i$ is the forecast load at time i.

Similarly, MAE is expressed as:

$$\varepsilon_{MAE} = \frac{\sum_{i=1}^{N}|y_i - \hat{y}_i|}{N}. \tag{6.49}$$

MSE can be calculated as:

$$\varepsilon_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2. \tag{6.50}$$

6.3.2 Model Setting

Short-term electricity load demand is affected by many factors, including time series, temperature factors, data types and emergency events. The impact of meteorological factors on the electricity load is particularly significant because the use of high-power electrical appliances such as air-conditioning, lighting, and heating is determined by environmental temperature and time factors [120]. In this chapter, our proposed DDBN model has the advantage of feature selection, and therefore the STLF model does not need to be initialised by feature selection and randomly parameter initialisation. The initialisation work will be finished by unsupervised RBM learning process. Here, weather factors are quantified based on the environment temperature. Also, the STLF has a significant daily periodicity, and therefore the history load in the first two days and one

week before the forecast date has also been processed as the input variables. The input and output variable indexes are shown in Table 6.1.

Table 6.1: STLF input and output information.

input features	input variables
1-7	actual loads at the previous week
8-14	temperatures at the previous week
15-21	day types at the previous week
22	actual load at 2 days ago
23	temperature at 2 days ago
24	day type at 2 days ago
25	actual load at 24 hours before
26	temperature value at 24 hours before
27	day type at 24 hours before
28	predicted temperature on the forecast day
29	day type on the forecast day
output	load value at time t on the forecast day

In order to achieve a unified analysis of various sampling data and improve the prediction accuracy, the raw data are pre-normalised. This chapter implements Gaussian normalisation to the input data of DDBN; process variables are normalised to zero mean and unit variance by the following linear transformation:

$$x_i^{\star}(k) = \frac{x_i(k) - \bar{x_i}}{\sigma_i}, \tag{6.51}$$

where $x_i(k)$ is the ith original input data, $\bar{x_i}$ and σ_i are the average and variance of ith original input data.

In the simulation studies, the hidden units of local DBN are set as 300, 500 and 200 neurons. The three system communication topologies under consideration are also shown in Fig. 6.2, each of them has a quarter of dataset.

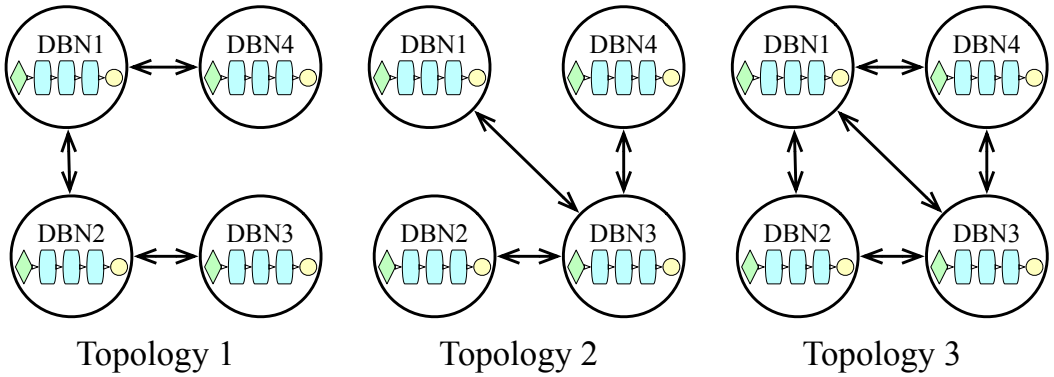

Figure 6.2: Communication typologies.

The transition probability matrix of this network is set as:

$$\gamma = \begin{bmatrix} 0.3 & 0.4 & 0.3 \\ 0.2 & 0.3 & 0.5 \\ 0.4 & 0.4 & 0.2 \end{bmatrix}. \tag{6.52}$$

6.3.3 Experimental Analysis and Comparison

Case 1

This case verifies the effectiveness of the proposed DDBN model. The data from January of 2017 to December of 2018 is selected as the training data, and we randomly chose four weeks data of 2019 as the testing data, which covers four seasons STLF in a year. The results are shown in Figs. 6.3 - 6.6 and Table 6.2. The bottom of Figs. 6.3 - 6.6 provides the absolute error at each hour of the test dataset. The blue line is the forecast results of proposed DDBN model, and it is evident that the proposed DDBN algorithm has less forecasting error at most of the data points.

From Figs. 6.3 - 6.6 and Table 6.2, the forecast results of the proposed DDBN model show better fitting to the real load data than the centralised DBN model. For different

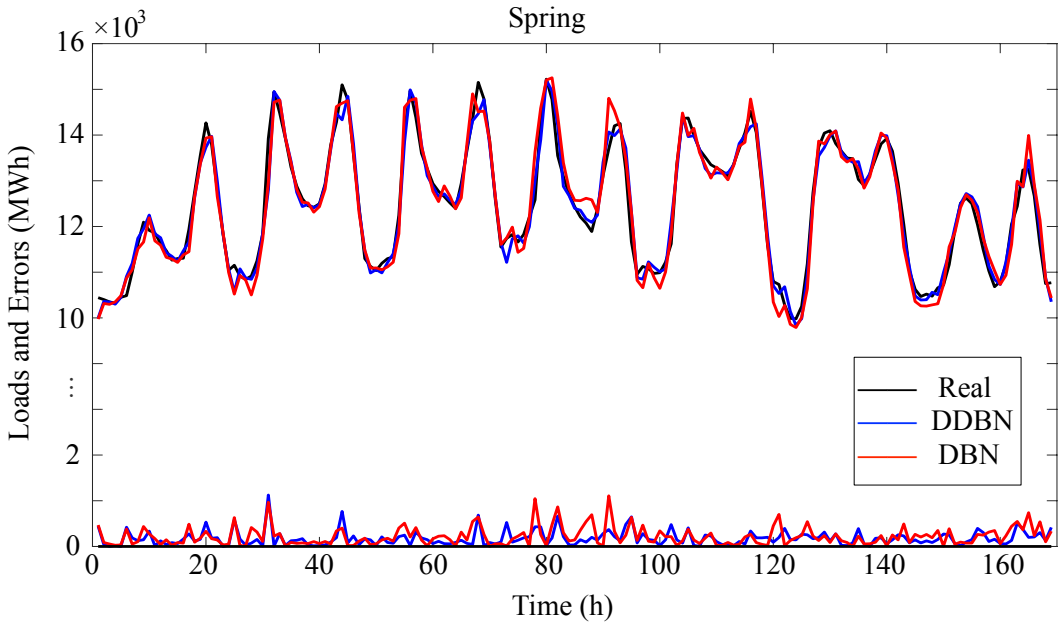

Figure 6.3: Spring short-term load forecasting results.

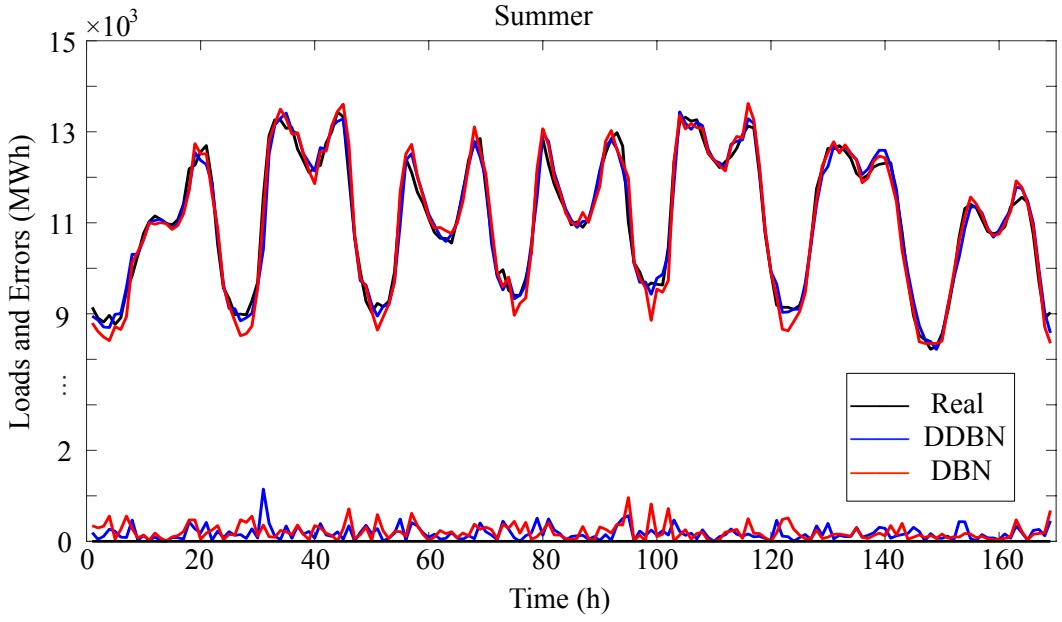

Figure 6.4: Summer short-term load forecasting results.

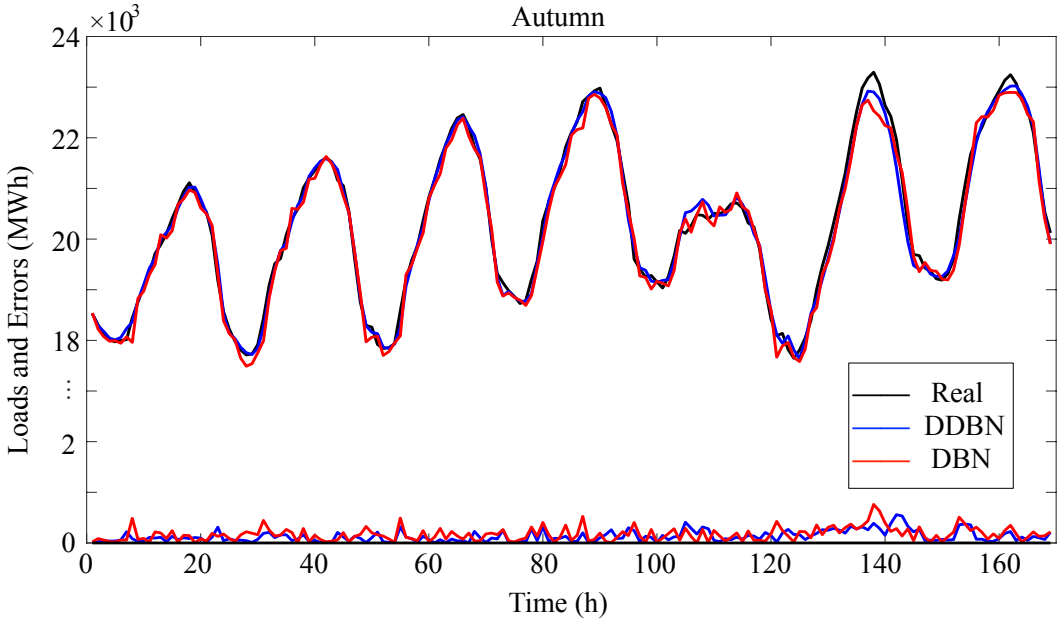

Figure 6.5: Autumn short-term load forecasting results.

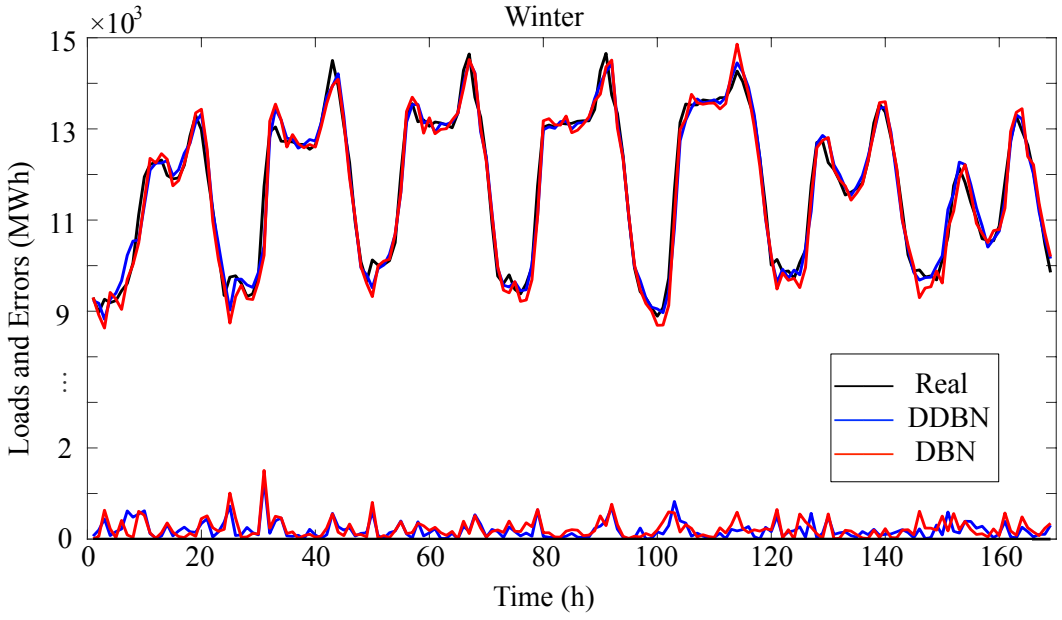

Figure 6.6: Winter short-term load forecasting results.

Table 6.2: Comparison of DDBN and DBN models.

	MAPE (%)		MAE (kWh)		MSE (kWh^2)	
	DDBN	DBN	DDBN	DBN	DDBN	DBN
Spring	1.90	2.40	182.41	247.39	$6.31e^4$	$1.04e^5$
Summer	2.18	2.66	193.52	245.52	$6.83e^4$	$1.04e^5$
Autumn	2.04	2.54	211.53	281.65	$7.74e^4$	$1.36e^5$
Winter	2.08	2.54	194.98	250.56	$7.22e^4$	$1.08e^5$
Average	2.05	2.54	195.61	256.28	$7.03e^4$	$1.13e^5$

seasons and dates, DDBN model can generate satisfactory prediction models in general. In this case, the average training time of DBN model is 7357.61 seconds, while the DDBN model only costs 2502.86 seconds. Compared with the centralised DBN model, the training time consumption of DDBN decreases by 65.98%. Therefore, the proposed DDBN model meets the STLF requirements and has better accuracy than the centralised DBN model.

In order to provide results illustrating the distribution of the load forecasting percentage errors, box-plots are used to visualise the forecasting errors through five-number summaries: sample minimum, lower quartile, median, upper quartile, and sample maximum [121]. Fig. 6.7 shows the percentage error statistics of hours, weeks and months in the whole year test dataset among four computing centres, separately.

As depicted in Fig. 6.7, power consumption and forecasting error are different in different periods. Due to the useful unsupervised RBM learning process and distributed training process, our STLF model can select most of the data information and load characteristics. From the top figure, the forecast errors in different weekdays are similar, which means our model has stable performance over a week. The middle figure indicates that power consumptions in the morning and evening are higher than that at noon, and the variance of electricity consumption in the early morning is more significant. The bottom figure describes the short-term load observations for each month over

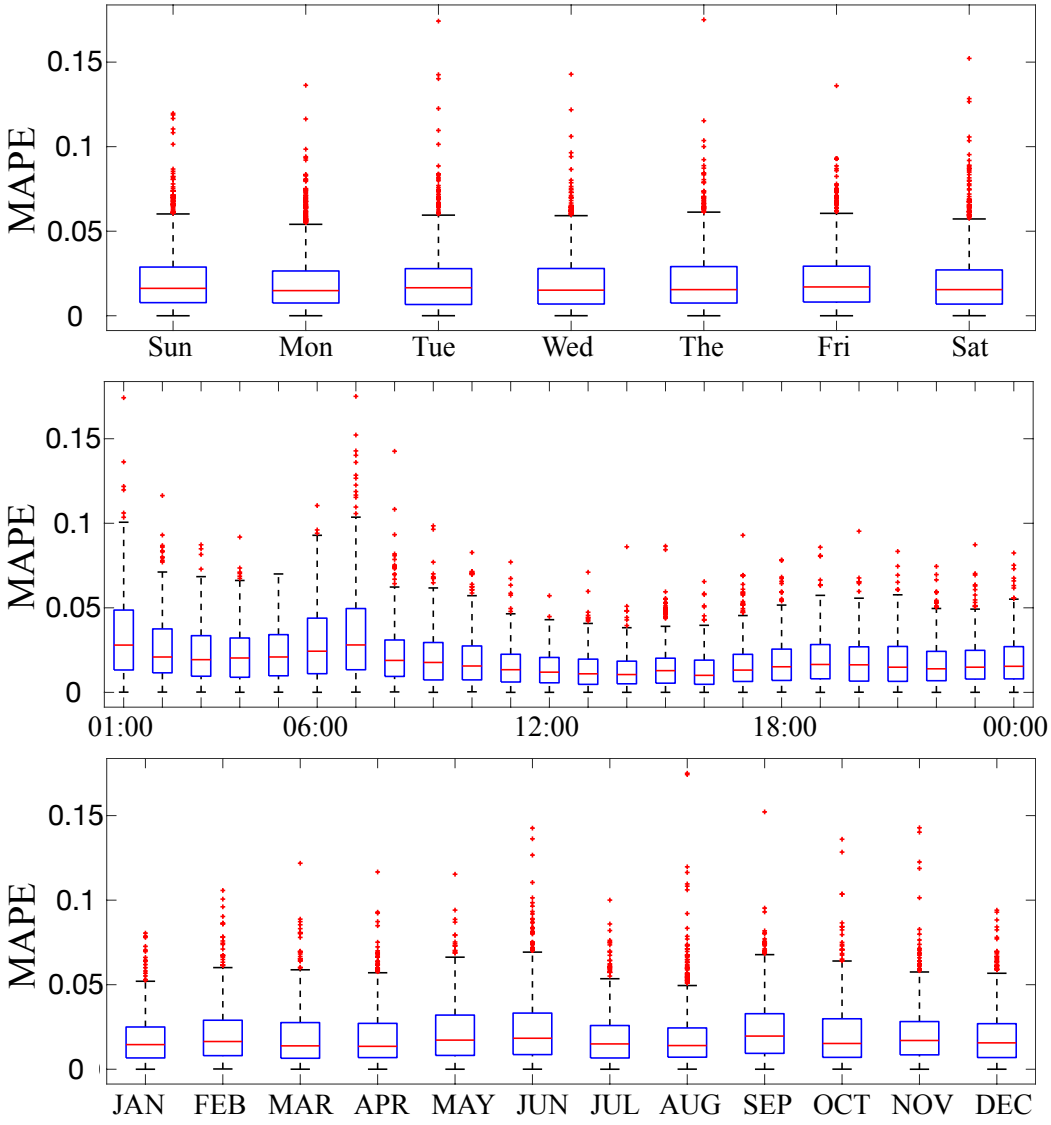

Figure 6.7: Error distribution of STLF.

the whole year. When the season changes and the hot and cold alternate, the percentage error of STLF will be increased (around June and September). However, in other months, the forecasting error is well controlled within a reasonable range.

Case 2

In this case, we investigate the learning performance of the proposed DDBN model. Fig. 6.8 depicts the MAPE in the training process among 4 local DDBN agents and traditional centralised DBN. During the training process, our local DDBN agent has only a quarter of the dataset of the traditional DBN model. We observe that the local computing agents of DDBN converge slower than centralised DBN, but can efficiently get similar accuracy after enough training iterations.

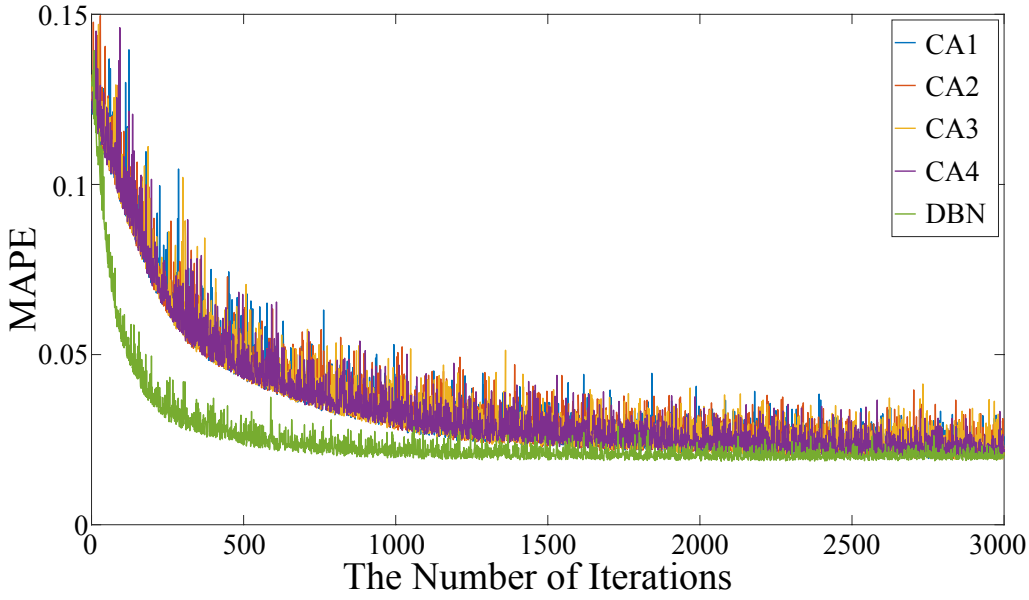

Figure 6.8: Learning performance comparison with DBN.

In the real power systems, the communication networks may be changed by many reasons, such as cyberattacks and routine maintenance, and therefore we further consider

the uncertain cyberattack in this case. It is assumed that there is a computing agent DBN3 which is attacked and loses its ability to communicate with the other computing agents at 500 iterations, and is re-connected at 1000 iterations after the emergency repairs.

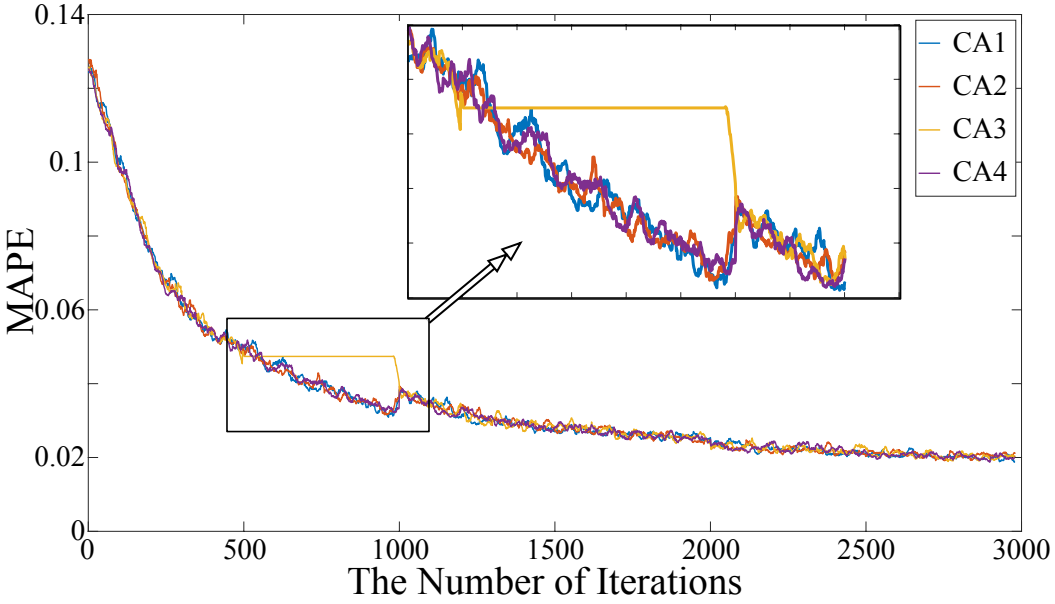

Figure 6.9: Learning performance during communication failure.

Fig. 6.9 shows the average MAPE values in successive 200 iterations. It can be seen that the DBN3 loses its computing power and will not respond to the other neighbouring agents when the cyberattack comes. However, the rest of computing agents will continue training through their load dataset and weights. After the computing agent remove the threat, the DBN3 will be reconnected into the training system so that all other computing agents can share and update their weight information to get better STLF model. Although the accuracy of STLF model is degraded in next few steps, all the computing agents can quickly tune the weight of DBN3 to reach a better model. The communication graph among the agents switches within a finite graph set following a Markovian chain. Therefore, the proposed algorithm can converge to its optimal location if the

finite graph set can coverage at least one connected graph. The proposed algorithm can adaptive to the communication network failures unless the communication graph set cannot coverage a connect graph. In the case of cyberattacks, the accuracy of STLF model will not be influenced much by the proposed Markovian switching topology.

Case 3

Here we summarise the different STLF models to illustrate the advantages of proposed model, three typical existing STLF models (ANN, Support Vector Regression (SVR), LSTM) are analysed for comparison. The results of different models are summarised in Fig. 6.10 and Table 6.3. The compared methods have same training data, test data with 3000 training iterations.

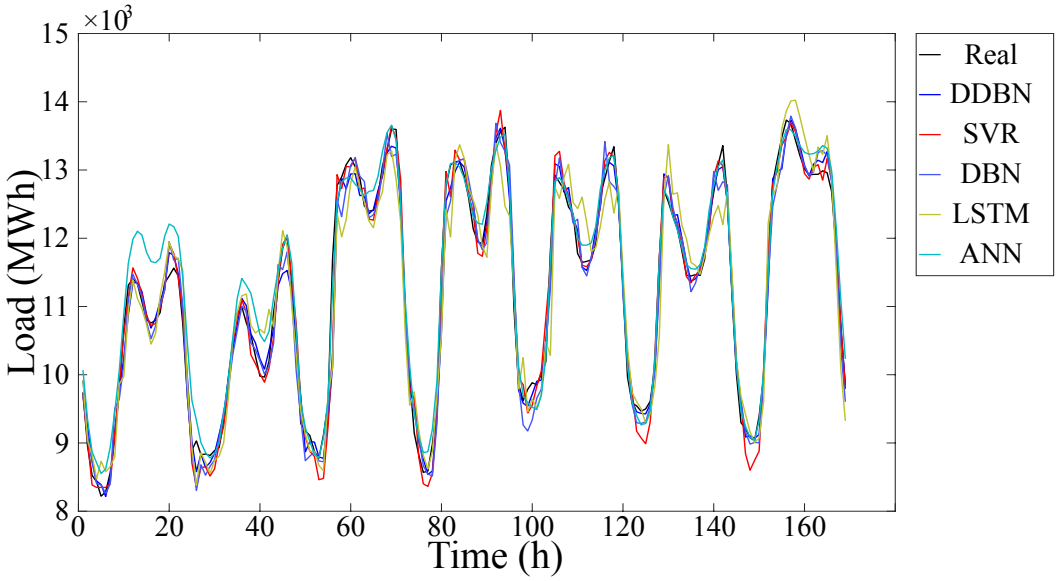

Figure 6.10: Comparison of different algorithms.

From the results given in Fig. 6.10 and Table 6.3, the DDBN algorithm meets the requirements and has better accuracy than other models. For the dataset used in this case,

Table 6.3: Comparison of different STLF models.

	ANN	DBN	DDBN	SVR	LSTM
MAPE (%)	3.64	2.00	1.98	2.44	1.95
MAE (W)	460.53	202.31	199.61	256.33	195.64
MSE (W^2)	$3.36e^5$	$7.27e^4$	$7.19e^4$	$1.13e^5$	$7.03e^4$

the DDBN algorithm proposed obtains better performance and less prediction errors.

Case 4

We carry out numerical experiments to verify the scalability of the proposed DDBN model. There are now more than 15.6 million smart and advanced meters operating across homes and businesses in Great Britain before June of 2019 [122]. The number of smart meters will continue to grow by around 5% a quarter. These smart meters will generate massive load data samples in different locations every day, and thus it is difficult for traditional STLF to collect and analyse those samples relying on a single computing centre.

Since the quantity of electricity load is developing rapidly, we test the scalability of the proposed DDBN model. Firstly, we compare the time consumption of the training process over DDBN, DBN, ANN, LSTM and SVR models. The training speed of the STLF varies according to the size of load dataset and the model of STLF. The dataset has been divided as 20000, 40000, 120000 pairs to test the different models. In this case, the basic structure of the neural networks is set as (250, 350, 200, 1) neurons, and the DDBN model has four computing agents. The comparison results are shown in Fig. 6.11.

Compared with the centralised DBN method, the DDBN model can realise distributed

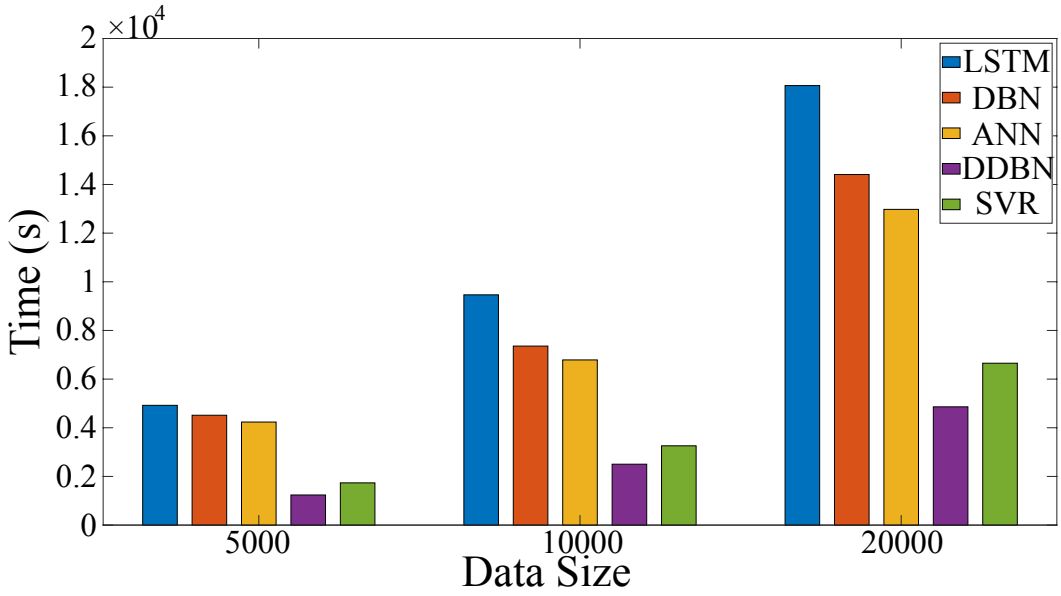

Figure 6.11: Comparison of time consumption.

architecture and solve the STLF problem with distributed computing agents. The distributed training process reduces the size of a dataset for training in each local network, and therefore it has advantages in training speed. From Fig. 6.11, it is evident that the training time of centralised DBN model is much higher than other models. Due to the pre-training process of RBM, the training speed of ANN and SVR is faster than DBN and DDBN if the size of the training dataset is small. However, with the increasing of load data quantity, DDBN model reflects the advantages of distributed structure, which can significantly accelerate the training speed of the model.

To illustrate the advantages of DDBN more clearly, we record the training speed under different topologies of DDBN models. The topologies of the distributed models are set as an undirected ring network [123]. Fig. 6.12 shows that the proposed DDBN model can reduce the training time with an increasing number of computing agents. It also reflects that the proposed DDBN model can be applied in STLF. At the same time, if the load dataset continues to increase, the aim of fast training can be achieved by adding

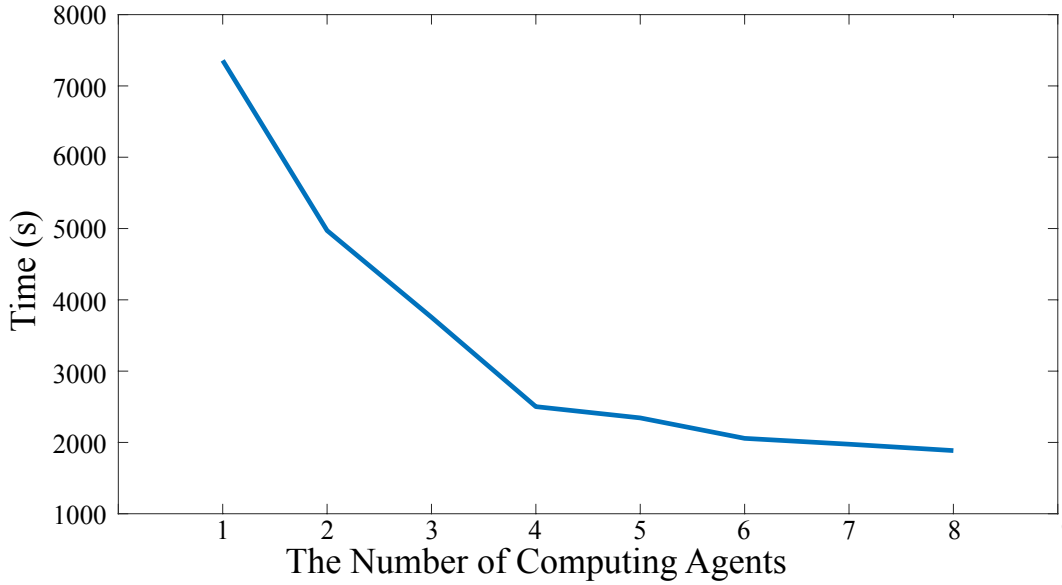

Figure 6.12: Time consumption with different computing agents.

more computing agents.

6.4 Summary

This chapter proposes a novel Markovian switching consensus based DDBN model to solve the STLF problem. The proposed DDBN model separates the massive load dataset into several local computing agents, which significantly reduces the training time of STLF and effectively avoids the over-fitting problem with its fully distributed structures. Meanwhile, the Markovian switching technology is applied in our DDBN model to improve the stability and robustness of STLF model.

In the numerical simulations, a real-world dataset has been applied to evaluate the performance of the DDBN model. The multiple seasonal variations can be forecast by the proposed approach, and the simulation results show the advantages in reducing time

consumption and improving forecast accuracy of STLF.

Chapter 7

Conclusions

In this chapter, we first conclude our main contribution, and then we provide some possible future works.

7.1 Conclusions

In this thesis, three important issues for the power system management have been addressed with different algorithms.

First, a distributed initialisation-free control algorithm is proposed for demand side management by maximising the total revenue of the power system while considering the system active power constraints and local physical constraints. The proposed distributed optimisation strategy is applied based on multi-agent system framework, which is robust to dealing with single point failures compared with centralised approaches. The proposed distributed manner can handle the initial errors. Therefore, the power units can plug-in or plug-off from any power allocations. Moreover, the proposed distributed

algorithm can balance the time-varying power mismatch between supply and demand side in an isolated power system.

To solve the bidding issue of BESS in power markets, we formulate a function approximation based reinforcement learning model for the BESS owners to maximise their daily bidding revenue in energy market and regulation market. This decision-making approach has taken into account the management of the BESS, bidding of the energy market auction and bidding of regulation market auction, which is seldom integrated in the previous papers. The ageing model of BESS is also considered in this thesis, which can protect the health of BESS while maximising individually profit. The proposed algorithm can provide a bidding strategy by analysing historical market data and local rewards without considering the information of its rivals. Therefore, the proposed FARL algorithm can effectively optimise the bidding strategy under an uncertain environment.

Finally, to handle the redundant data and provide accurate short-term load forecasting, we integrate the deep learning algorithm and the distributed optimisation algorithm together. A Markovian switching framework based distributed deep belief network algorithm is proposed for accurate short-term load forecasting. The introduced method separate the historical load data under a fully distributed multi-agent framework to reduce the training time and computing power. In the meantime, we develop the communication network by Markovian switching topology to enhance the cybersecurity of the power data. Moreover, simulation cases are studied to reveal the accuracy and effectiveness of our approach.

7.2 Future Research

In the future research, the following open problems could be heuristic topics for the distributed applications of power systems optimisation and management.

1. In the absence of power flow constraints, the current data-driven solutions cannot solve the optimal bidding problem of the optimal power flow type with power system operating constraints, including voltage, heat, current and loss. These can be further studied in the future.

2. It is possible to integrate the data-driven methods to demand side management problems. In a power system, there are numerical of smart meters in demand side, and they will generate a mass of data. With proper analysis, these data are expected to improve energy efficiency.

3. Most existing results for data-driven methods in power systems are designed and implemented by model-based algorithms, such as model predict control, machine learning and deep learning methods. For fast developing renewable energy applications, the power system needs time-varying operations and objectives, which is hard to forecast accurately by these methods. Therefore, online adjustment algorithms are supposed to overcome the error caused by the time-varying characteristics of the power systems.

Bibliography

[1] F. Slye, "Uk future energy scenarios: System operator," National Grid UK.

[2] T. Adefarati and R. Bansal, "Integration of renewable distributed generators into the distribution system: a review," *IET Renew. Power Gener.*, vol. 10, no. 7, pp. 873–884, 2016.

[3] T. Zhao, A. Parisio, and J. V. Milanović, "Distributed control of battery energy storage systems for improved frequency regulation," *IEEE Trans. Power Syst.*, vol. 35, no. 5, pp. 3729–3738, 2020.

[4] T. Zhao, B. Chen, S. Zhao, J. Wang, and X. Lu, "A flexible operation of distributed generation in distribution networks with dynamic boundaries," *IEEE Trans. Power Syst.*, vol. 35, no. 5, pp. 4127–4130, 2020.

[5] B. Bhandari, K.-T. Lee, G.-Y. Lee, Y.-M. Cho, and S.-H. Ahn, "Optimization of hybrid renewable energy power systems: A review," *Int. J. Precis. Eng. Man-Gt.*, vol. 2, no. 1, pp. 99–112, 2015.

[6] H. Zhao, Q. Wu, S. Hu, H. Xu, and C. N. Rasmussen, "Review of energy storage system for wind power integration support," *Appl. Energy*, vol. 137, pp. 545–553, 2015.

[7] Z. Li, Z. Ding, and M. Wang, "Optimal bidding and operation of a power plant with solvent-based carbon capture under a co2 allowance market: A solution with a reinforcement learning-based sarsa temporal-difference algorithm," *Engineering*, vol. 3, no. 2, pp. 257–265, 2017.

[8] T. Hong, P. Pinson, S. Fan, H. Zareipour, A. Troccoli, and R. J. Hyndman, "Probabilistic energy forecasting: Global energy forecasting competition 2014 and beyond," *Int. J. Forecast.*, vol. 32, no. 3, pp. 896 – 913, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169207016000133

[9] B. Gharesifard and J. Cortés, "Distributed continuous-time convex optimization on weight-balanced digraphs," *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 781–786, 2014.

[10] P. Yi, Y. Hong, and F. Liu, "Initialization-free distributed algorithms for optimal resource allocation with feasibility constraints and application to economic dispatch of power systems," *Automatica*, vol. 74, pp. 259–269, 2016.

[11] Y. Xu, Z. Yang, W. Gu, M. Li, and Z. Deng, "Robust real-time distributed optimal control based energy management in a smart grid," *IEEE Trans. Smart Grid*, vol. 8, no. 4, pp. 1568–1579, 2017.

[12] W. Lin, X. Jin, Y. Mu, H. Jia, X. Xu, X. Yu, and B. Zhao, "A two-stage multi-objective scheduling method for integrated community energy system," *Appl. Energy*, vol. 216, pp. 428–441, 2018.

[13] N. Statistics, "Digest of united kingdom energy statistics 2017," *U.K:Department of Energy and Climate Change*, pp. 1–264, 2017.

[14] B. P. Esther and K. S. Kumar, "A survey on residential demand side management

architecture, approaches, optimization models and methods," *Renew. Sustain. Energy Rev.*, vol. 59, pp. 342–351, 2016.

[15] N. Rahbari-Asr, U. Ojha, Z. Zhang, and M.-Y. Chow, "Incremental welfare consensus algorithm for cooperative distributed generation/demand response in smart grid," *IEEE Trans. Smart Grid*, vol. 5, no. 6, pp. 2836–2845, 2014.

[16] C. Gellings, "Enabling energy efficiency and demand response," *The Smart Grid*, pp. 43–46, 2009.

[17] A. Ipakchi and F. Albuyeh, "Grid of the future," *IEEE Power Energy Mag.*, vol. 7, no. 2, pp. 52–62, 2009.

[18] C. W. Gellings and J. H. Chamberlin, *Demand-side management: concepts and methods*. The Fairmont Press Inc., Lilburn, GA, 1987.

[19] A. G. Tsikalakis and N. D. Hatziargyriou, "Centralized control for optimizing microgrids operation," in *Power and Energy Society General Meeting, 2011 IEEE*. IEEE, 2011, pp. 1–8.

[20] Z. Fu, X. He, T. Huang, and H. Abu-Rub, "A distributed continuous time consensus algorithm for maximize social welfare in micro grid," *J. Frankl. Inst.*, vol. 353, no. 15, pp. 3966–3984, 2016.

[21] T. Zhao and Z. Ding, "Distributed agent consensus-based optimal resource management for microgrids," *IEEE Trans. Sustain. Energy*, vol. 9, no. 1, pp. 443–452, 2018.

[22] X. He, D. W. Ho, T. Huang, J. Yu, H. Abu-Rub, and C. Li, "Second-order continuous-time algorithms for economic power dispatch in smart grids," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 9, pp. 1482–1492, 2018.

[23] M. Nazari-Heris, B. Mohammadi-Ivatloo, G. B. Gharehpetian, and M. Shahidehpour, "Robust short-term scheduling of integrated heat and power microgrids," *IEEE Syst. J.*, vol. 13, no. 3, pp. 3295–3303, 2019.

[24] L. Luo, W. Gu, X.-P. Zhang, G. Cao, W. Wang, G. Zhu, D. You, and Z. Wu, "Optimal siting and sizing of distributed generation in distribution systems with pv solar farm utilized as statcom (pv-statcom)," *Appl. Energy*, vol. 210, pp. 1092–1100, 2018.

[25] H. Xin, Z. Qu, J. Seuss, and A. Maknouninejad, "A self-organizing strategy for power flow control of photovoltaic generators in a distribution network," *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 1462–1473, 2011.

[26] D. E. Olivares, C. A. Cañizares, and M. Kazerani, "A centralized optimal energy management system for microgrids," in *2011 IEEE Power and Energy Society General Meeting*. IEEE, 2011, pp. 1–6.

[27] T. Zhao and Z. Ding, "Distributed initialization-free cost-optimal charging control of plug-in electric vehicles for demand management," *IEEE Trans. Ind. Inform.*, vol. 13, no. 6, pp. 2791–2801, 2017.

[28] S. Z. Tajalli, T. Niknam, and J. Aghaei, "Maximizing social welfare considering the uncertainty of wind power plants using a distributed consensus-based algorithm," in *Electrical Engineering (ICEE), Iranian Conference on*. IEEE, 2018, pp. 1089–1093.

[29] G. Hug, S. Kar, and C. Wu, "Consensus+ innovations approach for distributed multiagent coordination in a microgrid," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1893–1903, 2015.

[30] N. Rahbari-Asr and M.-Y. Chow, "Cooperative distributed demand management

for community charging of phev/pevs based on kkt conditions and consensus networks," *IEEE Trans. Ind. Informat.*, vol. 10, no. 3, pp. 1907–1916, 2014.

[31] R. Deng, Z. Yang, F. Hou, M.-Y. Chow, and J. Chen, "Distributed real-time demand response in multiseller–multibuyer smart distribution grid," *IEEE Trans. Power Syst.*, vol. 30, no. 5, pp. 2364–2374, 2015.

[32] X. Xu, H. Jia, D. Wang, C. Y. David, and H.-D. Chiang, "Hierarchical energy management system for multi-source multi-product microgrids," *Renew. Energy*, vol. 78, pp. 621–630, 2015.

[33] C. Li, X. Yu, W. Yu, G. Chen, and J. Wang, "Efficient computation for sparse load shifting in demand side management," *IEEE Trans. Smart Grid*, vol. 8, no. 1, pp. 250–261, 2017.

[34] Y. Xu, W. Zhang, G. Hug, S. Kar, and Z. Li, "Cooperative control of distributed energy storage systems in a microgrid," *IEEE Trans. Smart Grid*, vol. 6, no. 1, pp. 238–248, 2015.

[35] C. Zhao, J. He, P. Cheng, and J. Chen, "Analysis of consensus-based distributed economic dispatch under stealthy attacks," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5107–5117, 2017.

[36] H. Han, H. Wang, Y. Sun, J. Yang, and Z. Liu, "Distributed control scheme on cost optimisation under communication delays for dc microgrids," *IET Gener. Transm. Dis.*, vol. 11, no. 17, pp. 4193–4201, 2017.

[37] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, "A survey of distributed optimization and control algorithms for electric power systems," *IEEE Trans. Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.

[38] M. H. Amini, S. Bahrami, F. Kamyab, S. Mishra, R. Jaddivada, K. Boroojeni, P. Weng, and Y. Xu, "Decomposition methods for distributed optimal power flow: Panorama and case studies of the dc model," in *Classical and Recent Aspects of Power System Optimization*. Elsevier, 2018, pp. 137–155.

[39] G. Wen, X. Yu, Z.-W. Liu, and W. Yu, "Adaptive consensus-based robust strategy for economic dispatch of smart grids subject to communication uncertainties," *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2484–2496, 2018.

[40] H. F. Habib, C. R. Lashway, and O. A. Mohammed, "A review of communication failure impacts on adaptive microgrid protection schemes and the use of energy storage as a contingency," *IEEE Trans. Ind Appl.*, vol. 54, no. 2, pp. 1194–1207, 2017.

[41] NextEra Energy, Inc, "Nextera energy annual report 2018," http://www.investor.nexteraenergy.com/~/media/Files/N/NEE-IR/reports-and-fillings/annual-reports/NextEra\%20Energy_Annual_Report_2018.pdf, December 2018.

[42] K. Divya and J. Østergaard, "Battery energy storage technology for power systems—an overview," *Electr. Power Syst. Res.*, vol. 79, no. 4, pp. 511–520, 2009.

[43] "Decision on multiple-use application issues," http://docs.cpuc.ca.gov/PublishedDocs/Published/G000/M204/K478/204478235.pdf, November 2018.

[44] Q. Shi, F. Li, Q. Hu, and Z. Wang, "Dynamic demand control for system frequency regulation: Concept review, algorithm comparison, and future vision," *Electr. Power Syst. Res.*, vol. 154, pp. 75–87, 2018.

[45] "Pjm manual 28: Operating agreement accounting," https://www.pjm.com/\~/media/documents/manuals/m28.ashx, December 2019.

[46] B. Xu, A. Oudalov, J. Poland, A. Ulbig, and G. Andersson, "Bess control strategies for participating in grid frequency regulation," *IFAC Proc. Vol.*, vol. 47, no. 3, pp. 4024–4029, 2014.

[47] M. ud din Mufti, S. A. Lone, S. J. Iqbal, M. Ahmad, and M. Ismail, "Supercapacitor based energy storage system for improved load frequency control," *Electr. Power Syst. Res.*, vol. 79, no. 1, pp. 226–233, 2009.

[48] J. Tan and Y. Zhang, "Coordinated control strategy of a battery energy storage system to support a wind power plant providing multi-timescale frequency ancillary services," *IEEE Trans. Sustain. Energy*, vol. 8, no. 3, pp. 1140–1153, 2017.

[49] Y. Cheng, M. Tabrizi, M. Sahni, A. Povedano, and D. Nichols, "Dynamic available agc based approach for enhancing utility scale energy storage performance," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 1070–1078, 2014.

[50] G. He, Q. Chen, C. Kang, and Q. Xia, "Optimal offering strategy for concentrating solar power plants in joint energy, reserve and regulation markets," *IEEE Trans. Sustain. Energy*, vol. 7, no. 3, pp. 1245–1254, 2016.

[51] G. He, Q. Chen, C. Kang, P. Pinson, and Q. Xia, "Optimal bidding strategy of battery storage in power markets considering performance-based regulation and battery cycle life," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2359–2367, 2015.

[52] M. Marzband, M. Javadi, S. A. Pourmousavi, and G. Lightbody, "An advanced retail electricity market for active distribution systems and home microgrid interoperability based on game theory," *Electr. Power Syst. Res.*, vol. 157, pp. 187–199, 2018.

[53] D. P. Chassin, J. C. Fuller, and N. Djilali, "Gridlab-d: An agent-based simulation framework for smart grids," *J. Appl. Math.*, vol. 2014, 2014.

[54] J. R. Vázquez-Canteli and Z. Nagy, "Reinforcement learning for demand response: A review of algorithms and modeling techniques," *Appl. Energy*, vol. 235, pp. 1072–1089, 2019.

[55] T. Krause, E. V. Beck, R. Cherkaoui, A. Germond, G. Andersson, and D. Ernst, "A comparison of nash equilibria analysis and agent-based modelling for power markets," *Int. J. Electr. Power Energy Syst.*, vol. 28, no. 9, pp. 599–607, 2006.

[56] H. Kebriaei, A. Rahimi-Kian, and M. N. Ahmadabadi, "Model-based and learning-based decision making in incomplete information cournot games: A state estimation approach," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 45, no. 4, pp. 713–718, 2014.

[57] Z. Li, Z. Ding, M. Wang, and E. Oko, "Model-free adaptive control for mea-based post-combustion carbon capture processes," *Fuel*, vol. 224, pp. 637–643, 2018.

[58] V. Nanduri and T. K. Das, "A reinforcement learning model to assess market power under auction-based energy pricing," *IEEE Trans. Power Syst.*, vol. 22, no. 1, pp. 85–95, 2007.

[59] E. Lakić, G. Artač, and A. F. Gubina, "Agent-based modeling of the demand-side system reserve provision," *Electr. Power Syst. Res.*, vol. 124, pp. 85–91, 2015.

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

[61] O. Abedinia, N. Amjady, and H. Zareipour, "A new feature selection technique for load and price forecast of electrical power systems," *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 62–74, 2016.

[62] K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, "Short-term load forecasting with deep residual networks," *IEEE Trans. Smart Grid*, 2018.

[63] T. Ouyang, Y. He, H. Li, Z. Sun, and S. Baek, "Modeling and forecasting short-term power load with copula model and deep belief network," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 3, no. 2, pp. 127–136, 2019.

[64] T. Hong and S. Fan, "Probabilistic electric load forecasting: A tutorial review," *Int. J. Forecast.*, vol. 32, no. 3, pp. 914–938, 2016.

[65] A. Ahmad, N. Javaid, M. Guizani, N. Alrajeh, and Z. A. Khan, "An accurate and fast converging short-term load forecasting model for industrial applications in a smart grid," *IEEE Trans. Ind. Informat.*, vol. 13, no. 5, pp. 2587–2596, 2016.

[66] R. Zhang, Z. Y. Dong, Y. Xu, K. Meng, and K. P. Wong, "Short-term load forecasting of australian national electricity market by an ensemble model of extreme learning machine," *IET Gener. Transm. Dis.*, vol. 7, no. 4, pp. 391–397, 2013.

[67] M. Barman, N. D. Choudhury, and S. Sutradhar, "A regional hybrid goa-svm model based on similar day approach for short-term load forecasting in assam, india," *Energy*, vol. 145, pp. 710–720, 2018.

[68] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on lstm recurrent neural network," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 841–851, 2017.

[69] H. Shi, M. Xu, and R. Li, "Deep learning for household load forecasting—a novel pooling deep rnn," *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 5271–5280, 2017.

[70] S. Khan, N. Javaid, A. Chand, A. B. M. Khan, F. Rashid, and I. U. Afridi, "Electricity load forecasting for each day of week using deep cnn," in *Workshops of*

the International Conference on Advanced Information Networking and Applications. Springer, 2019, pp. 1107–1119.

[71] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[72] A. Dedinec, S. Filiposka, A. Dedinec, and L. Kocarev, "Deep belief network based electricity load forecasting: An analysis of macedonian case," *Energy*, vol. 115, pp. 1688–1700, 2016.

[73] G. Fu, "Deep belief network based ensemble approach for cooling load forecasting of air-conditioning system," *Energy*, vol. 148, pp. 269–282, 2018.

[74] D. Liu, L. Zeng, C. Li, K. Ma, Y. Chen, and Y. Cao, "A distributed short-term load forecasting method based on local weather information," *IEEE Systems Journal*, vol. 12, no. 1, pp. 208–215, 2018.

[75] H. Jiang, Y. Zhang, E. Muljadi, Y. Gu, J. J. Zhang, and F. J. Solis, "Load forecasting based distribution system network reconfiguration–a distributed data-driven approach," National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2018.

[76] S. Li, L. Goel, and P. Wang, "An ensemble approach for short-term load forecasting by extreme learning machine," *Appl. Energy*, vol. 170, pp. 22–29, 2016.

[77] B. Wang, S. Zhao, and S. Zhang, "A distributed load forecasting algorithm based on cloud computing and extreme learning machine," *Power System Technology*, vol. 38, no. 2, pp. 526–531, 2014.

[78] H. Mohsenian-Rad, "Optimal bidding, scheduling, and deployment of battery systems in california day-ahead energy market," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 442–453, 2015.

[79] M. Cui, J. Wang, and M. Yue, "Machine learning-based anomaly detection for load forecasting under cyberattacks," *IEEE Trans. Smart Grid*, vol. 10, no. 5, pp. 5724–5734, 2019.

[80] G. Chaojun, D. Yang, P. Jirutitijaroen, W. M. Walsh, and T. Reindl, "Spatial load forecasting with communication failure using time-forward kriging," *IEEE Trans. Power Syst.*, vol. 29, no. 6, pp. 2875–2882, 2014.

[81] D. Ding, Q.-L. Han, Y. Xiang, X. Ge, and X.-M. Zhang, "A survey on security control and attack detection for industrial cyber-physical systems," *Neurocomputing*, vol. 275, pp. 1674–1683, 2018.

[82] N. Biggs, N. L. Biggs, and E. N. Biggs, *Algebraic Graph Theory*. Cambridge university press, 1993, vol. 67.

[83] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge university press, 2004.

[84] R. Bellman, "Dynamic programming," *Science*, vol. 153, no. 3731, pp. 34–37, 1966.

[85] Y. Zheng, J. Zhao, Y. Song, F. Luo, K. Meng, J. Qiu, and D. J. Hill, "Optimal operation of battery energy storage system considering distribution system uncertainty," *IEEE Trans. Sustain. Energy*, vol. 9, no. 3, pp. 1051–1060, 2018.

[86] K. Kim, K. Park, J. Lee, K. Chun, and S.-H. Lee, "Analysis of battery/generator hybrid container ship for co 2 reduction," *IEEE Access*, vol. 6, pp. 14 537–14 543, 2018.

[87] I. U. Nutkani, P. C. Loh, and F. Blaabjerg, "Cost-based droop scheme with lower generation costs for microgrids," *IET Power Electron.*, vol. 7, no. 5, pp. 1171–1180, 2014.

[88] M. A. Abido, "Environmental/economic power dispatch using multiobjective evolutionary algorithms," *IEEE Trans. Power Syst.*, vol. 18, no. 4, pp. 1529–1537, 2003.

[89] A. Farag, S. Al-Baiyat, and T. Cheng, "Economic load dispatch multiobjective optimization procedures using linear programming techniques," *IEEE Trans. Power Syst.*, vol. 10, no. 2, pp. 731–738, 1995.

[90] B. Qu, Y. Zhu, Y. Jiao, M. Wu, P. Suganthan, and J. Liang, "A survey on multi-objective evolutionary algorithms for the solution of the environmental/economic dispatch problems," *Swarm Evol. Comput.*, vol. 38, pp. 1–11, 2018.

[91] H. Shayeghi and M. Mahdavi, "Application of pso and ga for transmission network expansion planning," in *Analysis, Control and Optimal Operations in Hybrid Power Systems*. Springer, 2013, pp. 187–226.

[92] M. Fukushima, "Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems," *Math. Program.*, vol. 53, no. 1-3, pp. 99–110, 1992.

[93] P. Hota, A. Barisal, and R. Chakrabarti, "Economic emission load dispatch through fuzzy based bacterial foraging algorithm," *Int. J. Electr. Power Energy Syst.*, vol. 32, no. 7, pp. 794–803, 2010.

[94] N. P. 2017, "N2ex day ahead auction prices," https://www.nordpoolgroup.com/Market-data1/GB/Auction-prices/UK/Hourly/?view=table.

[95] W. Tasnin and L. C. Saikia, "Performance comparison of several energy storage devices in deregulated agc of a multi-area system incorporating geothermal power plant," *IET Renew. Power Gener.*, vol. 12, no. 7, pp. 761–772, 2018.

[96] PJM Staff, "Implementation and rationale for pjm's conditional neutrality regulation signals," https://www.pjm.com/\~/media/committees-groups/task-forces/rmistf/postings/regulation-market-whitepaper.ashx, January 2017.

[97] Y. Meng, M. Liang, J. F. DeCarolis, and N. Lu, "Design of energy storage friendly regulation signals using empirical mode decomposition," in *2019 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2019, pp. 1–5.

[98] Independent Market Monitor for PJM, "State of the market report for pjm, volume 2: Detailed analysis," https://legalectric.org/f/2020/03/2019-som-pjm-volume2.pdf, March 2020.

[99] Y. Xu, W. Zhang, G. Hug, S. Kar, and Z. Li, "Cooperative control of distributed energy storage systems in a microgrid," *IEEE Trans. Smart Grid*, vol. 6, no. 1, pp. 238–248, 2014.

[100] X. Zhou, J. L. Stein, and T. Ersal, "Battery state of health monitoring by estimation of the number of cyclable li-ions," *Control. Eng. Pract.*, vol. 66, pp. 51–63, 2017.

[101] S. Saxena, Y. Xing, D. Kwon, and M. Pecht, "Accelerated degradation model for c-rate loading of lithium-ion batteries," *Int. J. Electr. Power Energy Syst.*, vol. 107, pp. 438–445, 2019.

[102] B. Xu, A. Oudalov, A. Ulbig, G. Andersson, and D. S. Kirschen, "Modeling of lithium-ion battery degradation for cell life assessment," *IEEE Trans. Smart Grid*, vol. 9, no. 2, pp. 1131–1140, 2016.

[103] W. Ying, Z. Zhi, A. Botterud, K. Zhang, and D. Qia, "Stochastic coordinated operation of wind and battery energy storage system considering battery degradation," *J. Mod. Power Syst. Clean Energy*, vol. 4, no. 4, pp. 581–592, 2016.

[104] G. Li, J. Shi, and X. Qu, "Modeling methods for genco bidding strategy optimization in the liberalized electricity spot market–a state-of-the-art review," *Energy*, vol. 36, no. 8, pp. 4686–4700, 2011.

[105] Z. Li, Z. Ding, and M. Wang, "Operation and bidding strategies of power plants with carbon capture," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3244–3249, 2017.

[106] B. Bahmani-Firouzi and R. Azizipanah-Abarghooee, "Optimal sizing of battery energy storage for micro-grid operation management using a new improved bat algorithm," *Int. J. Electr. Power Energy Syst.*, vol. 56, pp. 42–54, 2014.

[107] R. D. Masiello, B. Roberts, and T. Sloan, "Business models for deploying and operating energy storage and risk mitigation aspects," *Proc. IEEE*, vol. 102, no. 7, pp. 1052–1064, 2014.

[108] W. Liu, P. Zhuang, H. Liang, J. Peng, and Z. Huang, "Distributed economic dispatch in microgrids based on cooperative reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2192–2203, 2018.

[109] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[110] M. Rashid and A. Gupta, "Mathematical model for combined effect of sei formation and gas evolution in li-ion batteries," *ECS Electrochem. Lett.*, vol. 3, no. 10, pp. A95–A98, 2014.

[111] E. Saiz-Marin, J. García-González, J. Barquin, and E. Lobato, "Economic assessment of the participation of wind generation in the secondary regulation market," *IEEE Trans. Power Syst.*, vol. 27, no. 2, pp. 866–874, 2012.

[112] J. Lee, J. Guo, J. K. Choi, and M. Zukerman, "Distributed energy trading in

microgrids: A game-theoretic model and its equilibrium analysis," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3524–3533, 2015.

[113] D. E. Aliabadi, M. Kaya, and G. Sahin, "Competition, risk and learning in electricity markets: An agent-based simulation study," *Appl. Energy*, vol. 195, pp. 1000–1011, 2017.

[114] A. D. Yucekaya, J. Valenzuela, and G. Dozier, "Strategic bidding in electricity markets using particle swarm optimization," *Electr. Power Syst. Res.*, vol. 79, no. 2, pp. 335–345, 2009.

[115] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems*, 2007, pp. 153–160.

[116] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[117] S. Scardapane, D. Wang, M. Panella, and A. Uncini, "Distributed learning for random vector functional-link networks," *Inform Sciences*, vol. 301, pp. 271–284, 2015.

[118] W. Chen and X. Li, "Observer-based consensus of second-order multi-agent system with fixed and stochastically switching topology via sampled data," *Int. J. Robust Nonlinear Control*, vol. 24, no. 3, pp. 567–584, 2014.

[119] B. Liu, Z. Ding, and C. Lv, "Distributed training for multi-layer neural networks by consensus," *IEEE Trans. Neural Netw. Learn. Syst.*, 2019.

[120] R. Hu, S. Wen, Z. Zeng, and T. Huang, "A short-term power load forecasting model based on the generalized regression neural network with decreasing step fruit fly optimization algorithm," *Neurocomputing*, vol. 221, pp. 24–31, 2017.

[121] C. Guan, P. B. Luh, L. D. Michel, Y. Wang, and P. B. Friedland, "Very short-term load forecasting: wavelet neural networks with data pre-filtering," *IEEE Trans. Power Syst.*, vol. 28, no. 1, pp. 30–41, 2012.

[122] M. Kerai, "Smart Meters, Quarterly Report to end September 2019, Great Britain," https://www.gov.uk/government/collections/smart-meters-statistics, 11 2019, accessed 2019.

[123] W. Ren and R. W. Beard, *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*, 1st ed. Springer Publishing Company, Incorporated, 2007.