

# Hierarchical Optimization of Smart Grids with Energy Storage Systems - A Model-Predictive Control and Auction-based Method

Chen-Chou Tsai

A Thesis Submitted to Institute of Computer Science and  
Information Engineering College of Engineering National  
Chung Cheng University for the Degree of Master in  
Computer Science and Information Engineering

June 2014

## Abstract

A smart grid is a modernized electrical grid that introduces a two-way communication where electricity and information can be exchanged between the utility company and micro-grids. Small scale generators constitute renewable energy resources including photovoltaic (PV), wind turbine, and fuel cell which are usually used to maintain the loads in a micro-grid. But intermittency of the generators caused by the unstable weather conditions reduced power quality, which can be improved by energy storage systems. Further, we can ease the stress of the utility company and reduce electricity cost by appropriately using energy storage systems (ESS) during peak electricity usage. Nevertheless, the use of ESS in micro-grids has introduced challenges of its own such as the prediction of electricity usage/generation, scheduling and control, battery lifetime, etc.

To address the ESS issues in this Thesis, we propose a Model-Predictive Control (MPC)-based scheduling method for ESS in a micro-grid. By using a high accuracy load prediction model, we can effectively charge/discharge when and what amount of energy as required. Through a time window-based optimization, the proposed MPC-based scheduling for ESS increases cost reduction of electricity in a micro-grid by taking amount of prediction power required by loads, amount of prediction power supplied by generators, charge/discharge operations for ESS, and dynamic electricity price declared by the utility company into consideration. Further, we present the trade off between cost reduction of electricity and lifetime of ESS. A multi-agent system is used to model a micro-grid. A micro-grid intelligent agent (MIA) can participate in the electricity bidding market which works via an auction mechanism.

Experiments show that the MPC-based scheduling method for ESS gives the highest cost reduction of 3.4% compared to other ESS strategies. Through bidding market, we can achieve an average cost saving of 35.25% with the first-price sealed auction and 34.86% with the second-price sealed auction.

**Keywords:** *Smart Grid, Model-Predictive Control, Energy Storage System Scheduling, Auction, Cost Reduction*

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Background . . . . .	4
1.1.1	Architecture of Smart Grid . . . . .	4
1.1.2	Energy Storage System . . . . .	4
1.1.3	Prediction model . . . . .	7
1.1.4	Example . . . . .	7
1.2	Motivation . . . . .	8
1.3	Thesis Organization . . . . .	9
<b>2</b>	<b>Related Work</b>	<b>10</b>
2.1	Multi-agent system modeling with energy storage system of smart grid .	10
2.2	Energy storage system optimization . . . . .	16
2.2.1	Energy Storage System Scheduling . . . . .	16
2.2.2	Prediction method . . . . .	17
<b>3</b>	<b>Preliminaries</b>	<b>19</b>
3.1	Terminology . . . . .	19
3.2	System Model . . . . .	20
3.3	Assumptions . . . . .	23
3.4	Problem Formulation . . . . .	23
3.5	Load Prediction Parameters . . . . .	24
3.6	Parameter for Model Predictive Control . . . . .	25
3.7	Parameter for the Bidding . . . . .	25
<b>4</b>	<b>MPC-based Scheduling</b>	<b>27</b>
4.1	Load/Generation Predictor Design . . . . .	27

4.1.1	Linear-regression based prediction method . . . . .	28
4.1.2	Computing the component temperatures . . . . .	28
4.1.3	Linear-regression based prediction framework . . . . .	34
4.2	Model-Predictive Control Scheduler in Micro-grid Level . . . . .	37
4.2.1	Model-predictive control scheduling on ESS . . . . .	37
4.2.2	Overall MPC scheduling framework . . . . .	39
4.3	Bidding Market at the Smart Grid Level . . . . .	42
4.3.1	Two traditional auction mechanisms . . . . .	43
4.3.2	Bidding Procedure . . . . .	43
<b>5</b>	<b>Experimental Results</b>	<b>48</b>
5.1	Experiment setup . . . . .	48
5.1.1	Experiment environment . . . . .	48
5.1.2	Mathematical programming solver . . . . .	49
5.1.3	Demand load data, generation prediction, specification of ESS, and dynamic electricity price . . . . .	49
5.1.4	Comparison with other ESS strategies . . . . .	50
5.2	Evaluation of linear regression-based load prediction . . . . .	54
5.3	ESS Scheduling Experiment . . . . .	58
5.4	Trade-off between Cost Reduction and Lifetime of ESS . . . . .	61
5.5	MPC-based scheduling with bidding market . . . . .	63
<b>6</b>	<b>Conclusions and Future Work</b>	<b>66</b>

# List of Figures

1.1	The traditional power grid . . . . .	2
1.2	The Smart Grid . . . . .	3
1.3	The Smart Grid model . . . . .	5
1.4	Without energy storage system. . . . .	6
1.5	With energy storage system. . . . .	7
1.6	Power output of microgrid . . . . .	8
2.1	Agent architecture with two micro-grids in a smart grid [1]. . . . .	12
2.2	15 minute auction flow. . . . .	14
2.3	Grid-connected system with two microgrids [2]. . . . .	15
2.4	MPC-based scheduling. . . . .	17
3.1	Three-level hierarchical smart grid model. . . . .	22
4.1	Load predictor framework. . . . .	29
4.2	Generation predictor framework. . . . .	30
4.3	Flow diagram of computing component temperatures. . . . .	32
4.4	Linear-regression based prediction. . . . .	35
4.5	MPC-Scheduler framework. . . . .	38
4.6	Optimization sliding window in MPC. . . . .	39
4.7	Flow diagram of the overall MPC framework. . . . .	40
4.8	Flow diagram of bidding market. . . . .	42
4.9	Flow diagram of the overall bidding market framework. . . . .	44
5.1	Dynamic electricity price of utility company in a day. . . . .	50
5.2	SoC in the exhausted method in a day. . . . .	51
5.3	The request of a micro-grid in a day. . . . .	52

5.4	The SoC of ESS in a day with backup strategy. . . . .	52
5.5	The demand load of a micro-grid in a day. . . . .	53
5.6	The SoC of ESS in a day with FR strategy. . . . .	53
5.7	Inaccuracy of residential area. . . . .	55
5.8	Inaccuracy of industrial area. . . . .	57
5.9	Cost reduction for different optimization time horizon. . . . .	58
5.10	24 intervals with 10 hours MPC optimization time horizon. . . . .	60
5.11	48 intervals with 10 hours MPC optimization time horizon. . . . .	60
5.12	96 intervals with 10 hours MPC optimization time horizon. . . . .	60
5.13	Scheduling time for different MPC optimization time horizons. . . . .	61
5.14	Cost reduction on electricity bill with different ESS management strategies. . . . .	62
5.15	LoL of different ESS strategies. . . . .	63
5.16	Cost saving with MPC-based scheduling of ESS with the first-price and the second-price sealed auctions. . . . .	64

# List of Tables

4.1	Example of relationship between $N_{itvl}$ and $L_{itvl}$ . . . . .	28
4.2	Example for temperature bounds. . . . .	31
4.3	Example for computing component temperatures. . . . .	34
4.4	Example for deciding the buyer and the seller before auction. . . . .	44
4.5	Example for adjusting <i>Request</i> after auction. . . . .	45
4.6	Example for shifting the priority of seller after auction. . . . .	45
5.1	Environment of experiment . . . . .	48
5.2	Daily 15-min demand load data of residential and industrial area. . . . .	49
5.3	Penetration for each generator module. . . . .	50
5.4	The specification of ESS. . . . .	50
5.5	Parameter settings for linear regression-based load prediction. . . . .	54
5.6	IR of residential area. . . . .	56
5.7	IR of industrial area. . . . .	56
5.8	Parameter settings for MPC-based scheduling of ESS. . . . .	58
5.9	Number of intervals required for MPC-based scheduling with different optimization time horizons. . . . .	59
5.10	Parameter settings for trade-off experiment. . . . .	61
5.11	Charge/Discharge time for all ESS strategies. . . . .	62
5.12	Parameter settings of bidding market. . . . .	64
5.13	The electricity bill of different auction mechanism for all MIAs. . . . .	65
5.14	Amount of surplus power and shortage power for all MIAs. . . . .	65

# Chapter 1

## Introduction

A smart grid is a modernized electrical grid that brings utility electrical delivery systems into the 21st century. It is a developing network of communications, controls, computers, automation and new technologies working together [3]. Also, the Smart Grid introduces a two way dialogue where electricity and information can be exchanged between the utility and its customers.

The Smart Grid is expected to exhibit the following important features:

- **Reliability:** Smart Grid will use some technologies to improve fault detection and allow self-healing of the network without the intervention of technicians. This will ensure more reliable supply of electricity, and reduce losses against natural disasters or power disturbance events.
- **Market-enabling:** Smart grid allows for systematic communication between suppliers and consumers. The suppliers will declare energy prices to consumers, while the consumers will be able to be more strategic in when they use energy. This is called *demand response*.
- **Security:** Smart Grid is designed to allow real-time contact between utilities and meters in customers' homes and businesses, there is a risk that these capabilities could be exploited for cyber-attack or even terrorist actions. Thus, measures are usually taken to ensure security in smart grids.
- **Smart power generation:** It is a concept of matching electricity production with demand using multiple identical generators and storages, each of which can start,

stop and operate efficiently at chosen load, independently of the others, thus making them suitable for base load and peak power generation.

- Platform for advanced services: With the development of technology and various requirements of consumers, smart grid is expected to support new products, services and markets.
- Efficiency: All of the assets in smart grid should be optimized and operate efficiently.

The framework of traditional power grid is as shown in Figure 1.1. The power grid was designed for utilities to deliver electricity to consumers and then bill them once a month or once every two months. This limited one-way interaction makes it difficult for the grid to respond to the ever-changing and rising energy demands of the 21st century.

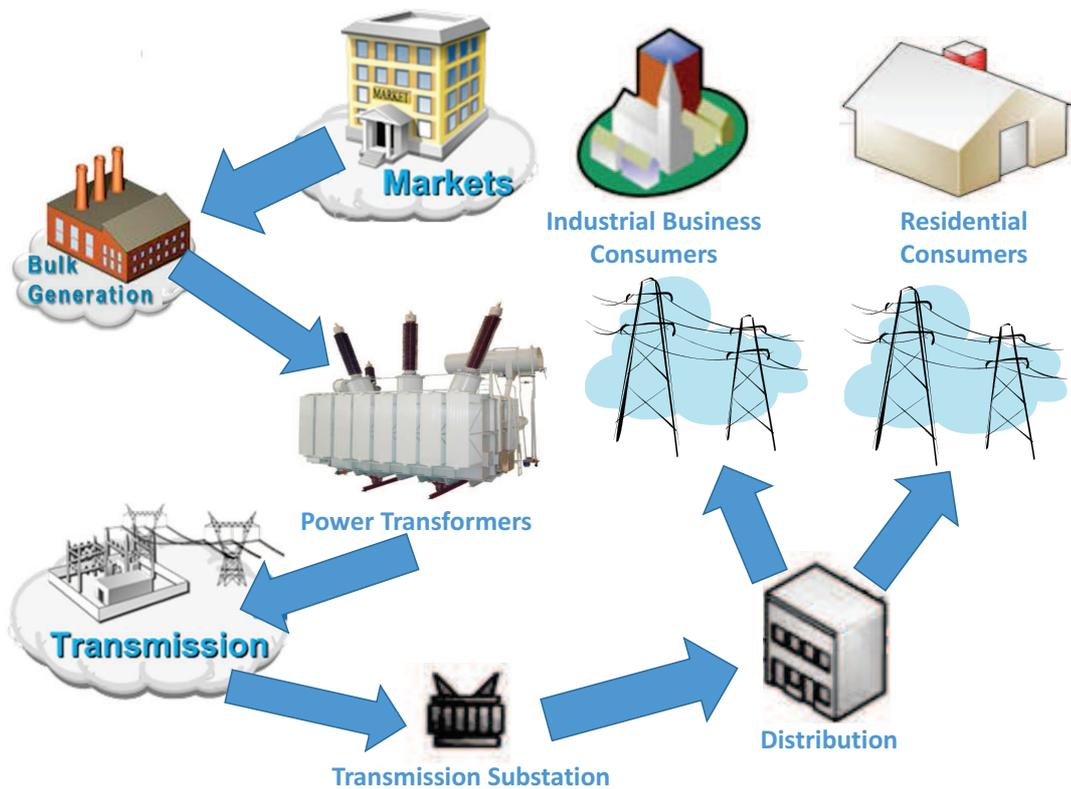


Figure 1.1: The traditional power grid

The framework of smart grid is as shown in Figure 1.2. Because the Smart Grid can provide two-way communication, many things can be easily realized, for example, the usage of electrical devices can be easily monitored by consumers.

The Smart Grid has many more benefits than the traditional power grid for our daily life, however its design still needs to be improved. There are still some problems existed in the Smart Grid, such as the high cost on electricity bill for consumers,  $CO_2$  pollution due to the high request for power generation of electricity company, and the single power generation source and fixed electricity price for user. Thus, we focus on the following issues in this Thesis:

- To make the Smart Grid more cost-effective.
- To reduce the  $CO_2$  pollution.
- To provide more information and automatic tools to the consumers to control the costs or the ways they use the electricity.

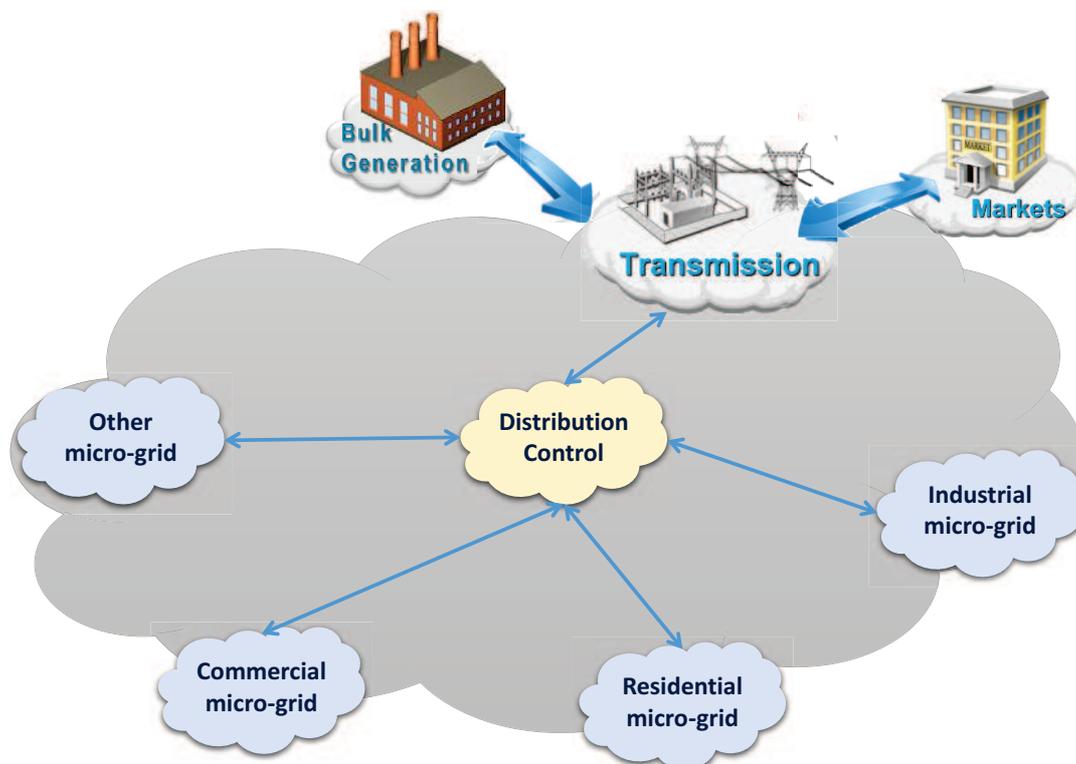


Figure 1.2: The Smart Grid

# 1.1 Background

## 1.1.1 Architecture of Smart Grid

A Smart Grid is made up of central generation, many micro-grids, and the distribution grid for electricity which links the above components. Figure 1.3 shows a general architecture of the Smart Grid. Consisting of a centralized generation, a distributed grid, and micro-grids. The central generation is the main supplies of electricity and the can support the demand of most loads in the micro-grids. Example of central generation is a nuclear power plant and thermal power plant. The grid on network electricity distribution allows bidirectional power flow between micro-grids.

A micro-grid is made up of some distributed energy resources (DER) such as micro-turbines, and renewable resources power, loads, and energy storage system (ESS). The concept of micro-grids was proposed by Lasseter- [4]. Micro-grids could be in grid-connected mode or in islanded mode. In the grid-connected mode, a single micro-grid can exchange electricity to other micro-grids or the central generation according to different situations of load balancing. In the islanded mode, the electricity supply from other micro-grids or the central generation is disabled. A micro-grid enters islanded mode either intentionally or unintentionally. Intentional islanding is due to unstable conditions outside the micro-grid such as high demands from some external power loads. Unintentional islanding occurs when there is some unexpected problem in the distribution grid of the smart grid leading to isolation of one or more micro-grids in islanded mode. There are some general issues in islanded mode:

- It is required to manage Critical/Non-critical<sup>1</sup> loads to available generation.
- It is required to optimize island operation for longevity.

## 1.1.2 Energy Storage System

Most natural resources will be consumed very soon if we do not change the way we use them [5][6]. Therefore, we look forward to renewable resources such as solar power generation, wind power generation, and hydroelectric power generation, which will not

---

<sup>1</sup>Critical loads represents the devices that cannot be shut down or be could damage the important data or harmful to human being, in the contract, Non-critical loads can be arbitrarily cut off.

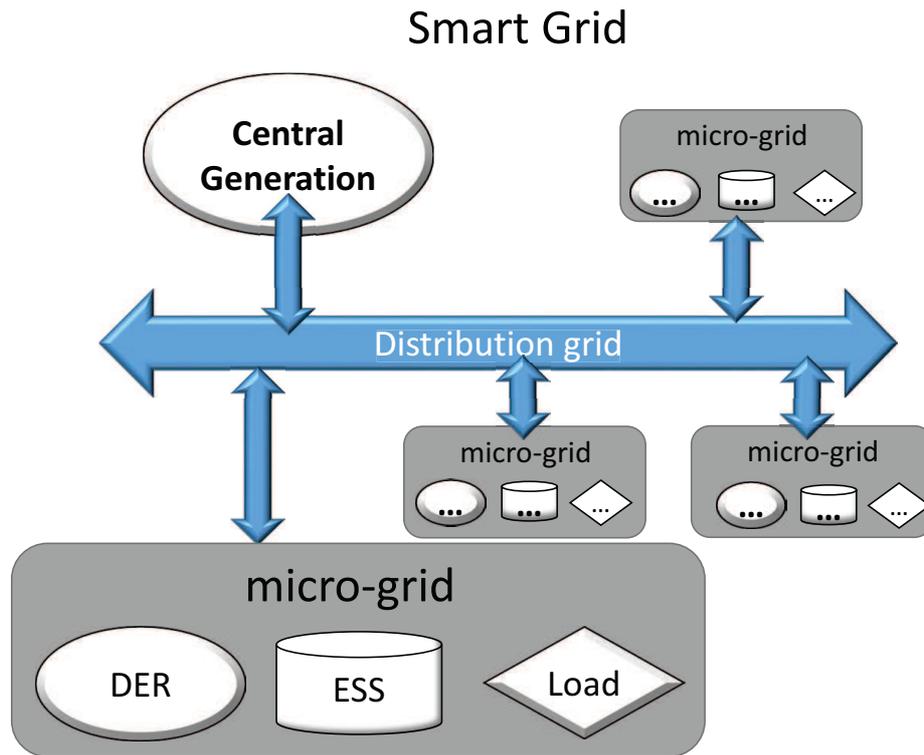


Figure 1.3: The Smart Grid model

only reduce energy consumption of fuel, with decrease in the emission of  $CO_2$  but will also save the world at the same time. However there is a problem of intermittency introduced by renewable energy resources. For example, the solar power generation may not work regularly through sunrise and sunset, such as on a cloudy day or a rainy day, when the production of electricity could be very unstable. As for wind power generation and hydroelectric power generation, the lack of wind and water will directly affect the production of electricity. As a solution, energy storage systems can be used to enhance power quality and improve controllability of power flow. Further, with the development of technology, more and more people are willing to set up some local small-scale renewable energy generation equipment that can ease the stress of electricity generation by utility company. Unexpected black-outs could thus be prevented and sustainable power generation can be guaranteed on a larger scale. But the ESS has some potential constraints such as over-charge or over-discharge for ESS will shorten the lifetime<sup>2</sup> itself, the high cost on updating new ESS equipments, and the lower electricity generation capacity cannot maintain the request for residential area, commercial area, and industrial area.

<sup>2</sup>Lifetime of ESS represents the number of charge/discharge cycles before the ESS fails.

Figure 1.4 and 1.5 respectively show how load balancing is performed without and with the energy storage systems. The demand load is fulfilled by the distributed energy resources (DER) in the micro-grid and utility company (Grid) outside the micro-grid without the ESS, in the contract, the power supply from utility company can be reduced with the using of ESS. When ESS is present in a smart grid, there are other issues such as how to increase the lifetime of battery and in a meantime increase the benefit of electricity bill. As a result, We have to schedule an energy storage system according to above issues.

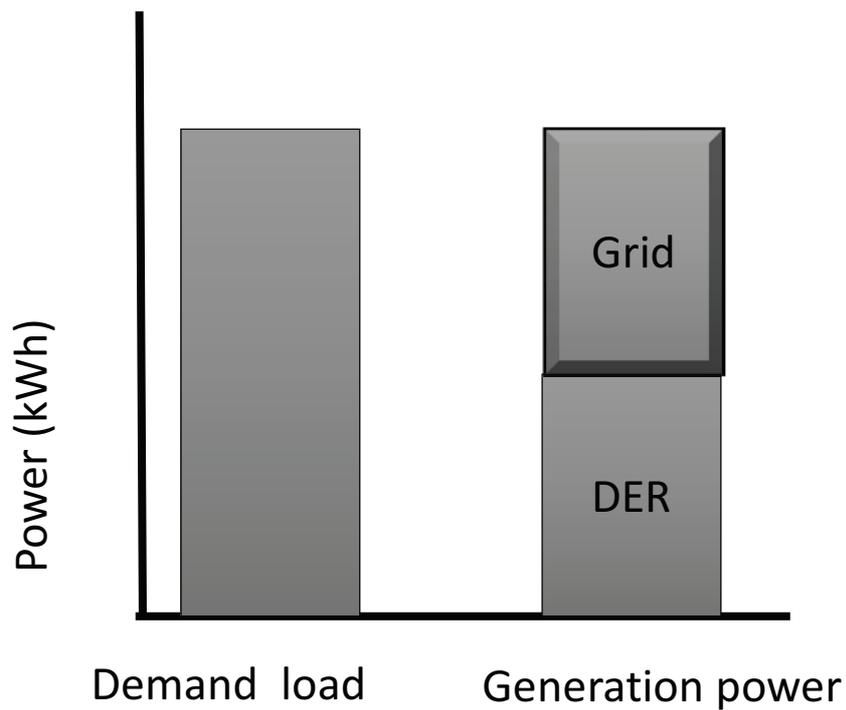


Figure 1.4: Without energy storage system.

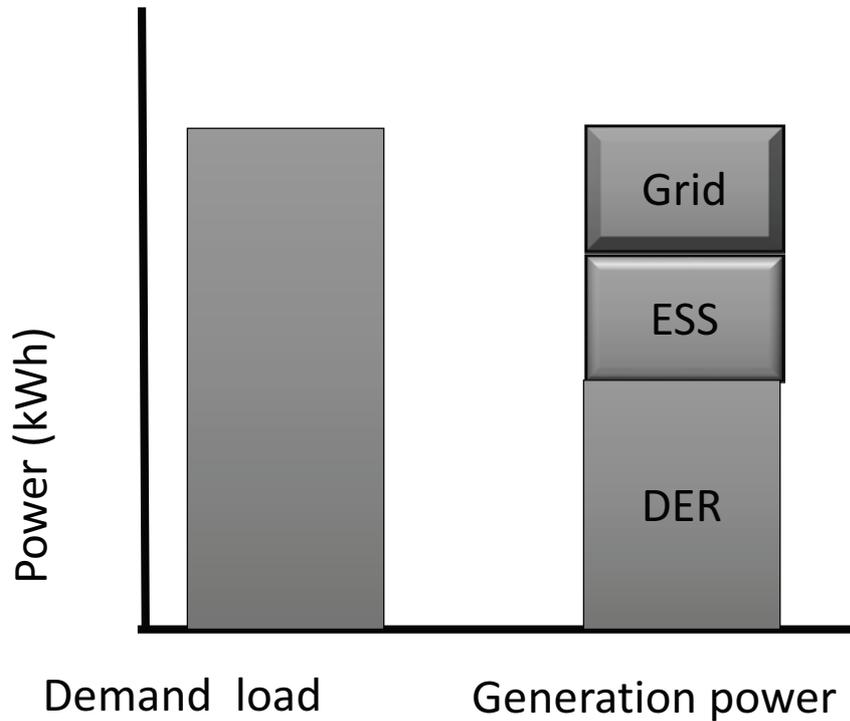


Figure 1.5: With energy storage system.

### 1.1.3 Prediction model

In order to effectively use ESS, the batteries need to be charged and discharged at appropriate time points such that excess energy is stored and stored energy is consumed at most beneficial period of time by appropriately chosen power loads. However, for such scheduling of battery charge/discharge to be effective, we need to predict when there is excess energy and when and what amount of energy required. Thus, prediction models are required for both generation (excess energy) and loads (energy demands). A general prediction model can be constructed from historical data, including power generation data and power load data. Accuracy of prediction models is very important for effective scheduling of batteries because an inaccurate prediction could result in wrong scheduling decisions, consulting in excess energy wanted (all batteries are fully charged ) or energy demands cannot be met (not enough energy stored in batteries).

### 1.1.4 Example

Let us look at an example [7], where we have historical data for one day. The micro-grid, including load demands, battery source, wind energy and distributed generators.

As shown in Figure 1.6, during 8th hour to the 20th hour of the day, demand loads are higher than the rest of the day. In other words, during this peak demand time slot, the price of electricity from the utility company would be higher. In such a period of high price, not only is the stress of generators increased, but the bills of consumers are also quite high. However, if we can schedule the energy storage system for each micro-grid in advance, the smart grid could be more cost-saving and energy-efficient.

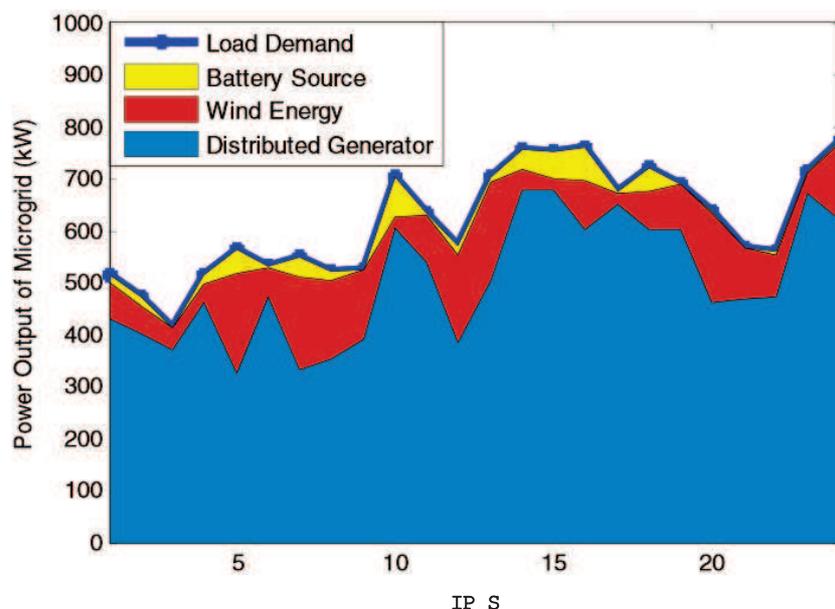


Figure 1.6: Power output of microgrid

## 1.2 Motivation

We will explain the motivation of this Thesis in this section. In order to reduce the cost on electricity bill with the use of ESS in the micro-grid, we have two goals as follows. The first goal is to propose a scheduling algorithm for the energy storage system in micro-grids, such that the demands for electricity and the electricity price from micro-grids can be considered, while a trade-off between the battery lifetimes and cost-saving for consumers is achieved. The second goal is to prepare suitable generation/load prediction methods and battery model to support the first goal. Case studies will be presented to illustrate how these two goals are achieved.

## 1.3 Thesis Organization

The rest of the Thesis is organized as follows. In Chapter 2, we introduce the related work on smart grid model, battery model, prediction method, and scheduling algorithm for batteries. In Chapter 3, we give some definitions used in our proposed models and algorithm, and then define some assumptions. Chapter 4 is the core of our Thesis, in which we describe the overall smart grid model and the details of the proposed scheduling algorithms and prediction method. In Chapter 5, we analyze the experimental results and present some case studies. In Chapter 6, we conclude the Thesis and give some future work.

# Chapter 2

## Related Work

In this chapter, we review some existing approaches for smart grid designs based on multi-agent systems (MAS), energy storage system optimization with load prediction methods, and energy storage scheduling.

### 2.1 Multi-agent system modeling with energy storage system of smart grid

Managing distributed resources in a smart grid with multiple micro-grids requires reliable and intelligent energy-management tools [8]. Smart grids are often modeled as a multi-agent system as evidenced by numerous literatures [1] [9] [2] [10] [11]. In MAS, each agent can manage the behaviours of individual unit mostly autonomously in a cooperative environment and the controls of various agents can implement asynchronously and in parallel.

A popular multi-agent system platform called Java Agent DEvelopment framework (JADE) [12] is often used to implement MAS design. JADE conforms to the IEEE's standard on Foundation for Intelligent Physical Agents (FIPA) [13], which helps to ensure interoperability among different systems and platforms that constitute a MAS design.

When a smart grid is modeled using multi-agent system [1] [9] [2], some common agents are as described in the following.

- Smart Grid Agent (SGA): A smart grid agent collects the power demand and sup-

ply information from micro-grid agents and automatically distributes electricity in the following way in order to secure loads.

- Among two or more micro-grid agents
- Micro-grid agent to utility agent
- Micro-Grid Agent (MGA): A micro-grid agent has some load agents (LA), generation agents (GA), and storage agents (SA). The responsibility of MGA is to aggregate inner load balance status periodically, then report to SGA. Two possible conditions could occur in each period.
  - Buy electricity: Amount of power required by loads is greater than the amount of power generated .
  - Sell electricity: Amount of power required by loads is less than the amount of power generated.
- Load Agent (LA): A load agent represents the electricity demand. In general, loads can be classified into residential area users, commercial area users and industrial area users.
- Generation Agent (GA): A generation agent represents a power generation equipment such as diesel generator, photovoltaic generation, wind turbine, and fuel cell.
- Storage Agent (SA): A storage agent represents a battery bank. A SA could represent a power load when the battery is charging and a power generator when the battery is discharging.
- Utility Agent (UA): A utility agent represents a massive generation unit. It is assumed that the UA can support most loads from all micro-grids. Moreover, UA declares time-of-use rate or hourly price of electricity price to the SGA periodically.

In [1], authors point out that the issue of supply-demand mismatch exists if generation sources are not enough to satisfy the load demand inside a micro-grid, and thus an agent-based energy-management system was proposed. The combination of

distributed resources and distributed storage brings more reliability into such systems. The proposed architecture is shown in Figure 2.1, which consist of 5 different agents as described in the following.

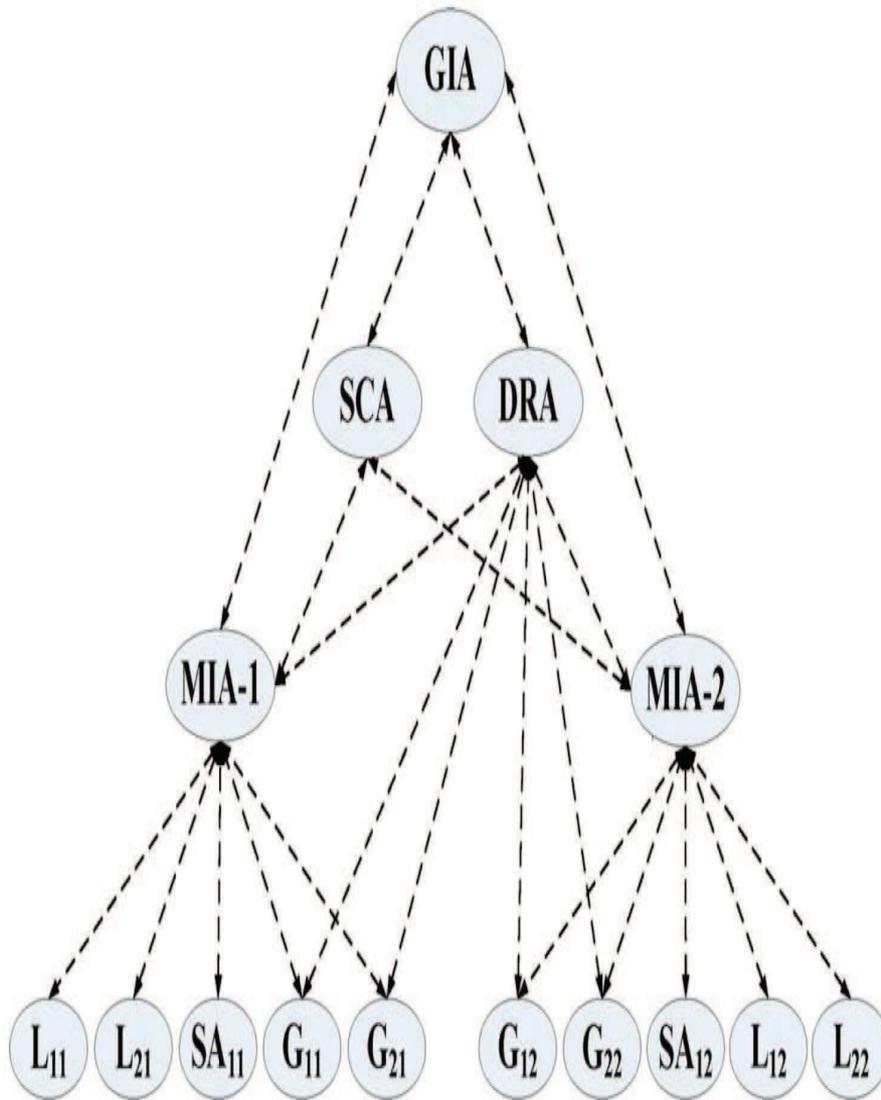


Figure 2.1: Agent architecture with two micro-grids in a smart grid [1].

- Global Intelligent Agent (GIA): GIA is similar to SGA, the GIA coordinates storage cluster agent (SCA), demand response agent (DRA), and micro-grid intelligent agent (MIA) to conduct global auction.
- Storage Cluster Agent (SCA): SCA participates in the global market on behalf of SAs.
- Demand Response Agent (DRA): DRA creates a bilateral negotiation with GAs.
- Micro-grid Intelligent Agent (MIA): MIA is similar to MGA, also MIA is responsible for conducting auction among local agents and updating the priorities of loads.
- Load (L) and Generation (G) agent: The load and generation agents collect their owner power information and inform MIA.
- Storage Agent (SA): The SAs represent the storage systems in a micro-grid. These agents are responsible for maintaining the storage system such that the state-of-charge is within the range  $(SoC_{Max}, SoC_{Min})$ , where  $SoC_{Max}$  represents the upper bound SoC for ESS and  $SoC_{Min}$  represents the lower bound SoC for ESS.

The auction for power-exchange is performed in intervals of 5, 15, 30 and 60 minutes. For example, if a 15 minute interval is considered for auction a day would be divided into 96 intervals (or block). In each interval, the initial 5 minutes is considered an auction period, as shown in Figure 2.2. The result of this auction is then implemented in the next interval.

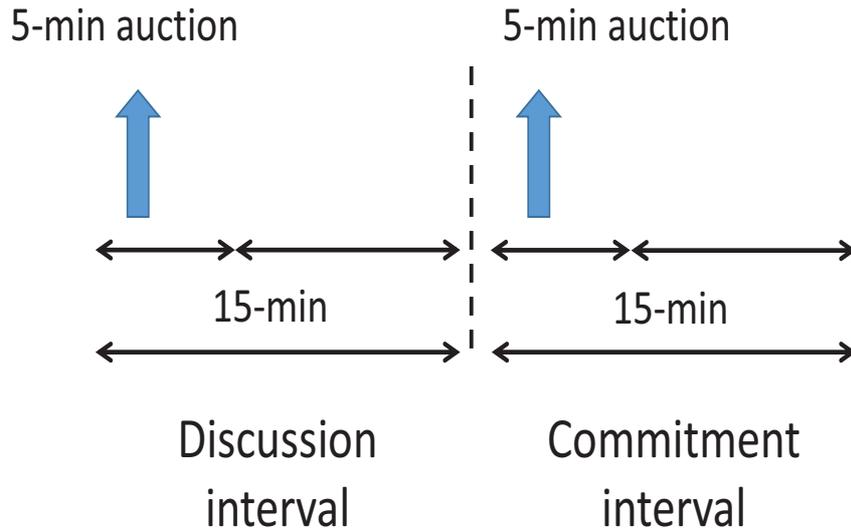


Figure 2.2: 15 minute auction flow.

Dou and Liu [2] proposed a multi-agent based hierarchical hybrid control architecture as shown in Figure 2.3. This architecture consists of three-level agents as described in the following.

- The upper level energy management agent: This agent is similar to SGA. The upper level agent is responsible for energy management strategy with multi-objective optimization (MOO) targeted at minimizing cost, pollution emission, and network loss. Tasks for generating electricity are then assigned to the unit agents of distributed energy resources through the middle level agent.
- The middle level coordinated control agent: The voltage in a micro-grid must be maintained by all distributed energy resources. Since a micro-grid system usually runs under different conditions to meet the changes in load demands, the suitable switch operation should be done so as to adapt to the different changes and maintain the micro-grid voltage level within a safety range.
- The lower level unit control agent: This agent works individually for micro-source, energy storage, renewable energy resources and load demand and ensures the control commands from the upper level are realized appropriately.

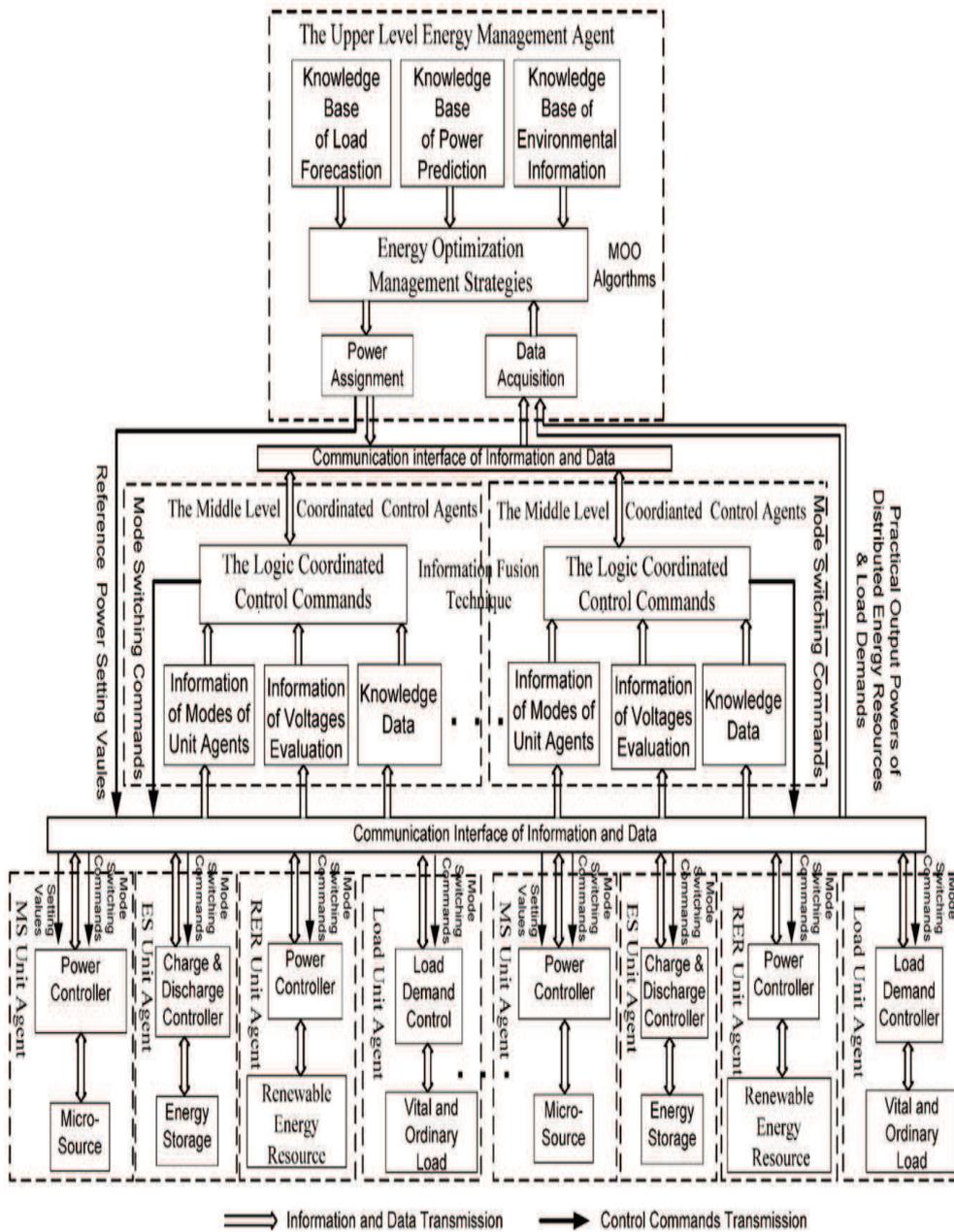


Figure 2.3: Grid-connected system with two microgrids [2].

## 2.2 Energy storage system optimization

For energy management in ESS, some common objectives are as follows:

- Improvement of energy efficiency.
- Extension of battery lifetime.
- Compliance to the constraints of energy storage modules, for example, never over-charge and over-discharge.

For customers, an ESS should not only handle the dynamically changing load demands and the unintentional black-out of the microgrid, but should also reduce the electricity cost via a smart exchange of power generation from other generating units. Nevertheless, there is a trade-off between the lifetime of battery and the cost reduction benefit for customers. A higher usage of the battery module in ESS, the larger is the reduction in electricity cost for customers increases, but the lifetime of battery module decreases. The opposite also holds.

### 2.2.1 Energy Storage System Scheduling

Scheduling for ESS is mainly to control the state-of-charge (SOC) of the battery. In addition, each energy storage module is characterized by its rate-of-charge, and rate-of-discharge. For example, a lead-acid battery is generally recommended to operate with its SOC maintained above 20% so as not to damage the battery. Tran et al. [14] [15] used Stochastic Dynamic Programming (SDP) to perform energy storage scheduling, targeting at three objectives including energy efficiency improvement, lifetime extension and compliance with constraints of battery module. The authors also proposed a model of battery lifetime, called Peukert Lifetime Energy Throughput (PLET) model that applies Peukert's Law for cycle life and depth-of-discharge (DOD) of battery. The advantage of this model is that it can calculate the loss of life during the scheduling even with incomplete half cycle or incomplete full cycle. In this model, value of PLET ( $C_{PLET}$ ) is calculated as:

$$C_{PLET} = d^{k_p} n \quad (2.1)$$

where  $d$  is the DOD (%),  $k_p$  is the Peukert Lifetime constant, and  $n$  is the number of cycles. For any DOD, the total PLET throughout the battery life ( $C_{PLET}^{life}$ ) is almost

constant. As a result, cumulative PLET ( $c_{PLET}^t$ ) stands for the battery life loss during usage at different DOD in the time interval  $t$ . Then Loss-of-Life ( $LoL$ ) of battery is calculated as:

$$LoL(\%) = \frac{c_{PLET}^t}{C_{PLET}^{life}} \times 100\% \quad (2.2)$$

Chen et al. [16] used the model predictive control (MPC) method for scheduling optimization with the objective for minimizing the energy cost. The MPC method is employed in scheduling, as shown in Figure 2.4. The time-line is divided into slots for scheduling, slot has a duration of  $\Delta$  minutes. At each time slot  $t$ , the MPC process gets the current electricity price and forecasts power information (amount required and generated) from slot  $t+1$  to  $t+(N-1)$ . The MPC process solves the optimization problem over this  $N$ -slot horizon time and applies it to time slot  $t$ . At time  $t+1$ , the process updates information for the next  $N$ -slot time and solves the optimization problem again for time slot  $t+1$ .

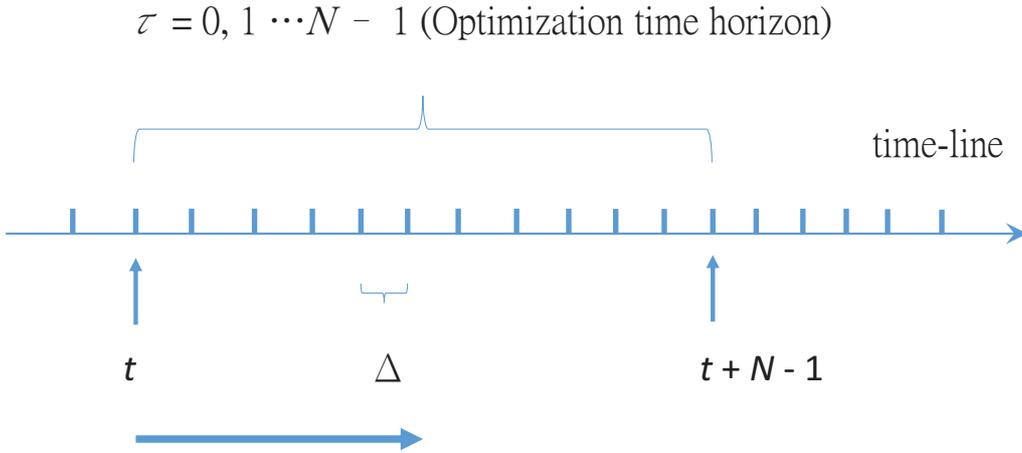


Figure 2.4: MPC-based scheduling.

## 2.2.2 Prediction method

Accurate prediction and modeling of the uncertainties associated with demand load has always been an important issue in smart grids. Li and Jayaweera [17] proposed two types of approaches to model customer load demand. The first one is based on a first order non-stationary Markov chain. The advantage of this advantage is that it can estimate how the distribution of load demands evolves over time and the disadvantage is that there is not enough immediate historical load data. The second method is based on

the time series analysis technique, that is, time-varying autoregressive (TVAR) process. The advantage of this method is that it significantly increases the prediction efficiency with less coefficient estimations, and the disadvantage is that it does not consider the day of the week, month and seasonal effects. Mathieu et al. [18] proposed a regression-based electricity load model that uses a time-of-week indicator variable and continuous temperature dependence. The advantage of this method is that when constructed appropriately, it provides a good fit to load data, and the results are easy to interpret, modify and compute.

# Chapter 3

## Preliminaries

In this chapter, we give definitions used in this Thesis including the terminology, a system model for smart grid, the proposed load prediction method, and the MPC-based scheduling algorithm for ESS. The terminology includes the main terms used in this Thesis. A model is proposed for smart grid that supports multiple micro-grids and use a global bidding mechanism for buying/selling electricity. A regression-based load prediction method is proposed based on 1) a time-of-week indicator variable, and 2) piecewise linear and continuous temperature. A model-predictive control (MPC)-based scheduling method is proposed that uses the predicted demand load, forecasted power generation, and the dynamic electricity prices along a time horizon of  $N$  time slots such that a scheduling operation for ESS that is optimal for the next time is found. We also give some assumptions and formulate the target problem in this Chapter. The details are explained as follows.

### 3.1 Terminology

The terminologies used in this Thesis are introduced as follows.

1. Energy Storage System (*ESS*): It is the energy storage equipment in micro-grid. The operation mode of *ESS* can be 1) charge mode, 2) no action, 3) discharge mode.
2. State-of-Charge (*SoC*): It is the equivalent of fuel gauge for *ESS*. Take for example,  $SoC = 100\%$  when *ESS* is fully charged. In other words,  $SoC = 0\%$  when *ESS* is fully discharged.

3. Rate-of-Charge (*ROC*) and Rate-of-Discharge (*ROD*): They are denoted in  $C$ -rate unit. Take for example,  $0.2C$  is equal to the 20% capacity of the energy storage charged/discharged in 1 hour.
4. Demand load (*DL*): It represents the demand for electricity in the micro-grid. Different types of consumers for electricity demand vary greatly. Also, these demands have different variations within one day.
5. Generation power (*Gen*): It represents the amount of power generated from PV, wind turbine, fuel cell, or other generators in the micro-grid.
6. Interval: A day can be cut into many intervals. Take for an example, a day can be cut into 24 intervals, where each interval is of 60 minutes.
7. N-sliding window: At any interval  $t$ , forecast data from  $t+1$  to  $t+N$  are taken into consideration.

## 3.2 System Model

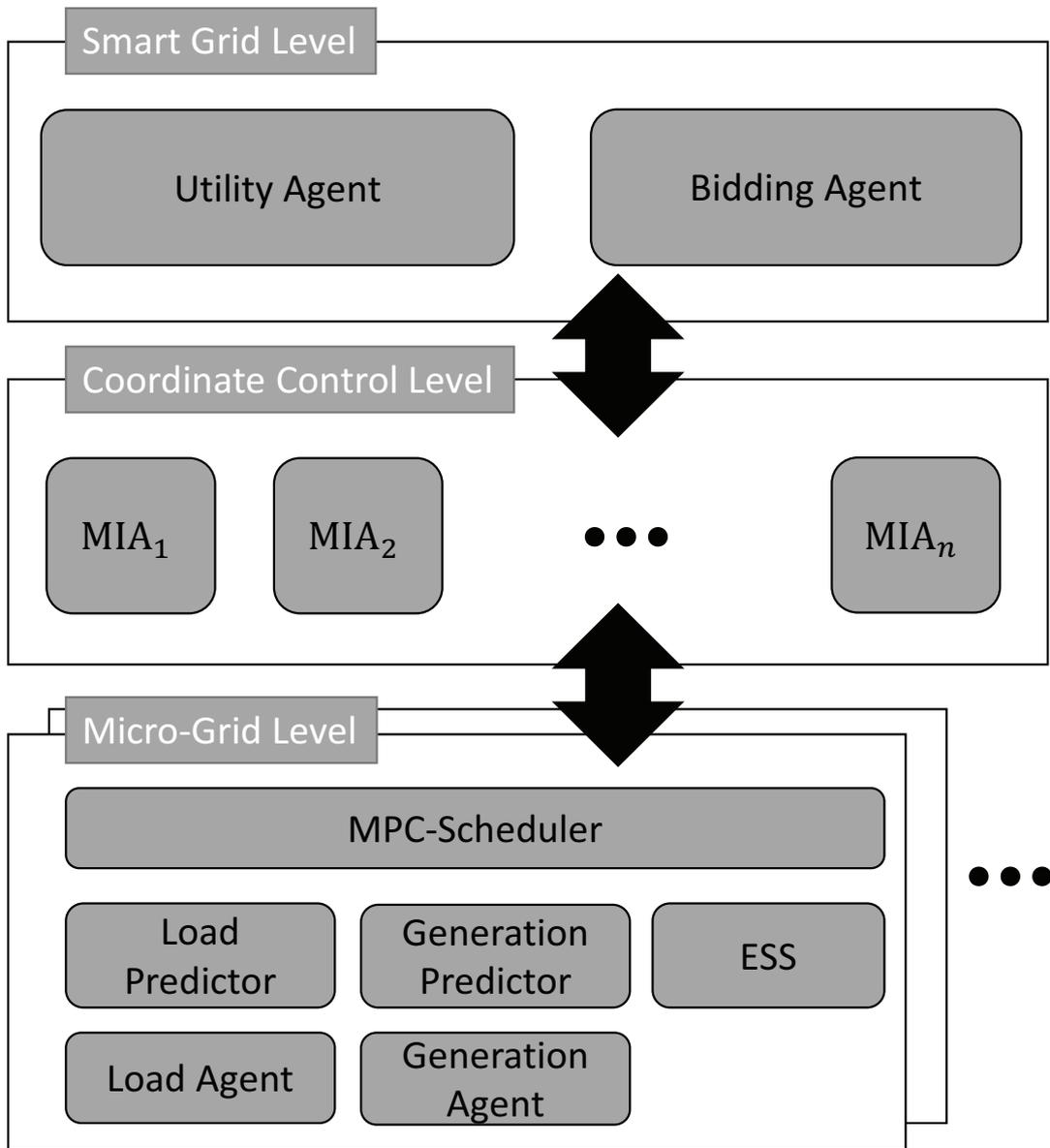
To support multiple micro-grids in global bidding for electricity within the smart grid, we propose a three-level hierarchical smart grid model based on the Multi-Agent System (MAS). The smart grid has three levels, namely a smart grid level, a coordinate control level, and a micro-grid level. As shown in Fig. 3.1, there are two agents at the smart grid level, including a utility agent and a bidding agent. At the coordinate control level, there are one or more micro-grid intelligent agents (MIA). As for micro-grid level, we design a MPC-Scheduler that collects power information from load predictors and generation agents and then decide a suitable operation mode for ESS according to some predefined rules. In the following time period, ESS will implement this operation mode and repeat the above process again.

At the smart grid level, the utility agent not only announces the dynamic electricity prices for each time interval, but also handles the inadequate or surplus power from all MIAs in each time interval. The other agent-bidding agent performs the market bidding of power-related commodities for all MIAs. In the bidding process, the electricity consumers can satisfy their load demands that have surplus electricity by buying elec-

tricity. As a result, at a cheaper price from the MIAs, the sellers do not waste the surplus electricity and gain the profit by transmitting power to other MIAs.

At the coordinate control level, each MIA can transmit/receive power to/from other MIAs. In other words, all the MIAs have the ability of bidirectional power dispatch. Moreover, all the MIAs can further transfer/receive power to/from the utility agent if supply-demand mismatch occurs among all MIAs. MIA should record the bidding information in each time interval.

At the micro-grid level, the MPC-Scheduler not only schedules the ESS, but also uploads the local supply-demand mismatch information to MIA for bidding at each time interval. The load predictor uses the historical load demands from a load agent, and predicts a new load demand at each time interval and sends the predicted load information to the MPC-Scheduler. The generation agent does likewise and sends the predicted generation information to the MPC-Scheduler at each time interval. The ESS receives the operation mode decided by the MPC-Scheduler, makes necessary changes (charge/discharge) as required, and records all such operation information for estimating the lifetime of battery.



MIA: micro-grid intelligent agent

Figure 3.1: Three-level hierarchical smart grid model.

### 3.3 Assumptions

- **Smart Grid Level:** It is assumed that all MIA requests arrive at the middle of a time interval and the bidding agent must deal with each request. When the next time interval starts, all of the MIAs should transfer/receive power to/from the another one according to the bidding result and the above process is repeated. After a matching between loads and generation within a micro-grid, an MIA may or may not participate in bidding at each time interval. A request for a “zero” or a “non-zero” amount of power will be made at each time interval by each time interval by each MIA. An micro-grid with a “zero” amount of power demand will not take part in bidding . The micro-grid with a request for a no-zero amount of power will participate in bidding. A request with a positive amount of power implies there is surplus electricity, which can be sold to the micro-grids that require power in the following time interval, that is, those who with a request for negative amount of power. During bidding, electricity are met by the surplus electricity. After bidding, if there is still unused surplus electricity then it is sold to the utility company. If there is deficit of electricity after bidding, then the unsatisfied requests for electricity demand are met by the utility.
- **Coordinate Control Level:** Anyone of the MIAs can transfer/receive power to/from the other MIAs without the power loss rate caused by long distance power transmission in conventional power grids. The energy conversion rate depends on the equipment capability along the power transmission path.
- **Micro-Grid Level:** It is assumed that there is no loss of power within a micro-grid and between micro-grids, irrespective of whether the power transfer/receive is to/from the other MIA, load agent, generation agent or ESS. Before each bidding process, the MPC-scheduler should finish all load/generation prediction, ESS scheduling, and upload its power request to its corresponding MIA for bidding.

### 3.4 Problem Formulation

Our target problem is defined as follows. Given the status at time interval  $t$  of a set of  $k$  micro-grids  $\{MIA_1(t), \dots, MIA_k(t)\}$  that comprise a smart grid, where  $MIA_i(t)$

$$= \langle DL_i(t), Gen_i(t), SP_i(t), uPrice(t) \rangle.$$

- $DL_i(t)$  denotes the demand load of the  $MIA_i(t)$  at the time interval  $t$ ,
- $Gen_i(t)$  denotes the generation power of the  $MIA_i(t)$  at the time interval  $t$ ,
- $SP_i(t)$  denotes the storage power of the  $MIA_i(t)$  at the time interval  $t$ ,
- $uPrice(t)$  denotes the utility electricity price at the time interval  $t$ ,

the target problem is to schedule and control the operation mode of ESS in all micro-grids such that the overall cost of electricity trading in the smart grid is minimized.

### 3.5 Load Prediction Parameters

We use the regression-based load prediction in our Thesis. In this section, we list the parameters and variables used in regression-based load prediction.

1.  $t_i$ : It represents the temperature at time interval  $i$ .
2.  $L_{itvl}$ : It represents the length of a time interval in minutes.
3.  $N_{itvl}$ : It represents the number of time intervals in a day.
4.  $lData_i$  and  $plData_i$ : It represents the historical demand load data and predicted demand load data at time interval  $i$ .
5.  $gData_i$  and  $pgData_i$ : It represents the historical power generation data and predicted power generation data at time interval  $i$ .
6.  $\pi_i$ : It represents the regression value coefficient for each time-of-week, which is used in Section 4.1.
7.  $N_B$ : It represents the number of bounds for the temperature intervals, which is used in Section 4.1.
8.  $\lambda_j$ : It represents the  $j^{\text{th}}$  temperature parameter, where  $j = 1 \dots N_B$ , which is used in Section 4.1.
9.  $B_k$ : It represents the  $k^{\text{th}}$  temperature bound, where  $k = 1 \dots N_B$ , which is used in Section 4.1.

10.  $T_{c,j}(t_i)$ : It represent the  $j^{\text{th}}$  component temperature of  $t_i$ , which is used in Section 4.1.
11.  $\hat{E}(t_i)$ : It represents the predicted demand load/generation power for  $t_i$ .

### 3.6 Parameter for Model Predictive Control

We use the model-predictive control method in our Thesis. In this section, we list the parameters and variables used in our MPC optimization algorithm.

1.  $i$ : It represents the  $i^{\text{th}}$  time interval in the MPC.
2.  $N$ : It represents the optimization time window length in terms of the number of time intervals.
3.  $DL_i$ : It represents the amount of demand load in a micro-grid at time interval  $i$ .
4.  $Gen_i$ : It represents the amount of electricity power supply in a micro-grid at time interval  $i$ .
5.  $uPrice_i$ : It represents the electricity price at time interval  $i$
6.  $U_i$ : It represents operation mode for  $ESS$  at time interval  $i$ .

$$U_i = \begin{cases} 1 & \text{if discharge mode;} \\ 0 & \text{if no action;} \\ -1 & \text{if charge mode;} \end{cases} \quad (3.1)$$

7.  $C_{ESS}$ : It represents the capacity of ESS in terms of kilowatt hour (kWh).
8.  $R_{ESS}$ : It represents the ROC or ROD of ESS.

### 3.7 Parameter for the Bidding

In this section, we list the parameters and variables used in the bidding market.

1.  $i$ : It represents the  $i^{\text{th}}$  time interval in the bidding market.

2.  $Request_i$ : It represents the power request of a MIA at time interval  $i$ , Its value is a positive number if there is surplus power for selling to other MIAs; it is negative if there is shortage of power that can be satisfied by other MIA.
3.  $N_{MIA}$ : It represents the number of MIAs in a smart grid.
4.  $N_{seller,i}$ : It represents the number of sellers at time interval  $i$ .
5.  $N_{buyer,i}$ : It represents the number of buyers at time interval  $i$ .
6.  $Priority_{MIA_i}$ : It represents the priority of  $i^{th}$  MIA for selling power.
7.  $P_r$ : It represents the retail price of electricity from a utility company.
8.  $P_f$ : It represents the feed-in price of electricity sold to a utility company.
9.  $pExchange$ : It represents the the electricity cost should be paid for all MIAs after auction.

# Chapter 4

## MPC-based Scheduling

We have introduced some basic concepts about our proposed three-level hierarchical smart grid model in Chapter 3. In Chapter 4, the most important chapter in this Thesis, we will explain the relationship among the load/generation predictor, MPC-scheduler, and auction-based marketing. Also, we will introduce the core techniques of each component in detail.

### 4.1 Load/Generation Predictor Design

Figure 4.1 illustrates the flow of estimating the amount of demand load by load predictor in a micro-grid and Figure 4.2 illustrates the flow of estimating the amount of power generation by generation predictor in a micro-grid. Load/generation predictor requires two inputs and produces one output as described in the following.

The first input is the historical weekly demand load/power generation data ( $lData/gData$ ) acquired from the load agent/generation agent. This data has 7 rows (days), where each row comprises 1 micro-grid ID, 1 date information, and  $N_{itvl}$  demand load / power generation columns which are in kilowatt hour (kWh) unit, such that each column is the accumulated volume within  $L_{itvl}$  minutes.

The second input is the historical weekly temperature data ( $tData$ ). This data has 7 rows, each row comprises 1 micro-grid ID, 1 date information, and  $N_{itvl}$  outdoor temperature columns, such that each column is the average temperature within  $L_{itvl}$  minutes.

The output is the predicted demand load ( $elData$ ) or predicted power generation

(*egData*) data of a week. This data has 7 rows (days), each row comprises 1 micro-grid ID, 1 date information, and  $N_{itvl}$  demand load / power generation columns which are in kilowatt hour (kWh) unit, each column is the accumulation volume within  $L_{itvl}$  minutes.

The first term  $N_{itvl}$  is the number of intervals and the second term  $L_{itvl}$  is the length of an interval, their relationship can be calculated by Equation (4.1). Consider the example shown in Table 4.1, if the length of a interval ( $L_{itvl}$ ) is 15 minutes, then the number of intervals ( $N_{itvl}$ ) is 96.

$$N_{itvl} = \frac{60 \times 24}{L_{itvl}} \quad (4.1)$$

Table 4.1: Example of relationship between  $N_{itvl}$  and  $L_{itvl}$ .

Number of intervals ( $N_{itvl}$ )	Length of a interval ( $L_{itvl}$ )
96	15 minutes
48	30 minutes
24	60 minutes

#### 4.1.1 Linear-regression based prediction method

The load / generation predictor has the dependence between a time-of-week indicator variable and a piecewise linear and continuous outdoor temperature dependence. Therefore, the predicted demand load / power generation  $\hat{E}(t_i)$  is calculated by Equation (4.2).

$$\hat{E}(t_i) = \pi_i + \sum_{j=1}^{N_B} \lambda_j T_{c,j}(t_i) \quad (4.2)$$

The parameters  $\pi_i$  for  $i = 1 \dots (7 \times N_{itvl})$ ,  $\lambda_j$  for  $j = 1 \dots N_B$ . To achieve a piecewise and continuous out door temperature at time interval  $i$ ,  $t_i$ , is broken into  $N_B$  component temperatures  $T_{c,j}(t_i)$  with  $j = 1 \dots N_B$ .

#### 4.1.2 Computing the component temperatures

Before computing the component temperatures, we have to create the temperature bounds in advance. In Equation (4.3), we divide the temperature into  $N_B$  equally sized

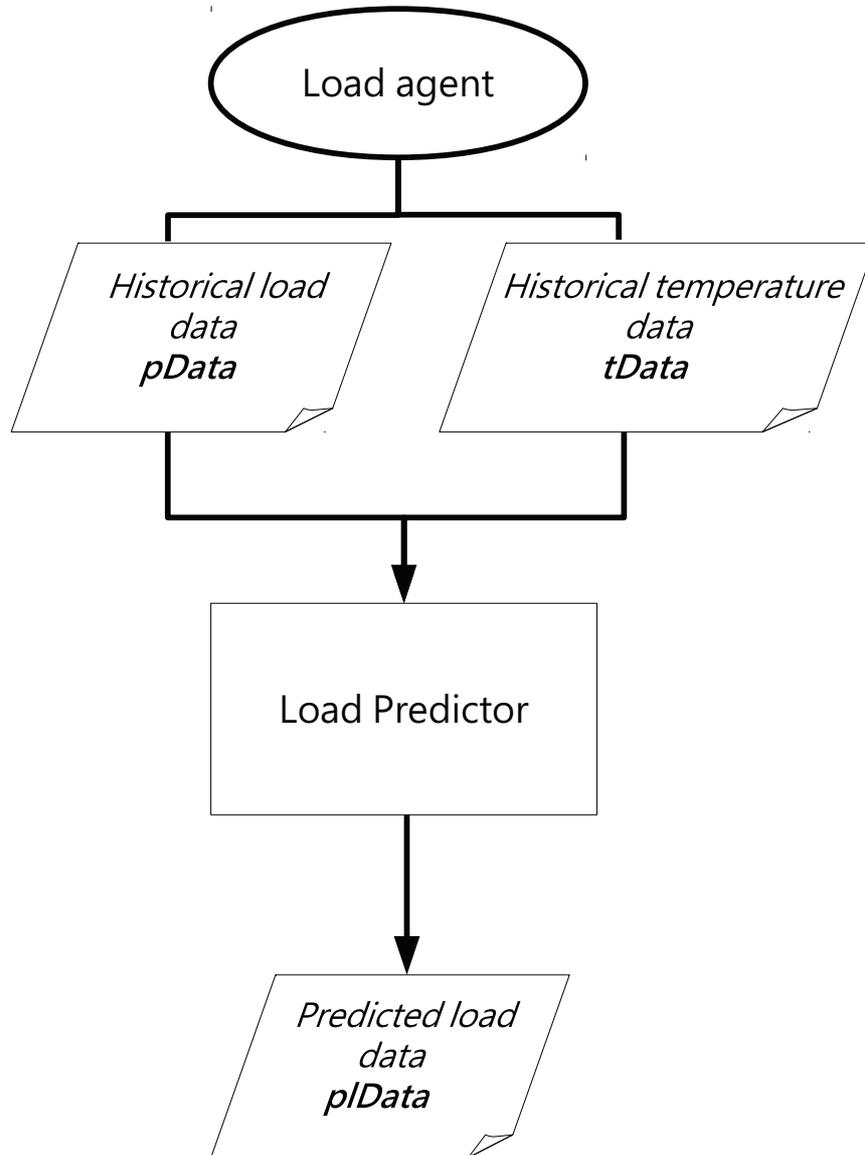


Figure 4.1: Load predictor framework.

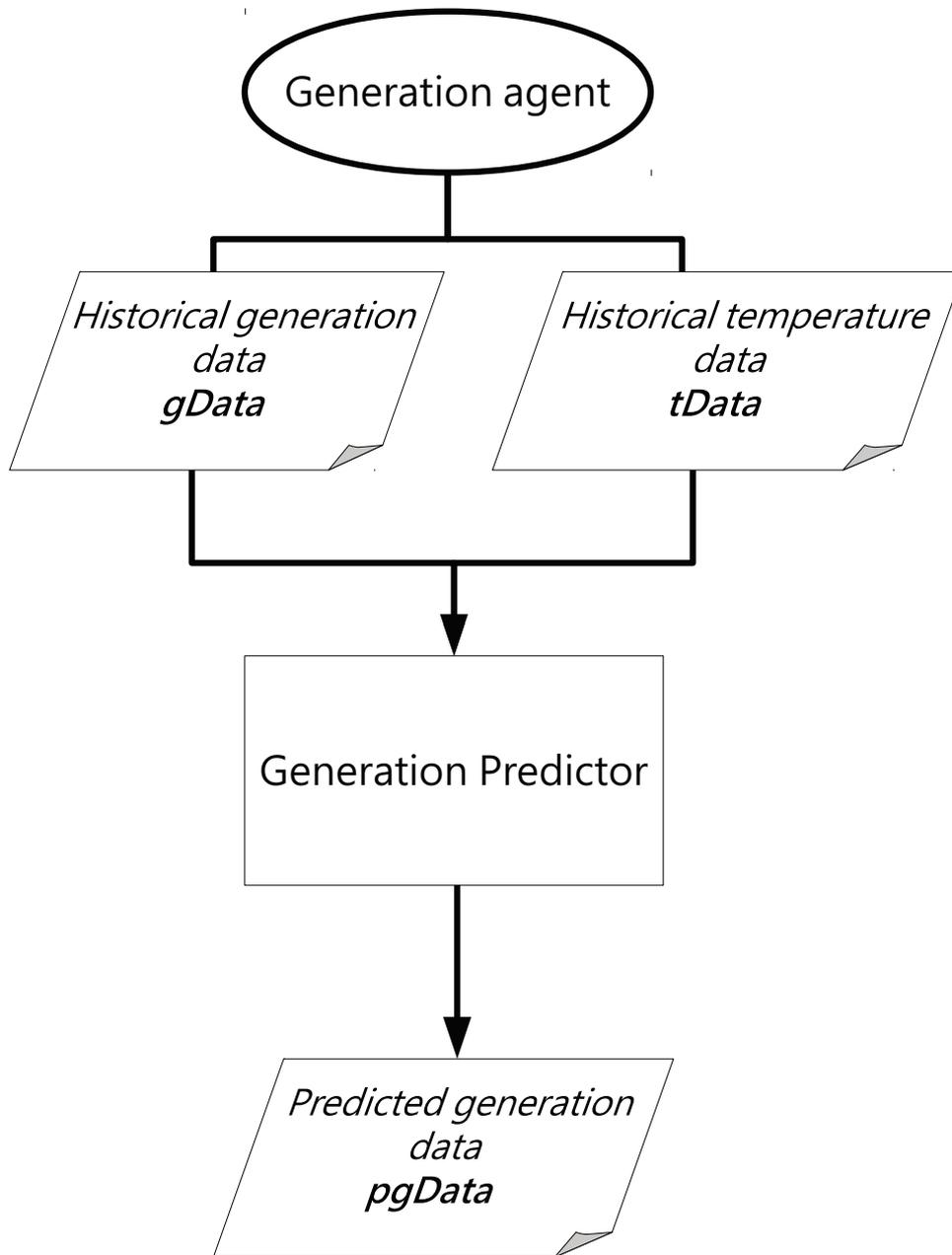


Figure 4.2: Generation predictor framework.

temperature bounds  $B_k$  for  $k = 1 \dots N_B$ . An example is shown in Table 4.2, if the minimum temperature  $T_{min}$  is 10 °C, the maximum temperature  $T_{max}$  is 35 °C and the number of temperature bounds  $N_B$  are 5, the temperature intervals would be [10, 15] °C, (15, 20] °C, (20, 25] °C, (25, 30] °C, (30, 35] °C.

$$B_k = T_{min} + \frac{T_{max} - T_{min}}{N_B} \times k \quad (4.3)$$

Table 4.2: Example for temperature bounds.

$B_1$	$B_2$	$B_3$	$B_4$	$B_5$
[10, 15] °C	(15, 20] °C	(20, 25] °C	(25, 30] °C	(30, 35] °C

Then, we can start to compute the component temperature. In Figure 4.3, the target temperature  $T$  is divided into  $N_B$  component temperatures  $T_{c,i}$  according to the ascending temperature bound  $B_i$  with  $i = 1 \dots N_B$ . With the increase of  $i$ ,  $T_{c,i}$  is calculated by Equation (4.4) when  $T$  is greater than  $B_i$ , otherwise,  $T_{c,i}$  is calculated by Equation (4.5) and turns on the signal variable ( $sig = 1$ ) which is a variable for recording whether the target temperature is greater than temperature bound or not. As shown in Table 4.3, the target temperature  $T$  is 24 in the fourth row, for  $i = 1$ ,  $T$  is greater than  $B_1$ , then  $T_{c,1}$  is equal to  $B_1$  (15); for  $i = 2$ ,  $T$  is greater than  $B_2$  (20), then  $T_{c,2}$  is the difference (5) of  $B_2$  and  $B_1$ ; for  $i = 3$ ,  $T$  is less than  $B_3$  (25), then  $T_{c,3}$  is the difference (4) of  $T$  and  $B_2$ , and  $sig = 1$ ; for  $i = 4$  and 5,  $T_{c,i}$  is equal to 0, other target temperatures are calculated in the same way. Algorithm 1 illustrates how to compute the component temperatures employed by the load / generation predictor.

$$T_{c,i} = \begin{cases} B_i & \text{if } i = 1, T > B_i \\ B_i - B_{i-1} & \text{if } i > 1, T > B_i; \end{cases} \quad (4.4)$$

$$T_{c,i} = \begin{cases} T & \text{if } i = 1, T \leq B_i; \\ T - B_{i-1} & \text{if } i > 1, T \leq B_i; \end{cases} \quad (4.5)$$

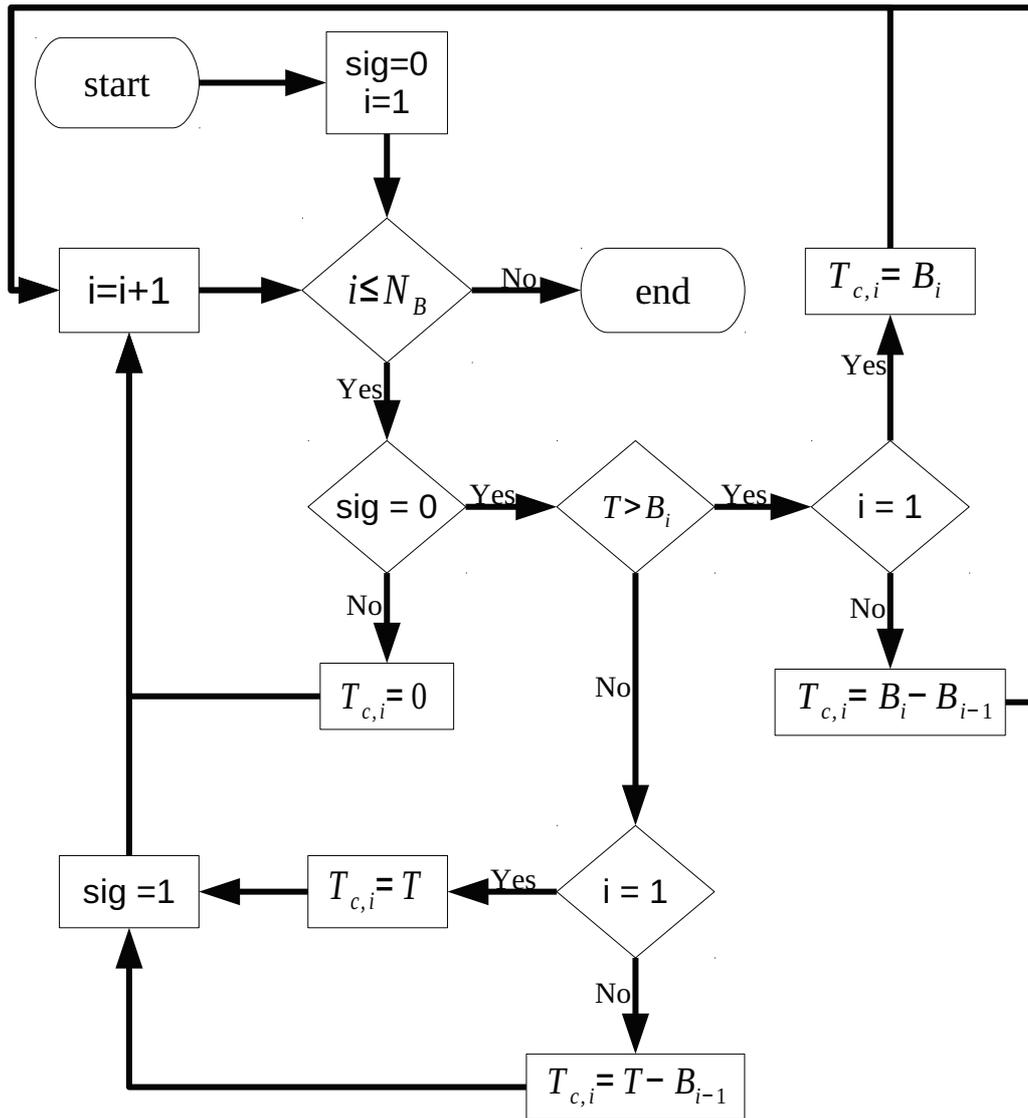


Figure 4.3: Flow diagram of computing component temperatures.

---

**Algorithm 1:** Compute the component temperatures

---

**Input:**

$pData$  : The historical weekly demand load / power generation data

$tData$  : The historical weekly temperature data

$tmpBnd$  : The array of temperature bound

$col$  : The number of columns of  $rMatrix$

$row$  : The number of rows of  $rMatrix$

**Output:**

$rMatrix$  : The matrix comprises component temperatures and load/generation data for linear regression

```
// Initialization
1  $sig \leftarrow 0$ ;
2 for  $j \leftarrow 1$  to  $row$  do
3   for  $i \leftarrow 1$  to  $col$  do
4      $rMatrix_{j,i} \leftarrow 0$ ;
5 for  $j \leftarrow 1$  to  $row$  do /* The  $j^{\text{th}}$  training week */
6   for  $i \leftarrow 1$  to  $col - 1$  do /* The  $i^{\text{th}}$  component temperature */
7     if  $tData_j > tmpBnd[i]$  and  $sig = 0$  then
8       if  $i = 1$  then
9          $rMatrix_{j,i} = tmpBnd[i]$ ;
10      else
11         $rMatrix_{j,i} = tmpBnd[i] - tmpBnd[i - 1]$ ;
12      else if  $tData_j \leq tmpBnd[i]$  and  $sig = 0$  then
13        if  $i = 1$  then
14           $rMatrix_{j,i} = tData_j$ ;
15        else
16           $rMatrix_{j,i} = tData_j - tmpBnd[i - 1]$ ;
17         $sig \leftarrow 1$ ;
18      else if  $sig = 1$  then
19         $rMatrix_{j,i} \leftarrow 0$ ;
20     $rMatrix_{j,col} = pData_j$ ;
21 return  $rMatrix$ ;
```

---

Table 4.3: Example for computing component temperatures.

		$i$				
		1	2	3	4	5
$T_{c,i}$	$B_i$	15	20	25	30	35
$T$						
11		11	0	0	0	0
20		15	5	0	0	0
24		15	5	4	0	0
27		15	5	5	2	0
31		15	5	5	5	1
35		15	5	5	5	5

### 4.1.3 Linear-regression based prediction framework

We can simply separate the prediction framework into three parts as shown in Figure 4.4, including initialization, computing the component temperature, and solving the linear regression problem. In first part, the load/generation predictor will read the historical weekly demand load/power generation data and the temperature data from  $N_{tweek}$  weeks. In the second part, load/generation predictor will get the minimum and maximum temperature in advance, then we can compute the component temperatures according to the temperature bounds. In the third part, we solve the linear regression problem by using the Ordinary Least Squares Principle (OLS). Eventually, we can get the predicted values of  $\pi_i$  and  $\lambda_j$  at  $i^{\text{th}}$  time interval for  $j = 1 \dots N_B$  used in the Equation (4.2). For the forecast temperature, we compute the component temperature by Subsection 4.1.2 first, and then get the predicted power  $\hat{E}(t_i)$ . Algorithm 2 illustrates the linear prediction method employed by load/generation predictor.

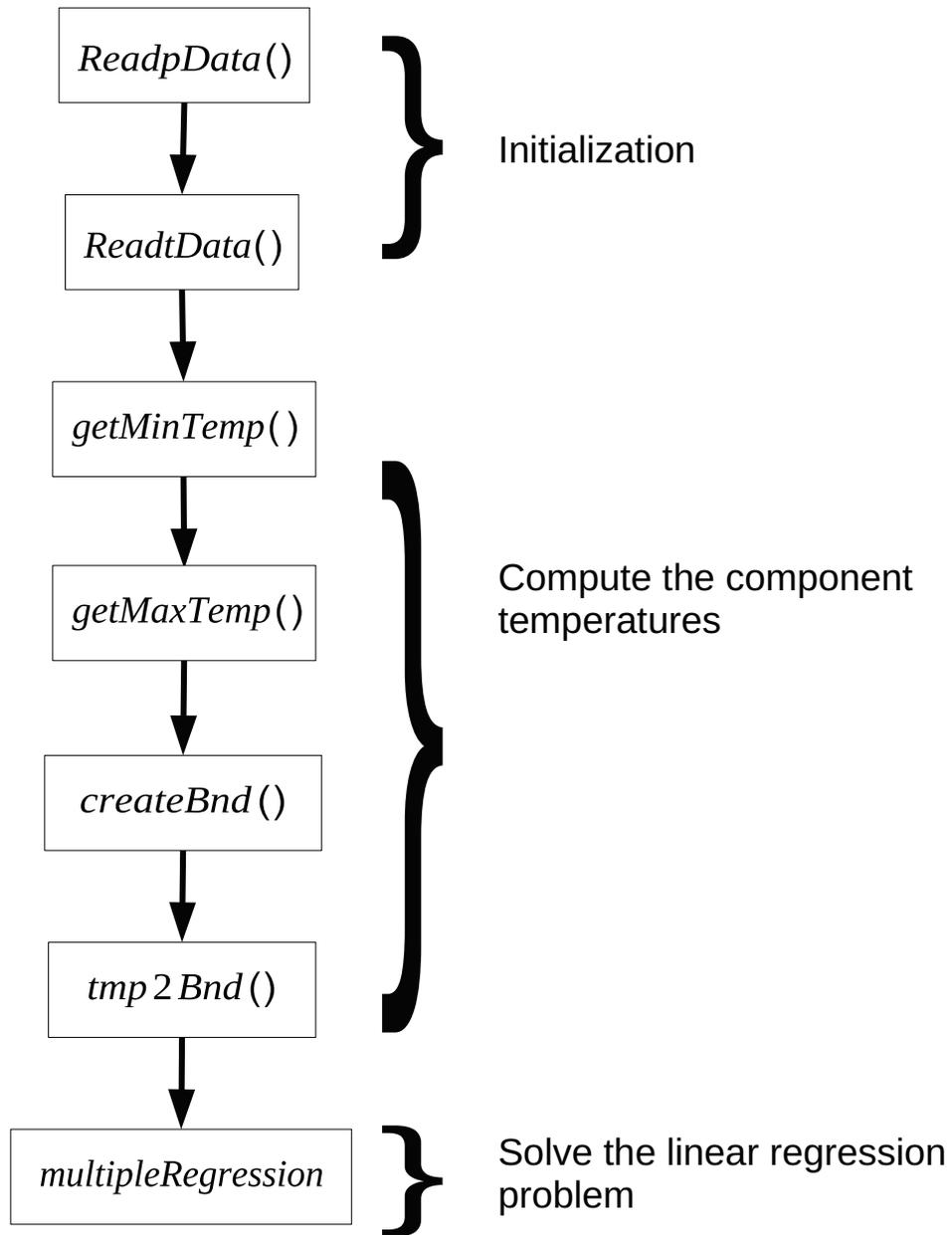


Figure 4.4: Linear-regression based prediction.

---

**Algorithm 2:** Linear regression-based prediction

---

**Input:**

$hData$  : The historical weekly demand load/power generation data

$tData$  : The historical weekly temperature data

$N_{tweek}$  : The number of training weeks

$N_{itvl}$  : The number of intervals a day

$N_B$  : The number of temperature bounds

**Output:**

$pData$  : The predicted weekly demand load/power generation data

```
1 for  $j \leftarrow 1$  to 7 do                                /* The  $j^{\text{th}}$  day of the week */
2   for  $i \leftarrow 1$  to  $N_{itvl}$  do                    /* The  $i^{\text{th}}$  interval of the day */
3     // Read data
4      $hData_{j,i} = \text{ReadpData}(N_{tweek}, N_{itvl});$ 
5      $tData_{j,i} = \text{ReadtData}(N_{tweek}, N_{itvl});$ 
6     // Get minimum and maximum temperature
7      $minTemp_{j,i} = \text{getMinTemp}(tData_{j,i}, N_{tweek});$ 
8      $maxTemp_{j,i} = \text{getMaxTemp}(tData_{j,i}, N_{tweek});$ 
9     // Start Multiple Linear Regression
10     $tmpBnd_{j,i} = \text{createBnd}(N_B, minTemp_{j,i}, maxTemp_{j,i}, N_{tweek});$ 
11     $rMatrix_{j,i} =$ 
12     $\text{temp2Bnd}(hData_{j,i}, tData_{j,i}, tmpBnd, N_B + 1, N_{tweek});$ 
13    /* Algorithm 1 */
14     $pData_{j,i} = \text{multipleRegression}(rMatrix_{j,i});$ 
15 return  $pData;$ 
```

---

## 4.2 Model-Predictive Control Scheduler in Micro-grid Level

Figure 4.5 illustrates the flow of scheduling the operation mode of ESS by the proposed MPC scheduler in a micro-grid. MPC scheduler requires three inputs and produces one output as described in the following.

The first input is the weekly predicted demand load data ( $plData$ ) from the load predictor. The second input is the weekly predicted power generation data ( $pgData$ ) from the generation predictor. The third input is the SoC constraints related to ESS such as the initial SoC ( $initSoC$ ), the upper bound of SoC ( $SoC_{max}$ ), the lower bound of SoC ( $SoC_{min}$ ), the rate-of-charge or rate-of-discharge of ESS ( $R_{ESS}$ ), and the capacity of ESS ( $C_{ESS}$ ). The output is the operation mode of ESS in the next interval ( $U_1$ ).

### 4.2.1 Model-predictive control scheduling on ESS

The MPC scheduler controls the operation mode of ESS the next time slot  $t+1$  by considering the future  $N$  slots, called the time horizon. If the time horizon is from slot  $t + 1$  to  $t + N$ , we need to predict the amount of power generation ( $Gen_i$ ), the amount of demand load ( $DL_i$ ), the amount of to be used from ESS or to be stored into ESS, and the dynamic utility price ( $uPrice_i$ ). As shown in Figure 4.6, at each time slot  $t$ , the MPC scheduler solves the optimization problem over the  $N$ -slot horizon time. The optimization problem is to maximize the benefits which would be reduce the electricity cost or increase the cost for selling electricity as shown by Equation (4.6) under two constraints. One constraint is ensure that the  $SoC$  of ESS is between the lower bound ( $SoC_{min}$ ) and the upper bound ( $SoC_{max}$ ) at each time slot  $t$ . The other constraint is to ensure that the operation mode ( $U_i$ ) of ESS should be either charge ( $-1$ ), discharge ( $1$ ), or no action ( $0$ ).

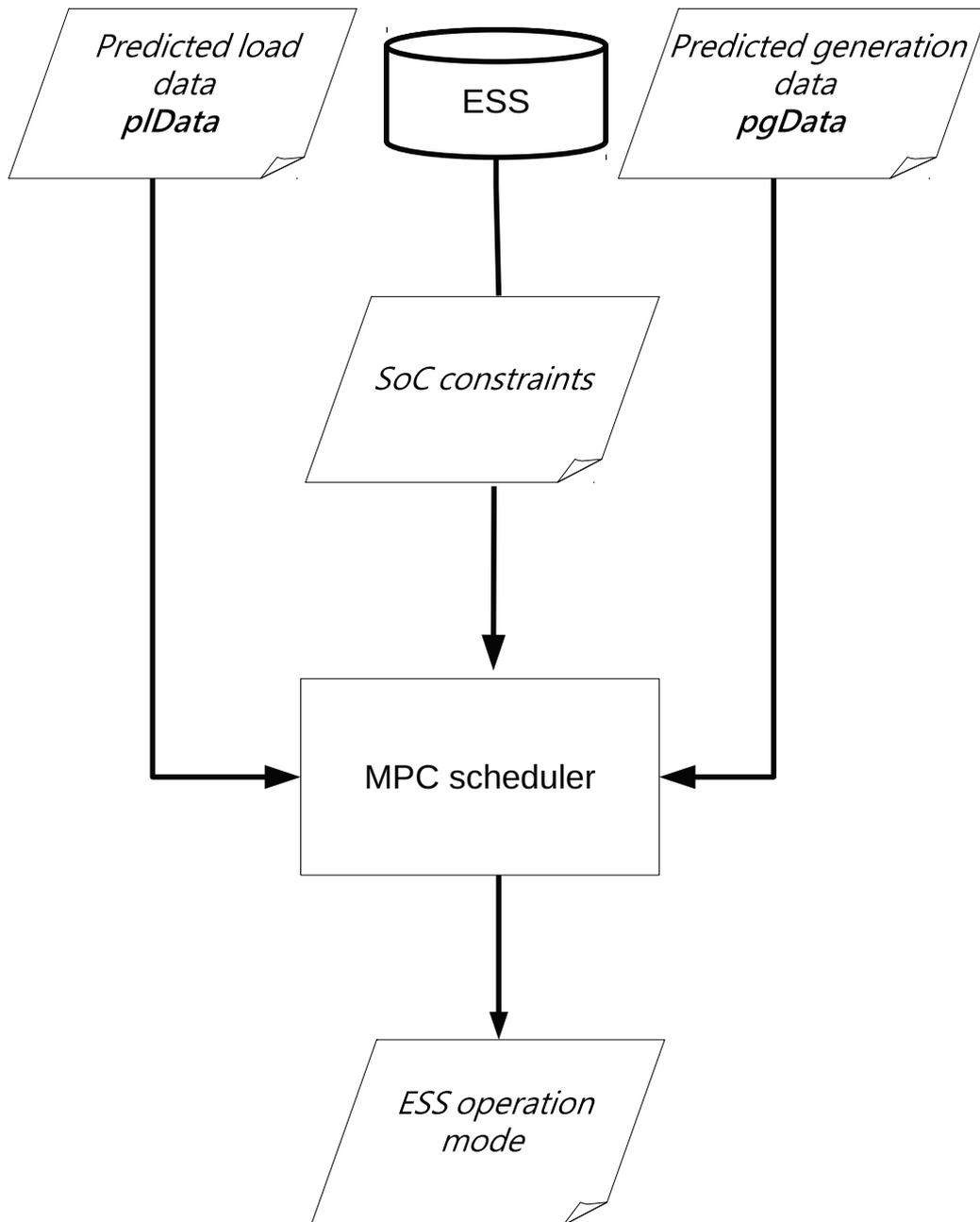


Figure 4.5: MPC-Scheduler framework.



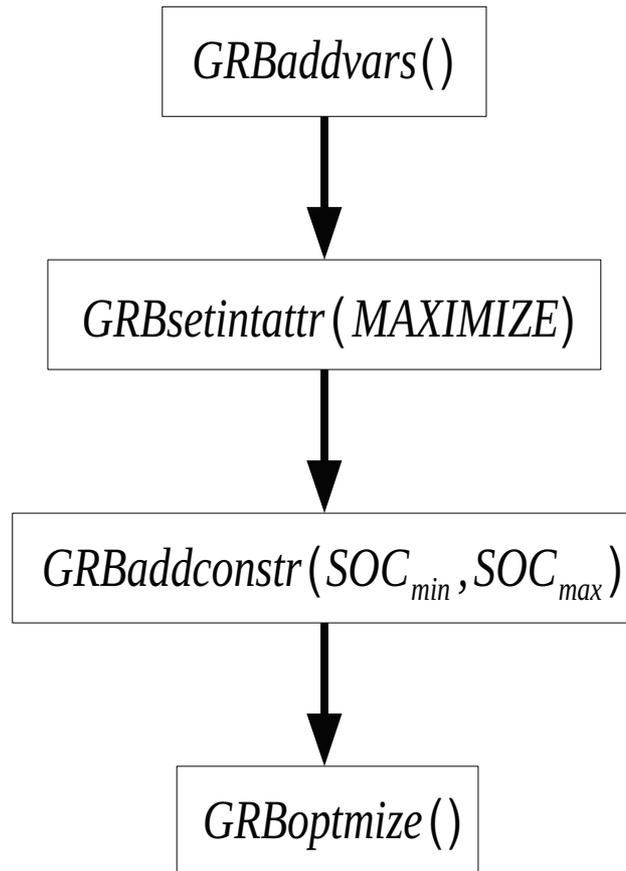


Figure 4.7: Flow diagram of the overall MPC framework.

---

**Algorithm 3:** Model-predictive control scheduling

---

**Input:**

$DL_j$  : The predicted demand load data  
 $Gen_j$  : The predicted power generation data  
 $uPrice_j$  : The dynamic utility prices  
 $initSoC_j$  : The initial SoC of ESS  
 $SoC_{min}$  : The lower bound of SoC  
 $SoC_{max}$  : The upper bound of SoC  
 $R_{ESS}$  : The rate-of-charge or rate-of-discharge of ESS  
 $C_{ESS}$  : The capacity of ESS  
 $N$  : The length of model-predictive control window

**Output:**

$U_j$  : The operation mode of ESS

```
1 for  $j \leftarrow 1$  to  $N$  do
  | // Create objective function
2    $obj_j = (Gen_j - DL_j + U_j \times R_{ESS} \times C_{ESS}) \times uPrice_j$ ;
3    $lb_j = -1$ ;
4    $ub_j = 1$ ;
5    $GRBaddvars(obj_j, lb_j, ub_j, GRB_INTEGER)$ ;
  | // Maximization Problem
6  $GRBsetintattr(GRB_MAXIMIZE)$ ;
  | // Set constraints
7 for  $j \leftarrow 1$  to  $N$  do
8   | for  $k \leftarrow 1$  to  $j$  do
9     |  $ind[k] = k$ ;
10    |  $val[k] = 1$ ;
11     $GRBaddconstr(ind, val, GRB_LESS_EQUAL, SoC_{max} - initSoC_j)$ ;
12     $GRBaddconstr(ind, val, GRB_GREATER_EQUAL, SoC_{min} -$ 
  |  $initSoC_j)$ ;
  | // Optimization-Integer Linear Programming
13 return  $GRBoptimize()$ ;
```

---

### 4.3 Bidding Market at the Smart Grid Level

Figure 4.8 illustrates the flow of auctioning off the surplus power and bidding for shortage power in the bidding market. The bidding market requires  $n$  power *request* from all of the MIAs and produces the power switching information (*pExchange*).

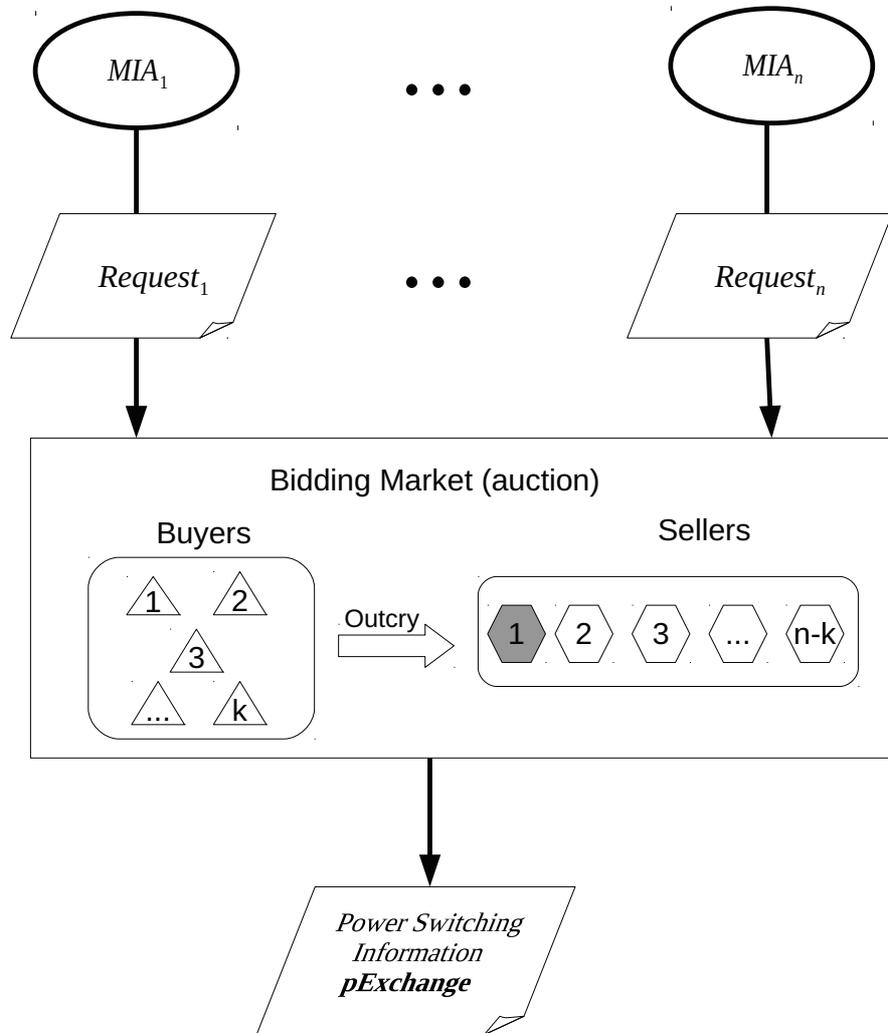


Figure 4.8: Flow diagram of bidding market.

In Figure 4.8, a *Request* could be either to buy or to sell power in a micro-grid. Namely, if the  $request_i$  is greater than 0, then  $MIA_i$  would be a seller in the bidding market and sell the surplus power to the other MIA with the higher price than utility feed-in price, otherwise, the  $MIA_i$  would be a buyer in the bidding market and fulfil the shortage power from the other MIA with the cheaper price than utility retail price. The output is the power switching information (*pExchange*) after the bidding process is over.

### 4.3.1 Two traditional auction mechanisms

We use two auction mechanism in our bidding market. The first-price sealed auction has two important features. One is that no bidder knows the bid of any other participant, and the other is that the highest bidder pays the price they submitted.

The second-price sealed auction is similar to the first-price sealed auction, the only one difference between them is that the highest bidder pays the second high submitted price. This auction mechanism will reduce more cost than the other one for the buyers such that buyers then increase the volume of power bidding in the market within the on-peak time.

### 4.3.2 Bidding Procedure

These is only one auction good selling at one time in the process of the bidding procedure. In order to maintain the fairness of all sellers when buyers bid for the good, we first give the non-repetitive random priority to them and then apply the Round-Robin (RR) rule to shift their priority.

As shown in Figure 4.9, the top half part is that the bidding market reads all of the MIA requests (*readRequest()*) and gives the non-repetitive priority to them (*setPriority()*), the bottom half part is flow of the main bidding procedure. In the lower left hand side of Figure 4.9, the bidding market will sell one good at one time until none of buyers or seller participating in the bidding, the dealer will decide who is the highest bidder (*getWinner()*), then sell the good to the highest priority seller (*getFirstPriority()*). After striking a bargain, both the buyer and the seller should adjust their request according to the transaction volume of power, and the priority is also shifted according to the RR rule. In the lower right hand side of Figure 4.9, the bidding procedure is over then the buyer should fulfil the rest of shortage power from utility company with retail price (*doReatail()*) and the seller should sell the rest of surplus power to utility company with feed-in price(*doFeedIn()*).

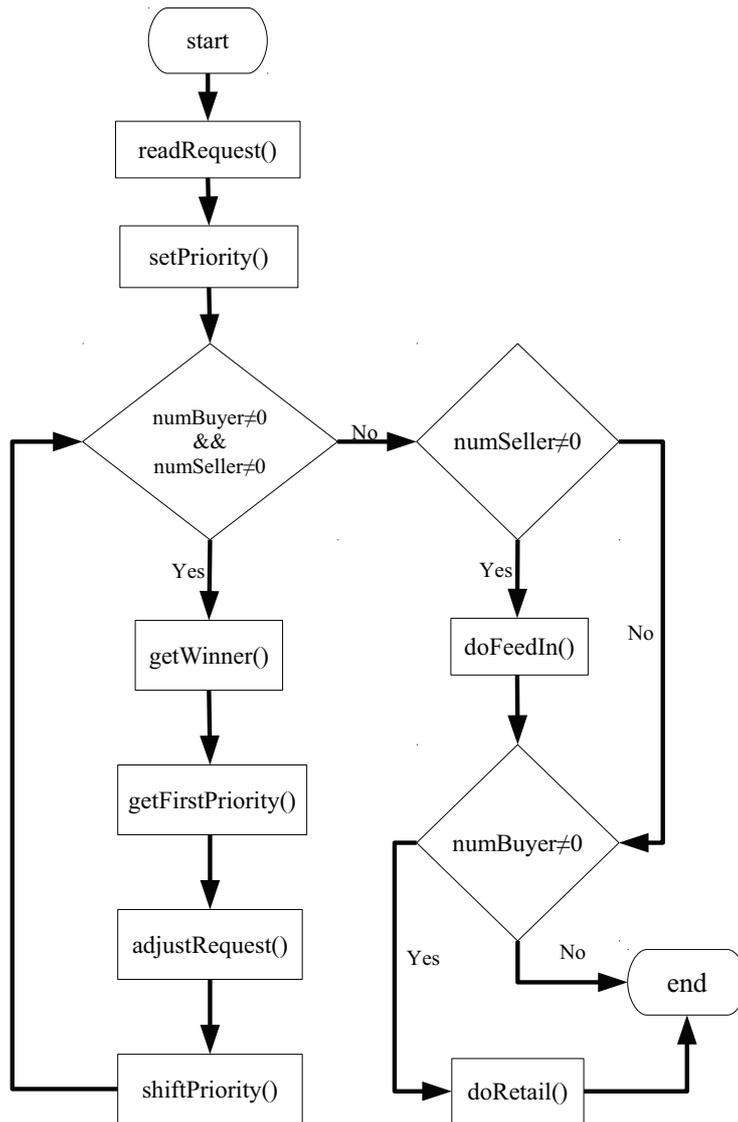


Figure 4.9: Flow diagram of the overall bidding market framework.

Take for example (See Table 4.4), the buyer E bids for the highest price \$60 than the others for a volume of 10kWh power, the highest priority seller B then sells out 10kWh to B. Hence, the priority of B is 3. The smaller number represents the higher priority.

Table 4.4: Example for deciding the buyer and the seller before auction.

Buyer	A	E	F	Seller	B	D
Outcry (\$)	50	<u>60</u>	45	Priority	<u>3</u>	5
Volume (kWh)	10					

Then the bidding market should adjust the request of buyer and seller. As shown in Table 4.5, we decrease the amount of surplus power of seller B and also decrease the

amount of shortage power of buyer E relatively according to the auction result.

Table 4.5: Example for adjusting *Request* after auction.

MIA	A	B	C	D	E	F
<i>Request</i> (kWh) – Before auction	-5	10	0	8	-12	-8
<i>Request</i> (kWh) – After auction	-5	<b>0</b> (-10)	0	8	<b>-2</b> (+10)	-8

Finally, the bidding market should shift the priority of seller. As shown in Table 4.6, we shift the seller B to the lowest priority of all MIAs, that is to say, the priority of B is changed from the 3<sup>th</sup> to the 6<sup>th</sup> and the MIA priority between 4<sup>th</sup> and 6<sup>th</sup> are also decreased by one position.

Table 4.6: Example for shifting the priority of seller after auction.

Priority	1	2	3	4	5	6
MIA – Before auction	C	E	<b><u>B</u></b>	A	D	F
MIA – After auction	C	E	A ( $\leftarrow$ 1)	D ( $\leftarrow$ 1)	F ( $\leftarrow$ 1)	<b><u>B</u></b> ( $\rightarrow$ 3)

Algorithm 4 illustrates the first-price sealed mechanism employed by bidding market and Algorithm 5 illustrates the second-price sealed mechanism employed by bidding market.

---

**Algorithm 4:** Auction–The first-price sealed auction

---

**Input:**

*Request* : A set of MIA requests

$N_{MIA}$  : The number of MIAs

$P_f$  : The feed-in price of utility company

$P_r$  : The retail price of utility company

**Output:**

*pExchange* : A set of power exchange information

```
// Initialize
1 readRequest(Request,  $N_{MIA}$ ,  $N_{buyer}$ ,  $N_{seller}$ );
2 setPriority( $N_{MIA}$ );
3 while  $N_{buyer} \neq 0$  and  $N_{seller} \neq 0$  do
    | // Start Auction
4     | getWinner( $MIA_{win}$ ,  $bPrice_{first}$ );
5     | getFirstPriority( $MIA_{first}$ );
6     | adjustRequest( $MIA_{first}$ ,  $MIA_{win}$ ,  $bPrice_{first}$ ,  $N_{buyer}$ ,  $N_{seller}$ )
    | shiftPriority( $MIA_{win}$ )
7 if  $N_{seller} \neq 0$  then
8   | doFeedIn( $P_f$ );
9 if  $N_{buyer} \neq 0$  then
10  | doRetail( $P_r$ );
11 return pExchange
```

---

---

**Algorithm 5:** Auction–The second-price sealed auction

---

**Input:**

*Request* : A set of MIA requests

$N_{MIA}$  : The number of MIAs

$P_f$  : The feed-in price of utility company

$P_r$  : The retail price of utility company

**Output:**

*pExchange* : A set of power exchange information

```
// Initialize
1 readRequest(Request,  $N_{MIA}$ ,  $N_{buyer}$ ,  $N_{seller}$ );
2 setPriority( $N_{MIA}$ );
3 while  $N_{buyer} \neq 0$  and  $N_{seller} \neq 0$  do
    | // Start Auction
4     | getWinner( $MIA_{win}$ , bPricesecond);
5     | getFirstPriority( $MIA_{first}$ );
6     | adjustRequest( $MIA_{first}$ ,  $MIA_{win}$ , bPricesecond,  $N_{buyer}$ ,  $N_{seller}$ )
    | shiftPriority( $MIA_{win}$ )
7 if  $N_{seller} \neq 0$  then
8   | doFeedIn( $P_f$ );
9 if  $N_{buyer} \neq 0$  then
10  | doRetail( $P_r$ );
11 return pExchange;
```

---

# Chapter 5

## Experimental Results

This chapter presents the evaluation of load prediction method, ESS scheduling on cost reduction, trade off between cost reduction and lifetime of ESS, and the MPC-based scheduling with bidding market.

### 5.1 Experiment setup

In this section, we explain the environment, extra math library, and demand load data used in our experiments.

Table 5.1: Environment of experiment

CPU	Intel(R) Core(TM) i7-2600 CPU 4 cores @ 3.40GHz
Memory	DDR3 1333 8G*2 (16G)
Operating system (OS)	Linux Ubuntu 11.10 (64 bits)
Kernel	3.0.0-25-generic
Compiler	gcc-4.4.6

#### 5.1.1 Experiment environment

For the experiment environment as shown in Table 5.1, we use a PC with an Intel(R) Core(TM) i7-2600 CPU. There are four cores in the CPU, and the frequency is 3.4 GHz for each core. There are 16 GB memory. Furthermore, we use 64-bit Linux Ubuntu 11.10 as our operating system. For the programming, we implemented our proposed framework using the C programming language.

### 5.1.2 Mathematical programming solver

In order to solve the integer linear programming (ILP) problem in our proposed MPC method, we use the mathematical programming solver – GUROBI-5.6.2 [19]. GUROBI is a free and easy-to-use software, and supports interfaces for a variety of programming languages such as C, C++, Java, .NET, Python, Matlab, and R. Also, it includes the following solvers: linear programming solver (LP solver), quadratic programming solver (QP solver), quadratically constrained programming solver (QCP solver), mixed-integer linear programming solver (MILP solver), mixed-integer quadratic programming solver (MIQP solver), and mixed-integer quadratically constrained programming solver (MIQCP solver).

### 5.1.3 Demand load data, generation prediction, specification of ESS, and dynamic electricity price

We use the ERCOT [20] demand load data including residential area and industrial area as our historical data in our experiments. Characteristics of daily 15-minute demand load are shown in Table 5.2, where base load represents the minimum power required by loads in 15-min interval for a day.

Table 5.2: Daily 15-min demand load data of residential and industrial area.

	<b>Base load (kWh)</b>	<b>Avg. load (kWh)</b>	<b>Peak load (kWh)</b>
<b>Residential</b>	22.87	26.23	30.26
<b>Industrial</b>	574.12	683.94	771.2

We use Matlab/Simulink R2013a generator module including PV, wind turbine, and fuel cell to predict generation power by importing the forecast weather data such as wind speed and irradiance. Penetration is used for setting up the maximum generation power for each generator module and is calculated by Equation 5.1. Table 5.3 lists the penetration for each generator module.

$$Penetration = \frac{Maximum\ generation\ power\ (kWh)}{Average\ load\ (kWh)} \quad (5.1)$$

The specification of ESS set up in our experiments is shown in Table 5.4.

Table 5.3: Penetration for each generator module.

	PV	Fuel cell	Wind turbine
Penetration	30%	20%	200%

Table 5.4: The specification of ESS.

Initial SoC	SoC range	RoC/RoD	Penetration
50%	20%~80%	0.1C/hr	20%

The dynamic electricity price of utility company used in our experiments is shown in Figure 5.1. The highest electricity price is \$27.35 at the 15<sup>th</sup> hour and the lowest electricity price is \$8.10 at the 5<sup>th</sup> hour.

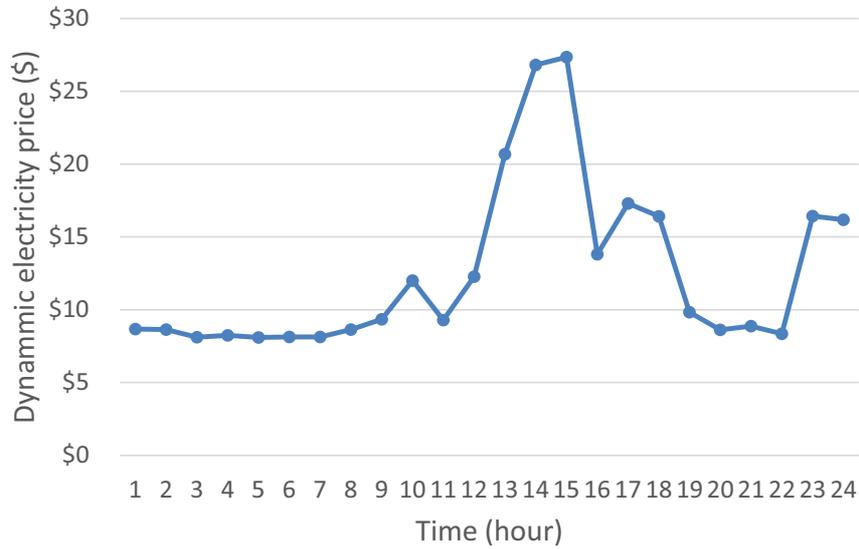


Figure 5.1: Dynamic electricity price of utility company in a day.

### 5.1.4 Comparison with other ESS strategies

We compare to three different strategies using ESS in our experiment including exhausted, backup [9] [1], and frequency regulation (FR) [21], and explain each strategy in the following.

The first strategy is that micro-grid uses ESS in exhausted way. As shown in Figure 5.2, the  $x$ -axis is the time from 1<sup>st</sup> hour to 24<sup>th</sup> hour in a day and the  $y$ -axis is SoC of ESS. The initial SoC is 50% and discharges 10% power in every hour until the lower bound (20%) of ESS is reached at 4<sup>th</sup> hour. Then, ESS charges 10% power in every hour until the upper bound (80%) of ESS is reached at 11<sup>th</sup> hour and repeats the above

flow of operating the ESS.

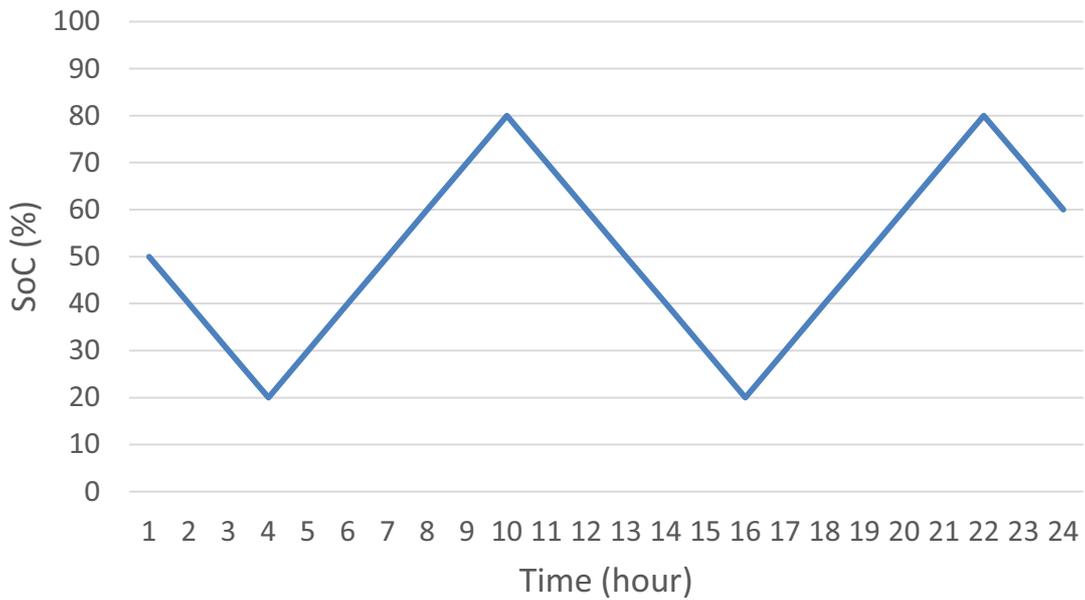


Figure 5.2: SoC in the exhausted method in a day.

The second strategy is that ESS discharges 10% power when micro-grid is in power shortage, and ESS charges 10% power when micro-grid has surplus power. Also, ESS could not discharge/charge when the lower bound (20%) / upper bound (80%) is reached. As shown in Figure 5.3 and Figure 5.4, the first five hours, the micro-grid has surplus power and the ESS charges 10% in first three hours and maintains 80% SoC at the 4<sup>th</sup> and 5<sup>th</sup> hours. Then, the micro-grid is in power shortage three hours in a row and ESS discharges 10% power in every hour. At the 24<sup>th</sup> hour, the micro-grid is in power shortage, but the lower bound (20%) is reached and thus ESS maintains 20% SoC.

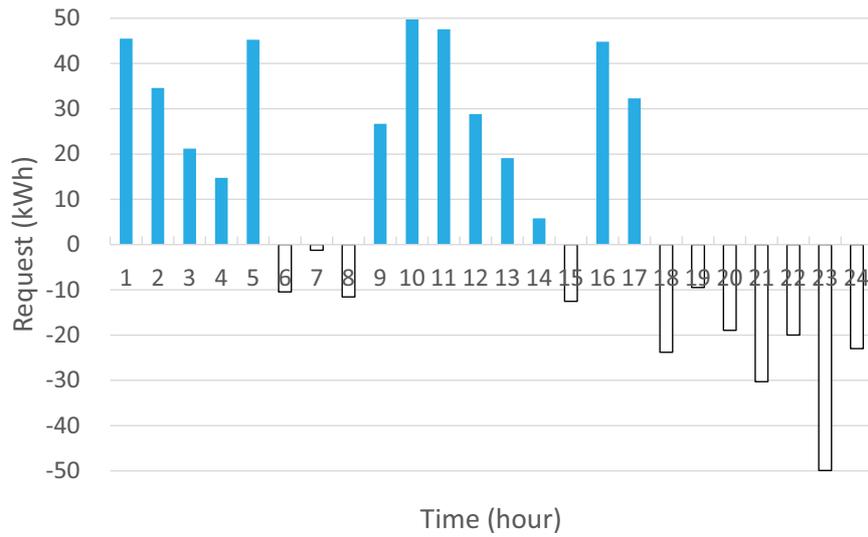


Figure 5.3: The request of a micro-grid in a day.

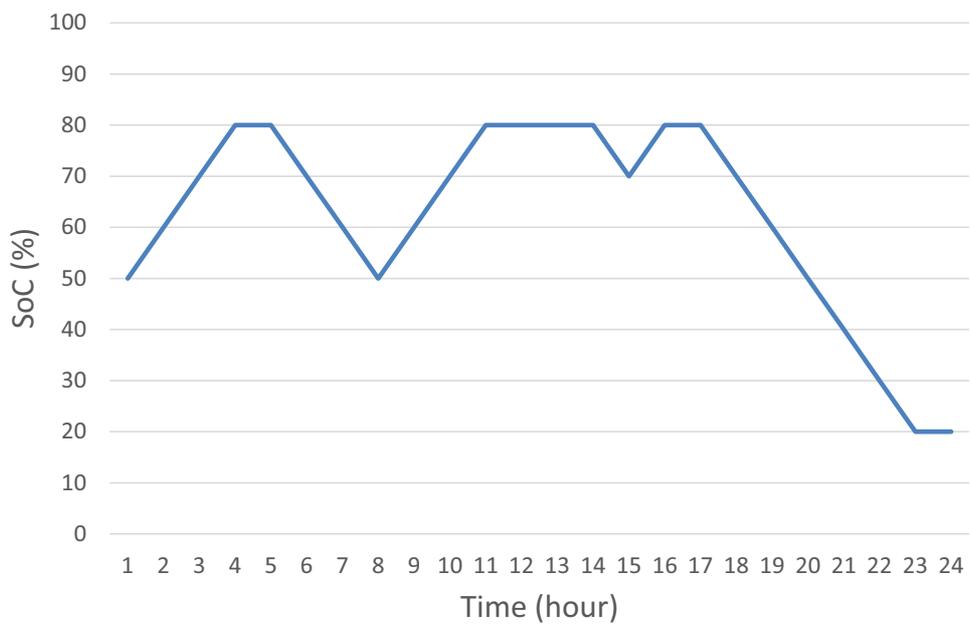


Figure 5.4: The SoC of ESS in a day with backup strategy.

The third strategy is to profile the frequency regulation (FR) in a micro-grid in advance, namely, ESS will discharge 10% power every hour during the consecutive time of highest demand load, and charge 10% power every hour during the consecutive time of lowest demand load.

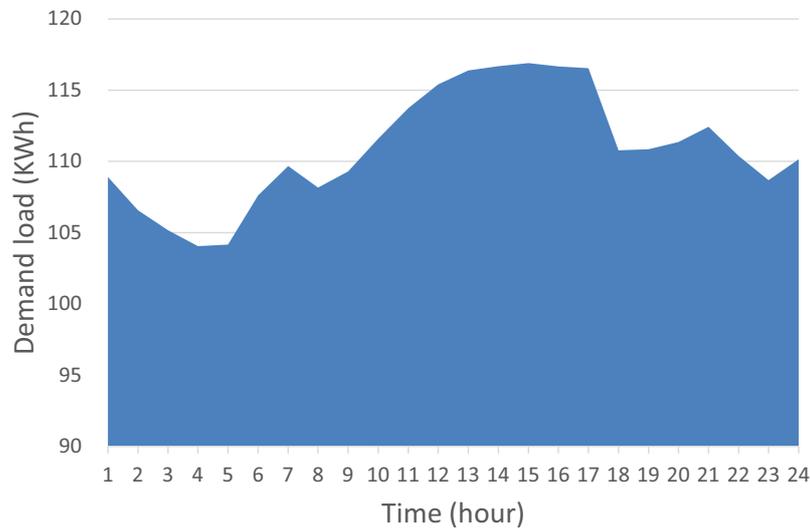


Figure 5.5: The demand load of a micro-grid in a day.

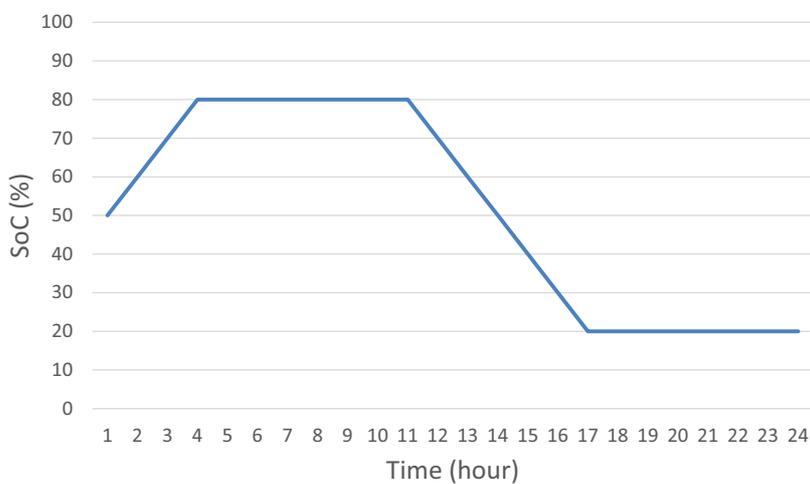


Figure 5.6: The SoC of ESS in a day with FR strategy.

As shown in Figure 5.5 and Figure 5.6, the initial SoC is 50% and the consecutive time of lowest demand load is from 1<sup>st</sup> hour to 7<sup>th</sup> hour, thus ESS will charge in the first three hours and maintains 80% SoC in the rest three hours. Further, the consecutive time of highest demand load is from 11<sup>st</sup> hour to 17<sup>th</sup> hour, thus ESS will discharge 10% power every hour until the lower bound is reached.

## 5.2 Evaluation of linear regression-based load prediction

Table 5.5 shows the parameter settings of load prediction method for the different number of temperature bounds, training weeks, and areas.

Table 5.5: Parameter settings for linear regression-based load prediction.

<b>Number of temperature bounds</b>	2~6
<b>Number of training weeks</b>	3~12
<b>Number of intervals in a day</b>	96
<b>Area</b>	Residential, industrial

We use Equation 5.2 to evaluate the accuracy of load prediction method, the lower *Inaccuracy* value is calculated, the predicted demand load is more accurate.

$$Inaccuracy = |Predicted\ load - historical\ load| \quad (5.2)$$

Furthermore, we use Equation 5.3 to choose proper acceptable parameter settings for load prediction.

$$InaccuracyRate(IR) = \frac{Inaccuracy}{historical\ load} \times 100\% \quad (5.3)$$

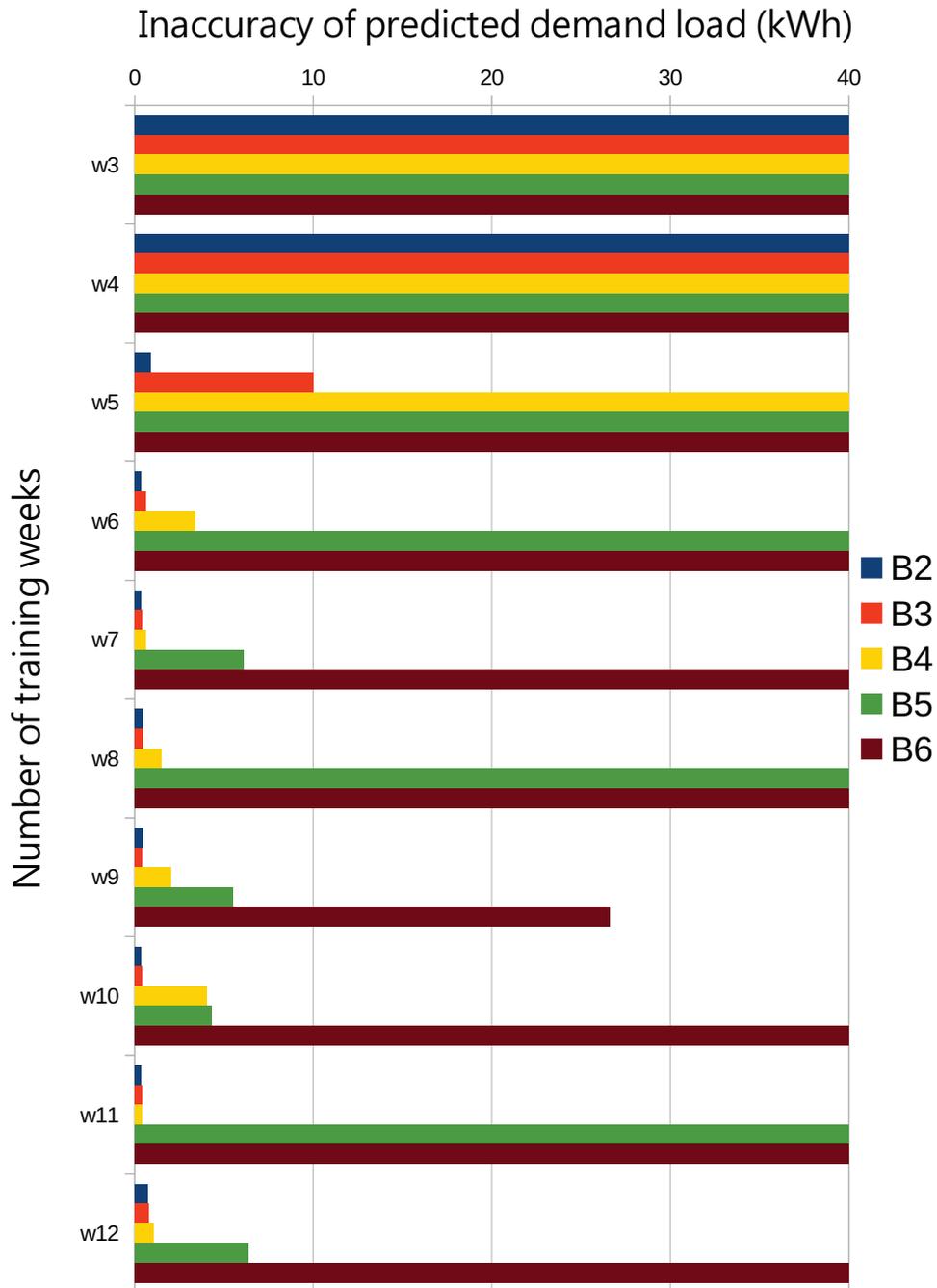


Figure 5.7: Inaccuracy of residential area.

Figure 5.7 shows *Inaccuracy* of residential area. The *Inaccuracy* from w3 to w5 is quite high no matter what temperature bound is used, whereas the *Inaccuracy* from w6 to w12 is relatively low when the temperature bounds are two or three. As shown in Table 5.6, the acceptable *IR* is marked with bold text, and all of marked *IR* is under 3%.

Figure 5.8 shows *Inaccuracy* of industrial area. The *Inaccuracy* from w3 to w7 is

Table 5.6: IR of residential area.

	B2	B3	B4	B5	B6
w3	376.55%	695.99%	601.10%	746.60%	950.53%
w4	1031.91%	827.24%	940.73%	723.09%	1092.23%
w5	3.41%	38.28%	660.17%	1433.29%	2219.81%
w6	<b>1.36%</b>	<b>2.29%</b>	12.91%	788.53%	851.95%
w7	<b>1.43%</b>	<b>1.50%</b>	2.46%	23.25%	957.73%
w8	<b>1.78%</b>	<b>1.84%</b>	5.79%	277.87%	750.82%
w9	<b>1.69%</b>	<b>1.62%</b>	7.85%	21.06%	101.48%
w10	<b>1.37%</b>	<b>1.51%</b>	15.50%	16.58%	278.73%
w11	<b>1.43%</b>	<b>1.59%</b>	1.57%	389.83%	511.60%
w12	<b>2.89%</b>	<b>2.98%</b>	4.07%	24.41%	389.84%

quite high no matter what temperature bound is used, whereas the *Inaccuracy* from w8 to w12 is relatively low when the temperature bounds are two or three. As shown in Table 5.7, the acceptable *IR* is marked with bold text, and all of marked *IR* is under 5%.

Table 5.7: IR of industrial area.

	B2	B3	B4	B5	B6
w3	5.92%	316.53%	766.07%	2072.81%	1619.39%
w4	5.15%	247.38%	828.60%	555.58%	1938.41%
w5	24.64%	201.81%	361.55%	1025.42%	2091.90%
w6	9.81%	9.96%	531.53%	1838.41%	1859.43%
w7	9.50%	5.43%	235.03%	1624.66%	1915.75%
w8	<b>1.85%</b>	<b>2.61%</b>	5.38%	1265.52%	1727.10%
w9	<b>2.20%</b>	<b>4.00%</b>	33.27%	967.33%	1571.28%
w10	<b>2.09%</b>	<b>4.20%</b>	6.10%	795.63%	1371.84%
w11	<b>1.74%</b>	<b>2.91%</b>	6.01%	673.60%	1430.71%
w12	<b>0.81%</b>	<b>1.05%</b>	6.60%	18.67%	569.56%

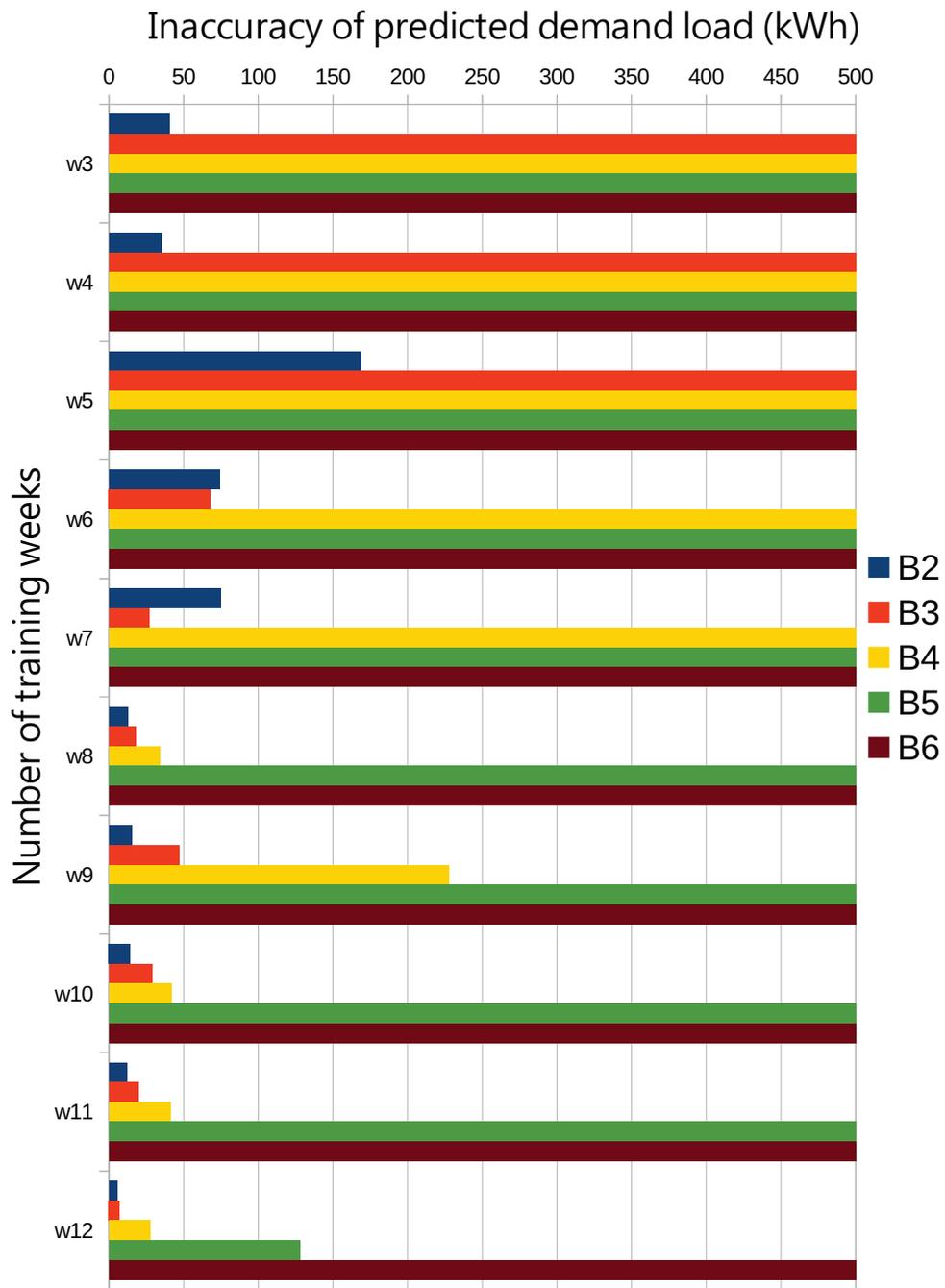


Figure 5.8: Inaccuracy of industrial area.

### 5.3 ESS Scheduling Experiment

Table 5.8: Parameter settings for MPC-based scheduling of ESS.

<b>Number of intervals in a day</b>	I24, I48, and I96
<b>Length of MPC optimization windows</b>	2~10 hours
<b>Area</b>	Residential
<b>Dynamic electricity price (\$)</b>	\$8.10~\$27.35

In this section, we use Equation 5.4 to evaluate how the proposed MPC-based scheduling improves the electricity bills with different optimization window, where  $Mbill$  represents the electricity bill under MPC-based scheduling and  $woESSbill$  represents electricity bill without the use of ESS. Table 5.8 lists the parameter settings for MPC-based scheduling of ESS including number of intervals in a day and length of MPC optimization windows.

$$Cost\ Reduction\ (CR) = \frac{|Mbill - woESSbill|}{woESSbill} \times 100\% \quad (5.4)$$

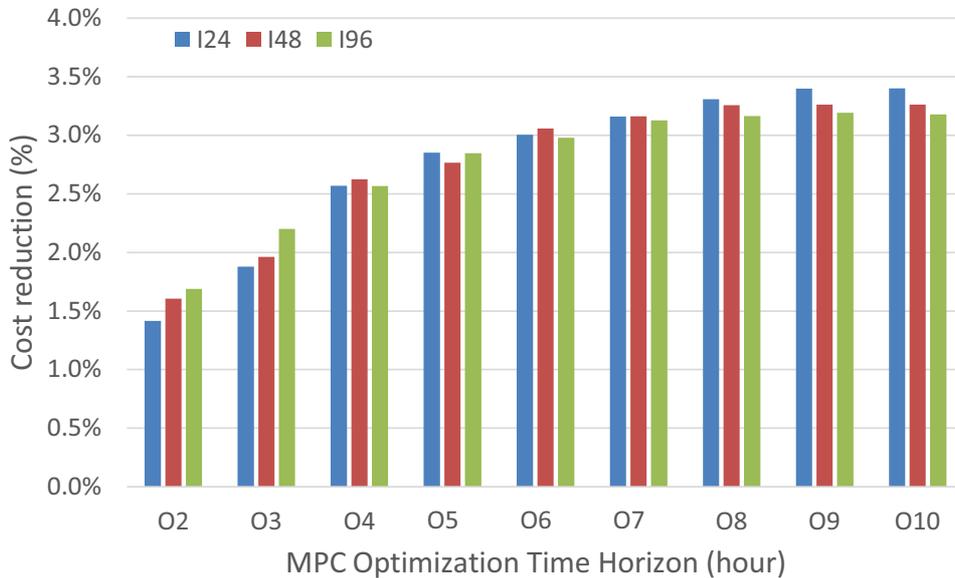


Figure 5.9: Cost reduction for different optimization time horizon.

As shown in Figure 5.9, cost reduction (CR) is increased with increase in the MPC optimization time horizon, O10 is 2.4 times greater than O2 with I24, O10 is 2.03 times greater than O2 with I48, O10 is 1.88 times greater than O2 with I96. But CR is totally

different for I24, I48, and I96 even with the same MPC optimization time horizon, for instance, when the time interval is made smaller, namely from 60 minutes (I24) to 15 minutes (I96), the more cost reduction increases. Take for an example, I24 is 1.04 times less than I48 and is 1.17 times less than I96 when MPC optimization time horizon is 3 hours. The numbers of time intervals that needs to be taken into consideration for MPC-based scheduling are as shown in Table 5.9. From Figure 5.9, we can conclude that the more the number of intervals considered, the more is the cost reduction.

Table 5.9: Number of intervals required for MPC-based scheduling with different optimization time horizons.

	<b>I24</b>	<b>I48</b>	<b>I96</b>
<b>2 hours</b>	2	4	8
<b>3 hours</b>	3	6	12
<b>4 hours</b>	4	8	16
<b>5 hours</b>	5	10	20
<b>6 hours</b>	6	12	24
<b>7 hours</b>	7	14	28
<b>8 hours</b>	8	16	32
<b>9 hours</b>	9	18	36
<b>10 hours</b>	10	20	40

But there are some irregular sign on cost reduction when MPC optimization time horizon increases, that is, the more the number of intervals considered, the less is the cost reduction. Take for an example, as shown in Figure 5.10 and Figure 5.11, due to the different charge operation during 19<sup>th</sup> to 21<sup>th</sup> hour results that the cost reduction of I24 is better than I48, as shown in 5.10 and Figure 5.12, due to the different charge operation during 8<sup>th</sup> to 12<sup>th</sup> hour results that the cost reduction of I24 is better than I96.

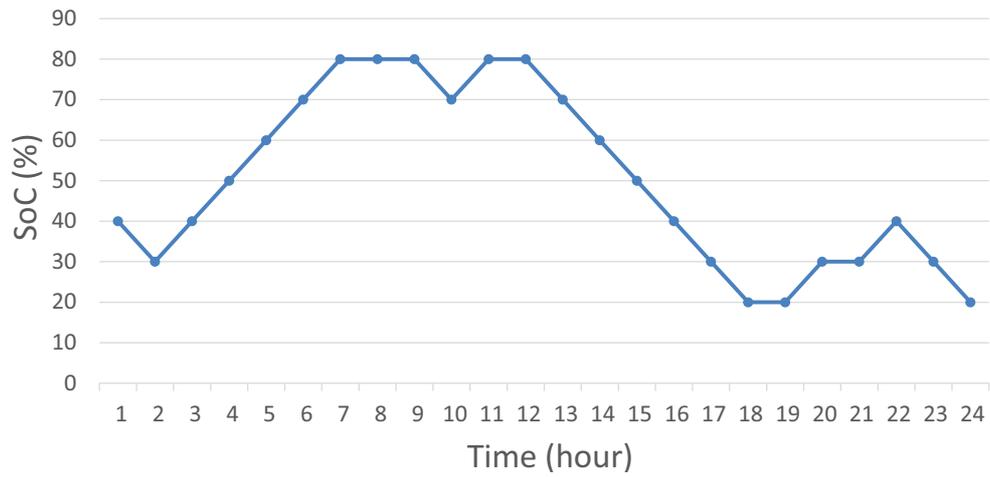


Figure 5.10: 24 intervals with 10 hours MPC optimization time horizon.

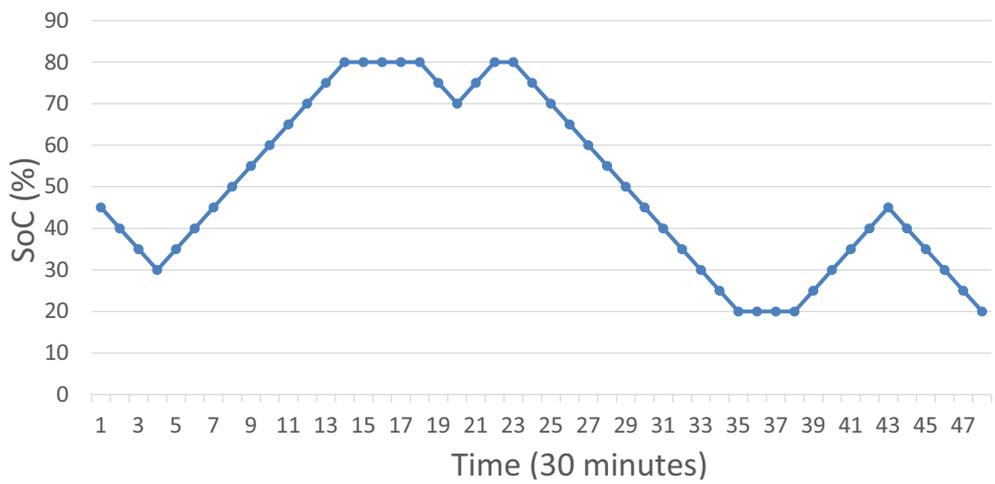


Figure 5.11: 48 intervals with 10 hours MPC optimization time horizon.

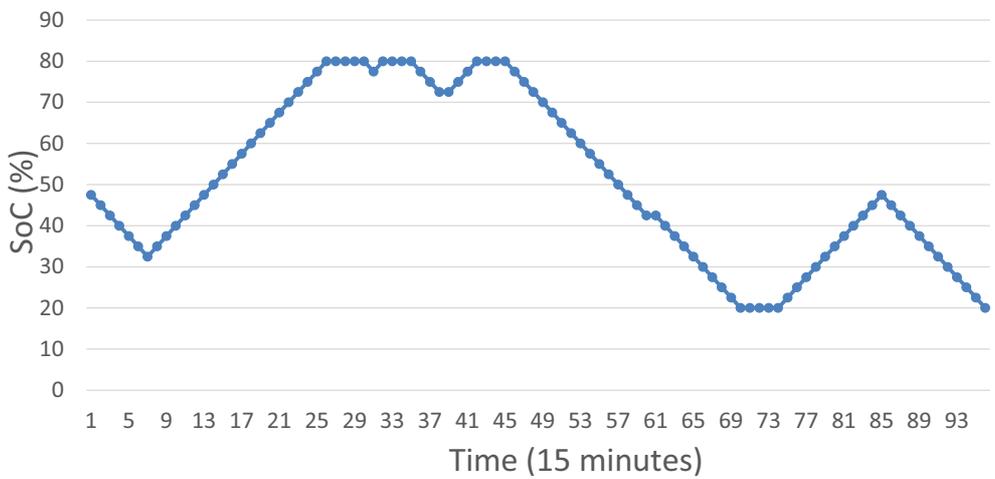


Figure 5.12: 96 intervals with 10 hours MPC optimization time horizon.

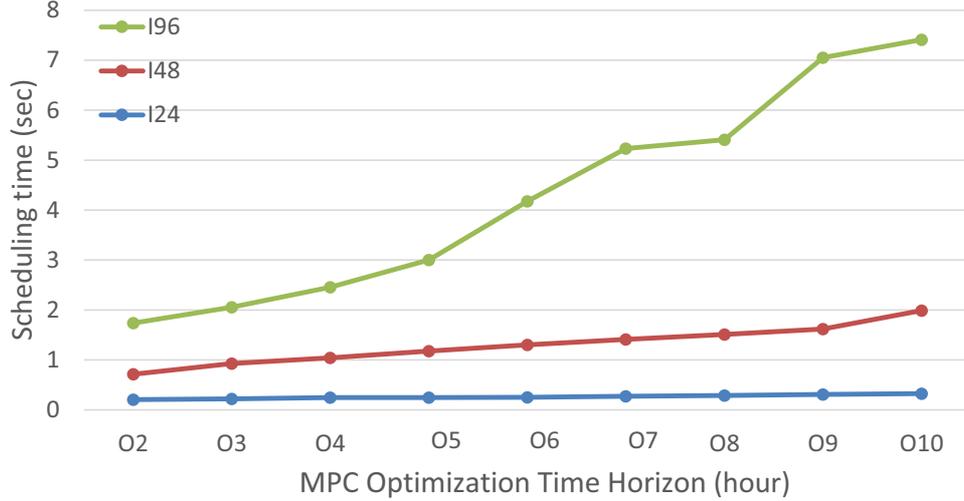


Figure 5.13: Scheduling time for different MPC optimization time horizons.

Furthermore, as shown in Figure 5.13, the scheduling time increases with increase in the MPC optimization time horizon, for instance, I24 is 4.83 times faster than I48 and is 15.5 times faster than I96 in average for all optimization time.

In summary, we suggest that a suitable MPC optimization time horizons is 10 hours and the number of intervals of a day is 24, in such case, the CR is highest and the execution time is relatively faster.

## 5.4 Trade-off between Cost Reduction and Lifetime of ESS

Table 5.10: Parameter settings for trade-off experiment.

<b>Runtime</b>	One week
<b>Area</b>	Residential
<b>Dynamic electricity price (\$)</b>	\$8.10~\$27.35

In this section, we will explain the trade off between the cost reduction on electricity bill and the lifetime of ESS. Table 5.10 lists the parameter settings in this experiment.

For estimating the lifetime of ESS, we use the Perkert lifetime energy throughput [1] (PLET) model to calculate the lifetime for different ESS management strategies including exhausted, FR, backup, and MPC as shown in Table 5.11.

Table 5.11: Charge/Discharge time for all ESS strategies.

Strategy	Charge	Discharge
<b>Exhausted</b>	During 4~10 hour, 16~22 hour	During 10~16 hour, 22~4 hour
<b>FR</b>	During 11~17 hour	During 1~7 hour
<b>Backup</b>	Surplus power	Power shortage
<b>MPC</b>	Off-peak time, surplus power	Peak time, Power shortage

$$cum_{PLET} = d^{\kappa_P} n \quad (5.5)$$

In Equation 5.5,  $cum_{PLET}$  represents the cumulative PLET battery loss, where  $d$  is depth-of-discharge (DOD),  $\kappa_P$  is the Peukert Lifetime constant (typically in the range 1.1 to 1.3), and  $n$  is the number of charge/discharge cycles.

$$Loss\ of\ Life\ (LoL) = \frac{cum_{PLET}^t}{C_{PLET}^{life}} \times 100\% \quad (5.6)$$

In Equation 5.6, LoL represents the loss-of-life during the usage at different DOD, where  $cum_{PLET}^t$  represents the cumulative PLET battery loss at different time  $t$ , and  $C_{PLET}^{life}$  represents the PLET total lifetime. The  $\kappa_P$  is set as 1.1 and the  $C_{PLET}^{life}$  is set as 100,000 in our experiment. We use Li-ion battery and its  $C_{PLET}^{life}$  is 102,760 [14].

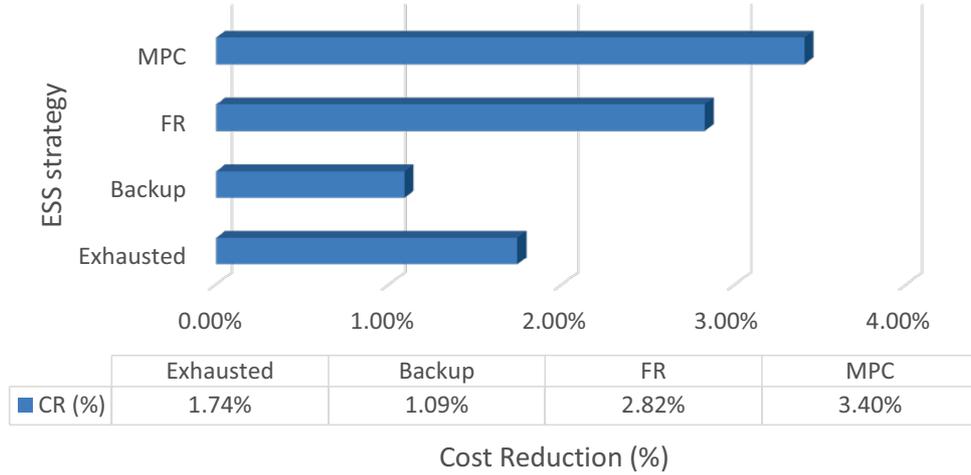


Figure 5.14: Cost reduction on electricity bill with different ESS management strategies.

As shown in Figure 5.14 and Figure 5.15, the MPC-based scheduling gives the highest cost reduction among all strategies, but the corresponding  $LoL$  is also the highest

one due to the strategy of MPC much as possible to alleviate the intermittency of renewable energy resources. MPC-based scheduling indeed can save cost on electricity bill, but a lot more power from ESS discharged during on-peak time or during the time periods of power shortage, thus the loss-of-life is significant. The FR strategy discharges ESS power only at on-peak time, and charges power at off-peak time. FR can indeed not only save more cost on electricity bill than the backup and exhausted strategies, but can also reduce the loss-of-life, which is the lowest among all strategies. The backup strategy only discharges power when the micro-grid is in power shortage, and charges power when the micro-grid has surplus power, the cost of electricity bill can not be reduced significantly in a such strategy because ESS is not charged at lower electricity prices and is not discharged power at higher electricity prices. The exhausted strategy uses ESS frequently, namely discharges power upto the preset lower bound and charges power upto the preset upper bound twice a day. As a result, the LoL of ESS in the exhausted method is quite high than the FR and the backup strategies but the cost reduction is not increased which is smaller than MPC and FR strategy.

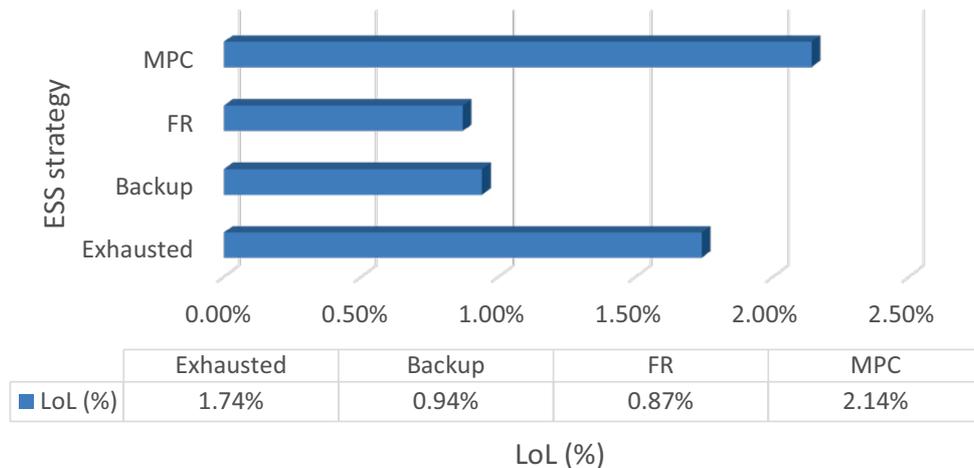


Figure 5.15: LoL of different ESS strategies.

## 5.5 MPC-based scheduling with bidding market

In this section, we apply the MPC-based scheduling of ESS for four MIAs with two auction mechanism including the first-price [22] [23] and the second-price [24] [25] sealed auction. The goal is cost saving on electricity bill. Table 5.12 lists the parameter settings used in our experiments.

Table 5.12: Parameter settings of bidding market.

<b>Number of MIAs</b>	4
<b>Runtime</b>	One week
<b>Area</b>	Residential
<b>Dynamic electricity price (\$)</b>	\$8.10~\$27.35

The cost saving is calculated by Equation 5.7, where  $aucbill$  represents the electricity bill with auction mechanism and  $woBIDbill$  represents electricity bill without the bidding market.

$$Cost\ saving\ (CS) = \frac{|aucbill - woBIDbill|}{woBIDbill} \times 100\% \quad (5.7)$$



Figure 5.16: Cost saving with MPC-based scheduling of ESS with the first-price and the second-price sealed auctions.

As shown in Figure 5.16, the cost saving (CS) is 35.25% in average with the first-price sealed auction and is 34.86% in average with the second-price sealed auction for all MIAs. The bidding market indeed brings more benefit on electricity bill. Table 5.13 lists the electricity bill of all MIAs with different auction mechanism.

On one hand, for the MIA with the amount of the generated power is greater than consumed power, the first-price sealed auction is suitable. On the other side, the second-price sealed auction is suitable for the MIA with the amount of demand power greater than the generated power. Take for an example, MIA 2 is the former one that gains more

Table 5.13: The electricity bill of different auction mechanism for all MIAs.

	<b>MIA 1</b>	<b>MIA 2</b>	<b>MIA 3</b>	<b>MIA 4</b>
<b>w/o bidding (\$)</b>	30240.6	31335.2	29208.7	32564.6
<b>First-price (\$)</b>	25734.3	17995.1	11103.6	25555.6
<b>Second-price (\$)</b>	23418.5	20986.6	15950.8	20032.8

Table 5.14: Amount of surplus power and shortage power for all MIAs.

	<b>MIA 1</b>	<b>MIA 2</b>	<b>MIA 3</b>	<b>MIA 4</b>
<b>+: Surplus power (kWh)</b>	280.7	1768.7	3035.6	320.1
<b>-: Shortage power (kWh)</b>	1912.9	597.0	1030.4	1864.8

benefits on bill for higher electricity price when selling surplus power and MIA 1 is the latter one that pays for the bill for cheaper electricity price when fulfilling the shortage power. Table 5.14 lists the amount of surplus power and shortage power for all MIAs.

# Chapter 6

## Conclusions and Future Work

In this Thesis, we have proposed a system model for smart grid architecture including micro-grid level, coordinate control level, and smart grid level and an Model-Predictive Control (MPC) based scheduling method for energy storage systems (ESS) in a micro-grid. The proposed MPC-based scheduling method can achieve an average of 3.40% cost reduction for electricity bill but the loss-of-life for ESS is 2.14% in one week which is higher than other ESS strategy including exhausted, backup, frequency regulation (FR). Further, with the participation in bidding market for all MIAs can achieve 35.25% cost saving in average with the first-price sealed auction and 34.86% cost saving in average with the second-price sealed auction.

The accuracy of load prediction method is bad when the number of temperature bounds are greater than 3, we will divide the temperature unequally according to the amount of power required by loads on different temperature bound, and thus improves the accuracy in the future. The MPC-based scheduling for ESS has a side-effect on loss-of-life, the ESS can not operate for a long time, and thus the cost for updating the ESS equipments increases as well. In the future, our work will try to take the loss-of-life of ESS into consideration, instead of reducing only the electricity bill. As a result, we will find the balance of saving the cost on electricity bill and extending the battery life for micro-grid. In the bidding market, we will use the different amount of power and fixed price rather the fixed amount of power and different price for auction mechanism, and saves more electricity cost.

# Bibliography

- [1] K. Nunna and S. Doolla, “Energy management in microgrids using demand response and distributed storagea multiagent approach,” *IEEE Transactions on Power Delivery*, vol. 28, no. 2, pp. 939–947, April 2013.
- [2] C.-X. Dou and B. Liu, “Multi-agent based hierarchical hybrid control for smart microgrid,” *IEEE transactions on smart grid*, vol. 4, no. 2, pp. 771–778, June 2013.
- [3] “Smart Grid,” <http://energy.gov/oe/technology-development/smart-grid>, Dec. 2008.
- [4] R. Lasseter, A. Akhil, C. Marnay, J. Stephens, J. Dagle, R. Guttromson, A. Meliopoulos, R. Yinger, and J. Eto, “The certs microgrid concept,” *White paper for Transmission Reliability Program, Office of Power Technologies, US Department of Energy*, April 2002.
- [5] I. Gyuk, P. Kulkarni, J. Sayer, J. Boyes, G. Corey, and G. Peek, “The united states of storage [electric energy storage],” *Power and Energy Magazine, IEEE*, vol. 3, no. 2, pp. 31–39, March 2005.
- [6] D. Manz, O. Schelenz, R. Chandra, S. Bose, M. de Rooij, and J. Bebic, *Enhanced reliability of photovoltaic systems with energy storage and controls*. National Renewable Energy Laboratory, 2008, [Online] Available: <http://www1.eere.energy.gov/solar/pdfs/42299.pdf>.
- [7] R. Fazal, J. Solanki, and S. K. Solanki, “Demand response using multi-agent system,” in *In Proceedings of the North American Power Symposium (NAPS)*. IEEE Press, Sept. 2012, pp. 1–6.

- [8] J. Martinez, V. Dinavahi, M. Nehrir, and X. Guillaud, "Tools for analysis and design of distributed resources—part IV: Future trends," *IEEE Transactions on Power Delivery*, vol. 26, no. 3, pp. 1671–1680, July 2011.
- [9] N. Kumar and S. Doolla, "Multiagent-based distributed-energy-resource management for intelligent microgrids," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 4, pp. 1678–1687, April 2013.
- [10] S. Kahrobaee, R. A. Rajabzadeh, L.-K. Soh, and S. Asgarpoor, "A multiagent modeling and investigation of smart homes with power generation, storage, and trading features," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 659–668, June 2013.
- [11] R. Fazal, J. Solanki, and S. K. Solanki, "Demand response using multi-agent system," in *Proceedings of the North American Power Symposium (NAPS), 2012*. IEEE, Sept. 2012, pp. 1–6.
- [12] "JADE agent development toolkit," <http://jade.tilab.com/>, 2014.
- [13] "Standard FIPA specifications," <http://www.fipa.org/repository/standardspecs.html>, 2014.
- [14] D. Tran and A. M. Khambadkone, "Energy management for lifetime extension of energy storage system in micro-grid applications," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1289–1296, Sept. 2013.
- [15] D. Tran, H. Zhou, and A. M. Khambadkone, "Energy management and dynamic control in composite energy storage system for micro-grid applications," in *Annual Conference on IEEE Industrial Electronics Society*. IEEE, Nov. 2010, pp. 1818–1824.
- [16] C. Chen, J. Wang, Y. Heo, and S. Kishore, "MPC-based appliance scheduling for residential building energy management controller," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1401–1410, Sept. 2013.
- [17] D. Li and S. K. Jayaweera, "Uncertainty modeling and prediction for customer load demand in smart grid," in *2013 IEEE Energy Technology Conference*. IEEE, May 2013, pp. 1–6.

- [18] J. L. Mathieu, P. N. Price, S. Kiliccote, and M. A. Piette, “Quantifying changes in building electricity use, with application to demand response,” *IEEE Transactions on Smart Grid*, vol. 2, no. 3, pp. 507–518, Sept. 2011.
- [19] “Gurobi Optimizer - State-of-the-Art Mathematical Programming Solver,” 2014, <http://www.gurobi.com>.
- [20] “ERCOT-load profiling,” 2014, <http://www.ercot.com/mktinfo/loadprofile/>.
- [21] S. Beer, T. Gomez, D. Dallinger, I. Momber, C. Marnay, M. Stadler, and J. Lai, “An economic analysis of used electric vehicle batteries integrated into commercial building microgrids,” *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 517–525, March 2012.
- [22] R. Duan and G. Deconinck, “Future electricity market interoperability of a multi-agent model of the smart grid,” in *Proceedings of the International Conference on Networking, Sensing and Control (ICNSC)*. IEEE, April 2010, pp. 625–630.
- [23] L. Gkatzikis, I. Koutsopoulos, and T. Salonidis, “The role of aggregators in smart grid demand response markets,” *IEEE Journal on Selected Areas in Communication*, vol. 31, no. 7, pp. 1247–1257, July 2013.
- [24] S. Y. Rashida and H. Navidi, “Knowledge-based scheduling resource into grid system by second-price auctions with best response dynamics,” *World Applied Sciences Journal*, vol. 21, no. 3, pp. 351–359, March 2013.
- [25] B. HomChaudhuri and M. Kumar, “Market based allocation of power in smart grid,” in *Proceedings of the American Control Conference (ACC), 2011*. IEEE, June 2011, pp. 3251–3256.