Demand Side Management for the Future Smart Grid

by

Pedram Samadi Dinani

B.Sc., Isfahan University of Technology, Isfahan, Iran, 2006
M.Sc., Isfahan University of Technology, Isfahan, Iran, 2009

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

The Faculty of Graduate and Postdoctoral Studies

(Electrical and Computer Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA
(Vancouver)

March 2015

Abstract

To achieve a high level of reliability and robustness in power systems, the grid is usually designed for the peak demand rather than the average demand. This usually results in an under-utilized system. Demand side management (DSM) programs can be adopted to shape the load pattern of the users to better utilize the available power generation capacity and to prevent installing new generation and transmission infrastructures. In this thesis, we propose different algorithms for DSM. First, we focus on the problem of maximizing the social welfare of the users. We consider a scenario where the users are equipped with automated control units and are able to make price-responsive decisions. We propose a Vickrey-Clarke-Groves (VCG) mechanism to maximize the social welfare of the users. Subsequently, we focus on developing a novel automated load scheduling algorithm to minimize the energy expenses of the user. The proposed algorithm takes into account the effects of the load uncertainties in future time slots. Moreover, the operational constraints of different types of appliances including must-run appliances, and interruptible and non-interruptible controllable appliances are studied. Next, we study how the utility company can set price values for different times of a day such that the peak-to-average ratio (PAR) of the load is minimized. We also consider the effects of the uncertainty regarding the price-responsiveness of the users. To simulate the likely behavior of the users in response to different price values for different times of the day, we propose the use of a system simulator unit. We propose two pricing algorithms based on stochastic approximation aiming to minimize the PAR of the aggregate load. Finally, we consider systems with

high penetration of renewable energy resources. To tackle the reverse power flow problem associated with these systems, we propose a joint load scheduling and trading algorithm. This algorithm encourages the users to sell their excess generation to their neighboring users which mitigates the reverse power flow problem.

Preface

Hereby, I declare that I am the first author of this thesis. Chapters 2-5 encompass work that has been published or is under review. The corresponding papers are co-authored by Dr. Vincent Wong and Dr. Robert Schober who supervised me through this research. The papers corresponding to Chapters 2, 3, and 4 were also co-authored by Dr. Amir-Hamed Mohsenian-Rad, who provided valuable comments on these works. The following publications describe the work completed in this thesis.

Journal Papers, Published

- Pedram Samadi, Hamed Mohsenian-Rad, Vincent W. S. Wong, and Robert Schober, "Real-Time Pricing for Demand Response Based on Stochastic Approximation," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 789-798, March 2014.

- Pedram Samadi, Hamed Mohsenian-Rad, Vincent W. S. Wong, and Robert Schober, "Tackling the Load Uncertainty Challenges for Energy Consumption Scheduling in Smart Grid," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 1007-1016, June 2013.

- Pedram Samadi, A. Hamed Mohsenian-Rad, Robert Schober, and Vincent W. S. Wong, "Advanced Demand Side Management for the Future Smart Grid Using Mechanism Design," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1170-1180, September 2012.

Journal Paper, Submitted

- Pedram Samadi, Vincent W. S. Wong, and Robert Schober, "Local Scheduling and Power Trading in Systems with High Penetration of Renewable Energy Resources," submitted, November 2014.

Conference Papers, Published

- Pedram Samadi, Hamed Mohsenian-Rad, Vincent W. S. Wong, and Robert Schober, "Utilizing Renewable Energy Resources by Adopting DSM Techniques and Storage Facilities," in *Proc. of IEEE International Conference on Communications (ICC)*, Sydney, Australia, June 2014.

- Pedram Samadi, Hamed Mohsenian-Rad, Vincent W. S. Wong, and Robert Schober, "Adaptive Energy Consumption Scheduling with Load Uncertainty for the Smart Grid," in *Proc. of IEEE International Conference on Communications (ICC)*, Budapest, Hungary, June 2013.

- Pedram Samadi, Robert Schober, and Vincent W. S. Wong, "Optimal Energy Consumption Scheduling Using Mechanism Design for the Future Smart Grid," in Proc. of *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Brussels, Belgium, October 2011.

- Pedram Samadi, Hamed Mohsenian-Rad, Robert Schober, Vincent W. S. Wong, and Juri Jatskevich, "Optimal Real-time Pricing Algorithm Based on Utility Maximization for Smart Grid," in *Proc. of IEEE International Conference on Smart Grid Communications (SmartGridComm)*, Gaithersburg, Maryland, October 2010.

Book Chapter

- Pedram Samadi, Hamed Mohsenian-Rad, Vincent W. S. Wong, and Robert Schober, "Demand Side Management for Smart Grid: Opportunities and Challenges," in *Smart Grid Communications and Networking*, Edited by Vincent Poor, Zhu Han, and Ekram Hossain, Cambridge University Press, 2011.

Table of Contents

Appendices

List of Tables

List of Figures

List of Abbreviations

AMI	Advanced metering infrastructure
DAP	Day-ahead pricing
DG	Distributed generator
DLC	Direct load control
DSM	Demand side management
ECC	Energy consumption controller
FDPS	Finite-deference price selection
GBD	Generalized Benders decomposition
IBR	Inclining block rates
IPM	Interior point method
MCP	Market clearing price
PAR	Peak-to-average ratio
PC	Personal computer
PCU	Price control unit
PEV	Plug-in electric vehicle
PHEV	Plug-in hybrid electric vehicle
PLP	Peak load pricing
RER	Renewable energy resource
RTP	Real-time pricing

SPPS	Simultaneous perturbation price selection
SPSP	Simultaneous perturbation stochastic approximation
SSU	System simulator unit
TOU	Time-of-use (pricing)
TV	Television
VCG	Vickrey-Clarke-Groves (mechanism)

Acknowledgments

First of all, I want to express my deepest sense of gratitude to my supervisors Dr. Vincent Wong and Dr. Robert Schober without whom this research would not be possible. I want to thank them for their continuous support, patience, technical and non-technical guidance that helped me accomplish this research. I also want to thank them for their understanding of the conditions and problems that I was involved with as a new international student.

I want to declare my special thanks to Dr. Amir Hamed Mohsenian-Rad for his enormous help, advice, and comments through my research. It was not possible to accomplish some parts without his very helpful advice.

I am also thankful to my best friends Keivan Ronasi, Ahmad Ashouri, Nasim Arianpoo, and Anahita Keshavarzi for their always being supportive and accountable. I want to thank Ali, Nastaran, Paria, Mahsa, Pooya, Azadeh, Milad, Bahar, Farid, Gelareh, Nasim, Bahareh, Neshat, Adam, Toufiqul, Imtiaz and all my other friends that definitely I cannot name them all here.

Most importantly, I feel indebted to my great family, my father, my mother and my brothers for their everlasting love, support, and protectivity not only in years of doing PhD but also throughout my entire life.

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

Chapter 1

Introduction

Demand side management (DSM) is one of the key components of the future smart grid to enable more efficient and reliable grid operation. To achieve a high level of reliability and robustness in power systems, the grid is usually designed for the peak demand rather than the average demand. This usually results in an under-utilized system. With the increasing expectations of the customers both in quantity and quality of the electric power [1], the emergence of new types of loads such as plug-in hybrid electric vehicles (PHEVs) which can potentially double the average residential load, the limited energy resources, and the lengthy and expensive process of exploiting new resources, there is an essential need to improve the utilization of the power grids [2]. DSM programs comprise different techniques employed to better utilize the available power generation capacity and to prevent installing new generation and transmission infrastructures. These programs typically aim at one or both of the following design objectives: *reducing consumption* and *shifting consumption*. Adopting energy-aware consumption patterns and constructing energy efficient buildings can reduce the energy consumption of users. However, there is also a need to develop practical methods to encourage users to shift their usage of high-power household appliances to off-peak hours.

In the current power systems, there is also a high concern regarding environmental issues. To reduce the emission of greenhouse gases, the deployment of more environmentally friendly and sustainable distributed renewable energy resources (RERs) such as wind and solar attracts more attention. Despite the benefits obtained by integrating RERs into the

distribution system including improved reliability, power quality, and reduced distribution loss, the management of RERs is challenging. Considering the intermittent and random nature of RERs, it is difficult to match supply and demand. Utilizing backup power and energy storage facilities are among the well-known solutions which have been proposed to tackle this problem [3, 4]. Backup power plants are usually fast acting power generators. However, the cost of operation and maintenance of these facilities is very high, and the environmental impact of these power plants is significant. An alternative approach is to use DSM to shape the daily energy consumption of the users in order to reduce the gap between supply and demand [5–8].

In this thesis, we propose different algorithms for DSM purposes. The proposed algorithms aim at achieving different objectives including maximizing the social welfare, minimizing the peak-to-average ratio (PAR) of the aggregate load, and minimizing the energy expenses of the users. Mathematical tools, such as optimization theory, mechanism design, game theory, and dynamic programming are applied in the design and analysis of these algorithms. The rest of this chapter is organized as follows. In Section 1.1, we provide an overview of different techniques considered for DSM. We summarize the main contributions and results of this thesis in Section 1.2. Finally, the organization of the thesis is described in Section 1.3.

1.1 Demand Side Management Techniques

DSM has been considered since the early 1980s [9–13]. Most power utilities prefer to think of DSM as a tool for load shaping, where the electricity demand is re-distributed over a certain period of time (e.g., time-of-day, day-of-week). Broad categories of load shaping objectives include peak clipping, load shifting, valley filling, strategic conservation, and flexible load shaping [9]. For example, peak clipping includes reducing the customers'

load at the peak hours. Load shifting consists of shifting load from the peak hours to the off-peak periods. Direct load control (DLC) and price-based load control are two general categories of DSM programs. In DLC programs, based on a contract between the energy provider and the users, the energy provider can remotely control the operation and energy consumption of certain appliances for users [14–20]. DLC programs can be used effectively where there is a centralized control over the operation of different loads, batteries, and alternative storage facilities. Examples of such scenario include microgrids that can be managed autonomously. In contrast, in price-based programs, the energy provider provides economic incentives to consumers by reflecting the hourly changes in the wholesale electricity price to the demand side such that users are encouraged to consume wisely [21–26].

Wholesale prices (i.e., the prices set by the generators to regional electricity retailers through a bidding process) vary drastically between the low-demand times of a day and the high demand periods. In particular, the electricity prices are lower at low demand hours and higher at high demand hours. However, these changes in the wholesale prices are currently unnoticed by end users in most regions. That is, users are usually charged with some average prices, and thus, there is no incentive for them to change their power consumption to utilize the available generation capacity efficiently. Mapping the wholesale electricity prices to the retail users can mitigate this problem and encourage users to conserve energy at high demand hours or shift the operation of their high load appliances from peak hours to off-peak hours.

Several time differentiating pricing methods have already been proposed in the literature. In time differentiating pricing tariffs, the intended operation period is divided into several time slots, and the price of electricity varies across time slots. For example, the prices may correspond to off-peak, mid-peak, and on-peak hours. The prices are usually

higher in the afternoon on hot days in the summer and cold days in the winter. Examples of time differentiating pricing include real-time pricing (RTP), time-of-use (TOU) pricing, and day-ahead pricing (DAP) [22, 23]. These methods mainly differ in how frequently the utility company changes the pricing tariffs, which may vary from once or twice a year in TOU pricing to hourly changes in RTP.

1.1.1 DSM Challenges and Opportunities

Extracting Local Information

With the growing deployment of *advanced metering infrastructure* (AMI) [27] and automated energy consumption controller (ECC) devices [11, 28, 29], RTP is gradually becoming a feasible DSM solution. In general, it is difficult for power users to follow the RTP price variations to make appropriate decisions accordingly. In this regard, ECC devices can help by making such *price-responsive* decisions on behalf of users to achieve certain objectives. Examples of such objectives include minimizing the energy expenses [11], maximizing the social welfare [28, 29], minimizing both the energy expenses and the waiting time [12], and maintaining system stability with minimum curtailment [30]. However, achieving social objectives (e.g., maximizing social welfare) is difficult because of its computational complexity. It also requires collecting various information about the energy consumption behavior of the users, the price elasticity of the users, and the benefit that each user obtains by consuming a certain amount of energy. However, in general, users are not willing to reveal such information, unless there is an incentive for them to do so. Therefore, elaborate design rules (*mechanisms*) are needed such that it is in each user's self interest to reveal its local information.

In Chapter 2, we propose an efficient pricing method to tackle this problem. We assume that each user is equipped with an ECC as part of its smart meter. All smart meters are

connected to not only the power grid but also a communication infrastructure. This allows two-way communication among smart meters and the utility company. We analytically model each user's preferences and energy consumption patterns in form of a *utility function*. Based on this model, we propose a Vickrey-Clarke-Groves (VCG) mechanism which aims to maximize the *social welfare*, i.e., the aggregate utility functions of all users minus the total energy cost. Our design requires that each user provides some information about its energy demand. In return, the energy provider will determine each user's electricity bill payment. The structure of electricity payment for each user is such that users are willing to reveal their local demand information to the energy provider.

Load Uncertainty

To shape the load pattern of the users, the energy provider sets different price values for different times of a day. The ECC device of each user retrieves the price information from the energy provider, and based on the constraints set by the user, schedules the operation of the appliances such that the energy expenses of the user is minimized. In practice, the information about the list of the appliances that need to be scheduled is not known at the beginning of the day, and the decision whether to use some appliances or not is made throughout the day. Moreover, the constraints on the operation of individual appliances are decided by the user and may vary from time to time. On the other hand, making decisions about the operating state of different appliances for the current time slot depends on the information about the user's energy needs available at the current time slot and the expected schedule determined in the upcoming time slots. The scheduling problem with load uncertainty usually includes solving a complex problem in the form of mixed-integer programming or dynamic programming. However, in situations where computational complexity or convergence time of the algorithm are critical such as in real-time application, a suboptimal but faster schemes that establish a balance between simple

implementation and adequate performance are preferable.

In Chapter 3, we propose a novel optimization-based real-time residential load management algorithm that takes into account *load uncertainty* in order to minimize the energy payment for each user. The algorithm takes into account different types of constraints on the operation of different appliances such as must-run appliances, controllable appliances that are interruptible, and controllable appliances that are not interruptible.

Price Responsiveness

The level of success for different pricing methods depends on various factors such as the amount of information being provided to each user, the effectiveness of the mapping of the wholesale prices to the retail prices, and the knowledge and abilities of the users to respond to price information. Another factor is the effectiveness of the home automation system. While the use of ECC devices improves users' *rationality* in response to price changes, such ECC devices can also introduce new DSM challenges such as *load synchronization* [12]. Load synchronization is referred to as the concentration of a large portion of energy consumption in low-price hours. Therefore, the effect of automated ECC devices on users' price-responsiveness is not obvious and yet to be investigated. That is, to set price values for different times of the day, the energy provider needs to take into account the level of price-responsiveness of the users.

It has been shown that load synchronization can be prevented by using pricing tariffs with *inclining block rates* (IBR). For the IBR tariffs, the marginal price increases with the total consumed power [31]. That is, beyond a predetermined power consumption threshold, electricity is offered at higher rates. This provides incentives for the users to distribute their load across different times of the day. Southern California Edison and Pacific Gas & Electric in United States, and British Columbia Hydro in Canada currently use IBR with various two-level conservation rate structures [32, 33]. In Chapter 3, we consider

RTP combined with IBR tariffs. We also study the effect combined RTP and IBR tariffs on the electricity payment of the users as well as on the their load pattern. Moreover, in Chapter 4, we extend this idea and propose a new pricing algorithm to minimize the PAR in aggregate load demand. The key challenge that we seek to address in Chapter 4 is the energy provider's uncertainty about the impact of prices on users' load profiles, in particular when users are equipped with automated ECC devices.

Reverse Power Flow

Integrating more environmentally friendly RERs into the power grid is one of the main features of the smart grid. The renewable portfolio standard requires the utility companies and energy providers in the United States and the United Kingdom to serve a specific minimum amount of their customers' load with RERs [34]. RERs such as solar and wind are non-dispatchable, since they are random in nature. In systems with high penetration of RERs, the power may flow from the distributed generators (DGs) to the substation, which negatively impacts the stability of the system. If the reverse power flow exceeds a certain threshold, it causes the *voltage rise* problem, which is a major challenge in integrating a large number of DGs in the distribution network [35–37].

To tackle the reverse power flow problem, it is desirable that users consume their generating power locally rather than injecting the excess power back into the grid. Storage facilities and DSM techniques can be adopted to shape the load pattern of the users to better match supply and demand [3, 4, 24–26, 38, 39]. That is, DSM techniques can be adopted to encourage users to shift their load to time slots with high generation from RERs. In Chapter 5, we consider the problem of load scheduling in systems with high penetration of RERs. To mitigate the problem of reverse power flow, we assume that users are able to sell their excess power to other users. On the one hand, local power trading can benefit the users by providing monetary revenue for them. On the other hand, it helps

to absorb excess power and to mitigate the reverse power flow problem.

1.1.2 Related Works

Considering the advances of AMI and two-way communication capabilities of smart grid, it becomes possible to seek different objectives for the power grid. Examples of such objectives include maximizing the social welfare, minimizing the electricity expenses of the users, and minimizing the PAR of the aggregate load. Li *et al.* [29] adopted the concept of utility functions to quantify the amount of benefit that each user obtains from the operation of different appliances. They argued that the obtained benefit depends on the volume and the pattern of the power consumption of the appliance. The authors proposed a scheduling algorithm which maximizes the obtained benefit subject to various operational constraints. The authors in [40] also considered the benefit that each user obtains from different services. They proposed a control algorithm to schedule the available distributed energy resources such that the obtained benefit is maximized. To tackle the computational complexity of the algorithm, particle swarm optimization is adopted to solve the corresponding problem. Conejo *et al.* [16] considered the problem of load scheduling to maximize the utility of the consumer. They proposed a linear programming algorithm which takes into account the constraints regarding the minimum daily energy requirements of the user, the minimum and the maximum load levels within a one hour time slot, and the ramping limits for such load levels. A robust optimization technique is adopted to model the uncertainty about the price values in each time slot. The authors in [12] proposed a load scheduling algorithm which attempts to achieve a trade-off between minimizing the electricity payment and minimizing the waiting time for the operation of the appliances. The proposed algorithm takes into account the uncertainty about the price values in the upcoming time slots. To predict the price values, a weighted average filter is adopted, and its coefficients are adjusted

based on real data. Kim *et al.* [41] considered the problem of power scheduling with price uncertainty. The key assumption in [41] is that the price values are independent from the load level in each time slot. As a result, the algorithm in [41] is based on a Markov decision process, where the decision on the operation of each appliance is made independently from other appliances. The authors in [42] proposed a heuristic algorithm which schedules the operation of the major appliances in a residential unit. The proposed algorithm benefits from a simple implementation. Moreover, it tries to balance the user comfort and the consumer price preferences. Xiong *et al.* [26] devised a scheduling algorithm which aims to keep the total power consumption below a target value. The proposed setting in [26] is suitable for DSM programs in which due to the complexity of the algorithm a top-down control approach is devised. That is, first, the total power consumption of each user in each time slot is determined. This is referred to as user level control. Then, each user tries to schedule the operation of its own appliances to meet the desired power consumption level. This is referred to as appliance level control. The authors in [30] proposed a multi-objective scheduling algorithm. The proposed algorithm adopts binary particle swarm optimization technique to schedule the operation of interruptible loads. The design aims to achieve the level of curtailment required by the system while satisfying the operational constraints of the interruptible loads. In addition, the proposed algorithm minimizes the joint payments of the users and the frequency of the load interruptions. In [43], Kishore *et al.* devised a two-stage control algorithm. In the first stage, based on price values and the preferences of the user, the operational schedule of different appliances is determined. In the second stage, neighboring users coordinate to reduce the peak demand.

Two-way communication capabilities of the smart grid makes it possible that users and utility companies interact with each other to provide users with improved customer services. Some of the works in the literature adopt game theoretic approaches to model

this interaction. Mohsenian-Rad *et al.* [11] devised a game theoretic approach to model the interaction between users. In their proposed system model, each user schedules its load to minimize its energy expenses. On the other hand, the utility company adopts a time differentiating pricing tariff which also takes into account the aggregate load level of the users. The price values are set by the utility company such that the generation cost or the PAR of the load is minimized. In [28], Chen *et al.* devised a Stackelberg game approach in which the energy provider acts as a leader and users are followers. The algorithm in [28] requires detailed information about users' energy consumption needs. This design intends to jointly maximize the social welfare of all users and the revenue of the energy provider. While most of the existing works focused on a scenario with a single utility company, the authors in [44] considered a system with multiple utility companies. In this study, the interaction between utility companies is modeled as a non-cooperative game, while an evolutionary game model is adopted to model the interaction between the users. The authors also studied the criteria required for the convergence of both games. The authors in [45] studied the impact of the consumers' price elasticity on the performance of the electricity market. In this study, users are modeled as agents who are well equipped with smart grid technologies. The authors concluded that the impact of price elasticity depends on the price level at which users are consuming energy. Moreover, users can benefit more if they are more price responsive.

Concerns about environmental issues and the need to reduce greenhouse gas emissions have attracted considerable attention to more environmentally friendly RERs. In the literature, different DSM programs are proposed to deal with uncertainties associated with RERs. The authors in [46] considered the effects of storage units as independent entities on the electricity market. They formulated the problem of selecting optimal energy and reserve bids for the storage units as a stochastic program that takes into account the

fluctuations of market prices due to the random nature of RERs. In [47], to deal with the uncertainties associated with RERs in matching supply and demand, a bidding strategy based on robust optimization is proposed. The authors in [47] showed that as the forecast error in the electricity price increases, the bidding strategy based on robust optimization outperforms the bidding strategy based on a deterministic approach. Kahrobaee *et al.* [48] proposed a system model in which users are modeled as independent agents making rational decisions to buy, sell, or store electricity. To make such decisions, the agents take into account their current load and generation, and their expected load and generation in the upcoming time slots. The authors also showed that the objective of the users to minimize their electricity expenses is aligned with the objective of the energy provider to flatten the load curve. In [49], Yang *et al.* proposed a DSM program in which users are equipped with RERs. Each user schedules the operation of its appliances to minimize the electricity payment. Moreover, users are able to sell their excess generation back to the grid.

1.2 Summary of Results and Contributions

This thesis covers several algorithms for DSM. The results are divided into four chapters. The contributions in each chapter are as follows:

- In Chapter 2, a DSM program is proposed to encourage efficient energy consumption among users. The proposed algorithm maximizes the aggregate utility of all users and minimizes the total cost imposed on the energy provider. We focus on systems where users are equipped with automated control units with an enhanced level of rationality. That is, users no longer accept the price as a fixed parameter and they can anticipate the impact of their actions on price values. Moreover, we assume that users have local information, and this information is essential for social planner to achieve the

maximum social welfare. We propose a VCG mechanism to elicit local information from rational users. We investigate some of the desired properties of our problem formulation including *truthfulness*, *efficiency*, and *nonnegative transfer*. We compare our efficient VCG DSM method with the case where users are price taker. We study the differences of these two systems, especially from the user payment perspective, and show that for the VCG mechanism users have to pay less. Simulation results confirm that both the users and the energy provider will benefit from the proposed scheme. The work in Chapter 2 is published in [50].

- In Chapter 3, we propose a real-time residential load management algorithm. The proposed algorithm takes into account the uncertainty about the future load demand, and it does not require the perfect knowledge of users' energy needs. Our algorithm aims at minimizing the electricity payment of residential users. Our design is multi-stage. As the demand information of the appliances is gradually revealed over time, the operation schedule of controllable appliances is updated accordingly. We study operation constraints to model a variety of appliances including must-run appliances, and interruptible and non-interruptible controllable appliances. Simulation results show that our proposed scheduling algorithm with load uncertainty reduces the energy payment of users compared to the case where no scheduling algorithm is adopted. Our proposed scheme also improves the overall power system performance by reducing the PAR in aggregate load demand. The work in Chapter 3 is published in [51].

- In Chapter 4, we address the problem of minimizing the PAR in the aggregate load demand through appropriately selecting price values for different time slots. The proposed algorithms take into account the uncertainty about the level of price-responsiveness of the users. We propose two iterative algorithms to be implemented

in a *price control unit* (PCU) based on the information provided by the *system simulator unit* (SSU). The first algorithm, called finite-difference price selection (FDPS), uses a variation of the finite-difference technique [52] to approximate the gradient of the PAR minimization objective function. The second algorithm, called simultaneous perturbation price selection (SPPS), is based on the simultaneous perturbation technique [52]. Moreover, an approximate dynamic programming scheme is proposed to simulate the likely behavior of the users in response to various price values at different time slots. Simulation results show that our proposed pricing algorithms reduce the PAR in aggregate load. In addition, we show that adopting the new pricing algorithms is also beneficial for the users if they are equipped with automated control units. The work in Chapter 4 is published in [53].

- In Chapter 5, we focus on the problems of load scheduling and power trading in systems with high penetration of RERs. We consider a game theoretic approach to model the interaction of users with excess power generation. Users compete to sell their extra generation to other local users. The revenue of competing users depends on their own marginal cost and the offers advertised by other competing users. Thus, each competing user chooses its offered price and output generation such that its revenue is maximized. We formulate the problem of selecting the offered price and output generation (i.e., the trading problem) as a linear mixed-integer program. To tackle the complexity of the trading problem, we adopt the generalized Benders' decomposition approach. Moreover, an approximate dynamic programming approach is adopted to schedule the operation of different types of appliances including must-run appliances, and interruptible and non-interruptible controllable appliances. Simulation results show that our proposed algorithm reduces the energy payment of users compared to the case where scheduling is not applied. Our proposed scheme fa-

cilitates the integration of RERs and mitigates the reverse power flow problem by providing the users an opportunity to trade their excess generation locally. The work in Chapter 5 has been submitted for publication in *IEEE Transactions on Smart Grid*.

The VCG mechanism proposed In Chapter 2 is a user level control algorithm. That is, this algorithm does not aim to schedule the operation of individual appliances, and the details of the power consumption of each appliance are hidden from the proposed algorithm. The load control algorithm proposed in Chapter 3 schedules the operation of individual appliances. Moreover, the proposed algorithm takes into account the operational constraints of different types of appliances. Thereby, we assume that the price values for different time slots are determined by the energy provider, and are known at the user side. The idea developed in Chapter 3 is extended in Chapter 4, where we address the question of how to determine the price values for each time slot. In Chapter 5, we focus on systems where most of the users are equipped with RERs. The load scheduling and trading algorithm proposed in Chapter 5 reduces the reverse power flow which is a main barrier for utilizing the RERs.

1.3 Thesis Organization

The rest of the thesis is organized as follows. In Chapter 2, we propose a VCG mechanism for DSM in the future smart grid. The proposed mechanism aims to maximize the aggregate utility of all users while minimizing the total cost of power generation. We investigate some of the main properties of the proposed mechanism such as truthfulness, efficiency, and nonnegative transfer. In Chapter 3, we propose an optimal residential load control algorithm for DSM in presence of load uncertainty. We formulate an optimization problem to minimize the electricity payment of the users in situations where only an estimate of

the future demand is available. We focus on a scenario where real-time pricing is combined with IBRs to balance residential load to achieve a low PAR. In Chapter 4, we propose two pricing algorithms based on stochastic approximation technique to minimize the PAR of the aggregate load. The proposed algorithms eliminate the need to know the structure of the objective function. In our proposed pricing algorithms, we take into account the way users will respond to different price values. We also consider the effect of control decisions of the ECC unit on the users' load profile. In Chapter 5, we propose a load control algorithm for DSM. We consider the problem of joint load scheduling and power trading. An approximate dynamic program is proposed to schedule the operation of different types of appliances. Furthermore, we adopt a game theoretic approach to model the interaction of users with excess power generation. Users with excess power generation choose their offered price and output generation such that they obtain a larger share of the market and their revenue is maximized. Finally, the thesis is concluded and some potential future work is introduced in Chapter 6. Each of the main chapters in this thesis is self-contained and included in separate journal articles or conference papers. The notations are defined separately for each chapter.

Chapter 2

Demand Side Management Using Mechanism Design

2.1 Introduction

As discussed in Chapter 1, in this chapter, we focus on developing a novel pricing method for DSM to encourage efficient energy consumption among users to achieve certain *social objectives*. However, it is difficult because of its computational complexity to achieve these social objectives, if all appliances of all users are to be jointly scheduled. To tackle this problem, a *top-down* control approach is devised. That is, first, the total power consumption of each user in each time slot is determined. This is referred to as user level control. Then, each user tries to schedule the operation of its own appliances to meet the desired power consumption level. This is referred to as appliance level control [26, 54]. In this chapter, we consider the problem of scheduling the total power consumption of each user at different time slots. We show that achieving social objectives is challenging even in user level control and requires collecting various information about the energy consumption behavior of the users, the price elasticity of the users, and the benefit that each user obtains by consuming a certain amount of energy. However, in general, users are not willing to reveal such information, unless there is an incentive for them to do so. Therefore, elaborate design rules (*mechanisms*) are needed such that it is in each user's self interest to reveal its local information. This problem has already been considered for smart grid [55] and in

other contexts such as in telecommunication networks [56–58]. However, the prior works assumed that users are *price taker* who accept the prices as fixed parameters. That is, they do not consider the possibility that their actions may affect the price. For systems with built-in automated control units, this assumption may no longer be valid. Therefore, here, we consider the case where users can anticipate the impact of their actions on price values.

VCG mechanism is a pricing method to elicit local information from rational users. For determining the price charged to each user, users are asked to declare their energy demand information. The payments of the users are then structured such that the users have incentive to declare their local information truthfully. We note that the VCG pricing mechanism has already been applied to resource management problems, e.g., in computer and communication networks [56]. However, since we consider a different problem formulation in the context of smart grid, many of the existing results, e.g., in [56] and [59], are not directly applicable and need to be revised as will be explained throughout this chapter. The contributions of this chapter are summarized as follows:

- We propose a VCG mechanism for DSM programs to encourage efficient energy consumption among users. In our system model, each user reveals its demand information to the energy provider. By running a centralized mechanism, the energy provider computes the optimal energy consumption level for each user, and advertises a specific electricity payment for each user.

- We formulate an optimization problem to maximize the aggregate utility of all users while minimizing the total cost imposed on the energy provider.

- We investigate some of the desired properties of our problem formulation. First, *truthfulness* and *efficiency* of the proposed mechanism are proved. Then, for our

model, we show the property of *nonnegative transfer* which means that the users always make nonnegative payments.

- We compare our efficient VCG DSM method with the case where users are price taker. We study the differences of these two systems, especially from the user payment perspective, and show that for the VCG mechanism users have to pay less.

- Simulation results confirm that both the users and the energy provider will benefit from the proposed scheme.

The rest of this chapter is organized as follows. The system model and problem formulation are presented in Section 2.2. The VCG mechanism and its different properties are discussed in Section 2.3. In Section 2.4, we provide a performance evaluation of the proposed pricing scheme, and the chapter is summarized in Section 2.5.

2.2 System Model and Problem Formulation

2.2.1 Power System

As in [11, 60, 61], we consider a smart power system with a single energy provider and several load subscribers or users as part of the general wholesale electricity market as shown in Fig. 2.1. For each user, we assume that there is an ECC unit which is embedded in the user's smart meter. The role of the ECC is to control the user's power consumption, and to coordinate each user with the energy provider. All ECC units are connected to the energy provider through a communication infrastructure such as a local area network.

The intended time cycle for the system's operation is divided into K time slots, where $K \triangleq |\mathcal{K}|$, and $\mathcal{K}$ is the set of all time slots. This division can be based on the behavior of the users and their power demand pattern: on-peak time slots, off-peak time slots, and

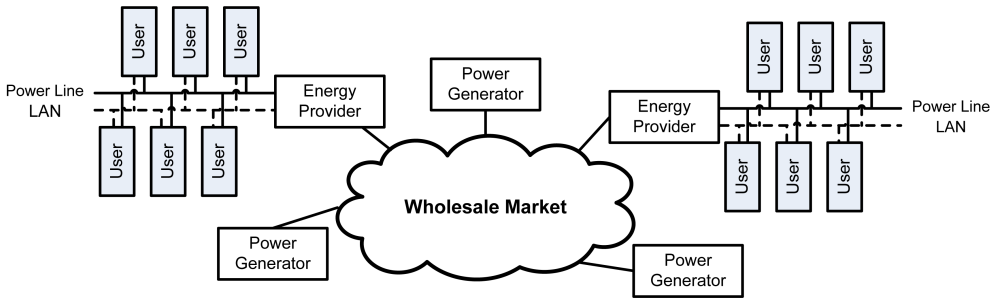

Figure 2.1: An illustration of the regional energy providers, several users, and multiple power generators as parts of the general wholesale energy market.

mid-peak time slots. Let $\mathcal{N}$ denote the set of all users and $N \triangleq |\mathcal{N}|$. In each time slot, we classify the load demand into two types, must-run loads and controllable loads [40]. Must-run loads are price-inelastic. For example, a refrigerator always needs to be on during the day. On the other hand, controllable loads can be stopped, adjusted, or shifted to other time slots and include the demand for services such as charging PHEVs. In each time slot k, we use M_n^k and m_n^k to denote the maximum and minimum power level for each user n, respectively. We use $\mathbf{M}_n \triangleq (M_n^1, \ldots, M_n^K)$ and $\mathbf{m}_n \triangleq (m_n^1, \ldots, m_n^K)$ to denote the vectors of the maximum and minimum power levels of user n in all time slots, respectively. We also define $\mathbf{M} \triangleq (\mathbf{M}_1, \ldots, \mathbf{M}_N)$ and $\mathbf{m} \triangleq (\mathbf{m}_1, \ldots, \mathbf{m}_N)$. We denote the minimum total energy requirements of user n as E_n and the vector of the minimum total energy requirements of all users as $\mathbf{E} \triangleq (E_1, \ldots, E_N)$, where for each user n, we have $E_n \geq \sum_{k \in \mathcal{K}} m_n^k$. For the users, it is difficult to determine their required demand information, i.e., the minimum and the maximum power requirement in each time slot, the minimum total energy requirement, and the benefit obtained by consuming a certain amount of energy. However, machine learning and stochastic signal processing techniques can be adopted in each user's ECC unit to help the user determine its required demand information. The normal pattern of the users' power consumption can be fed into appropriate machine learning algorithms to extract the demand information of the users. In order to provide the required energy for

each user n within the operation cycle, it is required that

$$\sum_{k\in\mathcal{K}} x_n^k \geq E_n, \tag{2.1}$$

where x_n^k is the power consumption level of user n in time slot k. Furthermore, we define $\mathbf{x}_n \triangleq (x_n^1, \ldots, x_n^K)$ as the vector of energy consumption of user n. The feasible energy consumption controlling set of user n is defined as

$$\mathcal{X}_n \triangleq \left\{ \mathbf{x}_n \,\middle|\, \sum_{k\in\mathcal{K}} x_n^k \geq E_n, \quad m_n^k \leq x_n^k \leq M_n^k, \quad \forall\, k \in \mathcal{K} \right\}. \tag{2.2}$$

Our key assumption is that users have price-elastic load. That is, they may shift or change their energy consumption in response to price values [62–64].

2.2.2 User Preference and Utility Function

Each user is assumed to be an independent decision maker. The energy demand of each user may vary based on different parameters. For example, we can take into account the climate conditions and the price of electricity. The energy demand also depends on the type of the users. Residential users may have different *responses* to the same price than industrial users. Even users within the same category may not be identical. The different responses of different users to various price scenarios can be modeled by using *utility functions* from microeconomics [65]. In fact, we can model the behavior of different users through their different choices of utility functions [66]. For each user n, we represent the corresponding utility function as $U_n(\sum_{k\in\mathcal{K}} x_n^k) \triangleq U(\sum_{k\in\mathcal{K}} x_n^k, \omega_n)$, where x_n^k is the power consumption level of user n in time slot k and ω_n is a parameter, which may vary among users, representing the value of electricity for each user. For each user, the utility function represents the level of satisfaction obtained by the user as a function of its total

power consumption throughout the operation period[1].

For all the users, we define $\boldsymbol{\omega} \triangleq (\omega_1, \ldots, \omega_N)$. We assume that the utility functions $U(x, \omega)$ satisfy the following properties:

Property 2.1 Utility functions are *non-decreasing*. This implies that the marginal benefit is nonnegative:

$$\frac{\partial U(x,\omega)}{\partial x} \geq 0. \tag{2.3}$$

Property 2.2 The marginal benefit of users is a non-increasing function. That is,

$$\frac{\partial^2 U(x,\omega)}{\partial x^2} \leq 0. \tag{2.4}$$

In other words, the utility functions are *concave*. While the class of utility functions that satisfy (2.3) and (2.4) is very large, it is convenient to have a linear marginal benefit [66], [69].

Property 2.3 For a fixed consumption level x, a larger ω gives a larger $U(x, \omega)$, which can be expressed as

$$\frac{\partial U(x,\omega)}{\partial \omega} > 0. \tag{2.5}$$

Property 2.4 When the consumption level is zero,

$$U(0, \omega) = 0, \quad \forall \omega > 0. \tag{2.6}$$

[1]The value of electricity for the users may also vary at different times of a day. Large loads that participate in wholesale electricity markets discriminate the value of electricity in different time slots through their different choices of utility functions at different times of a day [67, 68]. Considering the advances in home automation systems, it is conceivable that in the near future, instead of specifying only the value of their total power consumption, residential users will be able to determine the value of electricity for each time slot. In that case, the utility function of each user n can be replaced by $\sum_{k\in\mathcal{K}} \tilde{U}_n^k(x_n^k)$, where $\tilde{U}_n^k(\cdot)$ is the utility function of user n in time slot k.

We note that the operation of each individual appliance is meant to achieve a goal or to finish a task. For example, the air conditioning system is used to keep the temperature in a predetermined range. Thus, the total power consumption of each user can be considered as the aggregate power consumption required to complete different tasks. In this chapter, since we define the utility functions for the aggregate load of different tasks, rather than for the power consumption of each individual appliance, the utility functions do not decrease. This is because users can complete more tasks if they consume more power. Furthermore, it is reasonable to assume that users prioritize their tasks. Therefore, as the prices increase, they tend to ignore some less important tasks or switch to a different mode of operation with lower power consumption. This implies a decreasing marginal benefit and a concave and increasing utility function for the total power consumption of different tasks. In addition, we assume that users are able to specify how much they value energy through the proper choice of parameter ω, i.e., a higher ω implies a higher utility value. Finally, as utility functions quantify the level of satisfaction of the users, intuitively zero power consumption should result in a zero utility value. Recent reports indicate that the behavior of power users can indeed be accurately modeled by certain utility functions [69]. In this chapter, we consider *quadratic utility* functions corresponding to *linearly decreasing marginal benefit* [15]:

$$U(x,\omega) = \begin{cases} \omega x - \frac{\alpha}{2}x^2, & \text{if} \quad 0 \leq x < \frac{\omega}{\alpha}, \\ \\ \frac{\omega^2}{2\alpha}, & \text{if} \qquad x \geq \frac{\omega}{\alpha}, \end{cases} \tag{2.7}$$

where α is a pre-determined parameter. A few example utility functions from this class are shown in Fig. 2.2. The point where the utility function gets saturated and does not change corresponds to the maximum power requirement of the user.

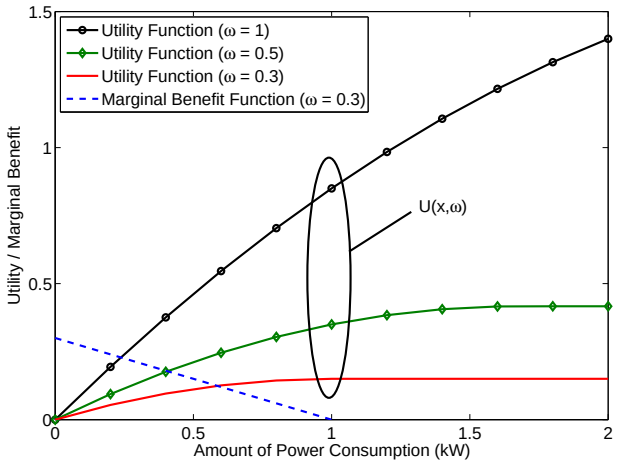

Figure 2.2: Sample utility functions for power users ($\alpha = 0.3$).

2.2.3 Energy Cost Model

We consider a cost function $C_k(L_k)$ indicating the cost of providing L_k units of energy offered by the energy provider in each time slot k. We make the following assumptions:

Assumption 2.1 The cost functions are *increasing* with respect to the total offered energy capacity.

Assumption 2.2 The cost functions are *strictly convex*.

Assumption 2.3 There exists a differentiable, convex, nondecreasing function $p_k(q)$ over $q \geq 0$ for each $k \in \mathcal{K}$, with $p_k(0) \geq 0$ and $p_k(q) \to \infty$ as $q \to \infty$, such that for $q \geq 0$

$$C_k(q) = \int_0^q p_k(z)dz. \tag{2.8}$$

Note that *quadratic functions* are among several practical examples for cost functions that satisfy Assumptions 2.1-2.3, and are considered throughout this chapter [11, 70]:

$$C_k(L_k) = a_k L_k^2 + b_k L_k + c_k, \tag{2.9}$$

where $a_k > 0$, $b_k \geq 0$, and $c_k \geq 0$ are fixed parameters.

2.2.4 Problem Formulation and Efficient Allocations

In this section, we consider the problem of power consumption level selection. From a social fairness point of view, it is desirable to utilize the available generated power provided by the energy provider in such a way that the sum of the utility functions of all users is maximized and the cost imposed on the energy provider is minimized. If centralized control is feasible and we can collect all information about the users' utility functions, an efficient energy consumption schedule can be characterized as the solution of the following problem:

$$\underset{\mathbf{x}_n \in \mathcal{X}_n,\, n \in \mathcal{N}}{\text{maximize}} \sum_{n \in \mathcal{N}} U_n \left(\sum_{k \in \mathcal{K}} x_n^k \right) - \sum_{k \in \mathcal{K}} C_k \left(\sum_{n \in \mathcal{N}} x_n^k \right), \tag{2.10}$$

where $\mathbf{x}_n$ is the vector of power consumptions of user n, $U_n(\cdot)$ is as in (2.7), and $C_k(\cdot)$ is defined in (2.9). The objective function in problem (2.10) is the sum of all utility functions minus the total energy cost in the system.

Problem (2.10) is a concave maximization problem and can be solved in a centralized fashion using *convex programming* techniques such as the interior point method (IPM) [71]. Since it is assumed that parameters ω_n, $\mathbf{m}_n$, $\mathbf{M}_n$, and E_n for each user n are local information, the energy provider may not have sufficient information to solve problem (2.10). Each user aims to optimize its local objective. To align these individual objectives with the social objective, some elaborately designed pricing scheme is needed. In general, users may have different approaches in responding to the price values set by the energy provider. This can lead to different equilibriums among users. We are interested in analyzing *competitive equilibrium* and *Nash equilibrium*. In competitive equilibrium, each user acts as a *price taker*. That is, it does not consider the effect of its actions on the price. However, in Nash equilibrium, we assume that users are price anticipator, i.e., they consider the effect

of their actions on the price set by the energy provider.

Price Taking Users

If users are price taker, i.e., they do not consider the effect of their actions on the price, then we need to analyze the competitive equilibrium among the users and the energy provider. Given a price vector $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K)$, where λ_k is the price in time slot k, a user who consumes x_n^k kW electricity in time slot k is charged $\lambda_k x_n^k$ dollars for that time slot. Since users treat the prices as fixed values, the payoff function for each user n becomes

$$P_n(\mathbf{x}_n) = U_n\left(\sum_{k\in\mathcal{K}} x_n^k\right) - \sum_{k\in\mathcal{K}} \lambda_k x_n^k, \tag{2.11}$$

where the first term represents the utility of user n as a function of its power consumption, and the second term represents its payment to the energy provider.

We call a pair $(\mathbf{x}, \boldsymbol{\lambda})$, where $\mathbf{x} \triangleq (\mathbf{x}_1, \ldots, \mathbf{x}_N)$, a competitive equilibrium if each user n maximizes its own payoff function defined in (2.11) for a given price vector $\boldsymbol{\lambda}$, i.e.,

$$P_n(\mathbf{x}_n) \geq P_n(\bar{\mathbf{x}}_n), \quad \bar{\mathbf{x}}_n \in \mathcal{X}_n,\ n \in \mathcal{N}, \tag{2.12}$$

where vector $\mathbf{x}$ is the solution to the problem defined in (2.10). It has been shown that under Properties 2.1-2.4 and Assumptions 2.1-2.3, a competitive equilibrium always exists [55, 72], and the results are summarized in the following proposition.

Proposition 2.1 There exists a competitive equilibrium $(\mathbf{x}, \boldsymbol{\lambda})$, where $\mathbf{x}$ is an optimal solution to problem (2.10).

The assumption that users are price taker is usually considered when the number of users is large, the amount of information provided to each user is limited, and the computer programs running the decentralized algorithm are embedded in the computer operating

system and are not tampered with by the vast majority of the users. However, as some individual users such as large industrial users may have a significant impact on the power system, or some parts of the power system may act autonomously as in microgrids for household users in which the number of users is much lower than the number of users in the whole grid, the price taking assumption may not always be valid. When the price taking assumption is violated, the model changes into a game, and the assumptions required for the validity of Proposition 2.1 do not hold. We investigate this scenario in the next sub-section.

Price Anticipating Users

If users are price anticipator, i.e., they do consider the effect of their actions on the price, then we need to analyze the Nash equilibrium of the *game* which is played among multiple users who compete for the available power provided by the energy provider. In this game theoretic model [73], the strategies of the users represent their power consumption level. We consider the following pricing scheme for resource allocation. Given $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$, the energy provider sets a single price $\mu_k(\mathbf{x}) = p_k(\sum_{n\in\mathcal{N}} x_n^k)$ for time slot k. User n then pays $x_n^k p_k(\sum_{n\in\mathcal{N}} x_n^k)$ for that time slot. We use the notation $\mathbf{x}_{-n}$ to denote the vector of all consumption powers chosen by users other than user n, i.e., $\mathbf{x}_{-n} = (\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \mathbf{x}_{n+1}, \ldots, \mathbf{x}_N)$. Then, given $\mathbf{x}_{-n}$, the payoff of each user n is obtained as

$$Q_n(\mathbf{x}_n; \mathbf{x}_{-n}) = U_n\left(\sum_{k\in\mathcal{K}} x_n^k\right) - \sum_{k\in\mathcal{K}} x_n^k p_k\left(\sum_{m\in\mathcal{N}} x_m^k\right). \tag{2.13}$$

The payoff function Q_n is similar to P_n, defined for price-taking users in (2.11). The only difference is that while the payoff function P_n takes the price λ_k as a fixed parameter, price anticipating users realize that the price is set according to $p_k(\sum_{m\in\mathcal{N}} x_m^k)$, and adjust their payoffs accordingly.

From (2.13), the payoff of each user depends on its power consumption as well as the power consumptions of other users. Hence, we have the following game among the users:

- Players: Registered users in set $\mathcal{N}$.
- Strategies: Each user $n \in \mathcal{N}$ selects its energy consumption level $\mathbf{x}_n \in \mathcal{X}_n$ to maximize its payoff.
- Payoffs: $Q_n(\mathbf{x}_n; \mathbf{x}_{-n})$ for each user $n \in \mathcal{N}$ as in (2.13).

A Nash equilibrium of the game defined by $(Q_1, \ldots, Q_N)$ is a vector $\mathbf{x}$ such that for all $n \in \mathcal{N}$,

$$Q_n(\mathbf{x}_n; \mathbf{x}_{-n}) \geq Q_n(\bar{\mathbf{x}}_n; \mathbf{x}_{-n}), \quad \bar{\mathbf{x}}_n \in \mathcal{X}_n. \tag{2.14}$$

It can be shown that a Nash equilibrium exists for this game, and the results are summarized in the following proposition. However, the details of the proof can be found in [73].

Proposition 2.2 Suppose that Properties 2.1-2.4 and Assumptions 2.1-2.3 hold. There exists a Nash equilibrium $\mathbf{x}$ for the game defined by $(Q_1, \ldots, Q_N)$.

In general, the Nash equilibrium of a resource allocation game may not be optimal [73, 74]. That is, the energy consumption profile obtained at the Nash equilibrium in a distributed pricing scenario may not necessarily be the same as the optimal solution of the optimization problem in (2.10). Next, we investigate how the price values can be set carefully by the utility company such that the system performance becomes optimal at the aforementioned Nash equilibrium.

2.3 Applying the Vickrey-Clarke-Groves Mechanism

In the previous section, we considered a method (mechanism) which uses only a single price in each time slot for all users to allocate the provided power. Despite its simplicity, the introduced mechanism suffers from a loss in efficiency if users are indeed price anticipator, and evaluate the effect of their actions on the price function. As mentioned before, the main obstacle in solving problem (2.10) is the lack of information about the utility functions of the users and their feasible set of power consumptions. However, if we remove the restriction that the mechanism only chooses a single price, we can elicit the local information of the users. One possible approach to convince users to declare their utility functions and constraint parameters truthfully is the VCG mechanism [75].

2.3.1 VCG Mechanism

In the VCG class of mechanisms, each user is asked to specify its feasible set of power consumption and its utility function, which in case of the utility functions in (2.7) reduces to revealing a utility parameter ω_n. For each user n, we use $U_n(\sum_{k\in\mathcal{K}} x_n^k) \triangleq U(\sum_{k\in\mathcal{K}} x_n^k, \omega_n)$ and $\hat{U}_n(\sum_{k\in\mathcal{K}} x_n^k) \triangleq U(\sum_{k\in\mathcal{K}} x_n^k, \hat{\omega}_n)$ to denote the true and declared utility function and $\mathcal{X}_n$ and $\hat{\mathcal{X}}_n$ to denote the true and declared feasible set of power consumptions, respectively. We define

$$\mathbf{I}_n \triangleq \{\omega_n, \mathbf{M}_n, \mathbf{m}_n, E_n\} \tag{2.15}$$

and

$$\hat{\mathbf{I}}_n \triangleq \{\hat{\omega}_n, \hat{\mathbf{M}}_n, \hat{\mathbf{m}}_n, \hat{E}_n\} \tag{2.16}$$

to denote the true and declared demand parameters, respectively, where $\hat{\omega}_n$, $\hat{\mathbf{M}}_n$, $\hat{\mathbf{m}}_n$, and $\hat{E}_n$ are the declared values for ω_n, $\mathbf{M}_n$, $\mathbf{m}_n$, and E_n, respectively. For notational simplicity,

we also define

$$\mathbf{I} \triangleq \{\boldsymbol{\omega}, \mathbf{M}, \mathbf{m}, \mathbf{E}\} \tag{2.17}$$

and

$$\hat{\mathbf{I}} \triangleq \{\hat{\boldsymbol{\omega}}, \hat{\mathbf{M}}, \hat{\mathbf{m}}, \hat{\mathbf{E}}\}, \tag{2.18}$$

where $\hat{\boldsymbol{\omega}}$, $\hat{\mathbf{M}}$, $\hat{\mathbf{m}}$, and $\hat{\mathbf{E}}$ are the declared values for vectors $\boldsymbol{\omega}$, $\mathbf{M}$, $\mathbf{m}$, and $\mathbf{E}$, respectively. If user n has a consumption vector $\mathbf{x}_n$, but has to pay t_n, then the payoff function of user n is

$$U_n\left(\sum_{k\in\mathcal{K}} x_n^k\right) - t_n. \tag{2.19}$$

On the other hand, the social objective is in the form of

$$U_n\left(\sum_{k\in\mathcal{K}} x_n^k\right) + \sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}} x_m^k\right), \tag{2.20}$$

where $\mathcal{N}_{-n}$ is the set of all users except user n. For a given vector of declared demand information $\hat{\mathbf{I}}$, the VCG mechanism chooses the energy consumption allocation $\mathbf{x}(\hat{\mathbf{I}})$ as an optimal solution to problem (2.10) and calculates optimal energy consumption vectors as

$$\mathbf{x}(\hat{\mathbf{I}}) = \underset{\mathbf{x}_n\in\hat{\mathcal{X}}_n,\, n\in\mathcal{N}}{\arg\max} \left\{ \sum_{n\in\mathcal{N}} \hat{U}_n\left(\sum_{k\in\mathcal{K}} x_n^k\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{n\in\mathcal{N}} x_n^k\right) \right\}, \tag{2.21}$$

and the payments are structured such that

$$t_n(\hat{\mathbf{I}}) = -\left(\sum_{m\in\mathcal{N}_{-n}} \hat{U}_m\left(\sum_{k\in\mathcal{K}} x_m^k\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}} x_m^k\right) \right) + h_n(\hat{\mathbf{I}}_{-n}), \tag{2.22}$$

where h_n is an arbitrary function of $\hat{\mathbf{I}}_{-n}$, i.e., the declared demand information of the users with user n excluded from the system. The true demand information of the users other

than n is denoted by $\mathbf{I}_{-n}$. The definition of the payments in (2.22) aligns user objectives with the social planner's objective.

Remark 2.1 We note that the information in (2.15) is similar to the type of information submitted by large purchasers of electricity in a wholesale electricity market. Each purchaser in a wholesale electricity market makes a day-ahead bid based on its demand curve. However, in contrast to (2.21) and (2.22), the power share of each purchaser and the price of electricity in day-ahead markets are determined by clearing the demand against the supply offers. The dispatch of the power is then balanced in real-time on the day of dispatch [67, 68]. As a result, the proposed schemes in this chapter can find interesting applications also in the wholesale electricity market.

Remark 2.2 We note that the VCG energy allocation in (2.21) determines the optimum level of energy consumption for the users in each time slot. However, in practice, the realization of random events can influence the level of energy consumption of the users. In this case, each user can be penalized by the energy provider for any deviation from the allocated level of energy consumption.

The cost term $C_k(\cdot)$ in (2.20) couples the consumption power variables of all users $\mathbf{x}$. This term makes the whole problem not only a utility maximization but also a cost minimization problem, and thus, the system objective is different from the normal objective of VCG mechanisms studied in other contexts [56, 76, 77]. These changes in our problem formulation require the verification of some desired properties of the proposed VCG mechanism for the new scenario. To this end, we make the following proposition.

Proposition 2.3 If the VCG mechanism defined in (2.21) and (2.22) is used to select electricity payment values, then declaring $\hat{\mathbf{I}}_n = \mathbf{I}_n$ is a dominant strategy for each user n, and following this strategy results in an efficient allocation.

The proof of Proposition 2.3 is given in Appendix A. Proposition 2.3 highlights two main features of the proposed VCG mechanism. First, the payment of each user is structured such that regardless of other users' strategies, the intended user cannot do better than truthfully declaring its demand information. This feature significantly reduces the communication requirements of the method and eliminates the need for interaction among users. Second, if all users declare their demand truthfully, the proposed VCG system results in an efficient system, i.e., the utilities of all users are maximized and the cost imposed on the energy provider is minimized. For the following, we need to determine function h_n introduced in (2.22). Here, we will use a popular choice for this function which is referred to as Clarke tax [75],

$$h_n(\hat{\mathbf{I}}_{-n}) = \sum_{m\in\mathcal{N}_{-n}} \hat{U}_m\left(\sum_{k\in\mathcal{K}} x_m^k(\hat{\mathbf{I}}_{-n})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}_{-n}} x_m^k(\hat{\mathbf{I}}_{-n})\right), \tag{2.23}$$

where $\mathbf{x}(\hat{\mathbf{I}}_{-n})$ is the VCG allocation choice in (2.21), but when user n is excluded from the system. Thus, the payment of user n is

$$\begin{aligned} t_n(\hat{\mathbf{I}}) = &-\left(\sum_{m\in\mathcal{N}_{-n}} \hat{U}_m\left(\sum_{k\in\mathcal{K}} x_m^k(\hat{\mathbf{I}})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}} x_m^k(\hat{\mathbf{I}})\right)\right) \\ &+\left(\sum_{m\in\mathcal{N}_{-n}} \hat{U}_m\left(\sum_{k\in\mathcal{K}} x_m^k(\hat{\mathbf{I}}_{-n})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}_{-n}} x_m^k(\hat{\mathbf{I}}_{-n})\right)\right). \end{aligned} \tag{2.24}$$

The payment of user n is the difference in the social welfare of the other users with and without the presence of user n.

2.3.2 VCG Mechanism and Nonnegative Transfer

In general, if users can serve as a source of electricity at some time instances during the day, e.g., because they have local generation capability or they can transfer the power stored in their local batteries back to the grid, then such users may receive payments from the grid. Such payments can also be interpreted as negative payments made by the users. However, in the problem formulation considered in this chapter, since users are only electricity consumers, this case does not arise, and the users' payments to the grid are always nonnegative. We will refer to this property as nonnegative transfer. In the following theorem, we show that for our problem formulation the nonnegative transfer property holds.

Theorem 2.1 Suppose Properties 2.1-2.4 and Assumptions 2.1-2.3 hold. Then, the VCG mechanism in (2.21) and (2.24) has the property of nonnegative transfer.

The proof of Theorem 2.1 is given in Appendix B.

2.3.3 VCG Mechanism and Market Clearing Price

The following theorem shows that the electricity payment of each user in the proposed VCG mechanism is less than its payment in a system which has price taking users and uses marginal cost pricing, i.e., the $\boldsymbol{\lambda}$ term in Proposition 2.1.

Theorem 2.2 Suppose Properties 2.1-2.4 and Assumptions 2.1-2.3 hold. For the VCG mechanism in (2.21) and (2.24), the payment of each user is $t_n \leq \sum_{k\in\mathcal{K}} \lambda_k^* x_n^k(\mathbf{I})$, where $\boldsymbol{\lambda}^* = (\lambda_1^*, \ldots, \lambda_K^*)$ is the vector of market clearing prices for problem (2.10).

The proof is based on the assumptions that the utility functions are concave and the cost function is convex. Optimality conditions of the VCG allocation (2.21) are adopted to

relate the VCG payment of each user to the market clearing price. The proof of Theorem 2.2 can be found in Appendix C.

2.4 Performance Evaluation

In this section, we present simulation results and assess the performance of our proposed mechanism and the impact of different system parameters. In our simulations, we assume that all users have concave quadratic utility functions as described in (2.7), where parameter α is chosen as 0.5. We set the parameters of the cost function in (2.9) for each time slot to $a_k > 0$, $b_k = 0$, and $c_k = 0$.

2.4.1 Performance Gains from Real-time Interaction with Users

To have a baseline scheme to compare with, we consider a *peak load pricing* (PLP) method in which the price value for each time slot is calculated based on the average power consumption of the users in each time slot to maximize the payoff of the energy provider which is its revenue minus total energy cost. For the PLP method, we assume that the energy provider has some prior information about the distribution of parameter ω of the users. Here, we assume a uniform distribution. We assume there are $N = 50$ users. We consider $K = 24$ representing a 24-hour period. Parameter ω of each user is selected from the set $\{5, 6, \ldots, 15\}$. However, random events are modeled via a small perturbation in the ω value of each user. We set the parameter a_k of the cost function equal to 0.02, 0.3, and 0.5 for off-peak, mid-peak, and on-peak hours, respectively. We assume that each user has a minimum required energy in each operation period, E_n, which varies from 9 kWh to 21 kWh. The minimum power requirements of each user in each time slot, m_n^k, are set on average to 0.1 kW, 0.5 kW, and 1 kW for off-peak, mid-peak, and on-peak hours, respectively.

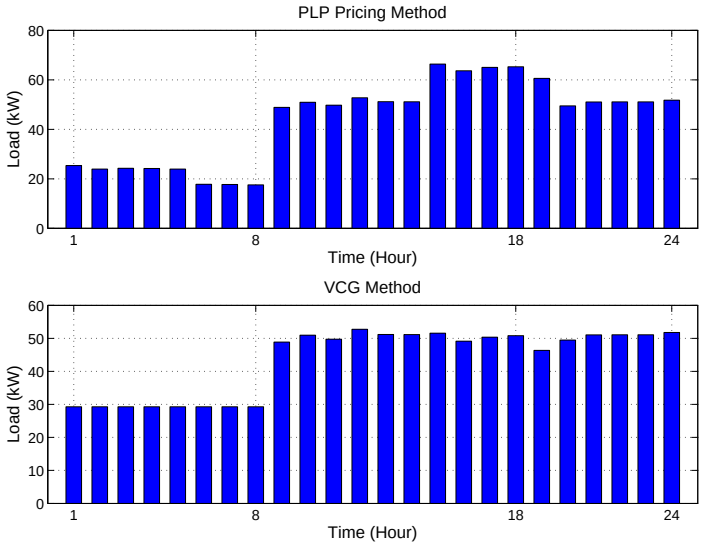

Figure 2.3: Power consumption for the proposed VCG method and a peak load pricing (PLP) method.

As illustrated in Fig. 2.3, the proposed VCG mechanism improves the performance of the system not only by reducing the power consumption of users but also by reducing the PAR from 1.51 to 1.21.

2.4.2 The Impact of Reflecting the Generating Cost

The proposed VCG mechanism is used to maximize the social welfare. Maximizing the aggregate utility of all users while minimizing the cost imposed on the energy provider is beneficial for both users and energy provider. The opportunity of reflecting the fluctuations of the wholesale price into the customer side is one of the main advantages of the proposed VCG mechanism. This aspect becomes more important in situations where the cost imposed on the energy provider is high. To have a baseline scheme to compare with, we consider a system which has price anticipating users and employs marginal cost pricing. It has been shown that in a system with price taking users, marginal cost pricing not only

maximizes the social welfare, but also maximizes the payoff of the energy provider [55]. As an upper bound on the payoff of the energy provider, we consider a system which has price taking users and employs marginal cost pricing. We assume there are 50 users, and parameter ω of each user is selected from the set $\{15, 25, 30, 40\}$. We assume that for each user n, parameter E_n varies from 10 kWh to 15 kWh and for different time slots, parameter m_n^k is set on average to 0.1 kW, 0.5 kW, and 1 kW for off-peak, mid-peak, and on-peak hours, respectively.

Furthermore, we assume that parameter a_k of the cost function is constant in all three time slots. The payoffs of the energy provider for the proposed VCG system, the system with price anticipating users, and the system with price taking users for different values of parameter a_k of the cost function are presented in Fig. 2.4. We can see that, since the VCG payment (2.24) is structured to consider the cost imposed on the energy provider, the payoff of the energy provider is higher compared to the system with price anticipating users. Note that the proposed VCG system and the price taking system are both efficient systems with the same power allocation. Hence, they have the same total power consumption.

2.4.3 Communication Requirements of the VCG System

The communication requirements are among the main aspects considered for any pricing method. In this section, the number of messages exchanged between users and also the energy provider is considered as a measure to compare the proposed VCG system with a system which has price anticipating users. In the VCG system, each user is asked to declare its parameter ω and its feasible set of power consumption to the energy provider, and in return, the energy provider determines the payment and the allocated power of each user. In practice, it may be preferable for the users to communicate only with a trusted entity such as the energy provider. However, when users are price anticipator, they form a game

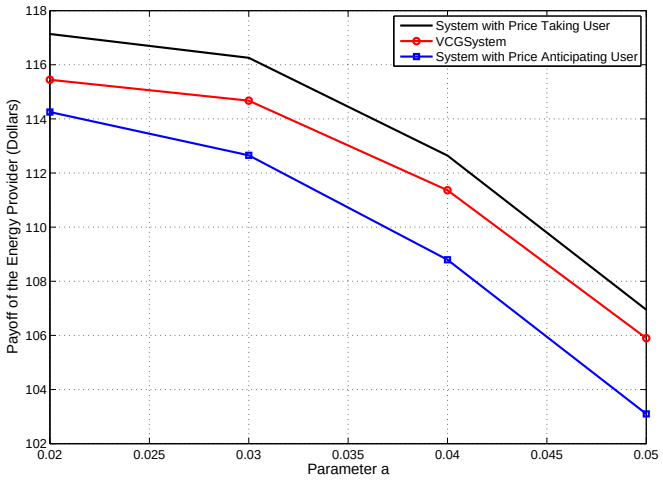

Figure 2.4: Payoff of the energy provider for the proposed VCG system, the system with price anticipating users, and the system with price taking users.

and have to exchange messages with each other. Communication requirements become an important feature specially in situations where the cost imposed on the energy provider is low, and most of the users can compete in the power consumption game. In a system where users are price anticipator, we use the myopic best-response algorithm [75, Ch. 6] to compute the Nash equilibrium. In this system, each user informs other users whenever it changes its power consumption. Each time one of the users updates its power consumption information, a *message* is sent. We set the parameter a_k of the cost function equal to 0.02, 0.3, and 0.5 for off-peak, mid-peak, and on-peak hours, respectively. We assume that for each user n, parameter E_n varies from 10 kWh to 20 kWh and for different time slots, parameter m_n^k is set on average to 0.1 kW, 0.5 kW, and 1 kW for off-peak, mid-peak, and on-peak hours, respectively.

The average number of messages exchanged between the various entities in the VCG system and the system with price anticipating users for $K = 24$ is presented in Table 2.1. As illustrated in Table 2.1, the method used in the system with price anticipating users

Table 2.1: Average number of exchanged messages in VCG system and system with price anticipating users.

Number of Users N	Price Anticipating System	VCG System
10	42767	20
20	79624	40
30	87808	60
40	135290	80
50	145718	100

requires much more message exchanges to converge than the VCG mechanism.

2.4.4 Effect of Parameter ω

In this section, we explore the effect of parameter ω on different aspects of the power system for $N = 50$ users and $K = 3$ time slots. In this regard, we mainly focus on the power consumption of the system and the payments of the users. To understand how changes in the parameter ω of a single user can affect others, we consider the same ω for all the users and change it for the first user starting from 5 while keeping this parameter fixed equal to 20 for the other users. We set parameter a_k of the cost function equal to 0.02 for all time slot. For each user n, parameter E_n selected to be 12, and parameter m_n^k is set to zero for all time slots.

The simulation results for the power consumption of the users are presented in Fig. 2.5. We notice that as we increase the ω of the first user, the power consumption of the other users decreases until they reach their minimum power requirements. Intuitively, as the first user values energy more by increasing its ω value, it has a negative impact on the power consumption of its rivals in the system. For the VCG mechanism, the payments of the users are closely related to their power consumption, i.e., as the first user increases its ω, the power consumption of the other users reduces as well as their aggregate payment until they reach the point where they consume their minimum power

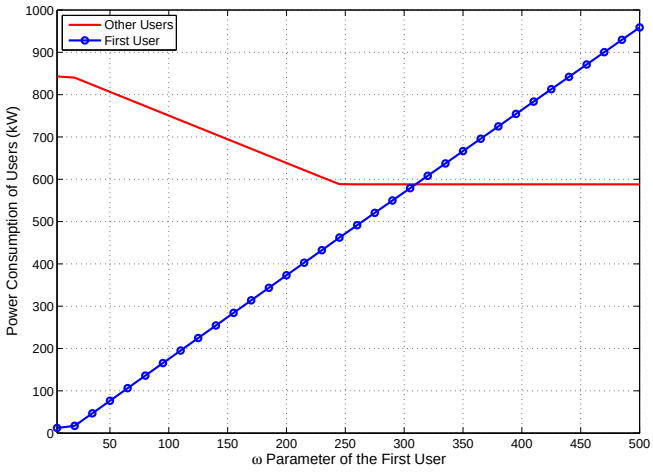

Figure 2.5: Power consumption of the first user and aggregate power consumption of other users when ω of the first user is increased.

requirements. After reaching this point, as they have a guaranteed amount of power consumption, their aggregate payment increases.

2.4.5 Exploring the Truthfulness Property

Truthfulness in dominant strategy for the proposed VCG mechanism means that regardless of other users' strategy, the intended user cannot do better than truthfully declare its demand information. In this section, we consider a system where there are $N=10$ users and $K=3$ time slots. We set parameter a_k of the cost function equal to 0.02 for all time slot. For each user n, parameter E_n is equal to 15 kWh and for different time slots, parameter m_n^k is set to zero for all time slots. We assume the true ω parameter of the users is $\boldsymbol{\omega} = [12, 6, 8, 8, 10, 10, 12, 12, 16, 20]$ and $E_1=15$. We explore the best response of the first user while other users declare their demand information truthfully. As illustrated in Fig. 2.6, the considered user (first user) with $\omega_1=12$ and $E_1=15$ cannot do better than truthfully declare $\hat{\omega}_1=12$ and $\hat{E}_1=15$.

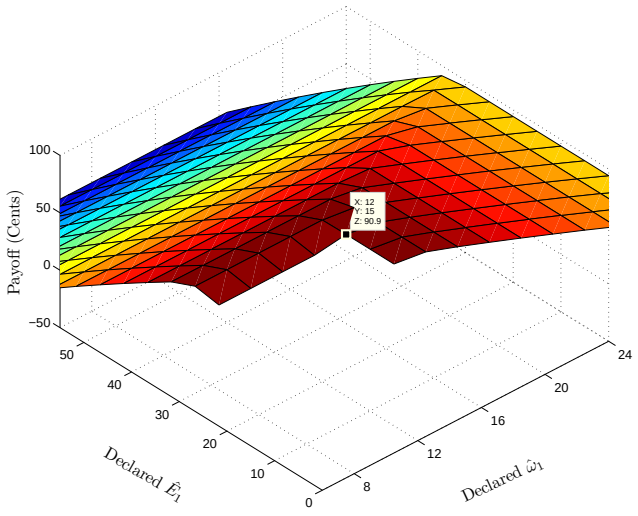

Figure 2.6: The payoff of the first user for different values of *declared* $\hat{\omega}_1$ and $\hat{E}_1$ (the *true* ω_1 is equal 12 and the *true* E_1 is equal 15).

2.5 Summary

In this chapter, we proposed a VCG mechanism for DSM in the future smart grid. The proposed mechanism aims to maximize the aggregate utility of all users while minimizing the total cost of power generation. We investigated some of the main properties of the proposed mechanism such as truthfulness, efficiency, and nonnegative transfer. Through simulation, we showed that the proposed VCG mechanism improves the performance of the system by encouraging users to reduce their power consumption and shift their loads to off-peak hours. The proposed VCG mechanism significantly reduces the communication overhead. We also analyzed the impact of some key parameters on our model through simulations. The simulations confirmed that by using our proposed VCG mechanism, in addition to maximizing the social welfare, the energy provider can benefit as well.

Chapter 3

Energy Consumption Scheduling with Load Uncertainty

3.1 Introduction

As discussed in Chapter 1, despite its importance, the effect of load uncertainties on DSM has not been well-studied in the smart grid literature [11, 16, 29, 40, 41, 78–81]. In this chapter, we focus on developing a novel automated optimization-based residential load scheduling algorithm in a retail electricity market with load uncertainties. We aim to minimize each user's electricity payment by optimally scheduling the operation of its appliances in real-time, subject to the operational constraints defined by the users. As in [12], we adopt RTP combined with IBR to better reflect the fluctuation of the wholesale electricity prices and to avoid *load synchronization*.

Our design can be partly compared with [41]. The problem tackled in this chapter is different from that in [41] in two aspects. First, the work in [41] addresses uncertainty in price values while we tackle uncertainty in load and users' energy consumption needs. Second, the key assumption in [41] is that the price values are independent from the load level in each time slot. Here, we relax this assumption. Our work is also different from the heuristic home automation schemes in [26, 42], as we use an optimization-based approach with elaborate mathematical modeling and take into account estimates of the future load to make better decisions. The contributions of this chapter can be summarized as follows:

- We propose a real-time residential load management algorithm with *load uncertainty* for DSM purposes. Our algorithm is based on solving an optimization problem that aims to minimize the electricity payment of residential users. Each appliance sends an admission request to the ECC unit to start operation. The operation of each appliance is subject to acceptance of its request. By running a centralized algorithm, the control unit determines the optimal operation schedule of each appliance in each time slot.

- We study operation constraints to model a variety of appliances including must-run appliances, and interruptible and non-interruptible controllable appliances. The last item refers to those appliances whose operation can be postponed, but once they start operation, they should stay on until they finish their task.

- Simulation results show that our proposed scheduling algorithm with load uncertainty reduces the energy payment of users compared to the case where no scheduling algorithm is adopted. Our proposed scheme also improves the overall power system performance by reducing the PAR in aggregate load demand.

The rest of this chapter is organized as follows. The system model is introduced in Section 3.2. The problem formulation and algorithm description are presented in Section 3.3. Simulation results are provided in Section 3.4. The chapter is summarized in Section 3.5.

3.2 System Model

In this section, we present a mathematical model for real-time residential load scheduling when combined RTP and IBR tariffs are implemented. We assume that price values are

informed by the retailer to end users through a digital communication infrastructure. Furthermore, we assume that each user is equipped with a smart meter, which has an ECC unit capable of scheduling and adjusting the household energy consumption.

Consider a residential unit that participates in a DSM program. Let $\mathcal{A}$ denote the set of all appliances in this unit. Each appliance $a \in \mathcal{A}$ can work either as must-run or controllable. Must-run appliances need to start working immediately. For example, we can classify TV and personal computer (PC) as must-run appliances. Clearly, the user should have the freedom to turn on or turn off the TV whenever he wants without the interference of the ECC. In contrast, the operation of controllable appliances can be delayed or interrupted if necessary. Each controllable appliance can be either interruptible or non-interruptible. For a controllable appliance a, if it is non-interruptible, then the ECC may only delay its operation. However, for interruptible appliances, it is not only possible to postpone the operation but also to interrupt the operation when needed and then restore the operation later on. Plug-in electric vehicle (PEV) and washing machine are examples of interruptible and non-interruptible controllable appliances, respectively. We assume that based on the demand requirements of the user, each appliance can be set as must-run or controllable. This setting is decided by the user and can vary from time to time. That is, depending on the preferences of the user, an appliance can be set as a must-run appliance in one day and as a controllable appliance in another day.

We divide the intended operation cycle into T time slots. Each time slot begins with an *admission control* phase. In this phase, to start the operation of an appliance, an *admission request* is sent to the ECC unit. Once an admission request is submitted, the state of the appliance changes from *sleep* to *awake*. The appliance remains awake until its operation is *finished*. However, the operation of an awake appliance is subject to acceptance of its admission request and specification of its operation schedule by the ECC unit. The

decisions regarding the admission of the requests and the adjustment of the operation of different awake appliances are updated periodically in each admission control phase.

An awake appliance a can be either *inactive* (with zero power consumption) or *active* (operating at nominal power γ_a). We note that the power consumption of each appliance could be different at different cycles of its operation due to the changes in the amount of current being absorbed. However, considering the exact load profile of each appliance adds to the complexity of the model and makes real-time implementation difficult. To tackle this implementation difficulty, similar to [40, 79], and [41], we consider an average power consumption γ_a for each appliance. Different operating states of must-run and controllable appliances are shown in Fig. 3.1.

We note that the operation of different appliances is influenced by the preferences of the user. Different parameters of our model may be considered to capture different types of preferences. For example, our model takes into account the time and the frequency at which each appliance sends admission requests to the ECC unit. Furthermore, we assume that the mode of operation of each appliance, i.e., whether it is must-run or controllable, is *not* pre-determined. That is, based on the preference of the user, each appliance can work either as must-run or controllable. Moreover, for controllable appliances, the deadline before which the operation of the appliance has to be finished is also determined based on the preference of the user. Other aspects of user preferences, such as the desirable room temperature, can also be considered to enhance energy consumption scheduling; however, adding those aspects will also make the design more complex and less appealing for real-time implementation in practice.

The admission request of each appliance a specifies the total energy E_a needed to finish the operation of the appliance, the operating power γ_a, and whether the appliance is must-run or controllable. For controllable appliances, the deadline before which the operation

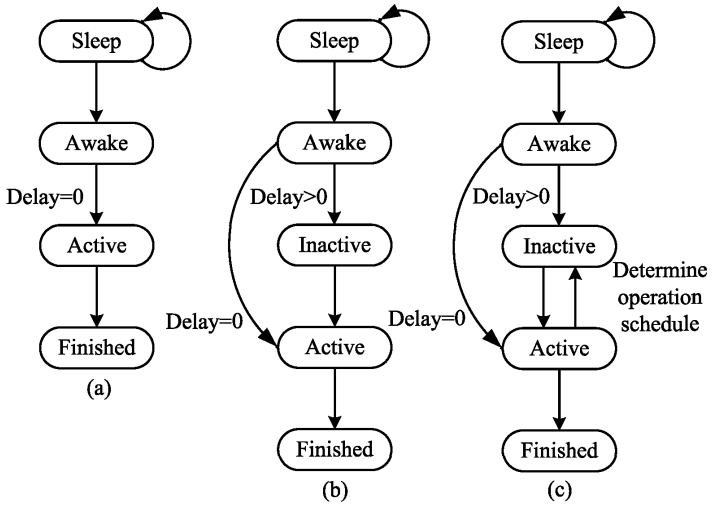

Figure 3.1: Different operating states of (a) must-run, (b) non-interruptible controllable, and (c) interruptible controllable appliances.

of the appliance has to be finished, denoted by β_a, and whether it is interruptible or not, are the additional information to be included in the admission request. For a controllable appliance a, if it is not interruptible, the ECC may only delay its operation. However, for interruptible appliances, it is not only possible to postpone the operation but also to interrupt the operation when needed.

We define binary variable $x_t^a \in \{0, 1\}$ as the state of power consumption of appliance $a \in \mathcal{A}$ at time slot $t \in \{1, \ldots, T\}$. We set $x_t^a = 1$ if appliance a is admitted to operate in time slot t (i.e., active), otherwise, we set $x_t^a = 0$ (i.e., inactive). Let E_t^a denote the amount of energy required to finish the operation of appliance a while the current time slot is t. Note that given E_t^a, for each future time slot $k > t > 0$, we have

$$E_k^a = \left[E_t^a - \gamma_a \sum_{i=t}^{k-1} x_i^a \right]^+ . \tag{3.1}$$

For controllable appliances that are non-interruptible, their operation can be delayed, but once they become active, they must remain active until the end of their operation.

Thus, for each non-interruptible controllable appliance a, we have

$$x_k^a = 1, \quad \forall\, k \in \{t, \ldots, \beta_a\},\ 0 < E_k^a < E_a. \tag{3.2}$$

Let $l_t \triangleq \sum_{a \in \mathcal{A}} \gamma_a x_t^a$ denote the total household power consumption at time slot t. We consider a pricing function $\lambda_t(l_t)$ which represents the price of electricity in each time slot t as a function of the user's power consumption in that time slot. For combined RTP and IBR pricing tariffs, the price function $\lambda_t(l_t)$ is defined as [12]:

$$\lambda_t(l_t) = \begin{cases} m_t, & \text{if } \ 0 \leq l_t \leq b_t, \\ n_t, & \text{if } \ l_t > b_t, \end{cases} \tag{3.3}$$

where m_t, n_t, and b_t are pre-determined parameters. We have $m_t \leq n_t$. Recall from Section 3.1 that a combined RTP and IBR pricing model can effectively avoid load synchronization [12].

3.3 Problem Formulation and Algorithm Description

In this section, we consider the problem of efficient power scheduling such that the electricity payment of each user is minimized. We assume that only some statistical demand information are known ahead of time. The exact information about the list of appliances that are awake in each time slot, whether they are must-run or controllable, and the deadline by which the operation of each appliance should be finished is revealed only gradually over time. We note that different sets of awake appliances, i.e., must-run and controllable, at future time slot $k > t$, can be separated into two disjoint subsets. The first subset includes those appliances that are awake at the current time slot t and remain awake at time slot $k > t$, i.e., the information about this subset of appliances is known at the current time

slot. The second subset consists of those appliances that are asleep at the current time slot t and will be awake at time slot $k>t$. However, at the current time slot t, only some statistical information about this subset of appliances is available. An update is received by the ECC unit at the beginning of each time slot, and the energy consumption schedule of each controllable appliance is adapted accordingly.

3.3.1 Problem Formulation

The optimum operation schedule can be determined if the demand information of all appliances is available at the beginning of the scheduling horizon. However, we assume here that the demand information of the appliances is not known and instead only stochastic information regarding the demand is available a priori. Thus, we formulate a scheduling problem that minimizes the *expected* energy payment of the user with respect to demand uncertainties. In each time slot t, as the demand information of the appliances is updated, the operation schedule of each controllable appliance can be rescheduled. The power scheduling can be identified in real-time as the solution of the following optimization problem:

$$\begin{aligned}
\underset{\substack{\mathbf{x}_t^a,\ \forall\, a\,\in\,\tilde{\mathcal{C}}_k,\\ \forall\, k\in\{t,\ldots,T\}}}{\text{minimize}} \quad & \mathbb{E}\left\{ L_t\lambda_t\big(L_t\big) + \sum_{k=t+1}^{T} L_{k,t}\lambda_k\big(L_{k,t}\big) \right\} \\
\text{subject to} \quad & x_k^a \in \{0,1\},\ \ \forall\, a\in\tilde{\mathcal{C}}_k,\ k\in\{t,\ldots,T\}, \\
& \gamma_a \sum_{k=t}^{\beta_a} x_k^a = E_t^a, \quad \forall\, a\in\tilde{\mathcal{C}}_k, \\
& x_k^a = 1,\ 0<E_k^a<E_a, \quad \forall\, a\in\tilde{\mathcal{N}}_k,\ k\in\{t,\ldots,\beta_a\},
\end{aligned} \tag{3.4}$$

where $\mathbb{E}\{\cdot\}$ denotes the expectation,

$$L_t = \sum_{a \in \tilde{\mathcal{M}}_t} \gamma_a + \sum_{a \in \tilde{\mathcal{C}}_t} \gamma_a x_t^a, \tag{3.5}$$

$$L_{k,t} = \sum_{a \in \mathcal{M}_{k,t}} \gamma_a + \sum_{a \in \hat{\mathcal{M}}_{k,t}} \gamma_a + \sum_{a \in \mathcal{C}_{k,t}} \gamma_a x_k^a + \sum_{a \in \hat{\mathcal{C}}_{k,t}} \gamma_a x_k^a, \tag{3.6}$$

$\mathbf{x}_t^a \triangleq (x_t^a, \ldots, x_T^a)$, E_k^a is as in (3.1), and the definitions of the different sets of appliances $\mathcal{M}_{k,t}$, $\hat{\mathcal{M}}_{k,t}$, $\tilde{\mathcal{M}}_t$, $\mathcal{C}_{k,t}$, $\hat{\mathcal{C}}_{k,t}$, $\tilde{\mathcal{C}}_k$, and $\tilde{\mathcal{N}}_k$ are presented in Table 3.1.

We note that E_t^a is known at time slot t, must-run appliances are active as long as they are awake, $\mathcal{C}_{k,t} = \mathcal{C}_{t,t}$ for all $k \geq t$, and as the demand information is known up to time slot t, $\hat{\mathcal{M}}_{t,t} = \hat{\mathcal{C}}_{t,t} = \emptyset$. The first term in the objective function in (3.4) is the payment of the user in the current time slot t for the known load L_t, while the second term is the expected cost of energy in the upcoming time slots. Each appliance can be either *on* or *off*. This is indicated by the first constraint. The second constraint implies that the operation of each appliance should be finished by its deadline. The last constraint guarantees that the operation of non-interruptible appliances will continue after they become active until they finish their job.

In our stochastic model, it is possible to devise different objectives and different strategies to schedule the operation of different appliances. The performances of different scheduling strategies are different. However, their different performances may be compared based on their average performance and their worst case performance for different demand requirements of the user. Problem (3.4) in its current form is difficult to solve as it requires the computation of the expected schedule for currently sleeping appliances[2]. To tackle

[2]One option to solve problem (3.4) is to formulate it as a dynamic programming problem. Considering the amount of information required to describe the state of each appliance, i.e., whether the appliance is awake or not, the remaining energy requirements, and the number of time slots remaining to reach the deadline, we may be faced with a huge state space of outcomes. Dynamic programming may suffer from

Table 3.1: List of variables used in this chapter.

$\mathcal{A}$	Set of appliances
T	Number of time slots
γ_a	Nominal power of appliance a
E_a	Total required energy of appliance a
β_a	Operating deadline of appliance a
x_t^a	State of power consumption of appliance a at time slot t
E_t^a	Remaining required energy of appliance a at time slot t
l_t	Total household power consumption at time slot t
$\lambda_t(\cdot)$	Price function at time slot t
m_t	Price parameter at time slot t
n_t	Price parameter at time slot t
b_t	Price parameter at time slot t
$\mathcal{M}_{k,t}$	Set of must-run appliances that are awake at time slot t and remain awake at time slot $k \geq t$
$\hat{\mathcal{M}}_{k,t}$	Set of must-run appliances that are asleep at time slot t and will be awake at time slot $k \geq t$
$\tilde{\mathcal{M}}_k$	Set of all must-run appliances that are awake at time slot k $(\mathcal{M}_{k,t} \cup \hat{\mathcal{M}}_{k,t})$
$\mathcal{C}_{k,t}$	Set of controllable appliances that are awake at time slot t and remain awake at time slot $k \geq t$
$\hat{\mathcal{C}}_{k,t}$	Set of controllable appliances that are asleep at time slot t and will be awake at time slot $k \geq t$
$\tilde{\mathcal{C}}_k$	Set of all controllable appliances that are awake at time slot k $(\mathcal{C}_{k,t} \cup \hat{\mathcal{C}}_{k,t})$
$\mathcal{N}_{k,t}$	Set of non-interruptible appliances of $\mathcal{C}_{k,t}$
$\hat{\mathcal{N}}_{k,t}$	Set of non-interruptible appliances of $\hat{\mathcal{C}}_{k,t}$
$\tilde{\mathcal{N}}_k$	Set of all non-interruptible appliances that are awake at time slot k $(\mathcal{N}_{k,t} \cup \hat{\mathcal{N}}_{k,t})$
$\mathcal{S}_t$	Set of all appliances that are sleeping at time slot t
y_k^a	Auxiliary variable for each non-interruptible appliance a at each time slot k
M	Auxiliary large number used in the problem formulation
ν_k	Auxiliary variable for each time slot k
p_t^a	Probability with which appliance a becomes awake at time slot t
q_a	Probability that appliance a does not become awake at any time slot
$p_{\tau,t}^a$	Probability that appliance a becomes awake at time slot $\tau > t$ given that it has not become awake until time slot t
$\delta_{\tau,t}^a$	Probability that must-run appliance a which is sleeping at time slot t will be active in time slot $\tau > t$
T_a	Number of time slots required to finish the operation of appliance a

the curse of dimensionality [82].

this problem, we minimize an *upper bound* of the objective function. We assume all appliances that become awake in future time slots are must-run appliances. In this case, the *risk of loss* for the user is minimized. That is, from the the user's electricity payment point of view, the worst performance of the ECC unit for different scheduling strategies is minimized.

$$\begin{aligned} \underset{\mathbf{x}_t^a,\, \forall\, a\in\tilde{\mathcal{C}}_t}{\text{minimize}} \quad & L_t\lambda_t\big(L_t\big) + \sum_{k=t+1}^{T} \mathbb{E}\left\{\bar{L}_{k,t}\lambda_k\big(\bar{L}_{k,t}\big)\right\} \\ \text{subject to} \quad & x_k^a \in \{0,1\}, \quad \forall\, a \in \tilde{\mathcal{C}}_t,\ k \in \{t,\ldots,T\}, \\ & \gamma_a \sum_{k=t}^{\beta_a} x_k^a = E_t^a, \quad \forall\, a \in \tilde{\mathcal{C}}_t, \\ & x_k^a = 1,\ 0 < E_k^a < E_a, \quad \forall\, a \in \tilde{\mathcal{N}}_t,\ k \in \{t,\ldots,\beta_a\}, \end{aligned} \tag{3.7}$$

where

$$\bar{L}_{k,t} = \sum_{a\in\mathcal{M}_{k,t}} \gamma_a + \sum_{a\in\hat{\mathcal{M}}_{k,t}\cup\hat{\mathcal{C}}_{k,t}} \gamma_a + \sum_{a\in\mathcal{C}_{k,t}} \gamma_a x_k^a \tag{3.8}$$

denotes the load at time slot $k > t$.

Problem (3.7) is still difficult to solve in its current form since the last constraint is conditioned on the value of E_k^a for $k \in \{t+1,\ldots,\beta_a\}$. From (3.1), for $k \in \{t+1,\ldots,\beta_a\}$, E_k^a depends on variable x_i^a for $i \in \{t,\ldots,k-1\}$, which is unknown and should be determined. However, by introducing an auxiliary variable $y_k^a \in \{0,1\}$ for each appliance $a \in \tilde{\mathcal{N}}_t$ and at each time slot $k \in \{t+1,\ldots,\beta_a\}$, the problem formulation in (3.7) can be re-written in a more tractable form. Here, the auxiliary variable y_k^a indicates whether the operation of appliance a is finished ($y_k^a = 1$) or not ($y_k^a = 0$) at a particular time slot $k \in \{t+1,\ldots,\beta_a\}$.

Thus, we can re-write problem (3.7) as

$$\underset{\substack{\mathbf{x}_t^a,\ \forall\, a\,\in\,\tilde{\mathcal{C}}_t\\ \mathbf{y}_t^a,\ \forall\, a\,\in\,\tilde{\mathcal{N}}_t}}{\text{minimize}} \quad L_t\lambda_t\big(L_t\big)+\sum_{k=t+1}^{T}\mathbb{E}\left\{\bar{L}_{k,t}\lambda_k\big(\bar{L}_{k,t}\big)\right\}+M\sum_{k=t}^{T}\sum_{a\in\tilde{\mathcal{C}}_t}y_k^a \tag{3.9a}$$

$$\text{subject to} \quad x_k^a\in\{0,1\},\ \forall\, a\in\tilde{\mathcal{C}}_t,\ \ k\in\{t,\ldots,T\}, \tag{3.9b}$$

$$y_k^a\in\{0,1\},\ \forall\, a\in\tilde{\mathcal{N}}_t,\ \ k\in\{t,\ldots,T\}, \tag{3.9c}$$

$$\gamma_a\sum_{k=t}^{\beta_a}x_k^a=E_t^a,\quad \forall\, a\in\tilde{\mathcal{C}}_t, \tag{3.9d}$$

$$y_k^a+\frac{E_t^a-\gamma_a\sum_{i=t}^{k-1}x_i^a}{E_a}\geq\epsilon,\quad \forall\, a\in\tilde{\mathcal{N}}_t,\ k\in\{t,\ldots,\beta_a\}, \tag{3.9e}$$

$$x_k^a+y_k^a+\frac{E_t^a-\gamma_a\sum_{i=t}^{k-1}x_i^a}{E_a}\geq 1,\quad \forall\, a\in\tilde{\mathcal{N}}_t,\ k\in\{t,\ldots,\beta_a\}, \tag{3.9f}$$

where $\mathbf{y}_t^a\triangleq(y_t^a,\ldots,y_T^a)$, M is a constant, and $0<\epsilon<\min\{\frac{\gamma_1}{E_1},\ldots,\frac{\gamma_{|\mathcal{A}|}}{E_{|\mathcal{A}|}}\}$ is a small constant. We can justify the new constraints as follows. In (3.9e), when $E_k^a=E_t^a-\gamma_a\sum_{i=t}^{k-1}x_i^a=0$, y_k^a becomes 1. However, as long as the operation is not finished, i.e., $E_k^a>0$, since y_k^a appears in the objective of the minimization problem, we have $y_k^a=0$. This is true, since for any value $E_k^a>0$, we have $\frac{E_k^a}{E_a}>\epsilon$. In (3.9f), when $E_k^a=E_a$, we have $y_k^a=0$, and x_k^a can be either 0 or 1, when $E_a>E_k^a>0$, we have $y_k^a=0$ and x_k^a has to be 1. However, when $E_k^a=0$, we have $y_k^a=1$, and since x_k^a appears in the objective of the minimization problem, it has to be 0.

For the price function in (3.3), since $m_t\leq n_t$, for a total load l_t at time slot t, the user's payment $l_t\times\lambda_t(l_t)$ is determined as the maximum of the two intersecting lines [12]:

$$l_t\times\lambda_t(l_t)=\max\big\{m_tl_t,\ n_tl_t+(m_t-n_t)b_t\big\}. \tag{3.10}$$

Thus problem (3.9) can be reformulated as

$$\begin{aligned}\underset{\substack{\mathbf{x}_t^a,\ \forall\, a\in\tilde{\mathcal{C}}_t\\ \mathbf{y}_t^a,\ \forall\, a\in\tilde{\mathcal{N}}_t}}{\text{minimize}}\ & \max\left\{m_t\left(\sum_{a\in\tilde{\mathcal{C}}_t}\gamma_a x_t^a+\sum_{a\in\tilde{\mathcal{M}}_t}\gamma_a\right), n_t\left(\sum_{a\in\tilde{\mathcal{C}}_t}\gamma_a x_t^a+\sum_{a\in\tilde{\mathcal{M}}_t}\gamma_a\right)+(m_t-n_t)b_t\right\}\\ &+\sum_{k=t+1}^{T}\mathbb{E}\left\{\max\left\{m_k\left(\sum_{a\in\tilde{\mathcal{C}}_t}\gamma_a x_k^a+l_{k,t}\right), n_k\left(\sum_{a\in\tilde{\mathcal{C}}_t}\gamma_a x_k^a+l_{k,t}\right)+(m_k-n_k)b_k\right\}\right\}\\ &+M\sum_{k=t}^{T}\sum_{a\in\tilde{\mathcal{C}}_t}y_k^a \qquad (3.11)\end{aligned}$$

subject to (3.9b)-(3.9f),

where

$$l_{k,t}\triangleq\sum_{a\in\mathcal{M}_{k,t}}\gamma_a+\sum_{a\in\hat{\mathcal{M}}_{k,t}\cup\hat{\mathcal{C}}_{k,t}}\gamma_a. \qquad (3.12)$$

Finally, by introducing another auxiliary variable, ν_k, for each time slot k, and by adopting the *certainty equivalent* approximation technique, i.e., all uncertainties are fixed at their expected value [83], we can re-write problem (3.11) as

$$\begin{aligned}\underset{\substack{\boldsymbol{\nu}_t,\,\mathbf{x}_t^a,\ \forall\, a\in\tilde{\mathcal{C}}_t\\ \mathbf{y}_t^a,\ \forall\, a\in\tilde{\mathcal{N}}_t}}{\text{minimize}}\ & \sum_{k=t}^{T}\nu_k+M\sum_{k=t}^{T}\sum_{a\in\tilde{\mathcal{C}}_t}y_k^a \qquad (3.13)\\ \text{subject to}\ & \text{(3.9b)-(3.9f)},\\ & m_t\left(\sum_{a\in\tilde{\mathcal{C}}_t}\gamma_a x_t^a+\sum_{a\in\tilde{\mathcal{M}}_t}\gamma_a\right)\le\nu_t,\\ & n_t\left(\sum_{a\in\tilde{\mathcal{C}}_t}\gamma_a x_t^a+\sum_{a\in\tilde{\mathcal{M}}_t}\gamma_a\right)+(m_t-n_t)b_t\le\nu_t,\\ & m_k\left(\sum_{a\in\tilde{\mathcal{C}}_t}\gamma_a x_k^a+\hat{l}_{k,t}\right)\le\nu_k,\ \forall\, k\in\{t+1,\ldots,T\},\end{aligned}$$

$$n_k\left(\sum_{a\in\tilde{\mathcal{C}}_t}\gamma_a x_k^a+\hat{l}_{k,t}\right)+(m_k-n_k)b_k \leq \nu_k,\ \forall\, k\in\{t+1,\ldots,T\},$$

where $\boldsymbol{\nu}_t \triangleq (\nu_t,\ldots,\nu_T)$, and $\hat{l}_{k,t}\triangleq\mathbb{E}\{l_{k,t}\}$, the estimate of the power consumption of must-run appliances in an upcoming time slot $k \geq t$ will be calculated in the next sub-section. Problem (3.13) is a mixed-binary linear program and can be solved efficiently by using MOSEK [84]. The solution of optimization problem (3.13) determines the appropriate scheduling for the operation of controllable appliances. However, for interruptible appliances, only the operation schedule of the current time slot t will be executed, and the schedule of the future time slots $t+1,\ldots,T$ may change when the optimization problem is solved again in the next time slot as new information about the future load becomes available.

3.3.2 Load Estimation

In our system model, we assume that the demand information of the appliances is *not* known ahead of time, i.e., in (3.12), the set of awake appliances in the upcoming time slots $k > t$ that are currently sleeping, $\hat{\mathcal{M}}_{k,t}\cup\hat{\mathcal{C}}_{k,t}$, is not known. Instead, only the probability with which each appliance becomes awake at each time slot t, p_t^a, is known before the operation cycle begins. Such information can be calculated, for example, based on the sleep and awake history of each appliance. For this purpose, we can observe a window of N consecutive days and mark those days in which appliance a becomes awake in a particular time slot t. The ratio of the number of marked days to the total number of observed days determines the probability with which appliance a becomes awake in time slot t, $p_t^a \triangleq \mathbb{P}(\Delta_t^a = 1)$, where Δ_t^a is a random variable that is equal to one if appliance a becomes awake in time slot t, and equal to zero otherwise. In our model, each appliance can become awake only once. If an appliance becomes awake more often, we can simply

introduce *virtual* appliances to deal with this issue. Therefore, we have

$$\sum_{t=1}^{T} p_t^a + q_a = 1, \tag{3.14}$$

where q_a denotes the probability that appliance a does *not* become awake at *any* time within the DSM's operation period $[1, T]$. We define $p_{\tau,t}^a$ as the probability that appliance a becomes awake in time slot $\tau > t$ given that it has *not* become awake until time slot t. That is,

$$p_{\tau,t}^a = \mathbb{P}(\Delta_\tau^a = 1 \mid \Delta_1^a = 0, \ldots, \Delta_t^a = 0). \tag{3.15}$$

Based on Bayes rule, $p_{\tau,t}^a$ can be calculated as

$$p_{\tau,t}^a = \frac{\mathbb{P}(\Delta_1^a = 0, \ldots, \Delta_t^a = 0 \mid \Delta_\tau^a = 1)\mathbb{P}(\Delta_\tau^a = 1)}{\mathbb{P}(\Delta_1^a = 0, \ldots, \Delta_t^a = 0)}. \tag{3.16}$$

If appliance a becomes awake at time slot $\tau > t$, it implies that it has not become awake in previous time slots. Therefore, $\mathbb{P}(\Delta_1^a = 0, \ldots, \Delta_t^a = 0 \mid \Delta_\tau^a = 1) = 1$. On the other hand, we obtain $\mathbb{P}(\Delta_1^a = 0, \ldots, \Delta_t^a = 0) = \sum_{k=t+1}^{T} p_k^a + q_a$ based on (3.14). We also have $\mathbb{P}(\Delta_\tau^a = 1) = p_\tau^a$. Therefore, (3.16) becomes

$$p_{\tau,t}^a = \frac{p_\tau^a}{\sum_{k=t+1}^{T} p_k^a + q_a}. \tag{3.17}$$

Next, assume that all appliances that become awake in future time slots are must-run appliances, and must-run appliances start operation once they become awake, see Section 3.3.1. Let Λ_τ^a denote the random variable that indicates whether must-run appliance a is active ($\Lambda_\tau^a = 1$) or not active ($\Lambda_\tau^a = 0$) in time slot τ. Also, let $\delta_{\tau,t}^a$ denote the probability that a must-run appliance a which is sleeping in time slot t will be active in time slot $\tau > t$. By conditioning on the time slot in which must-run appliance a becomes awake, $\delta_{\tau,t}^a$ can

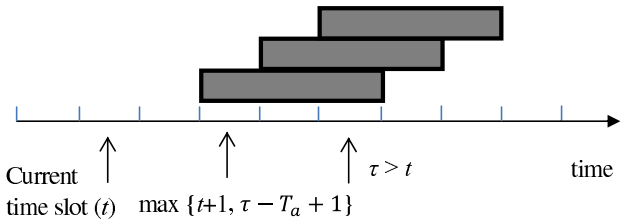

Figure 3.2: Must-run appliance $a \in \mathcal{S}_t$ with $T_a = 3$ will be active in $\tau > t$ if it starts operation within time interval $[\max\{t+1, \tau - T_a + 1\}, \tau]$.

be calculated as

$$\begin{aligned}\delta^a_{\tau,t} &= \mathbb{P}(\Lambda^a_\tau = 1 \mid \Delta^a_1 = 0, \ldots, \Delta^a_t = 0) \\ &= \sum_{k=t+1}^{T} \mathbb{P}(\Lambda^a_\tau = 1 \mid \Delta^a_1 = 0, \ldots, \Delta^a_t = 0, \Delta^a_k = 1) p^a_{k,t}, \end{aligned} \tag{3.18}$$

where $p^a_{k,t}$ is defined in (3.15). As illustrated in Fig. 3.2, a currently sleeping appliance will be active in time slot $\tau > t$, if it starts operation within time frame $[\max\{t+1, \tau - T_a + 1\}, \tau]$, where $T_a \triangleq \frac{E_a}{\gamma_a}$ is defined as the number of time slots required to finish the operation of appliance a while operating at power level γ_a. For simplicity, we assume T_a is integer. Therefore, $\mathbb{P}(\Lambda^a_\tau = 1 \mid \Delta^a_1 = 0, \ldots, \Delta^a_t = 0, \Delta^a_k = 1) = 1$ if $k \in [\max\{t+1, \tau - T_a + 1\}, \tau]$, and $\mathbb{P}(\Lambda^a_\tau = 1 \mid \Delta^a_1 = 0, \ldots, \Delta^a_t = 0, \Delta^a_k = 1) = 0$ otherwise. Thus, we have

$$\delta^a_{\tau,t} = \sum_{k=\max\{t+1, \tau - T_a + 1\}}^{\tau} p^a_{k,t}. \tag{3.19}$$

Finally, by conditioning on the event of observing a currently sleeping appliance active in an upcoming time slot τ, while the system is at time slot t, the estimate of the power consumption required in (3.13) becomes:

$$\hat{l}_{\tau,t} = \mathbb{E}\{l_{\tau,t}\} = \sum_{a \in \mathcal{M}_{\tau,t}} \gamma_a + \sum_{a \in \mathcal{S}_t} \gamma_a \delta^a_{\tau,t}, \tag{3.20}$$

Algorithm 3.1: Energy consumption scheduling algorithm in presence of load uncertainty executed at the beginning of each time slot t.

1: Receive admission requests.
2: Label received requests either as must-run or controllable.
3: Activate must-run appliances (start / continue operation).
4: Update $p^a_{\tau,t}$ according to (3.17).
5: Update $\delta^a_{\tau,t}$ according to (3.19).
6: Update $\hat{l}_{\tau,t}$ according to (3.20).
7: Update E^a_t according to (3.1).
8: Solve (3.13) to activate / deactivate controllable appliances.
9: **if** activated device is non-interruptible
10: Mark it as must-run.
11: **end if**

where $\mathcal{S}_t$ is defined in Table 3.1.

3.3.3 Algorithm Description

In this section, we explain the different steps of the proposed energy consumption scheduling algorithm in presence of load uncertainty (Algorithm 3.1) executed at each time slot t.

Step 1 At the beginning of the admission control phase at each time slot, all received admission requests are labeled as either must-run or controllable, c.f. Lines 1 and 2.

Step 2 Activate must-run appliances $a \in \tilde{\mathcal{M}}_t$ right away, c.f. Line 3. That is, start or continue their operation at the requested power γ_a. Their operation will not be interrupted, and they remain must-run until the end of their operation.

Step 3 In Line 4, considering the list of appliances that have already become awake, update the probabilities at which other appliances will send an admission request in the upcoming time slots as in (3.17). Adopt (3.19) to update the probabilities with which sleeping devices become active in upcoming time slots, c.f. Line 5.

Step 4 Use the current information to calculate the expected load in the upcoming time slots using (3.20) as indicated in Line 6. Update the remaining required energy of each appliance at the beginning of the current time slot, i.e., E_t^a, using (3.1), c.f. Line 7.

Step 5 Next, set the "on" / "off" state of each awake controllable appliance for the rest of the time slots by solving optimization problem (3.13), c.f. Line 8.

Step 6 In Lines 9 to 11, if any non-interruptible controllable appliance became active (i.e., it switched from *off* to *on*) in Step 5, remove it from the list of controllable appliances and add it to list of must-run devices as it should remain on until it finishes its operation.

3.4 Performance Evaluation

In this section, we present simulation results and assess the performance of our proposed DSM algorithm. We run the simulation multiple times with different patterns for the times at which the appliances become awake. We then present the average results. Unless stated otherwise, the simulation setting is as follows. We assume that the general RTP method combined with IBR is adopted as described in (3.3). In our system model, the retail price parameters, m_t, n_t, and b_t, are set by the retail energy provider to compensate the cost of providing energy and to shape the daily energy consumption of the user. However, these parameters are different from the load cost profile of the energy provider, as the load cost profile of the energy provider is determined in the wholesale electricity market. The exact load cost profile of the retailer is usually not known to the end users. Fig. 3.3 illustrates the variation of parameters m_t and n_t of the price function over one day. We consider a single household with various must-run and controllable appliances. Controllable appliances can be either interruptible or non-interruptible. Non-interruptible appliances include: electric

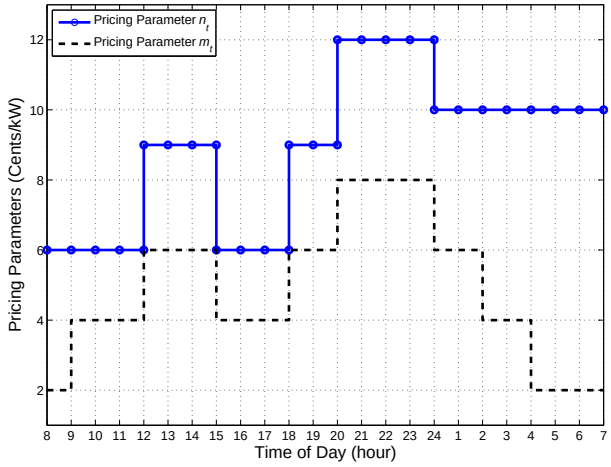

Figure 3.3: The pricing parameters used based on the combined RTP and IBR pricing model in (3.3). Parameter $b_t = 3.5$ kW is fixed for all time slots.

stove ($E_a = 4.5$ kWh, $\gamma_a = 1.5$ kW), clothes dryer ($E_a = 1$ kWh, $\gamma_a = 0.5$ kW), and vacuum cleaner ($E_a = 3$ kWh, $\gamma_a = 1.5$ kW). Interruptible appliances include: Refrigerator ($E_a = 2.5$ kWh, $\gamma_a = 0.125$ kW), air conditioner ($E_a = 6$ kWh, $\gamma_a = 1.5$ kW), dishwasher ($E_a = 2$ kWh, $\gamma_a = 1$ kW), heater ($E_a = 4$ kWh, $\gamma_a = 1$ kW), water heater ($E_a = 2$ kWh, $\gamma_a = 1$ kW), pool pump ($E_a = 4$ kWh, $\gamma_a = 2$ kW), and PEV ($E_a = 10$ kWh, $\gamma_a = 2.5$ kW). Must-run appliances include: Lighting ($E_a = 3$ kWh, $\gamma_a = 0.5$ kW), TV ($E_a = 1$ kWh, $\gamma_a = 0.25$ kW), PC ($E_a = 1.5$ kWh, $\gamma_a = 0.25$ kW), ironing appliance ($E_a = 2$ kWh, $\gamma_a = 1$ kW), hairdryer ($E_a = 1$ kWh, $\gamma_a = 1$ kW), and others ($E_a = 6$ kWh, $\gamma_a = 1.5$ kW). The details of the average annual energy consumption of different appliances and the average monthly energy consumption of residential users in the US can be found in [85] and [86]. The time slot at which each appliance becomes awake is selected randomly from a pre-determined time interval, e.g. [6:00, 14:00] for electric stove and [16:00, 24:00] for PEV.

3.4.1 Performance Gains of Users and Utility Company

To have a baseline to compare with, we consider a system without ECC deployment, where each appliance a is assumed to start operation right after it becomes awake at its nominal power γ_a. As an upper bound, we also consider a system with ECC deployment in which all the demand information of the appliances is available ahead of time. Simulation results for the average total power consumption for the proposed residential load control algorithm, the system without ECC deployment, and the system in which complete demand information is available ahead of time are depicted in Fig. 3.4. In our simulation model, we set $b_t = 3.5$ kW in (3.3) for all time slots. As illustrated in Fig. 3.4, to reduce electricity payment, the ECC unit shifts the load to time slots with lower prices such as the after midnight hours. However, the high price penalty for exceeding the b_t threshold prevents load synchronization as discussed in Section 3.1. The simulation results show that exploiting the use of the ECC unit reduces the average daily payment of the user from 4.76 Dollars/day to 4.01 Dollars/day. For the case, where complete information about the demand of each appliance is available ahead of time, the average daily payment of the user is 3.92 Dollars/day. To evaluate the PAR, the user's daily peak load is divided by his daily average load. That is, after running the algorithm, at the end of the operation period, we compute

$$PAR = \frac{T \max\{l_1, \ldots, l_T\}}{\sum_{k=1}^{T} l_k}, \tag{3.21}$$

where l_k is the total power consumption of the user at time slot k. The proposed algorithm also helps to reduce the average PAR of the system from 2.66 to 1.98 (25.5% PAR reduction) compared to the system without ECC deployment. The PAR of the system with ECC deployment if complete demand information is known a priori is 1.89.

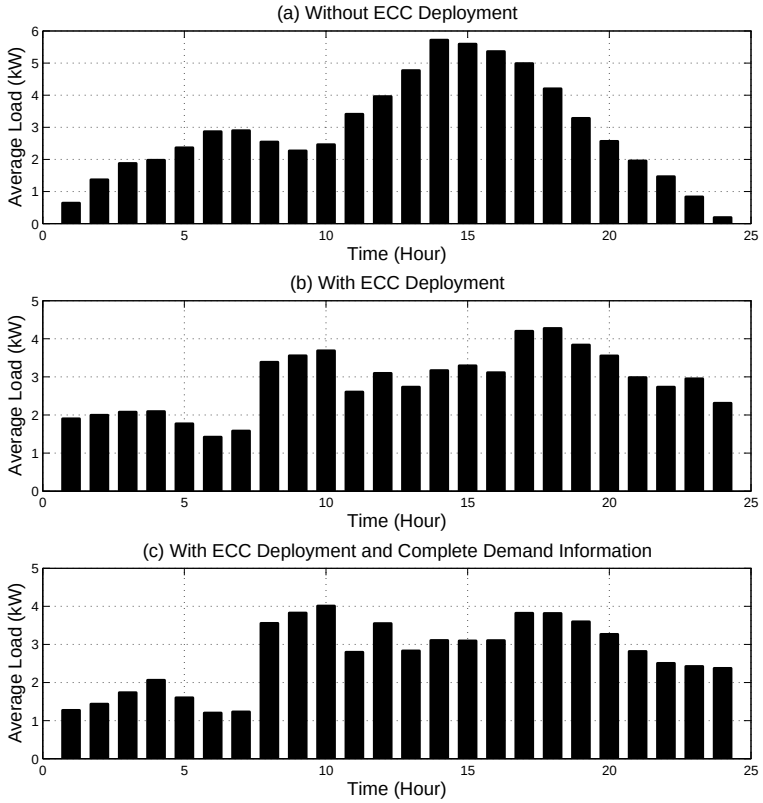

Figure 3.4: Power consumption for (a) the system without ECC deployment, (b) the system with ECC deployment, and (c) the system with ECC deployment in which complete demand information is available ahead of time.

3.4.2 Computational Complexity

In general, integer linear programs with n integer variables and m constraints are known to be NP-complete [87]. However, there exist pseudo-polynomial algorithms for solving $m \times n$ integer programs with fixed m which have an order of complexity of

$$O(n^{2m+2}(m\alpha)^{(m+1)(2m+1)} \log(n^2(m\alpha^2)^{2m+3})),$$

where α is the maximum coefficient in the set of constraints [88]. A complete discussion of the complexity of such algorithms is out of the scope of this chapter. To illustrate the

complexity of our proposed algorithm, simulation results for the average run time of the algorithm, the number of integer variables, and the number of constraints for different numbers of appliances and different time granularities are given for one time slot in Table 3.2. The order of complexity of the algorithm determines the maximum run-time or the maximum number of elementary operation required to solve the problem for any input scenarios. In practice, the times at which different appliances become awake are distributed over the operation horizon, and it is unlikely that all appliances become awake at the same time. Thus, at each time slot, the number of awake appliances required to be scheduled is limited. This can significantly reduce the average run time of the algorithm in most practical scenarios. By increasing the time granularity, the number of integer variables and the number of constraints are increased, since the number of time slots at which the operation of each appliance should be scheduled is increased. However, the effect of this increase is mitigated, since the times at which different appliances become awake are distributed over a larger number of time slots, and the number of awake appliances in each time slot is reduced.

3.4.3 The Impact of Price Control Parameter b_t

Considering the price function as described in (3.3), in each time slot, if the power consumption of the user exceeds a certain threshold defined as b_t, the user will be penalized by paying a higher price. The choice of parameter b_t has a significant impact on users' payments and the PAR. To have a baseline to compare with, similar to [41], we consider a system in which the effect of IBR is ignored and only the basic price in each time slot is taken into account to schedule the operation of different appliances in order to minimize the electricity payment of the user. Simulation results for the average payment of the user and the PAR of the system for different values of parameter b_t are shown in Figs. 3.5 and

Table 3.2: Performance measures and complexity analysis of the proposed algorithm.

Average run time of the algorithm (in seconds).

Time granularity	$\|\mathcal{A}\|$=20	$\|\mathcal{A}\|$=25	$\|\mathcal{A}\|$=35
1 hour	0.0287	0.0308	0.0350
30 minutes	0.0294	0.0316	0.0422
15 minutes	0.0302	0.0318	0.0988

Average number of integer variables.

Time granularity	$\|\mathcal{A}\|$=20	$\|\mathcal{A}\|$=25	$\|\mathcal{A}\|$=35
1 hour	80	156	296
30 minutes	121	230	440
15 minutes	168	316	619

Average number of constraints.

Time granularity	$\|\mathcal{A}\|$=20	$\|\mathcal{A}\|$=25	$\|\mathcal{A}\|$=35
1 hour	43	79	139
30 minutes	46	81	142
15 minutes	54	87	145

3.6, respectively. Intuitively, increasing the price parameter b_t for each time slot results in a reduction of the user's payment as shown in Fig. 3.5. Considering the average PAR, for the system without ECC deployment and the system without IBR consideration, the PAR does not change as the user does not respond to changes of parameter b_t. For the system with ECC deployment, for low values of parameter b_t, even the load of must-run appliances in most time slots exceeds this threshold. Thus, the ECC unit mainly considers the n_t price parameter to schedule the operation of controllable appliances. However, by increasing parameter b_t, since the load of must-run appliances lies below threshold b_t in some time slots, the user is encouraged to shift the controllable portion of its load to avoid paying higher price n_t rather than lower price m_t, which initially results in reducing the PAR. Later on, a further increase of parameter b_t reduces the effect of IBR, entails load synchronization effects, and increases the PAR of the system. That is, for large values of

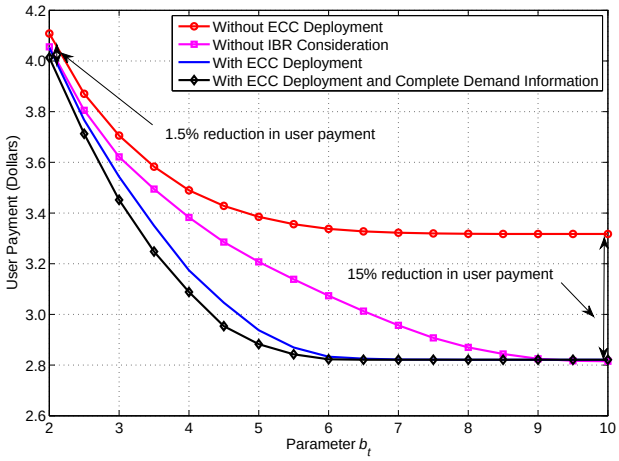

Figure 3.5: The daily payment of the user for different values of parameter b_t.

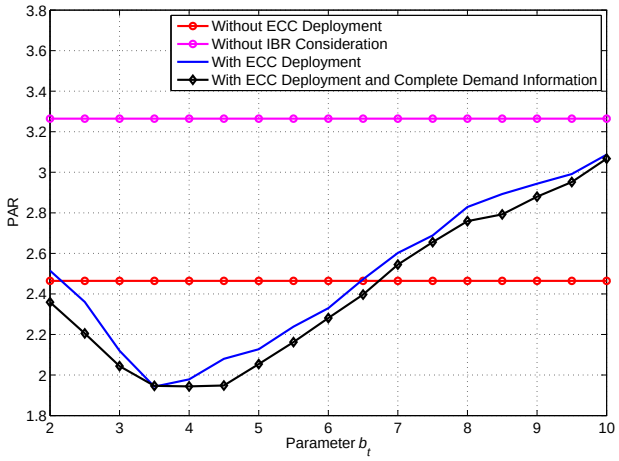

Figure 3.6: PAR of the system for different values of parameter b_t.

parameter b_t, it is less likely that the load of the user exceeds this threshold, and the ECC unit mainly pays attention to the value of the price parameter m_t in order to schedule the operation of controllable appliances. This results in shifting a large portion of the load to low price time slots. Therefore, for large values of parameter b_t, the performance of our proposed method approaches the one without IBR consideration as shown in Fig. 3.6.

3.4.4 The Impact of Adopting Inclining Block Rates

In this section, we examine how changes of the two parameters m_t and n_t of the price function will affect the performance of the system. In our simulation model, parameter m_t changes as illustrated in Fig. 3.3 and we set $b_t = 3$ kW for all time slots. However, parameter n_t is given by

$$n_t = \theta m_t, \quad \forall\, t \in \{1, \ldots, T\}. \tag{3.22}$$

Simulation results for the average daily payment of the user as well as the average PAR of the system for different values of parameter θ are depicted in Figs. 3.7 and 3.8, respectively. Intuitively, when θ is equal to one, i.e., when $m_t = n_t$ for all t, the performance of our proposed method is the same as the one in which the effect of IBR is ignored. However, by increasing parameter θ, the payment of the user will be increased, as the user has to pay more every time that its load exceeds threshold b_t as shown in Fig. 3.7. As indicated in Fig. 3.8, increasing parameter θ improves the PAR of the system, as load synchronization is prevented. That is, to avoid paying at higher price n_t, the ECC unit tries to distribute the load such that it does not exceed the b_t threshold. However, for the system without IBR consideration, changes of parameter θ do not affect the PAR.

3.5 Summary

In this chapter, we proposed an optimal residential load control algorithm for DSM in presence of load uncertainty. We formulated an optimization problem to minimize the electricity payment of the users in situations where only an estimate of the future demand is available. We focused on a scenario where real-time pricing is combined with IBRs to balance residential load to achieve a low PAR. Simulation results showed that the proposed algorithm reduces the energy cost of users, encouraging them to participate in

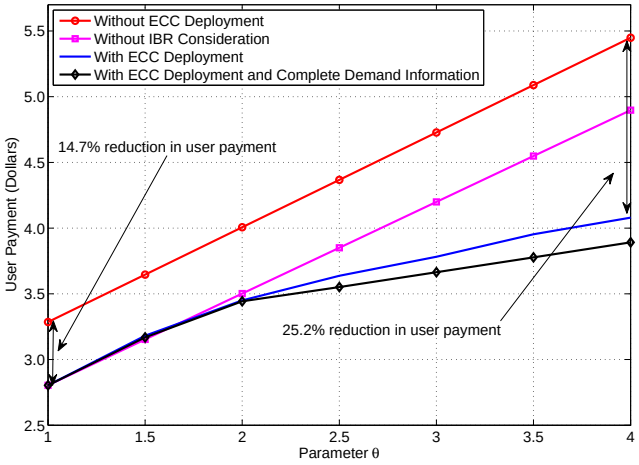

Figure 3.7: The daily payment of the user for different values of parameter θ.

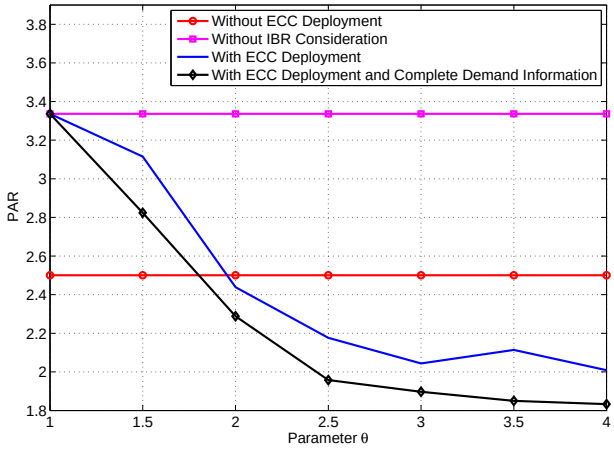

Figure 3.8: PAR of the system for different values of parameter θ.

DSM. Exploiting IBR with RTP tariffs can help to avoid load synchronization, and the combination of the general RTP method with our algorithm reduces the PAR of the total load. The latter provides incentives for utilities to support implementing the proposed algorithm.

Chapter 4

Real-Time Pricing for Demand Side Management

4.1 Introduction

With the growing deployment of AMI [27] and ECC devices [11, 28, 29, 50, 55, 89], RTP is gradually becoming a feasible DSM solution. In general, it is difficult for power users to follow the RTP price variations to make appropriate decisions accordingly. In this regard, ECC devices can help by making such price-responsive decisions on behalf of users to achieve certain objectives. However, while the use of ECC devices improves users' rationality in response to price changes, such ECC devices can also introduce new DSM challenges such as load synchronization [12] and price instability [43, 90]. Therefore, the effect of the automated ECC devices on users' price-responsiveness is not obvious and yet to be investigated.

In this chapter, we address minimizing the PAR in the aggregate load demand through pricing under the practical scenario that the utility is uncertain about users' price-responsiveness. While we assume that users are equipped with ECC devices, our approach is quite different from all prior works, e.g., in [11, 12, 28–30], which do not take into account the uncertainty in users' price-responsiveness. Note that such uncertainty is inevitable to preserve user privacy [91].

Some related literature can be summarized as follows. In [28], Chen *et al.* devised a

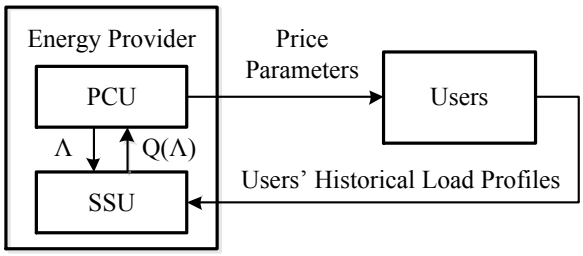

Figure 4.1: The block diagram of the proposed closed-loop pricing model.

Stackelberg game approach in which the energy provider acts as a leader and users are followers. This design intends to jointly maximize the social welfare of all users and the revenue of the energy provider. The algorithm in [28] requires detailed information about users' energy consumption needs. However, for the scheme proposed in this chapter, users are not required to submit their demand requirements at the beginning of the operation period. As a result, our design is more practical and preserves users' privacy. The authors of [11] proposed a game theoretic approach to minimize the PAR of the system, where users interact with each other and change their power consumption accordingly. However, such interactions may take a long time to converge, in particular in the presence of a large number of users. In contrast, our design does not involve direct user interaction. Therefore, it converges much faster.

The block diagram of our proposed real-time pricing model is shown in Fig. 4.1. Our main contributions are as follows:

- We propose two iterative algorithms to be implemented in a *price control unit* (PCU) for minimizing the PAR of the aggregate load based on the information provided by the *system simulator unit* (SSU). The first algorithm, called finite-difference price selection (FDPS), uses a variation of the finite-difference technique [52] to approximate the gradient of the PAR minimization objective function by making small one-at-a-time changes to each individual element of the input price parameter vector. The

FDPS algorithm requires only few iterations for convergence. However, it needs a large number of measurements of the objective function in each iteration.

- The second algorithm, called simultaneous perturbation price selection (SPPS), is based on the simultaneous perturbation technique [52]. Unlike the FDPS algorithm, all elements of the input variable are jointly and randomly perturbed to approximate the gradient. As a result, the SPPS algorithm significantly reduces the number of measurements of the objective function in each iteration, compared to the FDPS algorithm. Yet, we show that it achieves a similar performance.

- We propose an approximate dynamic programming scheme which simulates how users automatically respond to various price values to eliminate the need for users to reveal their detailed energy consumption needs to the energy provider. This assures user privacy.

- Simulation results show that our proposed pricing algorithms reduce the PAR in aggregate load. In addition, we show that adopting the new pricing algorithms is also beneficial for the users if they are equipped with automated control units.

The rest of this chapter is organized as follows. The system model and problem formulation are introduced in Section 4.2. The FDPS and SPPS algorithms are developed in Section 4.3. The SSU is explained in Section 4.4. Simulation results are presented in Section 4.5. The chapter is summarized in Section 4.6.

4.2 Problem Formulation

Let $\mathcal{U}$ denote the set of all users. Let $\mathcal{A}_u$ denote the set of all appliances of user $u \in \mathcal{U}$. We denote $\mathcal{M}_u$ as the set of must-run appliances of user u, $\mathcal{C}_u$ as the set of controllable

appliances of user u, and $\mathcal{N}_u$ as the non-interruptible subset of $\mathcal{C}_u$. For each user, we assume that there is an ECC unit which is embedded in the user's smart meter and controls the user's power consumption [24, 25]. The users' responses to the price changes are done automatically using the ECC units. All ECC units are connected to the energy provider through a two-way communication infrastructure. The operation period is divided into T time slots. We define binary variable $x_{u,a}^t \in \{0,1\}$ as the state of power consumption of appliance $a \in \mathcal{A}_u$ at time slot $t \in \mathcal{T}$, where $\mathcal{T} \triangleq \{1,\ldots,T\}$. We set $x_{u,a}^t = 1$ if appliance a operates in time slot t; otherwise, we have $x_{u,a}^t = 0$. For each user u, $E_{u,a}$ is the total energy requirement of appliance $a \in \mathcal{A}_u$, $\gamma_{u,a}$ is the nominal power consumption of appliance a, and $T_{u,a} = E_{u,a}/\gamma_{u,a}$.

4.2.1 Centralized Load Control Algorithm

Assuming that the energy provider is aware of all users' energy needs and is capable of remotely controlling the ECC devices of all users, the centralized load control problem to minimize the PAR in aggregate load can be formulated as

$$\underset{\mathbf{x}_{u,a} \in \tilde{\mathcal{X}}_u,\ \forall\, a \in \mathcal{A}_u,\ \forall\, u \in \mathcal{U}}{\text{minimize}} \quad \frac{T \max\{L_1,\ldots,L_T\}}{\sum_{t=1}^{T} L_t}, \tag{4.1}$$

where

$$\mathbf{x}_{u,a} \triangleq (x_{u,a}^1,\ldots,x_{u,a}^T), \tag{4.2}$$

$$L_t = \sum_{u\in\mathcal{U}} \sum_{a\in\mathcal{A}_u} \gamma_{u,a} x_{u,a}^t, \tag{4.3}$$

and the feasible set $\tilde{\mathcal{X}}_u$ is defined as

$$\tilde{\mathcal{X}}_u=\Bigg\{\mathbf{x}_{u,a} \;\Bigg|\; x^t_{u,a}\in\{0,1\},\ \forall a \in \mathcal{A}_u,\ \forall\, t \in \mathcal{T},$$
$$\gamma_{u,a} \sum_{t=\alpha_{u,a}}^{\beta_{u,a}} x^t_{u,a} = E_{u,a},\ \forall\, a \in \mathcal{C}_u,$$
$$\gamma_{u,a} \sum_{t=\alpha_{u,a}}^{\alpha_{u,a}+T_{u,a}-1} x^t_{u,a} = E_{u,a},\ \forall\, a \in \mathcal{M}_u,$$
$$x^t_{u,a} = 1,\ \forall\, a\in\mathcal{N}_u, \forall\, t \in\mathcal{T}_{u,a}, 0 < E^t_{u,a}<E_{u,a}\Bigg\}. \tag{4.4}$$

Here, $\alpha_{u,a}$ is the earliest time at which the operation of appliance a could be scheduled, $\beta_{u,a}$ is the deadline by which the operation of appliance a should be finished, $\mathcal{T}_{u,a} = \{\alpha_{u,a}, \ldots, \beta_{u,a}\}$, and $E^t_{u,a}$ is the amount of energy required to finish the operation of appliance $a \in \mathcal{A}_u$ while the system is at time slot t and is calculated as

$$E^t_{u,a} = \left[E_{u,a} - \gamma_{u,a} \sum_{k=1}^{t-1} x^k_{u,a}\right]^+. \tag{4.5}$$

The first constraint in (4.4) indicates that each appliance can be either *on* or *off*. The second constraint implies that the operation of each appliance should be scheduled within its feasible interval. The third constraint indicates that the operation of must-run appliances should be started immediately. The last constraint guarantees that the operation of non-interruptible appliances will continue, once they become active.

The ECC unit does not change the total load of the users, $\sum_{t=1}^{T} L_t$, and the denominator in (4.1) is a constant. We introduce an auxiliary variable Γ and rewrite problem (4.1) as

$$\underset{\substack{\Gamma,\, \mathbf{x}_{u,a} \in \tilde{\mathcal{X}}_u,\\ \forall\, a \in \mathcal{A}_u, \forall\, u \in \mathcal{U}}}{\text{minimize}} \quad \Gamma$$

$$\text{subject to} \quad \sum_{u\in\mathcal{U}}\sum_{a\in\mathcal{A}_u} \gamma_{u,a}x^t_{u,a} \le \Gamma, \quad \forall\, t \in \mathcal{T}. \tag{4.6}$$

Problem (4.6) is a linear mixed-integer program and can be solved using software such as MOSEK [84]. Its solution provides a performance benchmark for any load control algorithm that minimizes the PAR of the aggregate load while satisfying the demand requirements of all users.

4.2.2 Decentralized Price-Based Load Control Algorithm

In this section, we assume that the energy provider has no control over users' behavior and it may only influence the load by changing the price parameters. Recall from Section 4.1 that RTP and IBR are two non-flat pricing models that are used to encourage consumers to shift some of their load from peak hours to off-peak hours and also to prevent load synchronization.

Let $L^t_u \triangleq \sum_{a\in\mathcal{A}_u} \gamma_{u,a}x^t_{u,a}$ denote the total power consumption of user u at time slot t. Let $\lambda_t(L^t_u)$ denote the selected price of electricity in time slot t as a function of the user's power consumption in that time slot. By combining RTP and IBR [31, 32], the price function $\lambda_t(L^t_u)$ is defined as [12]:

$$\lambda_t(L^t_u) = \begin{cases} m_t, & \text{if } \ 0 \le L^t_u \le b_t, \\ n_t, & \text{if } \ L^t_u > b_t, \end{cases} \tag{4.7}$$

where m_t, n_t, and b_t are price parameters, and $m_t \le n_t$. Also, let $\Lambda_t \triangleq (m_t, n_t, b_t)$ and $\Lambda \triangleq (\Lambda_1, \ldots, \Lambda_T)$. The general pricing function in (7) represents an RTP structure that is combined with IBR. Based on this combined model, the price of electricity depends on the time of day and also the total load. We assume that the price parameters for each time slot are selected such that the PAR of the aggregate load in the system is minimized.

Thus, the best choice for the price parameters in each time slot is that obtained by solving the following optimization problem:

$$\begin{aligned} \underset{\Lambda}{\text{minimize}} \quad & Q(\Lambda) \\ \text{subject to} \quad & m_t^{min} \leq m_t \leq m_t^{max}, \; \forall \, t \in \mathcal{T}, \\ & n_t^{min} \leq n_t \leq n_t^{max}, \; \forall \, t \in \mathcal{T}, \\ & b_t^{min} \leq b_t \leq b_t^{max}, \; \forall \, t \in \mathcal{T}, \\ & m_t \leq n_t, \; \forall \, t \in \mathcal{T}, \end{aligned} \tag{4.8}$$

where

$$Q(\Lambda) = \max\{L_1(\Lambda), \ldots, L_T(\Lambda)\}, \tag{4.9}$$

$L_t(\Lambda)$ denotes the aggregate load at time slot t that depends on the price parameters. m_t^{min}, m_t^{max}, n_t^{min}, n_t^{max}, b_t^{min}, and b_t^{max} are lower and upper bounds for the price parameters m_t, n_t, and b_t, respectively. To devise an efficient pricing algorithm capable of minimizing the PAR, the energy provider needs to know the behavior of the users in response to the selected price parameters. With the deployment of the ECC devices, it is very challenging to anticipate the response of the users to different price parameters. Therefore, we *cannot* have an explicit expression for $Q(\Lambda)$ and consequently it is *not* possible to obtain a closed-form analytical solution for (4.8).

Due to the challenges explained above, we propose to solve problem (4.8) using an iterative algorithm that does not require a closed-form expression for $Q(\Lambda)$. That is, we follow a step-by-step procedure that moves from an initial guess to a final value which is close to the optimum solution of problem (4.8). There exist different methods to solve problem (4.8) iteratively. In this regard, we propose to equip the energy provider with an SSU, as shown in Fig. 4.1, that simulates the likely behavior of the users in response

to price parameters announced by the energy provider. The information produced by the SSU will then be used by the PCU to select prices.

4.3 Price Control Unit (PCU)

Recall from Section 4.2.2 that finding a closed-form solution for problem (4.8) is challenging. An alternative is an iterative algorithm using a gradient method. In this regard, we need to *approximate* the gradient from noisy measurements of $Q(\Lambda)$. Next, we propose two different methods for this purpose.

4.3.1 Finite-Difference Price Selection (FDPS)

Using the finite-difference technique [52, Ch. 6], the gradient of the objective function can be approximated by making small one-at-a-time changes to each of the individual elements of Λ. That is, the jth element of vector Λ is perturbed and the changes in the objective function are measured. The ratio of the changes in the objective function to the amount of the perturbation of the jth element of vector Λ approximates the jth element of the gradient vector of objective function $Q(\Lambda)$. The general recursive procedure of updating the price parameters in each time slot can be written as

$$\Lambda^{i+1} = \Lambda^i - \sigma^i \hat{g}^i(\Lambda^i), \tag{4.10}$$

where $p \times 1$ column vector $\hat{g}^i(\Lambda^i)$ is an estimate of the gradient of $Q(\Lambda)$, $\nabla Q(\Lambda)$, at iteration i based on the measurements of $Q(\Lambda)$[3], Λ^i is the input vector Λ at iteration i, and $p = 3T$ is the size of vector Λ^i. The step size $\sigma^i > 0$ is reduced as the number of

[3]For non-differentiable functions, to update the price parameters in (4.10), the subgradient of the objective function can be used instead of the gradient.

iterations increases to assure convergence. In our proposed FDPS algorithm, we use one-sided gradient approximations which involve evaluations of the form $Q(\Lambda^i + \text{perturbation})$ and $Q(\Lambda^i)$. That is, we obtain the gradient estimate as

$$\hat{g}^i(\Lambda^i) = \begin{bmatrix} \frac{Q(\Lambda^i + c^i\zeta_1) - Q(\Lambda^i)}{c^i} \\ \vdots \\ \frac{Q(\Lambda^i + c^i\zeta_p) - Q(\Lambda^i)}{c^i} \end{bmatrix}, \tag{4.11}$$

where ζ_j denotes a $p \times 1$ vector with a 1 in the jth position and zeros elsewhere, and $c^i > 0$ is the magnitude of the perturbations. Among different methods proposed for selecting coefficients σ^i and c^i, some specific forms have been suggested in practice which also satisfy the conditions required for convergence of the algorithm [52, Ch. 6]:

$$\sigma^i = \frac{\sigma}{(i+1+A)^\alpha}, \quad c^i = \frac{c}{(i+1)^\gamma}, \tag{4.12}$$

where σ, α, c, and γ are strictly positive constants, and $A \geq 0$ is added to improve the convergence of the algorithm.

4.3.2 Simultaneous Perturbation Price Selection (SPPS)

Next, we consider another method for approximating the gradient of the objective function $Q(\Lambda)$ which is known as *simultaneous perturbation stochastic approximation* [52, Ch. 7]. Similar to the FDPS algorithm, the SPPS algorithm updates the price parameters as in (4.10). However, unlike FDPS, the SPPS algorithm randomly and jointly perturbs *all* elements of Λ^i in order to obtain two different perturbed measurements of $Q(\cdot)$. Thus, the

two-sided simultaneous perturbation gradient approximation is given by

$$\hat{g}^i(\Lambda^i) = \begin{bmatrix} \frac{Q(\Lambda^i+c^i\Delta^i)-Q(\Lambda^i-c^i\Delta^i)}{2c^i\Delta_1^i} \\ \vdots \\ \frac{Q(\Lambda^i+c^i\Delta^i)-Q(\Lambda^i-c^i\Delta^i)}{2c^i\Delta_p^i} \end{bmatrix} = \frac{Q(\Lambda^i+c^i\Delta^i)-Q(\Lambda^i-c^i\Delta^i)}{2c^i}\left(\frac{1}{\Delta_1^i},\ldots,\frac{1}{\Delta_p^i}\right), \tag{4.13}$$

where $\Delta^i \triangleq \left(\Delta_1^i,\ldots,\Delta_p^i\right)$ is the perturbation vector, and $\Delta_j^i \in \{-1,1\}$ is a random number. We note that, for the SPPS algorithm, the number of measurements in each iteration is two, independent of the size parameter p. Thus, compared to FDPS, the SPPS algorithm provides large savings in the number of measurements in each iteration, especially if p is large. This lower per-iteration complexity is beneficial as long as the number of iterations required to converge to an optimal value of $\Lambda^\star$ does not increase significantly.

4.3.3 Algorithm Description

In this section, we explain the steps of the proposed FDPS and SPPS algorithms (Algorithm 4.1) executed at the PCU. At the beginning of the algorithm, the initial values for parameters σ, α, c, γ, A, and Λ^0 are selected, c.f. Line 1. At the ith iteration of the algorithm, the coefficients σ^i and c^i are updated as in (4.12), c.f. Line 3. For the FDPS algorithm, the gradient is approximated as in (4.11), c.f. Line 5. For the SPPS algorithm, the gradient is approximated as in (4.13), c.f. Line 7. Λ^i is updated as in (4.10), c.f. Line 9. The algorithm is stopped if the maximum number of allowed iterations is reached or the difference between two subsequent values of the objective function is less than a pre-determined threshold, c.f. Line 10.

Algorithm 4.1: Price selection algorithm executed at the PCU.

1: Select initial value for σ, α, c, γ, A, and Λ^0.
2: **repeat**
3: Update σ^i and c^i as in (4.12).
4: **if** (FDPS) **then**
5: Calculate $\hat{g}^i(\Lambda^i)$ as in (4.11).
6: **elseif** (SPPS)
7: Calculate $\hat{g}^i(\Lambda^i)$ as in (4.13).
8: **end if**
9: Update Λ^i as in (4.10).
10: **until** the stopping criteria

4.3.4 Convergence of the Algorithms

We now present the conditions for convergence of Algorithm 4.1. Convergence of different stochastic approximation based algorithms has been analyzed under various conditions. In particular, algorithms based on *simultaneous perturbation stochastic approximation* (SPSA) have attracted more attention, as they require fewer objective function evaluations. Spall showed the convergence of SPSA under a three times differentiability condition for the objective function [52]. However, it was shown later that weaker assumptions suffice for SPSA to converge [92–94]. The iterative updating step in (4.10) can be written as

$$\Lambda^{i+1} = \Lambda^i - \sigma^i\big(\tilde{g}^i(\Lambda^i) + \epsilon^i\big), \tag{4.14}$$

where $\tilde{g}^i(\Lambda^i)$ is any subgradient of the objective function at the ith iteration, and ϵ^i represents the observation noise and bias term. For differentiable functions, the subgradient $\tilde{g}^i(\Lambda^i)$ is identical to the gradient of the objective function. For non-differentiable functions, the sub-gradient of the objective function $Q(\cdot)$ at Λ^i is defined as any vector g that satisfies the inequality $Q(\Phi) \geq Q(\Lambda^i) + g^{\mathrm{T}}(\Phi - \Lambda^i)$ for all Φ. Following the discussion in [93, 94], the conditions for the iterative convergence of Λ^i to the optimum value Λ^* that minimizes the objective function are summarized as

A4.1 The domain of $Q(\cdot)$ is convex and closed. $Q(\cdot)$ is convex, and the expected value $\mathbb{E}[Q(\Lambda^i)]$ is uniformly bounded, where $\mathbb{E}\{\cdot\}$ denotes mathematical expectation.

A4.2 For the step-size parameters we must have: a) $\sigma^i > 0$, b) $c^i > 0$, c) $\sigma^i \to 0$, d) $c^i \to 0$, e) $\sum_{i=0}^{\infty} \sigma^i = \infty$, and f) $\sum_{i=0}^{\infty} (\sigma^i / c^i)^2 < \infty$.

A4.3 Let $\mathcal{I}^i \triangleq (\Lambda^0, \ldots, \Lambda^i)$. $\sum_{i=0}^{\infty} (\sigma^i)^2 \mathbb{E}\big[\|\epsilon^i\|^2 | \mathcal{I}^i\big] < \infty$.

A4.4 The subgradient $\tilde{g}^i$ is uniformly bounded.

A4.5 Δ_j^i must be independent for all i and j, identically distributed for all j at each iteration i, symmetrically distributed about zero, and uniformly bounded in magnitude for all i and j.

Condition A4.1 specifies the criteria required for the convergence of the algorithm to the global optimum. Condition A4.2 determines the rate at which the gain σ^i has to decay. The gain σ^i should decay neither too fast nor too slow. It has to approach zero fast enough to damp the effects of the noise as the algorithm gets closer to the solution Λ^*. However, it has to approach zero at a sufficiently slow rate to ensure full convergence of the algorithm. Condition A4.3 ensures that the algorithm is able to cope with the noise. In practice, for large numbers of users, the effect of each individual user on the aggregate load of the system is small and the variations in the demand requirements of different users help in making the load curve smooth which also reduces the effects of the noise term. Conditions A4.4 and A4.5 ensure that the algorithm is asymptotically an unbiased estimator of the optimum value Λ^* [94]. Condition A4.5 determines the randomization property of the perturbation vector such that the objective function can be effectively approximated by a smooth function at the points of non-differentiability [94].

Together, conditions A4.1-A4.5 specify the *ideal* requirements for the convergence of the algorithm. However, in practice, due to the lack of knowledge of the structure of $Q(\cdot)$, it is

very difficult or even impossible to check these conditions. To resolve this issue, *gradient-free* techniques are adopted to optimize the objective function in this chapter. This also reveals the difficulty of verifying the above mentioned conditions. However, despite the fact that some conditions may not be verifiable, it has been shown that the adopted techniques are among the most effective methods to optimize objective functions with an unknown formulation in practice [52]. Different methods have been proposed in the literature to ensure that the stochastic approximation methods converge to the global optimum among multiple local optima. One of the well-known approaches is to inject an additive noise in the right-hand side of the basic updating step in (4.10) [52, Ch. 6].

4.4 System Simulator Unit (SSU)

In this section, we explain the algorithm to be implemented in the SSU. This requires an understanding of how the ECC device may operate for each user. We assume that the operation of ECC devices in each time slot begins with an admission control phase, where appliances send admission requests to the ECC unit. Once an admission request is submitted, the state of the appliance changes from *sleep* to *awake*. The ECC unit schedules the operation of awake appliances such that the electricity expenses of the user are minimized.

To simulate the users' load patterns, the SSU simulates the time at which each appliance becomes awake and also the time by which the operation of each appliance has to be finished. Such information can be obtained based on the sleep and awake history of each appliance. To preserve the users' privacy, we assume that the actual data is manipulated such that the statistical information is preserved, but it is not possible to extract the exact information about the demand requirements of individual users [95, 96]. Various privacy aware smart metering techniques have been proposed in the literature, such as secure meter

data aggregation [97], and privacy aware home energy management system [98]. By using the manipulated data, the SSU simulates the likely control decisions of the ECC unit of each user based on the price indicated by the PCU. We note that the SSU simulates the likely behavior of *general users*, and each general user does not refer to any particular user.

In the following, we first explain the control algorithm running in the ECC device of each user and then how the SSU simulates the control decisions of the ECC devices.

4.4.1 Power Scheduling Done by ECC Devices

For each user u, the power scheduling is done by the ECC device at the current time slot t by solving the following optimization problem that is specific to user u and aims to minimize the *expected* energy cost in the upcoming time slots:

$$V_u^t(\mathbb{S}_u^t) = \underset{\substack{\mathbf{x}_{u,a}^t \in \mathcal{X}_u^t, \\ \forall\, a \in \mathcal{C}_u^k, \\ \forall\, k \in \mathcal{T}^t}}{\text{minimize}}\; g_t\big(\mathbb{S}_u^t, L_u^t\big) + \mathbb{E}\Big\{ V_u^{t+1}(\mathbb{S}_u^{t+1}) \mid \mathbb{S}_u^t \Big\}, \tag{4.15}$$

where $\mathcal{T}^t \triangleq \{t, \ldots, T\}$, $\mathbf{x}_{u,a}^t \triangleq (x_{u,a}^t \ldots, x_{u,a}^T)$, and we have

$$g_t\big(\mathbb{S}_u^t, L_u^t\big) \triangleq L_u^t \lambda_t\big(L_u^t\big), \tag{4.16}$$

$$L_u^t = \sum_{a \in \mathcal{M}_u^t} \gamma_{u,a} + \sum_{a \in \mathcal{C}_u^t} \gamma_{u,a} x_{u,a}^t. \tag{4.17}$$

We refer to $V_u^t(\cdot)$ as the value function of user u at time slot t, and $V_u^{T+1}(\cdot) \triangleq 0$. For each user u, we also define the *state* of the system at time slot t as $\mathbb{S}_u^t \triangleq (\mathbf{E}_u^t, \mathbf{I}_u^t)$, where $\mathbf{I}_u^t \triangleq (\mathcal{M}_u^t, \mathcal{C}_u^t)$ and $\mathbf{E}_u^t \triangleq (E_{u,1}^t, \ldots, E_{u,|\mathcal{A}_u|}^t)$. Here, $\mathcal{M}_u^t$ and $\mathcal{C}_u^t$ are the sets of must-run and controllable appliances of user u that are awake at time slot t, respectively. The feasible

set $\mathcal{X}_u^t$ in problem (4.15) is defined as

$$\mathcal{X}_u^t = \Bigg\{ \mathbf{x}_{u,a}^t \;\Bigg|\; x_{u,a}^k \in \{0,1\},\ \forall\, a \in \mathcal{C}_u^k,\ \forall\, k \in \mathcal{T}^t,$$
$$\gamma_{u,a} \sum_{m=k}^{\beta_{u,a}} x_{u,a}^m = E_{u,a}^k,\ \forall\, a \in \mathcal{C}_u^k,\ \forall\, k \in \mathcal{T}^t$$
$$x_{u,a}^k = 1,\ \forall\, a \in \mathcal{N}_u^k, \forall\, k \in \mathcal{T}_{u,a}^t, 0 < E_{u,a}^k < E_{u,a} \Bigg\}, \tag{4.18}$$

where $\mathcal{T}_{u,a}^t \triangleq \{t, \ldots, \beta_{u,a}\}$, and $\mathcal{N}_u^k$ denotes the non-interruptible subset of $\mathcal{C}_u^k$. The first term in the objective function in (4.15) is the payment of the user in the current time slot t for the *known* load L_u^t, while the second term is the expected cost of energy in the upcoming time slots, which we will refer to as the *cost-to-go*. The feasible set in (4.18) is similar to (4.4). However, it is based on the updated information which is available up to time slot t. An algorithm based on linear mixed-integer programming has been proposed in [51] to solve problem (4.15). However, its complexity makes it difficult to be used in the SSU.

4.4.2 Simulation of ECC Operation at SSU

In order to mimic the operation of the ECC devices, the energy provider needs to similarly solve optimization problem (4.15). However, this cannot be done because the energy provider does not have access to the details regarding the users' energy needs. To tackle this problem, we propose an approximate dynamic programming algorithm to estimate the solution of problem (4.15). First, we note that the state of user u in the next time slot, $\mathbb{S}_u^{t+1}$, depends on the current state $\mathbb{S}_u^t$, the decision which is made at the current time slot $\mathbf{x}_{u,a}^t$, and the exogenous information which arrives at the beginning of the next time slot

$\mathbf{I}_u^{t+1}$. We define

$$\mathbb{S}_{u,x}^t = \mathrm{S_x}(\mathbb{S}_u^t, \mathbf{x}_{u,a}^t), \tag{4.19}$$

$$\mathbb{S}_u^{t+1} = \mathrm{S_I}(\mathbb{S}_{u,x}^t, \mathbf{I}_u^{t+1}), \tag{4.20}$$

where $\mathbb{S}_{u,x}^t$ is the state of the system immediately after we make a decision and is referred to as *post-decision* state [82], $\mathrm{S_x}(\cdot)$ is the state transition function which takes into account the effect of decisions, and $\mathrm{S_I}(\cdot)$ is the state transition function which takes into account the effect of arrival information.

A well-known approach to approximate the cost-to-go is to represent it based on the post-decision state $\mathbb{S}_{u,x}^t$ [82]. Problem (4.15) can now be written as

$$\hat{V}_u^t(\mathbb{S}_u^t) = \underset{\substack{\mathbf{x}_{u,a}^t \in \mathcal{X}_u^t, \\ \forall\, a \in \mathcal{C}_u^t}}{\text{minimize}}\ g_t\big(\mathbb{S}_u^t, L_u^t\big) + \hat{V}_{u,x}^{t+1}(\mathbb{S}_{u,x}^t), \tag{4.21}$$

where $\hat{V}_u^t(\cdot)$ is the approximation of the cost of being in state $\mathbb{S}_u^t$, and $\hat{V}_{u,x}^{t+1}(\cdot)$ is the approximation of the cost-to-go by writing it as a function of post-decision state $\mathbb{S}_{u,x}^t$ rather than current state $\mathbb{S}_u^t$. Since $\mathbb{S}_{u,x}^t$ is a deterministic function of $\mathbf{x}_{u,a}^t$, problem (4.21) is a deterministic optimization problem. Among different techniques considered to approximate the cost-to-go, parametric models [82] are particularly popular, where the value function is replaced with a linear regression. Let $\phi_t(\cdot)$ be a basis function which captures some features of the underlying system at time slot t. We approximate the cost-to-go at the next time slot as

$$\hat{V}_{u,x}^{t+1}(\mathbb{S}_{u,x}^t) = \sum_{k=t+1}^{T} \theta_k \phi_k(\tilde{\mathbf{x}}_u^{t+1}), \tag{4.22}$$

where θ_k is the weight coefficient at time slot k, $\tilde{\mathbf{x}}_u^{t+1} = (\tilde{\mathbf{x}}_{u,1}^{t+1}, \ldots, \tilde{\mathbf{x}}_{u,|\mathcal{A}_u|}^{t+1})$, $\tilde{\mathbf{x}}_{u,a}^{t+1} = (\tilde{x}_{u,a}^{t+1}, \ldots, \tilde{x}_{u,a}^{T})$, and we have

$$\phi_k(\tilde{\mathbf{x}}_u^{t+1}) = g_k(\mathbb{S}_{u,x}^t, \tilde{L}_u^k), \tag{4.23}$$

$$\tilde{L}_u^k = \sum_{a \in \mathcal{M}_u^t} \gamma_{u,a} + \sum_{a \in \mathcal{C}_u^t} \gamma_{u,a} \tilde{x}_{u,a}^k. \tag{4.24}$$

Furthermore, we can calculate $\tilde{\mathbf{x}}_u^{t+1}$ as follows:

$$\tilde{\mathbf{x}}_u^{t+1} = \underset{\mathbf{x}_{u,a}^t \in \bar{\mathcal{X}}_u^t, \forall\, a \in \mathcal{C}_u^t}{\operatorname{argmin}} \sum_{k=t+1}^{T} \theta_k g_k\big(\mathbb{S}_{u,x}^t, l_u^k\big), \tag{4.25}$$

where $\bar{\mathcal{X}}_u^t$ is the feasible set defined by (4.18) while the state of the system is $\mathbb{S}_{u,x}^k$ and the first integer constraint in (4.18) is relaxed as $0 \leq x_{u,a}^k \leq 1$. l_u^k is defined as

$$l_u^k = \sum_{a \in \mathcal{M}_u^t} \gamma_{u,a} + \sum_{a \in \mathcal{C}_u^t} \gamma_{u,a} x_{u,a}^k. \tag{4.26}$$

The basis functions $\phi_k(\cdot)$ in (4.23) capture the estimate of the cost in future time slots based on the information which is available at the current time slot t. The cost-to-go then is approximated as a weighted sum of the estimated cost of all upcoming time slots. However, as the new observations about the true cost of each time slot are revealed, the weight coefficients $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_T)$ are updated accordingly, as we explain next.

4.4.3 Updating the Value Function Estimation

Assume that we have n different observations for the true value of being in different states (i.e., the observations from the real system at the end of the entire operation period) that can be written in vector form as $(V_u^m, \mathbb{S}_u^m)_{m=1}^n$. Let ϕ^m be the vector of basis functions evaluated at $\mathbb{S}_u^m$, and $\mathbf{\Phi}^n$ be a matrix with n rows, one corresponding to each observation,

and T columns, one for each feature. Let $\mathbf{V}_u^n$ be a column vector with elements V_u^m. By using *least square* batch linear regression [82], we can update vector $\boldsymbol{\theta}$ as

$$\boldsymbol{\theta} = \left(\left(\boldsymbol{\Phi}^n\right)^{\mathrm{T}}\boldsymbol{\Phi}^n\right)^{-1}\left(\boldsymbol{\Phi}^n\right)^{\mathrm{T}}\mathbf{V}_u^n, \tag{4.27}$$

where, T is the transpose operator. We note that at the end of the operation period, we have multiple observations for different states of the system. The estimate of the value function's parameters can be improved if the observations of multiple operating periods are used to update the $\boldsymbol{\theta}$. Moreover, the estimate of the value function's parameters can be further improved if users are able to communicate and share their observations to have more samples to update the parameters of the value function. In practice, it may not be possible to obtain the true observation of the cost-to-go from the real system because of privacy issues. To tackle this problem, the results produced by the SSU can be used to update the value function's parameters. We note that in a real system, users are making control decisions based on the partial information available at the beginning of each time slot. That is, the complete demand requirements in the future time slots are not known. The SSU simulates the behavior of each user for different scenarios. For each scenario, to better mimic the behavior of each user, the control decisions are similarly made based on partial information available at the beginning of each time slot. Thus, similar to the real system, at each time slot, the exact cost-to-go is not known and only some estimation of it is available. However, at the end of each scenario, the exact value of cost-to-go can be observed. These observations can be used instead of true observation from the real system to update the value function's parameters.

Algorithm 4.2: The algorithm executed at the SSU.

1: Initialize n.
2: Receive price parameters Λ.
3: Select initial value for vector $\boldsymbol{\theta}$.
4: **for** $u \in \mathcal{U}$
5: **for** $t \in \mathcal{T}$
6: Determine appliances that become awake and their demand requirements.
7: Receive new information $\mathbf{I}_u^t$.
8: Determine $x_{u,a}^t$ as the solution of (4.21).
9: Update $(E_{u,1}^t, \ldots, E_{u,|\mathcal{A}_u|}^t)$ as in (4.5).
10: **end for**
11: Update $\boldsymbol{\theta}$ as in (4.27).
12: **end for**
13: Determine the aggregate load of the system.

4.4.4 Algorithm Description

We now explain the steps of the proposed control algorithm (Algorithm 4.2) to be executed in the SSU. At the beginning, the value of n is initialized and the price parameters Λ are received from the PCU, c.f. Lines 1 and 2. Subsequently, the initial value for vector $\boldsymbol{\theta}$ is selected randomly, c.f. Line 3. For each user u and at the beginning of each time slot t, the appliances that become awake are determined. The SSU also determines the demand requirements of each appliance. That is, whether the appliance is must-run or controllable and also the deadline by which the operation of the appliance should be finished are determined. The lists of awake appliances are then updated, and the operation schedule of the awake appliances for the current time slot t is calculated as the solution of problem (4.21), c.f. Lines 4 to 10. At the end of the operation period, we update vector $\boldsymbol{\theta}$ as in (4.27). The aggregate load of the system in each time slot t is determined as $\bar{L}_t = \sum_{u \in \mathcal{U}} \sum_{a \in \mathcal{A}_u} \gamma_{u,a} \bar{x}_{u,a}^t$, where $\bar{x}_{u,a}^t$ is determined at time slot t as the solution of (4.21), c.f. Line 13.

4.5 Performance Evaluation

In this section, we present simulation results and assess the performance of our proposed price control algorithm. Unless stated otherwise, the simulation setting is as follows. We assume that the general RTP method combined with IBR is adopted as described in (4.7). We consider a system with $|\mathcal{U}| = 50$ users. Each user possesses various must-run and controllable appliances. We assume that the exact information about the energy requirements of the users is not known by the SSU. However, we assume that some statistical information about the energy requirements of the users in form of distribution functions is available at the SSU. This statistical information includes the number of appliances, the nominal power consumption of each appliance, the probabilities with which each appliance becomes awake in each time slot, and the deadline by which the operation of each appliance should be finished. The statistical information can be obtained from the operational history of the real system. In the SSU, for a typical household user, we consider on average 18 appliances. Some of the appliances and their operating specifications are summarized in Table 4.1. The time slot at which each appliance becomes awake is selected randomly from a pre-determined interval. Based on the demand requirements of the user, each appliance can be set as must-run or controllable. This setting is decided by the user and can vary from time to time.

In our simulation setting, we consider various must-run and controllable appliances [51]. For example, we consider electric stove, clothes dryer, and vacuum cleaner as non-interruptible appliances. Refrigerator and air conditioner are modeled as interruptible appliances, and must-run appliances include: lighting, TV, etc. In general, the operation of some appliances can be correlated. However, taking such correlations into account for algorithm design would make the implementation of the SSU significantly more complex, which may not be desirable in practice. Therefore, we assume that the operations of

appliances are independent. For controllable appliances, the operating deadline is selected randomly from the remaining feasible time slots.

We note that the SSU does not observe the demand requirements of the users in the real system. Instead, it simulates the behavior of each user by running multiple scenarios. To better simulates the decisions made by the user, for each scenario, the information about the demand requirement of the user is updated gradually over time. That is, the SSU mimics the control decisions of the user based on the partial information available at the beginning of each time slot. For each user u and at the beginning of each time slot t, we determine the appliances that become awake and their operating specifications. The lists of awake appliances are then updated, and the operation schedule of the awake appliances for the current time slot t is calculated as the solution of problem (4.21). The aggregate load of the system in each time slot t is determined as

$$\bar{L}_t = \sum_{u \in \mathcal{U}} \sum_{a \in \mathcal{A}_u} \gamma_{u,a} \bar{x}^t_{u,a}, \tag{4.28}$$

where $\bar{x}^t_{u,a}$ is obtained at time slot t as the solution of (4.21). This procedure is repeated for multiple scenarios of the demand requirements of each user and the average results are considered.

By testing different practical examples, it has been shown in [52] that $\alpha = 0.602$ and $\gamma = 0.101$ are good choices for (4.12). To mitigate the effect of the measurement noise, we set c at a level approximately equal to the standard deviation of the measurement noise in $Q(\Lambda)$. We set A equal to 10 percent of the maximum number of allowed iterations. Coefficient σ in (4.12) plays an important role in the convergence of the algorithm as it has a significant effect on the step size in the different iterations. To select σ, first for each element j of Λ, we determine the appropriate value of σ_j that keeps the range of changes in the jth element of Λ in an appropriate range. Second, to assure stability, we

Table 4.1: Operating specifications of different appliances.

	$E_{u,a}$ (kWh)	$\gamma_{u,a}$ (kW)	arrival interval
Electric stove	4.5	1.5	[06:00, 14:00]
Clothes dryer	1	0.5	[14:00, 22:00]
Vacuum cleaner	2	1	[06:00, 15:00]
Refrigerator	2.5	0.125	[06:00, 09:00]
Air conditioner	4	1	[12:00, 22:00]
Dishwasher	2	1	[15:00, 24:00]
Heater	6	1.5	[15:00, 03:00]
Water heater	3	1.5	[06:00, 23:00]
Pool pump	4	2	[12:00, 21:00]
PEV	10	2.5	[16:00, 24:00]
Lighting	3	0.5	[16:00, 24:00]
TV	1	0.25	[16:00, 01:00]
PC	1.5	0.25	[08:00, 24:00]
Ironing appliance	2	1	[06:00, 16:00]
Hairdryer	1	1	[06:00, 13:00]
Other	6	1.5	[06:00, 24:00]

set $\sigma = \min\{\sigma_1, \ldots, \sigma_p\}$.

4.5.1 Performance Gains for the Utility Company

To have a baseline to compare with, we consider a system without ECC deployment, where each appliance a starts operation right after it becomes awake at its nominal power $\gamma_{u,a}$. For the system without ECC deployment, users are not responding to the variations of the price parameters. Furthermore, as an upper bound on the performance of the energy provider in minimizing the PAR of the aggregate load, we consider a system in which the energy provider knows all the demand requirements of the users and is capable of controlling the ECC units of all the users. The energy provider schedules the operation of all the appliances of the users such that all the demand requirements of the users are met. This system with direct load control achieves the minimum PAR of the aggregate load,

and since the energy provider has full control over the operation of the users' appliances, the performance of the system is independent of the price parameters. We note that the existence of a pricing scheme that can achieve this performance bound is not guaranteed. Since optimization problem (4.6) is too complex, we calculate a lower bound on the PAR of the system with direct load control. That is, we treat all controllable appliances as if they are interruptible, and instead of solving the mixed integer program, we present the results for the corresponding continuous problem. Simulation results for the average total power consumption of the proposed load control algorithms, the system without ECC deployment, and the lower bound on the PAR of the system with direct load control are depicted in Fig. 4.2. Simulation results for the average PAR of the aggregate load at different iterations of the proposed SPPS pricing algorithm, the proposed FDPS pricing algorithm, the system without ECC deployment, and a system with direct load control are depicted in Fig. 4.3. The simulation results show that the PAR of the aggregate load for the system without ECC deployment is on average 1.92. Our proposed SPPS algorithm reduces the PAR of the aggregate load to 1.58 (i.e., 18% reduction). Our proposed FDPS algorithm reduces the PAR of the aggregate load to 1.49 (i.e., 22% reduction). The lower bound on the achievable PAR of the system with direct load control is on average 1.2. Considering the number of time slots and the number of price parameters for each time slot, the number of measurements of the FDPS algorithm is 72 times higher than for the SPPS algorithm.

4.5.2 Performance Gain of the Users

For the SSU, we propose a load control algorithm (Algorithm 4.2) which simulates the operation of the ECC unit of each user. Since the SSU has to be fast enough and has to deal with the complex system of multiple users, the proposed load control algorithm

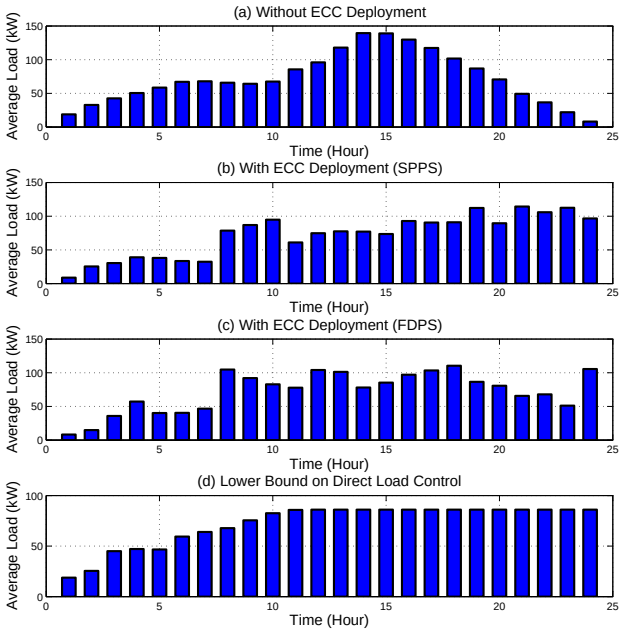

Figure 4.2: Aggregate load profile in different scenarios.

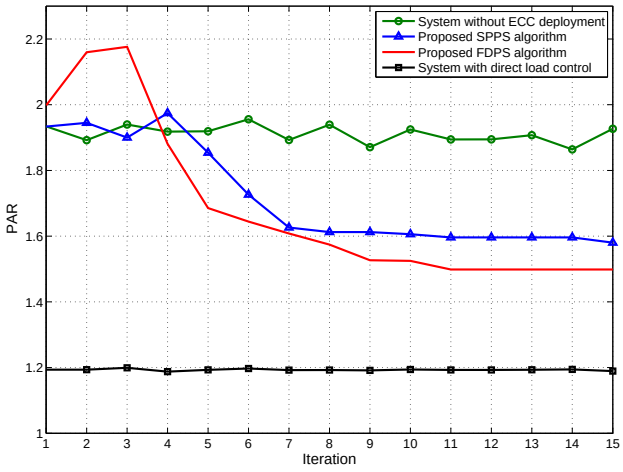

Figure 4.3: The PAR of the aggregate load in different scenarios.

is based on an approximate dynamic programming approach. The proposed load control algorithm can be adopted in the ECC units of the user. Therefore, in this section, we assess the performance of the proposed load control algorithm. To have a baseline to

compare with, we consider a system without ECC deployment, where each appliance a is assumed to start operation right after it becomes awake. As an upper bound, we also consider the scheme in [51] in which problem (4.21) is solved to schedule the operation of controllable appliances. Simulation results show that, to reduce electricity payment, the proposed control algorithm shifts the load to time slots with lower prices such as the few first hours after midnight. However, the high price penalty for exceeding the b_t threshold prevents load synchronization as discussed in Section 4.1. The simulation results show that the use of the proposed algorithm reduces the average daily payment of the user from \$4.85 to \$3.99. The average daily payment of the users for the load control algorithm in [51] is \$3.88. We can see that the efficiency loss in our proposed scheme compared to the one in [51] is small, although, our design has less computational complexity and is faster. The running times of the proposed FDPS and SPPS algorithms are directly influenced by the number of measurements of the objective function in each iteration and the running time of the SSU for each measurement. The SSU simulates the load pattern of each user to produce the aggregate load pattern of the users. This process can be done in parallel or sequentially. The running time of the SSU increases approximately linearly with the number of users if the load pattern of individual users is simulated sequentially. In the following, we evaluate the complexity of the load control algorithm (Algorithm 4.2) which simulates the load pattern of each user for different numbers of appliances. In general, integer linear programs with n integer variables and m constraints are NP-complete. That is, the order of complexity grows exponentially with respect to n and m [88]. A complete discussion of algorithm complexity is beyond the scope of this chapter. However, to provide a general idea about the complexity of our proposed algorithm compared to the one in [51], simulation results for the average running time and the average number of integer variables for both algorithms are presented in Table 4.2. The results were obtained by a computer

Table 4.2: Performance measures of different algorithms.
Average run time of the algorithm (in seconds).

	$\|\mathcal{A}\|$=20	$\|\mathcal{A}\|$=30	$\|\mathcal{A}\|$=40
Proposed algorithm for SSU	0.7324	0.7673	0.7919
Algorithm in [51]	2.1364	10.3071	69.5810

Average number of integer variables.

	$\|\mathcal{A}\|$=20	$\|\mathcal{A}\|$=30	$\|\mathcal{A}\|$=40
Proposed algorithm for SSU	4	6	10
Algorithm in [51]	57	90	129

system with Intel(R) Core(TM) i7 CPU 3.07 GHz processor, 12 GB RAM, and Windows 7 operating system. We note that the order of complexity is identified for the worst case scenario. However, in practice, by increasing the number of appliances, it is unlikely that all appliances become awake at the same time. Thus, the running time of the algorithm will not change significantly in each time slot for most practical scenarios.

4.6 Summary

In this chapter, we proposed two pricing algorithms based on stochastic approximation technique to minimize the PAR of the aggregate load. The proposed algorithms eliminate the need to know the structure of the objective function. In our proposed pricing algorithms, we took into account the way users will respond to different price values. We also considered the effect of control decisions of the ECC unit on the users' load profile. Moreover, we proposed the use of an SSU. A load control algorithm based on the approximate dynamic programming approach is also proposed and executed at the SSU to simulate the operation of the ECC unit at the demand side. The details of the demand requirements of the users at the appliance level are considered in the SSU. Simulation results showed that

our proposed algorithms reduce the PAR of the aggregate load. The proposed algorithms provide incentives for the users to reduce their energy expenses.

Chapter 5

Load Scheduling and Power Trading in Systems with High Penetration of Renewable Energy Resources

5.1 Introduction

RERs such as solar and wind are non-dispatchable, since they are random in nature. In systems with high penetration of RERs, the power may flow from DGs to the substation, which negatively impacts the stability of the system. If the reverse power flow exceeds a certain threshold, it causes the *voltage rise* problem, which is a major challenge in integrating a large number of DGs in the distribution network [35–37].

To tackle the reverse power flow problem, it is desirable that users consume their generating power locally rather than injecting the excess power back into the grid. Storage facilities and DSM techniques can be adopted to shape the load pattern of the users to better match supply and demand [3, 24, 25, 51, 55]. Moreover, users are able to sell their excess power to other users. On the one hand, local power trading can benefit the users by providing monetary revenue for them. On the other hand, it helps to absorb excess power and to mitigate the reverse power flow problem.

Different DSM programs have been designed to facilitate the integration of RERs into

the power grid, and the impact of different decision makers on the electricity market has been studied [44–49, 99–102]. Most of the existing work in the literature considers the case where users can sell their excess generation back into the grid [48, 49, 102]. Despite its importance, the possibility of trading energy among local users and the related benefits that users may obtain due to competition between local generators have not been well investigated.

In this chapter, we focus on modeling the interaction between users that can sell and buy locally generated electricity. In our model, users can offer their excess local generation capacity to other users with an appropriately selected price. For a given user, the selected price and the offered generation capacity depend on both the marginal cost of the user and the price offered by other users. Thus, users form a *game* in which they aim to maximize their own revenue. Due to the competition between multiple local generators, the consuming users may benefit from a lower price compared to the price advertised by the utility company. Our main contributions in this chapter are as follows:

- We consider a game theoretic approach to model the interaction of users with excess power generation. Users compete to sell their extra generation to other local users. The revenue of competing users depends on their own marginal cost and the offers advertised by other competing users. Thus, each competing user chooses its offered price and output generation such that its revenue is maximized.

- We formulate the problem of selecting the offered price and output generation (i.e., the trading problem) as a mixed-integer program. To tackle the complexity of the trading problem, we adopt the generalized Benders' decomposition approach.

- We propose an approximate dynamic programming approach to schedule the operation of must-run, interruptible controllable, and non-interruptible controllable appliances. The linearity of the proposed approximated scheduling problem makes it

possible to schedule the operation of different appliances independently. Independent scheduling of appliances significantly reduces the complexity of the scheduling algorithm and makes the real-time implementation of the algorithm possible.

- Simulation results show that our proposed algorithm reduces the energy payment of users compared to the case where trading is not applied. Our proposed scheme facilitates the integration of RERs and mitigates the reverse power flow problem by providing the users an opportunity to trade their excess generation locally.

The rest of this chapter is organized as follows. The system model is introduced in Section 5.2. In Section 5.3, the problem formulation and algorithm description are presented. Simulation results are provided in Section 5.4, and Section 5.5 concludes the chapter.

5.2 System Model

We consider a smart power system with a single utility company and several users. Each user is equipped with a renewable DG, such as photovoltaic cells or wind turbines. The demand requirements of the users are met by their local generation, imported power from other users, and imported power from the utility company. We assume that the structure of underlaying power grid is such that users have the option to export their excess power generation back to the utility company or to the other users. Examples of such grid topologies include micro-grids. The topology of these autonomous systems can be reconfigured by their own control infrastructure. Moreover, the adjustable configuration of these systems makes it possible for the users to export their excess power generation to the utility company or to the other users. We also assume that all users are equipped with advanced metering infrastructure such that their imported and exported power can be precisely measured. Let $\mathcal{U}$ denote the set of users. We assume that each user $u \in \mathcal{U}$ is

equipped with an energy consumption controller (ECC) capable of scheduling and adjusting the household energy consumption. The ECC units of all users are connected. We divide the intended operation cycle into $T \triangleq |\mathcal{T}|$ time slots, where $\mathcal{T} \triangleq \{1, \ldots, T\}$.

Let $\mathcal{A}_u$ denote the set of all appliances of user u. We assume that based on the demand requirements of the user, each appliance can be set as either must-run or controllable. This setting is decided by the user and can vary from time to time. The ECC has no control over the operation of must-run appliances. In contrast, the operation of controllable appliances can be *delayed* or *interrupted* if necessary. Each controllable appliance can be either interruptible or non-interruptible. For interruptible appliance a, the ECC may delay or interrupt its operation. However, for non-interruptible appliance a, it is only possible to delay its operation.

At the beginning of a time slot, each appliance of user u that is about to start operation sends an admission request to the ECC unit. The admission request specifies whether the appliance is must-run or controllable and its operational specifications, i.e., $T_{u,a} \triangleq [\alpha_{u,a}, \beta_{u,a}]$, where $\alpha_{u,a}$ is the earliest time at which appliance a can start operating (i.e., the time slot it becomes awake), and $\beta_{u,a}$ is the deadline by which the operation of appliance a has to be finished. The power consumption of each appliance can be different at different cycles of its operation due to changes in the amount of current being absorbed. We define vector $\mathbf{P}_a \triangleq (P_1^a, \ldots, P_{I_a}^a)$ as the power consumption pattern of appliance a if *no scheduling* is applied, where P_i^a is the power consumption of appliance a at its ith *operating cycle*, $I_a \triangleq |\mathcal{I}_a|$ is the number of operating cycles of appliance a, and $\mathcal{I}_a \triangleq \{1, \ldots, I_a\}$. We also assume that the duration of each operating cycle is one time slot. In practice, if the actual operating cycle lasts for more than one time slot, this can be modeled by introducing multiple consecutive operating cycles having identical values for the power consumption. An example for the pattern of power consumption of controllable appliance a before and

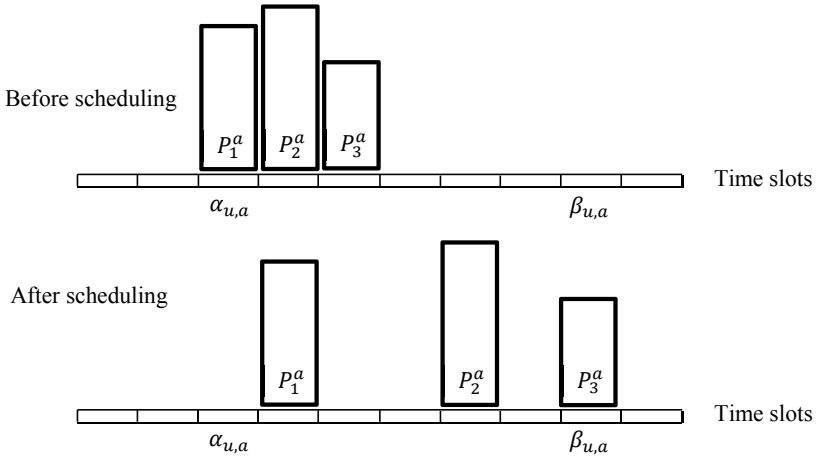

Figure 5.1: An example for the pattern of power consumption of interruptible controllable appliance a before and after scheduling.

after scheduling is illustrated in Fig. 5.1.

For each appliance a of user u, we define $\chi_t^{u,a} \triangleq (q_t^{u,a}, w_t^{u,a})$ as the state of appliance a at time slot t, where $q_t^{u,a}$ is the number of remaining operating cycles of appliance a, and $w_t^{u,a}$ is the number of time slots for which the operation of appliance a can be delayed. We define $x_t^{u,a} \in \{0, 1\}$ as an indicator which shows whether appliance a of user u is scheduled to operate at time slot t ($x_t^{u,a} = 1$) or not ($x_t^{u,a} = 0$). The state of appliance a at the next time slot $t + 1$, $\chi_{t+1}^{u,a}$, can be inferred from its current state $\chi_t^{u,a}$, its type, and the operating decision $x_t^{u,a}$. For a must-run appliance a, the initial state is $\chi_{\alpha_{u,a}}^{u,a} = (I_a, 0)$. A must-run appliance a starts operation immediately (i.e., $x_t^{u,a} = 1$, if $q_t^{u,a} \geq 1$) and its state $\chi_{t+1}^{u,a} = (q_{t+1}^{u,a}, 0)$ evolves as

$$\chi_{t+1}^{u,a} = \big(q_t^{u,a} - 1, 0\big), \qquad \text{for } q_t^{u,a} > 0. \tag{5.1}$$

For a non-interruptible appliance a, it is only possible to delay its operation. Once it has started operation, it is not possible to interrupt its operation. The initial state of a

non-interruptible appliance a is $\chi^{u,a}_{\alpha_{u,a}} = (I_a, \beta_{u,a} - \alpha_{u,a} - I_a + 1)$. The state evolves as

$$\chi^{u,a}_{t+1} = \begin{cases} \big(q^{u,a}_t, w^{u,a}_t - 1\big), & \text{if } \; x^{u,a}_t = 0, w^{u,a}_t \geq 1, \\ \big(q^{u,a}_t - 1, 0\big), & \text{if } \; x^{u,a}_t = 1, q^{u,a}_t \geq 1. \end{cases} \tag{5.2}$$

For an interruptible appliance a, it is not only possible to delay operation but also to interrupt operation if required. The initial state of an interruptible appliance a is $\chi^{u,a}_{\alpha_{u,a}} = (I_a, \beta_{u,a} - \alpha_{u,a} - I_a + 1)$, and it evolves as

$$\chi^{u,a}_{t+1} = \begin{cases} \big(q^{u,a}_t, w^{u,a}_t - 1\big), & \text{if } x^{u,a}_t = 0, w^{u,a}_t \geq 1, \\ \big(q^{u,a}_t - 1, w^{u,a}_t\big), & \text{if } x^{u,a}_t = 1, q^{u,a}_t \geq 1. \end{cases} \tag{5.3}$$

We also define $\chi^u_t = (\chi^{u,1}_t, \ldots, \chi^{u,|\mathcal{A}_u|}_t)$.

To better utilize the RERs and match demand and supply, we assume that each user is equipped with a storage device such as a battery. We define B_b as the storage capacity of the battery. We assume that the maximum charging and discharging rates of the battery are identical and denoted by e_b. For time slot t, we define variable $y^u_t \in [-e_b, e_b]$ as the charging / discharging rate of user u's battery. Moreover, Λ^u_t is the charging state of user u's battery at the beginning of time slot t. Thus, we have

$$\Lambda^u_t = \Lambda^u_0 + \sum_{k=1}^{t-1} y^u_k, \tag{5.4}$$

where Λ^u_0 is the initial charging state of the battery. At any time slot t, the stored energy cannot exceed the storage capacity B_b. Moreover, it is not possible to extract more energy from the storage unit than what is stored, i.e.,

$$0 \leq \Lambda^u_t \leq B_b. \tag{5.5}$$

Let g_t^u denote the total power exported by user u at time slot t. Users may utilize different technologies or different types of RERs to generate power. The intermittent nature of the RERs may cause problems regarding stability, voltage regulation, and power quality. In general, the quality of the generated power needs to be enhanced before it can be transmitted to other users. Different techniques have been proposed to improve the quality of power which results in additional costs for generators [103]. Therefore, the maintenance and operation cost is different for different users. We consider a cost function $\mathbb{C}_t^u(g_t^u)$ indicating the cost of providing g_t^u units of energy for user u at time slot t. We assume that the cost function is *increasing* and *strictly convex* in the offered energy. In this chapter, we consider a quadratic cost function:

$$\mathbb{C}_t^u(g_t^u) = a_t^u(g_t^u)^2 + b_t^u g_t^u + c_t^u, \tag{5.6}$$

where $a_t^u > 0$ and $b_t^u, c_t^u \geq 0$ are pre-determined parameters.

5.3 Problem Formulation and Algorithm Description

We consider the problems of *load scheduling* and *power trading*. Each user schedules the operation of its appliances to reduce its energy expenses. Primarily, each user uses its local generation to meet its own demand. If the local generation is not sufficient, then the user buys energy from its neighbors and the utility company. On the other hand, users with excess generation capacity compete with each other to sell their extra generations.

We define λ_t^h as the price value set by the utility company for each time slot t. We refer to users with excess generation as *competing* users. The advertised price of competing users should be less than λ_t^h to be economically reasonable for buyers. Competing users may have different marginal production cost. Thus, their advertised prices can be different.

Among the competing users, those with the smallest advertised price are selected to serve the demand. The *selected* competing users are referred to as *providing* users. Different approaches have been proposed in the literature to clear the market and pay the sellers. Providing users may sell their excess generation at their advertised price, or they may adopt a *market clearing price* (MCP) to serve the demand [104]. MCP is the highest advertised price among providing users. In our system model, we assume that the market is cleared by an MCP. Moreover, competing users who are not selected as providing users can still sell their extra generation back into the grid at a lower price $\lambda_t^l \leq \lambda_t^h$.

5.3.1 Scheduling Problem Formulation

We assume that the exact information about the list of appliances that are awake in each time slot, whether they are must-run or controllable, and the deadline by which their operation has to be finished is revealed only gradually over time to the ECC units. The operating schedule of controllable appliances depends on the price of electricity for the current time slot (i.e., the MCP). On the other hand, the MCP depends on the tentative load schedule of each user and whether the user has excess generation or not. In general, it is difficult to formulate and solve the joint optimization problem for determining the MCP and the operating schedule of the controllable appliances. To tackle this problem, we determine the operating schedule of the appliances and the MCP in two stages, i.e., the *scheduling* stage and the *trading* stage.

At the scheduling stage, we assume that the operating schedule of the controllable appliances is determined based on the *estimated* MCP for each time slot. Each user estimates the *average* MCP for each time slot based on its observations from past operating periods. This estimate can be different for different users as the effects of both selling and buying energy are taken into account. In this stage, users determine whether they have excess generation or have to acquire energy from neighboring users. However, at the

trading stage, competing users compete and the exact MCP for the current time slot t is determined. At the trading stage, competing users prefer to sell their excess generation to their neighbors rather than to the utility company, as they can make more profit by selling at a price higher than λ_t^l. Moreover, the consuming users may also benefit from local trading because the competing users may reduce their offered price compared to λ_t^h to obtain a larger share of the market. We define

$$L_t^u = \sum_{a \in \mathcal{A}_u} P^a_{I_a - q_t^{u,a}+1} x_t^{u,a} + y_t^u - \psi_t^u \tag{5.7}$$

as the load of user u at time slot t, where ψ_t^u is the amount of generated power of user u at time slot t. For user $u \in \mathcal{U}$ at time slot $t \in \mathcal{T}$, we define $v_t\big(\chi_t^u, L_t^u\big)$ as the payment of the user for the *known* load L_t^u:

$$\begin{aligned} v_t\big(\chi_t^u, L_t^u\big) &= \mu_t L_t^u j_t^u + \big(\mu_t f_t^u + \lambda_t^l (1 - f_t^u)\big) L_t^u (1 - j_t^u) \\ &= \Big(\mu_t j_t^u + \big(\mu_t f_t^u + \lambda_t^l (1 - f_t^u)\big)(1 - j_t^u)\Big) L_t^u, \end{aligned} \tag{5.8}$$

where μ_t is the MCP at time slot t, j_t^u is a binary variable specifying whether $L_t^u \geq 0$ ($j_t^u = 1$) or not ($j_t^u = 0$), and f_t^u denotes the portion of generated power which is processed and sold to neighboring users at price μ_t.

For each user u, the power scheduling is done by its ECC unit at current time slot t by solving the following optimization problem, which aims to minimize the *expected* energy cost in the upcoming time slots:

$$\mathbb{V}_t^u(\chi_t^u) = \underset{\substack{\mathbf{x}_t^u \in \mathbb{X}_t^u, \\ y_t^u \in \mathbb{Y}_t^u}}{\text{minimize}}\ v_t\big(\chi_t^u, L_t^u\big) + \mathbb{E}\Big\{\mathbb{V}_{t+1}^u(\chi_{t+1}^u) \mid \chi_t^u\Big\}, \tag{5.9}$$

where $\mathbb{E}\{\cdot\}$ denotes the expectation with respect to the demand and generation uncertainties in upcoming time slots, $\mathbf{x}_t^u \triangleq (x_t^{u,1} \ldots, x_t^{u,|\mathcal{A}_u|})$, $\mathbb{X}_t^u$ is the set of actions available for each appliance a at state χ_t^u, and $\mathbb{Y}_t^u$, which is the feasible operating set of the battery, is defined as

$$\mathbb{Y}_t^u = \Big\{ y_t^u \mid y_t^u \in [-e_b, e_b], 0 \leq \Lambda_t^u \leq B_b \Big\}. \tag{5.10}$$

$v_t\big(\chi_t^u, L_t^u\big)$ is as (5.8), while the second term on the right hand side of (5.9) is the expected cost of energy in the upcoming time slots, which we will refer to it as the *cost-to-go*. We refer to $\mathbb{V}_t^u(\cdot)$ as the value function of user u at time slot t, and $\mathbb{V}_{T+1}^u(\cdot) \triangleq 0$. Considering (5.8), we define

$$\theta_t^u(\chi_t) \triangleq \mu_t j_t^u + \big(\mu_t f_t^u + \lambda_t^l(1 - f_t^u)\big)(1 - j_t^u) \tag{5.11}$$

as the *effective* MCP for user u at time slot t. $\theta_t^u(\cdot)$ incorporates the effects of both electricity payment and electricity revenue for user u. $\theta_t^u(\cdot)$ depends on the scheduling and trading decisions of user u, the decisions of other users, and the realizations of random events (i.e., the power generation from RERs and the demand requirements of the users). Thus, it is very difficult to calculate $\theta_t^u(\cdot)$ directly. To tackle this problem, we propose an approximate dynamic programming approach to estimate the solution of problem (5.9). One approach to approximate the value function is to adopt parametric models [82]. The value function is replaced with a linear regression. Problem (5.9) can be approximated as

$$\hat{\mathbb{V}}_t^u(\chi_t^u) = \underset{\substack{\mathbf{x}_t^u \in \mathbb{X}_t^u, \\ y_t^u \in \mathbb{Y}_t^u}}{\text{minimize}} \ \eta_t^u L_t^u + \mathbb{E}\Big\{ \hat{\mathbb{V}}_{t+1}^u(\chi_{t+1}^u) \mid \chi_t^u \Big\}, \tag{5.12}$$

where η_t^u is the weight coefficient of user u at time slot t. A comparison of (5.9) and (5.12) reveals that the coefficient η_t^u approximates the function $\theta_t^u(\chi_t)$ in (5.8). However,

as new observations about the true value function for each time slot are revealed, the weight coefficients η_t^u are updated accordingly, as will be explained later in this section.

The linearity of (5.12) makes it possible to calculate the optimal values of $\mathbf{x}_t^{u,a}$ and y_t^u independently, and thus, to separate the scheduling process of individual appliances. That is, for each individual appliance a, we need to solve the following dynamic program:

$$\hat{\mathbb{V}}_t^{u,a}(\chi_t^{u,a}) = \underset{x_t^{u,a} \in \mathbb{X}_t^{u,a}}{\text{minimize}}\ \eta_t^u P^a_{I_a - q_t^{u,a}+1} x_t^{u,a} + \hat{\mathbb{V}}_{t+1}^{u,a}(\chi_{t+1}^{u,a}), \tag{5.13}$$

where $\mathbb{X}_t^{u,a}$ is the set of actions available for appliance a at state $\chi_t^{u,a}$, and $\hat{\mathbb{V}}_t^{u,a}(\chi_t^{u,a})$ is the approximate value function for appliance a in state $\chi_t^{u,a}$. When state $\chi_t^{u,a}$ is equal to $(0, w_t^{u,a})$, $\hat{\mathbb{V}}_t^{u,a}(\chi_t^{u,a}) = 0$ since the operation of the appliance is finished. Problem (5.13) can be solved by backward induction.

For operation of the battery, due to the continuity of the decision variables, it is more convenient to represent its scheduling process in the form of an optimization problem

$$\hat{\mathbb{V}}_t^{u,b}(\Lambda_t^u) = \underset{\substack{y_k^u \in \mathbb{Y}_k^u, \\ k \in \mathcal{T}_t}}{\text{minimize}} \sum_{k \in \mathcal{T}_t} \eta_k^u y_k^u, \tag{5.14}$$

where $\hat{\mathbb{V}}_t^{u,b}(\Lambda_t^u)$ is the approximate value function for the battery in state Λ_t^u, and $\mathcal{T}_t \triangleq \{t, \ldots, T\}$.

We assume that at the end of the operation period, the true value function for the whole operation period can be observed (i.e., the total electricity cost for all T time slots). We denote the true value function for user u as $\mathbb{V}_u$. However, based on the vector of value function parameters, $\boldsymbol{\eta}_u = (\eta_1^u, \ldots, \eta_T^u)$, and the total load vector in each time slot,

$\mathbf{L}_u = (L_1^u, \ldots, L_T^u)$, the approximated value function is

$$\hat{\mathbb{V}}_u = \mathbf{L}_u^{\mathrm{T}} \boldsymbol{\eta}_u, \tag{5.15}$$

where T is the transpose operator. After the true value function has been observed, this *new* information is used to adjust the *old* estimate of parameter $\boldsymbol{\eta}_u$. Let m denote the number of observations obtained so far. We define $\mathbb{V}_{u,m}$, $\hat{\mathbb{V}}_{u,m}$, $\boldsymbol{\eta}_{u,m}$, and $\mathbf{L}_{u,m}$ as the values of $\mathbb{V}_u$, $\hat{\mathbb{V}}_u$, $\boldsymbol{\eta}_u$, and $\mathbf{L}_u$ corresponding to the mth observation, respectively. As the new $(m+1)$th observation arrives, we update the value function parameters based on the recursive least square method, i.e.,

$$\boldsymbol{\eta}_{u,m+1} = \boldsymbol{\eta}_{u,m} + \frac{\mathbf{H}_m \mathbf{L}_{u,m+1}}{\frac{1}{\delta} + \mathbf{L}_{u,m+1}^{\mathrm{T}} \mathbf{H}_m \mathbf{L}_{u,m+1}} \left(\mathbb{V}_{u,m+1} - \hat{\mathbb{V}}_{u,m+1} \right),$$

where

$$\hat{\mathbb{V}}_{u,m+1} = \mathbf{L}_{u,m+1}^{\mathrm{T}} \boldsymbol{\eta}_{u,m},$$

and

$$\mathbf{H}_{m+1} = \mathbf{H}_m - \frac{\mathbf{H}_m \mathbf{L}_{u,m+1} \mathbf{L}_{u,m+1}^{\mathrm{T}} \mathbf{H}_m}{\frac{1}{\delta} + \mathbf{L}_{u,m+1}^{\mathrm{T}} \mathbf{H}_m \mathbf{L}_{u,m+1}}.$$

Here, $\mathbf{H}_0$ is a positive definite matrix, and $0 < \delta < 1$ is the observation weight. We note that the influence of the recent observations decreases more rapidly for smaller values of δ.

5.3.2 Trading Problem Formulation

In this section, we consider the trading stage and focus on the competing users and how they interact. If the aggregate excess generation of the competing users is more than the demand, the competing users choose their offered price and generation such that they will

be selected as *providing users*, and their payoff will be maximized. The competing users also take into account the offers advertised by other competing users. In our system model, users do consider the effect of their actions on the MCP. Thus, we need to analyze the Nash equilibrium of the *game* played among multiple competing users who compete to have some share of the market. In this game theoretic model, the strategy of each user consists of its offered price and generation for sale to other users. Let $\mathcal{G}_t = \{1, \ldots, N_t\}$ be the set of all competing users at time slot t, where $N_t \triangleq |\mathcal{G}_t|$. Moreover, $\mathcal{G}_t^{-u} \triangleq \mathcal{G}_t \setminus \{u\}$ is defined as the set of all competing users other than user u. We denote the *offer* of competing user u as $\mathcal{O}_t^u \triangleq (\pi_t^u, g_t^u)$, where π_t^u is the offered price and g_t^u is the offered generation capacity. We define $\mathbb{O}_t^{-u} \triangleq (\mathcal{O}_t^n \mid n \in \mathcal{G}_t^{-u})$ as the vector of offers advertised by all competing users other than user u. The information about the offers of the other users, i.e., $\mathbb{O}_t^{-u}$ is received by user u. Without loss of generality, we assume that the elements of vector $\mathbb{O}_t^{-u}$ are *sorted in an ascending order* based on the entries π_t^n, $n \in \mathcal{G}_t^{-u}$.

User u chooses its offer $\mathcal{O}_t^u$ such that it will be selected as a providing user and its payoff is maximized. To identify whether offer $\mathcal{O}_t^u$ will place user u among the providing users or not, the ranking of offer $\mathcal{O}_t^u$ among other offers $\mathbb{O}_t^{-u}$ has to be evaluated. We adopt the binary auxiliary variable z_t^n, $n \in \mathcal{G}_t^{-u}$, to indicate whether the offered price π_t^n is lower or equal to the selected price π_t^u ($z_t^n = 1$) or not ($z_t^n = 0$). Thus,

$$r_t^u = \sum_{n \in \mathcal{G}_t^{-u}} z_t^n \tag{5.16}$$

indicates the number of users with lower offered prices than user u (i.e., the rank of offer $\mathcal{O}_t^u$ among other offers $\mathbb{O}_t^{-u}$). To calculate binary variable z_t^n, the following constraint is added to the trading optimization problem

$$\frac{\pi_t^u}{\pi_t^n} > z_t^n, \ \forall\, n \in \mathcal{G}_t^{-u}. \tag{5.17}$$

If $\pi_t^n > \pi_t^u$, constraint (5.17) ensures that $z_t^n = 0$, i.e., the constraint is active. If the constraint is inactive, i.e., $\pi_t^n \leq \pi_t^u$, z_t^n can be either 0 or 1. To enforce $z_t^n = 1$ while the constraint is *inactive*, an *auxiliary* term is added to the objective of the optimization problem such that, z_t^n is set to 1. To this end, a term $-K_1 z_t^n$ is added to the objective of the user's minimization problem, where K_1 is a large positive constant. That is, while constraint (5.17) is inactive, z_t^n is set to 1 for minimization of the objective function.

Starting from the user with the smallest offered price, competing users will be added to the set of providing users until there is enough generation to serve the demand. In this case, the MCP will be equal to the highest offered price among the group of providing users. We define $\mathbb{S}_t^n$, $n \in \mathcal{G}_t$, as a group of competing users with offered price less than or equal to π_t^n. The MCP will be equal to π_t^n, $n \in \mathcal{G}_t$, if the following two conditions are satisfied:

C5.1 The aggregate generation of the users within $\mathbb{S}_t^n$, $n \in \mathcal{G}_t$, is sufficient to meet the demand.

C5.2 Among all groups that satisfy condition C5.1, $\mathbb{S}_t^n$ has the smallest number of members.

Condition C5.2 ensures that the process of adding competing users to the set of providing users will be stopped if we have enough generation to meet the demand. If the aggregate generation of all competing users is less than the total demand, all competing users will be selected as providing users and the MCP will be equal to the price advertised by the utility company.

We define S_t^n, $n \in \mathcal{G}_t$, as

$$S_t^n = \sum_{i \in \mathbb{S}_t^n} g_t^i \tag{5.18}$$

$$= \begin{cases} \sum_{i=1}^{n} g_t^i + (1 - z_t^n) g_t^u, & \text{if } \; n \in \mathcal{G}_t^{-u}, \\ \sum_{i \in \mathcal{G}_t^{-u}} z_t^i g_t^i + g_t^u, & \text{if } \; n = u \end{cases} \tag{5.19}$$

to denote the aggregate generation of the users within $\mathbb{S}_t^n$. For each group $\mathbb{S}_t^n$, $n \in \mathcal{G}_t$, we assign a binary auxiliary variable d_t^n to indicate whether the MCP is set to the offered price π_t^n ($d_t^n = 1$) or not ($d_t^n = 0$). We also define auxiliary variable d_t^0 which plays the same role as d_t^n for the utility company's price λ_t^h. The MCP is finally selected from one of the advertised prices π_t^n or λ_t^h. Thus, we have

$$\sum_{n \in \mathcal{G}_t} d_t^n + d_t^0 = 1. \tag{5.20}$$

To enforce condition C5.1 for evaluating the MCP, the constraint

$$\frac{S_t^n}{D_t} \geq d_t^n, \; \forall \, n \in \mathcal{G}_t \tag{5.21}$$

is added to the trading optimization problem, where D_t is the total demand at time slot t. If $S_t^n < D_t$, then constraint (5.21) is *active* and $d_t^n = 0$. If constraint (5.21) is *inactive*, i.e., $S_t^n \geq D_t$, d_t^n can be either 0 or 1. To satisfy constraint (5.20) and to enforce condition C5.2 for evaluating the MCP, one of the variables d_t^n, $n \in \mathcal{G}_t$, or d_t^0 has to be set to 1 if its corresponding constraint (5.21) is *inactive* and the associated set $\mathbb{S}_t^n$ has the smallest number of members. To this end, an auxiliary term $K_2\big(\sum_{n \in \mathcal{G}_t^{-u}} n d_t^n + (N_t + 1) d_t^0 + r_t^u d_t^u\big)$ is added to the objective function of the trading optimization problem, where K_2 is a large positive constant. For the added term, the coefficients of the binary variables d_t^n, $n \in \mathcal{G}_t$, are equal to the size of the groups $\mathbb{S}_t^n$. Thus, to minimize the objective function, among the variables d_t^n, $n \in \mathcal{G}_t$, for which the corresponding constraint (5.21) is inactive, the one with the smallest weight is set to 1. However, if the corresponding constraint (5.21) is active

for all variables d_t^n, $n \in \mathcal{G}_t$, variable d_t^0 will be set to 1. Finally, to evaluate the payoff of user u, we define the auxiliary variable f_t which indicates whether the offer $\mathcal{O}_t^u$ will place user u in the set of providing users ($f_t = 1$) or not ($f_t = 0$).

Given $\mathbb{O}_t^{-u}$, each competing user u solves the following optimization problem to calculate its offered price and generation capacity

$$\underset{\mathbf{I}_t,\, \mathbf{C}_t}{\text{minimize}}\ \mathbb{F}(\mathbf{I}_t, \mathbf{C}_t) \tag{5.22a}$$

$$\text{subject to } f_t, d_t^0, d_t^n, z_t^n, d_t^u \in \{0,1\},\ \forall\, n \in \mathcal{G}_t^{-u}, \tag{5.22b}$$

$$\text{constraints } (5.17), (5.20), (5.21), \tag{5.22c}$$

$$\frac{\sum_{n\in\mathcal{G}_t} d_t^n \pi_t^n + d_t^0 \lambda_t^h}{\pi_t^u} \geq f_t, \tag{5.22d}$$

$$\lambda_t^l \leq \pi_t^u \leq \lambda_t^h, \tag{5.22e}$$

$$0 \leq g_t^u \leq G_t^u, \tag{5.22f}$$

where $\mathbf{I}_t = (\mathbf{d}_t, \mathbf{z}_t, f_t)$, $\mathbf{C}_t = (g_t^u, \pi_t^u)$, $\mathbf{d}_t \triangleq (d_t^0, d_t^n \,|\, n \in \mathcal{G}_t)$, $\mathbf{z}_t \triangleq (z_t^n \,|\, n \in \mathcal{G}_t^{-u})$, and

$$\begin{aligned}\mathbb{F}(\mathbf{I}_t,\mathbf{C}_t) = \mathbb{C}_t^u(g_t^u) - \Big(\sum_{n\in\mathcal{G}_t} d_t^n \pi_t^n + d_t^0 \lambda_t^h\Big) g_t^u f_t - \lambda_t^l \big(G_t^u - g_t^u\big) \\ - K_1 \sum_{n\in\mathcal{G}_t^{-u}} z_t^n - K_2 f_t + K_2 \Big(\sum_{n\in\mathcal{G}_t^{-u}} n d_t^n + (N_t+1) d_t^0 + r_t^u d_t^u\Big),\end{aligned} \tag{5.23}$$

where $K_1 \gg K_2$, and G_t^u is the maximum generation capacity of user u at time slot t. The first term in (5.23) is the cost of providing g_t^u units of energy for the intended user u, c.f. (5.6). The second and the third terms reflect the revenue of user u for the offer (π_t^u, g_t^u). The term

$$\mu_t = \sum_{n\in\mathcal{G}_t} d_t^n \pi_t^n + d_t^0 \lambda_t^h \tag{5.24}$$

is equal to the MCP. Constraint (5.22d) checks whether or not the considered user u is

selected as a providing user. If the offered price π_t^u is greater than the MCP, then user u is not a providing user ($f_t = 0$). Otherwise, f_t can be either 0 or 1. In this case, as f_t appears in the objective function with a negative sign, it will be set to $f_t = 1$. The offered price cannot exceed the price advertised by the utility company and the amount of generation cannot exceed the total generation capacity as reflected by (5.22e) and (5.22f), respectively.

Problem (5.22) is a mixed-integer program. By adopting the generalized Benders' decomposition approach [105], problem (5.22) can be re-written as

$$\underset{\mathbf{I}_t}{\text{minimize}}\ \mathbb{V}(\mathbf{I}_t) \tag{5.25a}$$

$$\text{subject to } (5.20), (5.22\text{b}), \tag{5.25b}$$

$$\frac{\pi_t^{u*}}{\pi_t^n} \geq z_t^n,\ \forall\, n \in \mathcal{G}_t^{-u}, \tag{5.25c}$$

$$\frac{S_t^{n*}}{D_t} \geq d_t^n,\ \forall\, n \in \mathcal{G}_t, \tag{5.25d}$$

$$\frac{\sum_{n\in\mathcal{G}_t^{-u}} d_t^n \pi_t^n + d_t^u \pi_t^{u*} + d_t^0 \lambda_t^h}{\pi_t^{u*}} \geq f_t, \tag{5.25e}$$

where

$$\mathbb{V}(\mathbf{I}_t) = \underset{\mathbf{C}_t}{\text{minimize}}\ \mathbb{F}(\mathbf{I}_t, \mathbf{C}_t) \tag{5.26a}$$

$$\text{subject to}\ \ (5.22\text{c})\text{-}(5.22\text{f}), \tag{5.26b}$$

and problem (5.26) should be feasible for the set of *complicating* variables $\mathbf{I}_t$ that are held fixed. Here, $\mathbf{C}_t^* = (\pi_t^{u*}, g_t^{u*})$ is the solution of (5.26), and S_t^{n*} is as in (5.18), where $\mathbf{C}_t$ is set to $\mathbf{C}_t^*$.

The procedure for solving problem (5.22) is as follows:

Step 1 Let $\mathbf{I}_t^0$ be an initial vector of complicating variable $\mathbf{I}_t$ for which problem (5.26) is feasible. Solve subproblem (5.26) to obtain optimal vector $\mathbf{C}_t^*$. $BFS = \mathbb{F}(\mathbf{I}_t^0, \mathbf{C}_t^*)$ is the best value of problem (5.22) found so far, and $\mathbb{V}(\mathbf{I}_t) = \mathbb{F}(\mathbf{I}_t, \mathbf{C}_t^*)$ forms the objective function of problem (5.25). Select the convergence tolerance parameter $\epsilon > 0$.

Step 2 Solve problem (5.25) to obtain $\mathbf{I}_t^*$ and $\mathbb{V}(\mathbf{I}_t^*)$. If $|\mathbb{V}(\mathbf{I}_t^*) - BFS| < \epsilon$, terminate.

Step 3 Solve subproblem (5.26) to obtain $\mathbf{C}_t^*$ for the complicating variables found in Step 2, $\mathbf{I}_t^*$. If $|\mathbb{F}(\mathbf{I}_t^*, \mathbf{C}_t^*) - BFS| < \epsilon$, terminate. If $\mathbb{F}(\mathbf{I}_t^*, \mathbf{C}_t^*) \leq BFS$, update $BFS = \mathbb{F}(\mathbf{I}_t^*, \mathbf{C}_t^*)$. Return to Step 2.

We note that for a fixed value of $\mathbf{I}_t$, problem (5.26) is a convex quadratic optimization problem and can be solved efficiently by convex optimization techniques. Moreover, $\mathbb{V}(\mathbf{I}_t)$ is bounded. Furthermore, problem (5.25) is a quadratic integer program, which can be solved efficiently with optimization software such as MOSEK [84]. Following the discussion in [105], the adopted generalized Benders' decomposition procedure converges to the optimum value. The proof of the following theorem can be found in [105].

Theorem 5.1 For a finite discrete vector $\mathbf{I}_t$, the generalized Benders' decomposition procedure to solve (5.22) terminates in a finite number of steps for any given $\epsilon > 0$.

5.3.3 Market Clearing Game

From problem (5.22), the payoff of each user depends on its offer (π_t^u, g_t^u) as well as the offers of the other users. Hence, we have the following game among the users:

- Players: Competing users.
- Strategies: Each user selects its offered price and generating capacity (π_t^u, g_t^u) to minimize its cost.

- Costs: The cost of each user is determined based on optimization problem (5.22).

In the proposed market clearing game, first, each competing user assumes random offers for other competing users $\mathbb{O}_t^{-u}$. This assumption is required since at the beginning, competing user u has no prior information about the other users. Next, each competing user u solves its own local problem (5.22). That is, each user plays its best response to the offers advertised by other users. Each competing user broadcasts its new offer through the communication infrastructure. We note that each user broadcasts its offer only if it has been changed compared to its previous offer. Moreover, it will update its local information about the offers of the other users whenever it receives a broadcast message. This procedure continues until no competing user changes its offer.

5.3.4 Algorithm Description

In this section, we explain the steps of the proposed algorithm for load scheduling and power trading. At the beginning, the list of the appliances that should be scheduled is updated, c.f. Line 1. Based on the estimated MCP, the operating schedule of each appliance is determined as in (5.13). Moreover, the charging and discharging rate of the battery is calculated as in (5.14), c.f. Line 2. At the end of the scheduling stage, the user determines whether it has excess generation or not. Users with excess generation receive the information about the offers of other competing users. Each competing user solves (5.22) by adopting the generalized Benders' decomposition approach to obtain $\mathcal{O}_t^u$. Users update their offered price and generation in response to the advertised prices by other competing users. This process continues until the users reach a Nash equilibrium of the trading game as described in Section 5.3.3, c.f. Lines 4 to 7. Users that are selected as providing users sell g_t^u units of their extra generation to consuming users at the MCP, c.f. Line 9. Competing users that are not selected as providing users can sell their excess

Algorithm 5.1: Load scheduling and power trading algorithm executed at each time slot $t \in \mathcal{T}$ for user $u \in \mathcal{U}$.

1: Update the list of the appliances that are to be scheduled.
2: Schedule the appliances as in (5.13) and (5.14).
3: **if** there is excess generation
4: **repeat**
5: Receive offers of other users $\mathbb{O}_t^{-u}$.
6: Solve (5.22) to obtain the offer $\mathcal{O}_t^u$.
7: **until** the Nash equilibrium of the trading game is reached
8: **if** selected as a providing user
9: Sell g_t^u units of excess generation to other users at the MCP.
10: **end if**
11: Sell the remaining generation to the grid at price λ_t^l.
12: **end if**

generation back to the grid at the lower price λ_t^l, c.f. Line 11.

5.4 Performance Evaluation

In this section, we present simulation results and assess the performance of our proposed DSM program. We consider a system with $|\mathcal{U}| = 10$ users. Each user possesses various must-run and controllable appliances. We assume that the information about the energy requirements of the users is not known at the beginning of the operation period. That is, the list of appliances that are awake in each time slot, whether they are must-run or controllable, and the deadline by which the operation of each appliance has to be finished are not known a priori. We run the simulation multiple times with different patterns for the times at which the appliances become awake. We then present the average results. For a typical user, we consider on average 16 appliances. We consider different power consumptions for different operation cycles of the appliances. The pattern of power consumption of the appliances is known to the ECC. Some of the appliances and their

operating specifications are summarized in Table 5.1. In our simulation settings, we assume that λ_t^h varies between 12 cents/kWh and 24 cents/kWh for off-peak and on-peak time slots, respectively. The parameter λ_t^l is set to 4 cents/kWh. The cost function parameters a_t^u and b_t^u are different for different users varying between 0.1 and 0.6, and c_t^u is set to 0, c.f. (5.6).

To have a baseline to compare with, we consider a system without ECC deployment and trading opportunity, where each appliance a starts operation with its power pattern $\mathbf{P}_a$ directly after it has sent an admission request to the ECC unit. In this model, the excess generation of each user is sold to the utility company if it is not consumed or stored. For the system without ECC deployment, users are *not* responding to the variations of the price parameters. In order to be able to better evaluate the effect of trading, we also consider a system in which each users is equipped with an ECC unit to schedule the operation of its controllable appliances, but is not able to trade its excess generation with other users.

Simulation results for the average total power imported from the utility company for the proposed load scheduling algorithm, the system without ECC deployment and trading, and the system with ECC deployment but without trading are depicted in Fig. 5.2. The average imported energy for the system without ECC deployment and trading is 367.2 kWh, while this value is 349.0 kWh for the system in which users are equipped with ECCs but cannot trade. The patterns of power consumption of the latter two systems are different since the system with ECC deployment shifts most of the load to low price time slots. Our proposed load scheduling algorithm reduces the average imported energy to 222.8 kWh because of the trading among the users. The average electricity cost of the users for the system without ECC deployment and trading is \$60.93. For the system with ECC deployment but without trading, this value is reduced to \$53.81. Our proposed algorithm further reduces the electricity cost of the users to \$36.77. The advantages of the proposed algorithm are twofold. First, users are able to decrease their energy expenses by

Table 5.1: Operating specifications of different appliances.

	average P_i^a (kW)	arrival interval
Electric stove	1.5	[06:00, 14:00]
Clothes dryer	0.5	[14:00, 22:00]
Vacuum cleaner	1	[06:00, 15:00]
Refrigerator	0.125	[06:00, 09:00]
Air conditioner	1	[12:00, 22:00]
Dishwasher	1	[15:00, 24:00]
Heater	1.5	[15:00, 03:00]
Water heater	1.5	[06:00, 23:00]
Pool pump	2	[12:00, 21:00]
Electric vehicle	2.5	[16:00, 24:00]
Lighting	0.5	[16:00, 24:00]
TV	0.25	[16:00, 01:00]
PC	0.25	[08:00, 24:00]
Ironing appliance	1	[06:00, 16:00]
Hairdryer	1	[06:00, 13:00]
Other	1.5	[06:00, 24:00]

selling their excess generation to other users with a price higher than λ_t^l. Second, buyers may also benefit from the price reduction due to the competition between multiple sellers.

Our proposed DSM program encourages the utilization of RERs by providing a trading opportunity for the users. To evaluate the effect of the proposed algorithm on the amount of power being injected back into the grid (i.e., the reverse power flow) due to the mismatch between supply and demand, we show in Fig. 5.3 the reverse power flow as a function of the percentage of users that are equipped with RERs. Fig. 5.3 shows that the amount of reverse power flow is significantly reduced for the proposed algorithm.

Due to the competition between the users, the electricity may be offered at a price lower than the price advertised by the utility company. To better understand the effect of the competition between the users on the MCP, we focus on a simplified model in which a single time slot is considered, and three competing users try to sell their excess generation to serve a demand of $D_t = 10$ kW. We consider two different scenarios. In the first scenario, each

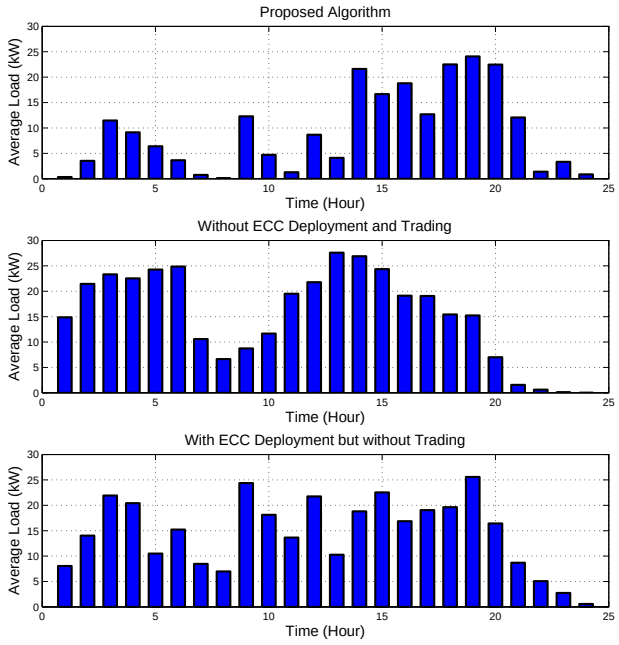

Figure 5.2: Average imported power from utility company for different scenarios.

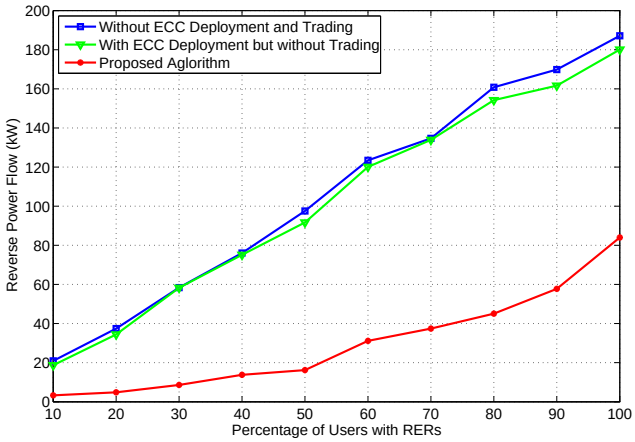

Figure 5.3: Reverse power flow vs. the percentage of users equipped with RERs.

competing user has enough generation to meet the demand (i.e., $G_t^u = 12$ kW), whereas in the second scenario, the generation capacity of each user is not sufficient to meet the total demand (i.e., $G_t^u = 6$ kW). We assume $\lambda_t^h = 12$ cents/kWh, and the cost parameters

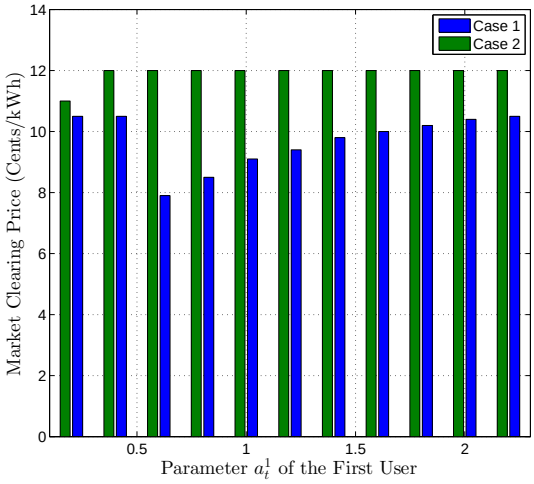

Figure 5.4: MCP for $G_t^u = 12$ kW (Case 1) and $G_t^u = 6$ kW (Case 2).

of the last two users are fixed. Simulation results for the MCP for different values of the first user's parameter a_t^u in (5.6) are depicted in Fig. 5.4. The parameter a_t^u for the last two users is set to 0.6. For the first scenario, if the production cost of the first user is low enough, the user prefers to reduce its production and shares a small portion of the market with other competing users with higher offered prices to keep the price as high as possible and maximize its revenue. On the other hand, if the production costs of the users are comparable, the users will also share the market. In this case, the MCP will drop due to the competition between the users. When the generating capacity of each individual user is not sufficient to meet the total demand, the users will share the market and try to keep the price high to maximize their revenue.

5.5 Summary

In this chapter, we proposed a load control algorithm for DSM. We considered the problem of joint load scheduling and power trading. An approximate dynamic program was proposed to schedule the operation of different types of appliances, and a game theoretic

approach was adopted to model the interaction of the users with excess power generation. Users with excess power generation choose their offered price and output generation such that they obtain a larger share of the market and their revenue is maximized. Simulation results showed that our proposed algorithm reduces the energy costs of the users. That is, competing users may sell their extra generation to local users at a price higher than the buying price of the utility company, and consuming users may buy the electricity from neighboring users at a price lower than the selling price of the utility company. Moreover, the possibility to trade facilitates the integration of RERs by encouraging the users to consume their excess generation locally which mitigates the reverse power flow problem.

Chapter 6

Conclusions and Future Work

In this chapter, we summarize the results and highlight the contributions of this thesis. We also suggest several topics for future work.

6.1 Research Contributions

- In Chapter 2, we proposed a VCG mechanism for DSM. The proposed algorithm encourages efficient energy consumption among users aiming at maximizing the aggregate utility of all users while minimizing the total cost of power generation. Some of the main properties of the proposed mechanism such as truthfulness, efficiency, and nonnegative transfer were studied. We also compared our proposed VCG mechanism with the case where users are price taker. We studied the differences of these two systems, especially from the user payment perspective, and showed that for the VCG mechanism users paid less. Simulation results confirmed that both the users and the energy provider benefited from the proposed scheme.

- In Chapter 3, we proposed an optimal residential load control algorithm for DSM in presence of load uncertainty. Our proposed algorithm aimed to minimize the electricity payment of the users while only some estimate of the future demand was available. We focused on a scenario where real-time pricing was combined with inclining block rate tariffs to balance residential load to achieve a low PAR. We studied the operational constraints of a variety of appliances including *must-run* appliances,

and *interruptible* and *non-interruptible* controllable appliances. Simulation results showed that our proposed scheduling algorithm with load uncertainty reduced the energy payment of users compared to the case where no scheduling algorithm was adopted. Our proposed scheme also improved the overall power system performance by reducing the PAR in aggregate load demand.

- In Chapter 4, we proposed two pricing algorithms based on stochastic approximation technique. The proposed algorithms aimed to minimizing the PAR in the aggregate load demand under the practical scenario where the utility company was uncertain about the users' price-responsiveness. The proposed algorithms were to be implemented in a price control unit based on the information provided the system simulator unit. The system simulator unit considered the effect of control decisions of the ECC unit on the users' load profile. That is, a load control algorithm based on the approximate dynamic programming approach was proposed to simulate the operation of the ECC unit at the demand side. Simulation results showed that our proposed algorithms reduce the PAR of the aggregate load. The proposed algorithms provided incentives for the users to reduce their energy expenses.

- In Chapter 5, we focused on the problem of joint *load scheduling* and *power trading* in systems with high penetration of renewable energy resources. We proposed an approximate dynamic programming approach to schedule the operation of must-run as well as the interruptible and non-interruptible controllable appliances. We showed that the operation of different appliances can be scheduled independently which significantly reduced the complexity of the scheduling algorithm and made the real-time implementation of the algorithm possible. Furthermore, we adopted a game theoretic approach to model the interaction of users with excess power generation. Users with excess power generation choose their offered price and output generation

such that they obtained a larger share of the market and their revenue was maximized. Simulation results showed that our proposed algorithm reduced the electricity cost of the users compared to the case where scheduling and trading were not applied. Our proposed scheme facilitated the integration of RERs and mitigated the reverse power flow problem by providing the users an opportunity to trade their excess generation locally.

6.2 Suggestions for Future Work

In the following, we discus several possibilities for extension of the current work.

1. **The effect of malicious users:** In Chapter 2, we proposed a VCG mechanism to maximize the social welfare of the users. The ideas developed in this chapter can be extended in several directions. For example, a system with multiple energy providers can be considered. The effect of malicious users can be explored as well.

2. **Suboptimal algorithm for load scheduling:** In Chapter 3, we proposed an optimal load control algorithm for DSM. The problem is formulated as a mixed-integer program. However, in situations where computational complexity or convergence time of the algorithm are critical such as in real-time applications, a suboptimal but faster scheme that establishes a balance between simple implementation and adequate performance is desirable. One direction could be designing and analyzing suboptimal algorithms. Approximate algorithms and heuristics are examples of suboptimal algorithms. The efficiency loss of these algorithms can be analyzed, and if possible some bounds on the performance of these algorithms are determined.

3. **Pricing algorithm in systems with different objectives:** In Chapter 4, we assumed that all users are equipped with ECC units and try to minimize their energy

expenses. In practice, some users may schedule their power consumption to achieve different objectives such as minimizing their energy expenses, maximizing the social welfare, etc. In general, some users may be equipped with automated control units while others make control decisions manually. To obtain a better estimate of the likely behavior of the users, for the SSU, considering users with different objectives and different levels of price-responsiveness is an interesting topic for future work. That is, the SSU needs to implement and simulate the likely load control algorithms for different types of users. These algorithms may seek different objectives with different levels of complexity. The SSU needs to establish a balance between the accuracy of estimations and the simple and fast implementation. Thus, different suboptimal algorithms can be adopted to mimic the DSM algorithms controlling the load pattern of the users.

4. **Different market structures for local trading:** In Chapter 5, we considered a system where users are able to trade their excess generation locally. One direction to extend this idea could be investigating the performance of systems with different market structures in which users trade their excess generation. For example the scenario in which the market is not cleared at a unified MCP and each seller can sell its excess generation as its offered price can be considered.

Bibliography

[1] "IEEE guide for smart grid interoperability of energy technology and information technology operation with the electric power system (EPS), end-use applications, and loads," *IEEE Std 2030-2011*, pp. 1–126, Oct. 2011.

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

[3] K. Wang, F. Ciucu, C. Lin, and S. H. Low, "A stochastic power network calculus for integrating renewable energy sources into the power grid," *IEEE J. on Selected Areas in Comm.*, vol. 30, no. 6, pp. 1037–1048, Jul. 2012.

[4] A. M. L. da Silva, L. C. Nascimento, M. A. da Rosa, D. Issicaba, and J. Lopes, "Distributed energy resources impact on distribution system reliability under load transfer restrictions," *IEEE Trans. on Smart Grid*, vol. 3, no. 4, pp. 2048–2055, Dec. 2012.

[5] H. Nunna and S. Doolla, "Demand response in smart distribution system with multiple microgrids," *IEEE Trans. on Smart Grid*, vol. 3, no. 4, pp. 1641–1649, Dec. 2012.

[6] J. M. Guerrero, F. Blaabjerg, T. Zhelev, K. Hemmes, E. Monmasson, S. Jemei, M. P. Comech, R. Granadino, and J. I. Frau, "Distributed generation: Toward a new energy paradigm," *IEEE Industrial Electronics Mag.*, vol. 4, no. 1, pp. 52–64, Mar. 2010.

[7] Y. Atwa, E. El-Saadany, M. Salama, and R. Seethapathy, "Optimal renewable resources mix for distribution system energy loss minimization," *IEEE Trans. on Power Systems*, vol. 25, no. 1, pp. 360–370, Feb. 2010.

[8] S. Suryanarayanan, F. Mancilla-David, J. Mitra, and Y. Li, "Achieving the smart grid through customer-driven microgrids supported by energy storage," *Proc. of IEEE Int'l. Conf. on Industrial Technology*, pp. 884–890, Mar. 2010.

[9] C. Gellings, "The concept of demand-side management for electric utilities," *Proceedings of the IEEE*, vol. 73, no. 10, pp. 1468–1470, Oct. 1985.

[10] B. Ramanathan and V. Vittal, "A framework for evaluation of advanced direct load control with minimum disruption," *IEEE Trans. on Power Systems*, vol. 23, no. 4, pp. 1681–1688, Oct. 2008.

[11] A. H. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," *IEEE Trans. on Smart Grid*, vol. 1, no. 3, pp. 320–331, Dec. 2010.

[12] A. H. Mohsenian-Rad and A. Leon-Garcia, "Optimal residential load control with price prediction in real-time electricity pricing environments," *IEEE Trans. on Smart Grid*, vol. 1, no. 2, pp. 120–133, Sep. 2010.

[13] M. Fahrioglu and F. Alvarado, "Designing incentive compatible contracts for effective demand management," *IEEE Trans. on Power Systems*, vol. 15, no. 4, pp. 1255–1260, Nov. 2000.

[14] N. Ruiz, I. Cobelo, and J. Oyarzabal, "A direct load control model for virtual power plant management," *IEEE Trans. on Power Systems*, vol. 24, no. 2, pp. 959–966, May 2009.

[15] R. Faranda, A. Pievatolo, and E. Tironi, "Load shedding: A new proposal," *IEEE Trans. on Power Systems*, vol. 22, no. 4, pp. 2086–2093, Nov. 2007.

[16] A. Conejo, J. Morales, and L. Baringo, "Real-time demand response model," *IEEE Trans. on Smart Grid*, vol. 1, no. 3, pp. 236–242, Dec. 2010.

[17] T. Jin and M. Mechehoul, "Ordering electricity via Internet and its potentials for smart grid systems," *IEEE Trans. on Smart Grid*, vol. 1, no. 3, pp. 302–310, Dec. 2010.

[18] M. Alizadeh, A. Scaglione, and R. Thomas, "From packet to power switching: Digital direct load scheduling," *IEEE J. on Selected Areas in Comm.*, vol. 30, no. 6, pp. 1027–1036, Jul. 2012.

[19] T. Logenthiran, D. Srinivasan, and T. Shun, "Demand side management in smart grid using heuristic optimization," *IEEE Trans. on Smart Grid*, vol. 3, no. 3, pp. 1244–1252, Sep. 2012.

[20] T. H. Yoo, H. G. Kwon, H. C. Lee, C. Rhee, Y. T. Yoon, and J. Park, "Development of reliability based demand response program in Korea," in *Proc. of IEEE Innovative Smart Grid Technologies*, Anaheim, CA, Jan. 2011.

[21] M. Shinwar and A. Youssef, "A water-filling based scheduling algorithm for the smart grid," *IEEE Trans. on Smart Grid*, vol. 3, no. 2, pp. 710–719, Jun. 2012.

[22] M. Crew, C. Fernando, and P. Kleindorfer, "The theory of peak-load pricing: A survey," *Journal of Regulatory Economics*, vol. 8, no. 3, pp. 215–248, Nov. 1995.

[23] S. Zeng, J. Li, and Y. Ren, "Research of time-of-use electricity pricing models in China: A survey," in *Proc. of IEEE Int'l. Conf. on Industrial Engineering and Engineering Management*, Singapore, Dec. 2008.

[24] C. Chen, K. G. Nagananda, G. Xiong, S. Kishore, and L. V. Synder, "A communication-based appliance scheduling scheme for consumer -premise energy management systems," *IEEE Trans. on Smart Grid*, vol. 4, no. 1, pp. 56–65, Mar. 2013.

[25] S. Salinas, M. Li, and P. Li, "Multi-objective optimal energy consumption scheduling in smart grids," *IEEE Trans. on Smart Grid*, vol. 4, no. 1, pp. 341–348, Mar. 2013.

[26] G. Xiong, C. Chen, S. Kishore, and A. Yener, "Smart (in-home) power scheduling for demand response on the smart grid," in *Proc. of IEEE PES Innovative Smart Grid Technologies Conf.*, Anaheim, CA, Jan. 2011.

[27] R. Krishnan, "Meters of tomorrow," *IEEE Power and Energy Magazine*, pp. 92–94, Mar. 2008.

[28] J. Chen, B. Yang, and X. Guan, "Optimal demand response scheduling with Stackelberg game approach under load uncertainty for smart grid," in *Proc. of IEEE Int'l Conf. on Smart Grid Communications*, Tainan, Taiwan, Nov. 2012.

[29] N. Li, L. Chen, and S. H. Low, "Optimal demand response based on utility maximization in power networks," in *Proc. of IEEE Power and Energy Society General Meeting*, Detroit, MI, Jul. 2011.

[30] M. Pedrasa, T. Spooner, and I. MacGill, "Scheduling of demand side resources using binary particle swarm optimization," *IEEE Trans. on Power Systems*, vol. 24, no. 3, pp. 1173–1181, Aug. 2009.

[31] P. Reiss and M. White, "Household electricity demand, revisited," *Review of Economic Studies*, vol. 72, no. 3, pp. 853–883, July 2005.

[32] S. Borenstein, "Equity effects of increasing-block electricity pricing," *Center for the Study of Energy Markets Working Paper*, vol. 180, Nov. 2008.

[33] BC Hydro, *Electricity Rates*, Mar. 10, 2015. [Online]. Available: http://www.bchydro.com/about/planning_regulatory/tariff_filings.html

[34] D. Hurlbut, *State Clean Energy Practices: Renewable Portfolio Standards*. National Renewable Energy Laboratory, 2008.

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

[36] R. A. Walling, R. Saint, R. C. Dugan, J. Burke, and L. A. Kojovic, "Summary of distributed resources impact on power delivery systems," *IEEE Trans. on Power Delivery*, vol. 23, no. 3, pp. 1636–1644, Jul. 2008.

[37] R. Tonkoski, D. Turcotte, and T. El-Fouly, "Impact of high PV penetration on voltage profiles in residential neighborhoods," *IEEE Trans. on Sustainable Energy*, vol. 3, no. 3, pp. 518–527, May 2012.

[38] S. Bahramirad, W. Reder, and A. Khodaei, "Reliability-constrained optimal sizing of energy storage system in a microgrid," *IEEE Trans. on Smart Grid*, vol. 3, no. 4, pp. 2056–2062, Dec. 2012.

[39] Z. Xu, X. Guan, Q. S. Jia, J. Wu, D. Wang, and S. Chen, "Performance analysis and comparison on energy storage devices for smart building energy management," *IEEE Trans. on Smart Grid*, vol. 3, no. 4, pp. 2136–2147, Dec. 2012.

[40] M. Pedrasa, T. Spooner, and I. MacGill, "Coordinated scheduling of residential distributed energy resources to optimize smart home energy services," *IEEE Trans. on Smart Grid*, vol. 1, no. 2, pp. 134–143, Sep. 2010.

[41] T. T. Kim and H. V. Poor, "Scheduling power consumption with price uncertainty," *IEEE Trans. on Smart Grid*, vol. 2, no. 3, pp. 519–527, Sep. 2011.

[42] S. Tiptipakorn and W. Lee, "A residential consumer-centered load control strategy in real-time electricity pricing environment," in *Proc. of North American Power Symposium*, Las Cruces, NM, Oct. 2007.

[43] S. Kishore and L. Snyder, "Control mechanisms for residential electricity demand in smartgrids," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersburg, MD, Oct. 2010.

[44] B. Chai, J. Chen, Z. Yang, and Y. Zhang, "Demand response management with multiple utility companies: A two-level game approach," *IEEE Trans. on Smart Grid*, vol. 5, no. 2, pp. 722–731, Mar. 2014.

[45] P. Thimmapuram and J. Kim, "Consumers' price elasticity of demand modeling with economic effects on electricity markets using an agent-based model," *IEEE Trans. on Smart Grid*, vol. 4, no. 1, pp. 390–397, Mar. 2013.

[46] H. Akhavan-Hejazi and H. Mohsenian-Rad, "Optimal operation of independent storage systems in energy and reserve markets with high wind penetration," *IEEE Trans. on Smart Grid*, vol. 5, no. 2, pp. 1088–1097, Mar. 2014.

[47] A. Thatte, L. Xie, D. Viassolo, and S. Singh, "Risk measure based robust bidding strategy for arbitrage using a wind farm and energy storage," *IEEE Trans. on Smart Grid*, vol. 4, no. 4, pp. 2191–2199, Dec. 2013.

[48] S. Kahrobaee, R. Rajabzadeh, L.-K. Soh, and S. Asgarpoor, "A multiagent modeling and investigation of smart homes with power generation, storage, and trading features," *IEEE Trans. on Smart Grid*, vol. 4, no. 2, pp. 659–668, Jun. 2013.

[49] K. Yang and A. Walid, "Outage-storage tradeoff in frequency regulation for smart grid with renewables," *IEEE Trans. on Smart Grid*, vol. 4, no. 1, pp. 245–252, Dec. 2012.

[50] P. Samadi, A. H. Mohsenian-Rad, R. Schober, and V. W. S. Wong, "Advanced demand side management for the future smart grid using mechanism design," *IEEE Trans. on Smart Grid*, vol. 3, no. 3, pp. 1170–1180, Sep. 2012.

[51] P. Samadi, A. H. Mohsenian-Rad, V. W. S. Wong, and R. Schober, "Tackling the load uncertainty challenges for energy consumption scheduling in smart grid," *IEEE Trans. on Smart Grid*, vol. 4, no. 2, pp. 1007–1016, Jun. 2013.

[52] J. Spall, *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. John Wiley & Sons, 2003, vol. 64.

[53] P. Samadi, H. Mohsenian-Rad, V. W. S. Wong, and R. Schober, "Real-time pricing for demand response based on stochastic approximation," *IEEE Trans. on Smart Grid*, vol. 5, no. 2, pp. 789–798, March 2014.

[54] M. Alizadeh, A. Scaglione, R. Thomas, and D. Callaway, "Information infrastructure for cellular load management in green power delivery systems," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Brussels, Belgium, Oct. 2011.

[55] P. Samadi, A. H. Mohsenian-Rad, R. Schober, V. W. S. Wong, and J. Jatskevich, "Optimal real-time pricing algorithm based on utility maximization for smart grid," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersburg, MD, Oct. 2010.

[56] S. Yang and B. Hajek, "VCG-Kelly mechanisms for allocation of divisible goods: Adapting VCG mechanisms to one-dimensional signals," *IEEE J. on Selected Areas in Comms.*, vol. 25, no. 6, pp. 1237–1243, Aug. 2007.

[57] S. Su and M. van der Schaar, "On the application of game-theoretic mechanism design for resource allocation in multimedia systems," *IEEE Trans. on Multimedia*, vol. 10, no. 6, pp. 1197–1207, Oct. 2008.

[58] J. Jia, Q. Zhang, Q. Zhang, and M. Liu, "Revenue generation for truthful spectrum auction in dynamic spectrum access," in *Proc. of the Tenth ACM Int'l Symposium on Mobile Ad Hoc Networking and Computing*, New Orleans, LA, May 2009.

[59] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, *Algorithmic Game Theory*. Cambridge University Press, 2007.

[60] S. Caron and G. Kesidis, "Incentive-based energy consumption scheduling algorithms for the smart grid," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersburg, MD, Oct. 2010.

[61] C. Ibars, M. Navarro, and L. Giupponi, "Distributed demand management in smart grid with a congestion game," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersburg, MD, Oct. 2010.

[62] S. Stoft, *Power System Economics: Designing Markets for Electricity*. Wiley-IEEE Press, 2002.

[63] ——, "The demand for operating reserves: Key to price spikes and investment," *IEEE Trans. on Power Systems*, vol. 18, no. 2, pp. 470–477, May 2003.

[64] P. Visudhiphan and M. Ilic, "Dynamic games-based modeling of electricity markets," in *IEEE Power Engineering Society 1999 Winter Meeting*, New York, NY, Feb. 1999.

[65] A. Mas-Colell, M. D. Whinston, and J. R. Green, *Microeconomic Theory*, 1st ed. USA: Oxford University Press, 1995.

[66] M. Fahrioglu and F. Alvarado, "Using utility information to calibrate customer demand management behavior models," *IEEE Trans. on Power Systems*, vol. 16, no. 2, pp. 317–322, May 2001.

[67] A. Philpott and E. Pettersen, "Optimizing demand-side bids in day-ahead electricity markets," *IEEE Trans. on Power Systems*, vol. 21, no. 2, pp. 488–498, May 2006.

[68] H. Chao, "Demand response in wholesale electricity markets: The choice of customer baseline," *J. of Regulatory Economics*, vol. 39, no. 1, pp. 1–21, Nov. 2010.

[69] M. Fahrioglu, M. Fern, and F. Alvarado, "Designing cost effective demand management contracts using game theory," in *Proc. of IEEE Power Eng. Soc. Winter Meeting*, New York, NY, Jan. 1999.

[70] A. Wood and B. Wollenberg, *Power Generation, Operation, and Control*. New York: Wiley-Interscience, 1996.

[71] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[72] S. Low and D. Lapsley, "Optimization flow control I: Basic algorithm and convergence," *IEEE/ACM Trans. on Networking*, vol. 7, no. 6, pp. 861–874, Dec. 1999.

[73] R. Johari, S. Mannor, and J. Tsitsiklis, "Efficiency loss in a network resource allocation game: The case of elastic supply," *IEEE Trans. on Automatic Control*, vol. 50, no. 11, pp. 1712–1724, Nov. 2005.

[74] R. Johari and J. Tsitsiklis, "A scalable network resource allocation mechanism with bounded efficiency loss," *IEEE J. on Selected Areas in Comms.*, vol. 24, no. 5, pp. 992–999, May 2006.

[75] Y. Shoham and K. Leyton-Brown, *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2008.

[76] R. Johari and J. Tsitsiklis, "Communication requirements of VCG-like mechanisms in convex environments," in *Proc. of Allerton Conference on Communication, Control, and Computing*, Monticello, IL, Sep. 2005.

[77] ——, "Efficiency of scalar-parameterized mechanisms," *Operations Research*, vol. 57, no. 4, pp. 823–839, May 2009.

[78] R. Ferrero and S. Shahidehpour, "Short-term power purchases considering uncertain prices," *IEE Proceedings of Generation, Transmission and Distribution*, vol. 144, no. 5, pp. 423–428, Sep. 1997.

[79] T. Hubert and S. Grijalva, "Realizing smart grid benefits requires energy optimization algorithms at residential level," in *Proc. of IEEE Innovative Smart Grid Technologies*, Anaheim, CA, Jan. 2011.

[80] M. Parvania and M. Fotuhi-Firuzabad, "Demand response scheduling by stochastic SCUC," *IEEE Trans. on Smart Grid*, vol. 1, no. 1, pp. 89–98, Jun. 2010.

[81] K. Cheung and R. Rios-Zalapa, "Smart dispatch for large grid operations with integrated renewable resources," in *Proc. of IEEE Innovative Smart Grid Technologies*, Anaheim, CA, Jan. 2011.

[82] W. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Wiley-Blackwell, 2007.

[83] D. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific Belmont, MA, 2005.

[84] "MOSEK," *http://www.mosek.com*, Mar. 10, 2015.

[85] U.S. Department of Energy, *The 2010 Buildings Energy Data Book*. Energy Efficiency and Renewable Energy, Mar. 2011.

[86] "U.S. Energy Information Administration," Mar. 10, 2015. [Online]. Available: http://www.eia.gov

[87] C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Courier Dover, 1998.

[88] C. Papadimitriou, "On the complexity of integer programming," *J. of the ACM*, vol. 28, no. 4, pp. 765–768, Oct. 1981.

[89] P. Samadi, R. Schober, and V. W. S. Wong, "Optimal energy consumption scheduling using mechanism design for the future smart grid," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Brussels, Belgium, Oct. 2011.

[90] M. Roozbehani, M. Dahleh, and S. Mitter, "Volatility of power grids under real-time pricing," *IEEE Trans. on Power Systems*, vol. 27, no. 4, pp. 1926–1940, Nov. 2012.

[91] J. Liu, Y. Xiao, S. Li, W. Liang, and C. Chen, "Cyber security and privacy issues in smart grids," *IEEE Comm. Surveys and Tutorials*, pp. 981–997, Fourth quarter 2012.

[92] L. Gerencser, G. Kozmann, and Z. Vago, "SPSA for non-smooth optimization with application in ECG analysis," in *Proc. of IEEE Conference on Decision and Control*, Tampa, FL, Dec. 1998.

[93] Y. He, M. C. Fu, and S. I. Marcus, "Convergence of simultaneous perturbation stochastic approximation for nondifferentiable optimization," *IEEE Trans. on Automatic Control*, vol. 48, no. 8, pp. 1459–1463, Aug. 2003.

[94] F. Yousefian, A. Nedić, and U. V. Shanbhag, "On stochastic gradient and subgradient methods with adaptive steplength sequences," *Automatica*, vol. 48, no. 1, pp. 56–67, Jan. 2012.

[95] Y. Kim, E. Ngai, and M. Srivastava, "Cooperative state estimation for preserving privacy of user behaviors in smart grid," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Brussels, Belgium, Oct. 2011.

[96] T. Chim, S. M. Yiu, L. C. K. Hui, and V. O. K. Li, "PASS: Privacy-preserving authentication scheme for smart grid network," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Brussels, Belgium, Oct. 2011.

[97] F. Li, B. Luo, and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersburg, MD, Oct. 2010.

[98] G. Kalogridis, C. Efthymiou, S. Denic, T. Lewis, and R. Cepeda, "Privacy for smart meters: Towards undetectable appliance load signatures," in *Proc. of IEEE Int'l. Conf. on Smart Grid Communications*, Gaithersburg, MD, Oct. 2010.

[99] Z. Ding, Y. Guo, D. Wu, and Y. Fang, "A market based scheme to integrate distributed wind energy," *IEEE Trans. on Smart Grid*, vol. 4, no. 2, pp. 976–984, Jun. 2013.

[100] I. Lampropoulos, N. Baghina, W. Kling, and P. Ribeiro, "A predictive control scheme for real-time demand response applications," *IEEE Trans. on Smart Grid*, vol. 4, no. 4, pp. 2049–2060, Dec. 2013.

[101] N. Gatsis and G. Giannakis, "Decomposition algorithms for market clearing with large-scale demand response," *IEEE Trans. on Smart Grid*, vol. 4, no. 4, pp. 1976–1987, Dec. 2013.

[102] I. Atzeni, L. Ordonez, G. Scutari, D. Palomar, and J. Fonollosa, "Demand-side management via distributed energy generation and storage optimization," *IEEE Trans. on Smart Grid*, vol. 4, no. 2, pp. 866–876, Jun. 2013.

[103] M. Singh, V. Khadkikar, A. Chandra, and R. K. Varma, "Grid interconnection of renewable energy sources at the distribution level with power-quality improvement features," *IEEE Trans. on Power Delivery*, vol. 26, no. 1, pp. 307–315, Jan. 2011.

[104] V. Krishna, *Auction Theory*, 2nd ed. Academic Press, 2009.

[105] A. M. Geoffrion, "Generalized Benders decomposition," *J. of Optimization Theory and Applications*, vol. 10, no. 4, pp. 237–260, 1972.

Appendix A

Proof of Proposition 2.3

Given the payment in (2.22), since user n cannot affect the term h_n by changing $\hat{\mathbf{I}}_n$, it declares $\hat{\mathbf{I}}_n$ only to maximize

$$W_n(\mathbf{x}_n(\hat{\mathbf{I}}), t_n(\hat{\mathbf{I}})) = U_n\Big(\sum_{k\in\mathcal{K}} x_n^k(\hat{\mathbf{I}})\Big) + \sum_{m\in\mathcal{N}_{-n}} \hat{U}_m\Big(\sum_{k\in\mathcal{K}} x_m^k(\hat{\mathbf{I}})\Big) - \sum_{k\in\mathcal{K}} C_k\Big(\sum_{m\in\mathcal{N}} x_m^k(\hat{\mathbf{I}})\Big).$$

However, the above expression is bounded above by

$$\underset{\substack{\mathbf{x}_n\in\mathcal{X}_n,\, \mathbf{x}_m\in\hat{\mathcal{X}}_m,\\ m\in\mathcal{N}_{-n}}}{\text{maximize}} \quad U_n\Big(\sum_{k\in\mathcal{K}} x_n^k\Big) + \sum_{m\in\mathcal{N}_{-n}} \hat{U}_m\Big(\sum_{k\in\mathcal{K}} x_m^k\Big) - \sum_{k\in\mathcal{K}} C_k\Big(\sum_{m\in\mathcal{N}} x_m^k\Big).$$

Note that $\mathbf{x}(\hat{\mathbf{I}})$ satisfies (2.21), and user n can achieve the maximum payoff by truthfully declaring $\hat{\mathbf{I}}_n = \mathbf{I}_n$ for solving (2.21). Since this optimal strategy does not depend on the demand information declared by other users, it confirms the result that for VCG mechanisms, truthful declaration is a dominant strategy. ■

Appendix B

Proof of Theorem 2.1

In the equilibrium, all users declare their demand information truthfully. Then, we can write the payment of user n as

$$t_n(\mathbf{I}) = -\left(\sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}} x_m^k(\mathbf{I})\right)\right) + \left(\sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I}_{-n})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}_{-n}} x_m^k(\mathbf{I}_{-n})\right)\right),$$

where $\mathbf{x}(\mathbf{I}_{-n})$ is the optimal solution for the social objective when user n is excluded from the system. So, we have

$$\sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I}_{-n})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}_{-n}} x_m^k(\mathbf{I}_{-n})\right) \geq \sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}_{-n}} x_m^k(\mathbf{I})\right). \tag{B.1}$$

Furthermore, from Assumption 2.1, $C_k(\cdot)$ is an increasing function. Therefore, we have

$$\sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I}_{-n})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}_{-n}} x_m^k(\mathbf{I}_{-n})\right) \geq \sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}} x_m^k(\mathbf{I})\right), \tag{B.2}$$

and thus (2.24) is nonnegative. ■

Appendix C

Proof of Theorem 2.2

In the equilibrium, all users declare their demand information truthfully. So, the payment of user n is

$$t_n(\mathbf{I}) = -\left(\sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}} x_m^k(\mathbf{I})\right)\right) + \left(\sum_{m\in\mathcal{N}_{-n}} U_m\left(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I}_{-n})\right) - \sum_{k\in\mathcal{K}} C_k\left(\sum_{m\in\mathcal{N}_{-n}} x_m^k(\mathbf{I}_{-n})\right)\right).$$

Since $\mathbf{x}(\mathbf{I})$ is the optimal solution for the social objective problem, the optimality conditions of (2.21) imply that

$$\begin{aligned} &\lambda_k^* = p_k\left(\textstyle\sum_{n\in\mathcal{N}} x_n^k(\mathbf{I})\right), \\ &U_n'\left(\sum_{k\in\mathcal{K}} x_n^k(\mathbf{I})\right) = \lambda_k^*, \quad \text{if } x_n^k(\mathbf{I}) > m_n^k \ \text{ and } \ \sum_{k\in\mathcal{K}} x_n^k(\mathbf{I}) > E_n, \\ &U_n'\left(\sum_{k\in\mathcal{K}} x_n^k(\mathbf{I})\right) \le \lambda_k^*, \quad \text{if } x_n^k(\mathbf{I}) = m_n^k \text{ or } \sum_{k\in\mathcal{K}} x_n^k(\mathbf{I}) = E_n, \end{aligned} \tag{C.1}$$

where $p_k(\cdot)$ has been introduced in (2.8), and λ_k^* is the market clearing price for the problem (2.10).

By concavity of U_n we have

$$\begin{aligned} U_n(x) &\geq U_n^{'}(x)x, \\ \tfrac{U_n(x)-U_n(y)}{U_n^{'}(x)} &\leq x-y, \end{aligned} \tag{C.2}$$

and by convexity of C_k, we have

$$\begin{aligned} C_k(q_1) &\leq C_k^{'}(q_1)q_1, \\ \tfrac{C_k(q_1)-C_k(q_2)}{C_k^{'}(q_1)} &\geq q_1-q_2. \end{aligned} \tag{C.3}$$

Then, from (C.1)-(C.3), we have

$$\begin{aligned} t_n &\leq \sum_{m\in\mathcal{N}_{-n}} \frac{\lambda_k^*\Big[U_m\big(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I})\big) - U_m\big(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I}_{-n})\big)\Big]}{U_m^{'}\big(\sum_{k\in\mathcal{K}} x_m^k(\mathbf{I})\big)} \\ &\quad - \sum_{k\in\mathcal{K}} \frac{\lambda_k^*\Big[C_k\Big(\sum_{m\in\mathcal{N}_{-n}} x_m^k(\mathbf{I}_{-n})\Big) - C_k\Big(\sum_{m\in\mathcal{N}} x_m^k(\mathbf{I})\Big)\Big]}{p_k\Big(\sum_{m\in\mathcal{N}} x_m^k(\mathbf{I})\Big)} \\ &\leq \sum_{k\in\mathcal{K}} \lambda_k^* x_n^k(\mathbf{I}), \end{aligned}$$

which completes the proof. ■