Distributed Optimization in Power Networks and General Multi-agent Systems

Thesis by

Na Li

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California

2013

(Defended May 23, 2013)

To my dear family.

Acknowledgements

I am deeply grateful to my advisor Professor John Doyle and co-advisor Professor Steven Low for the continuous support for my PhD study and research. Since September 2007, they have deeply influenced me in different aspects of my life, not only through their great guidance in research, but also through their patience, enthusiasm, encouragement, and genuine concern for students. John introduced me to a variety of areas, provided the vision, and gave me the freedom and support to pursue various projects. He has very broad interests and is expert at extracting essentials almost for any topic. His passion, enthusiasm, and belief in science and research have been a great source of inspiration and motivation to me. Steven taught me how to integrate theory and practice in order to make contributions to real systems. He never scarifies practical applications for theoretical beauty, or vice versa. His high standards for the quality of work has encouraged me to pursue the perfection of the work. John and Steven, thanks for being such good role models for my academic career. I could not have imagined having better advisors for my graduate study.

My sincere gratitude goes to my former college research advisor, Professor Jeff Shamma. The research experience at Jeff's lab in 2006 and 2007 introduced a new area, control and systems, to that little college girl and shape her later career path. Jeff has been helpful in providing advice many times during my graduate study and there were always so much for me to learn from any conversation we had. He was and remains one of my best role models for a scientist, mentor, and teacher.

I would like to sincerely thank Professor Lijun Chen, Professor Jason Marden for being great mentors and friends for me. The thesis would have not been possible if it were not for their guidance and collaborations. They have been very patient and detailed in teaching me discipline and precision in the research process. They were always willing to help and provide their best suggestions toward career and life and ultimately helped me grow as both an individual and a researcher. Lijun, thank you for introducing me to the area of

the smart grid and for providing the great vision. Your insights, especially physical insights, always lead to great problems and solutions. Without you, I could not have done a coherent work for power grids. Jason, our collaboration dates back to my senior year in college. I was lucky to join Caltech as a PhD student at same time as you joined Caltech as a postdoc. Thank you for introducing me to the area of game theory and for teaching me the joy of "playing games" in academia and life. Lijun and Jason, I remember and will alway remember how you guide my work and revise my paper step by step. You were and will be my dear mentors and friends forever.

I also would like to thank other professors who has helped me grow up in the past years. Professor Adam Weirman, thank you for always making time to provide me advice and give me feedback about my work. You have set a great model for me to learn as a junior faculty. Professor Richard Murray, thank you for being on my PhD defense committee and for being available every time when I need help and advice.

I am grateful to my intelligent colleagues in Control & Dynamical Systems (CDS) and RSRG. Special gratitude goes to some current and past group members for fruitful collaborations and intriguing discussions. I wish to acknowledge an incomplete list of group members: Andrea Censi, Chenghao Chien, Jerry Cruz, Masoud Farivar, Lingwen Gan, Dennice Gayme, Shuo Han, Vanessa Jonsson, Andy Lamperski, Javad Lavaei, Minghong Lin, Zhenhua Liu, Nikolai Matni, Somayeh Sojoudi, Changhong Zhao, etc.

Furthermore, I would like to thank all those who helped me, including my friends and teachers; without them I would not be where I am today. Special thanks go to my friends at Caltech: Qi An, Ting Chen, Mingyuan Huang, Rui Huang, Yu Huang, Guanglei Li, Piya Pal, Rangoli Sharan, Zhiying Wang, Mao Wei, Xi Zhang, Guoan Zheng, Hongchao Zhou, Zicong Zhou, Zhaoyan Zhu, etc.

Last but not least, I would like to give my deepest gratitude to my family, my mother Xinhua Xia, my father Chunsheng Li, my husband Pengcheng Luo, my sister Cheng Li and my brother Hui Li for their endless love and support of my career. I dedicate this thesis to my family as an inadequate appreciation of everything that they have done for me.

Abstract

The dissertation studies the general area of complex networked systems that consist of interconnected and active heterogeneous components and usually operate in uncertain environments and with incomplete information. Problems associated with those systems are typically large-scale and computationally intractable, yet they are also very well-structured and have features that can be exploited by appropriate modeling and computational methods. The goal of this thesis is to develop foundational theories and tools to exploit those structures that can lead to computationally-efficient and distributed solutions, and apply them to improve systems operations and architecture.

Specifically, the thesis focuses on two concrete areas. The first one is to design distributed rules to manage distributed energy resources in the power network. The power network is undergoing a fundamental transformation. The future smart grid, especially on the distribution system, will be a large-scale network of distributed energy resources (DERs), each introducing random and rapid fluctuations in power supply, demand, voltage and frequency. These DERs provide a tremendous opportunity for sustainability, efficiency, and power reliability. However, there are daunting technical challenges in managing these DERs and optimizing their operation. The focus of this dissertation is to develop scalable, distributed, and real-time control and optimization to achieve system-wide efficiency, reliability, and robustness for the future power grid. In particular, we will present how to explore the power network structure to design efficient and distributed market and algorithms for the energy management. We will also show how to connect the algorithms with physical dynamics and existing control mechanisms for real-time control in power networks.

The second focus is to develop distributed optimization rules for general multi-agent engineering systems. A central goal in multiagent systems is to design local control laws for the individual agents to ensure that the emergent global behavior is desirable with respect to the given system level objective. Ideally, a system

designer seeks to satisfy this goal while conditioning each agent's control on the least amount of information possible. Our work focused on achieving this goal using the framework of game theory. In particular, we derived a systematic methodology for designing local agent objective functions that guarantees (i) an equivalence between the resulting game-theoretic equilibria and the system level design objective and (ii) that the resulting game possesses an inherent structure that can be exploited for distributed learning, e.g., potential games. The control design can then be completed by applying any distributed learning algorithm that guarantees convergence to the game-theoretic equilibrium. One main advantage of this game theoretic approach is that it provides a hierarchical decomposition between the decomposition of the systemic objective (game design) and the specific local decision rules (distributed learning algorithms). This decomposition provides the system designer with tremendous flexibility to meet the design objectives and constraints inherent in a broad class of multiagent systems. Furthermore, in many settings the resulting controllers will be inherently robust to a host of uncertainties including asynchronous clock rates, delays in information, and component failures.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The dissertation focuses on the general area of complex networked systems that consist of interconnected and active heterogeneous components that usually operate in uncertain environments and with incomplete information. Problems associated with those systems are typically large-scale and computationally intractable, yet they are also very well-structured and have features that can be exploited by appropriate modeling and computational methods. The goal of this thesis is to develop foundational theories and tools to exploit those structures that can lead to computationally-efficient and distributed solutions, and apply them to improve systems operations and architecture.

Specifically, this dissertation focuses on two concrete areas. The first one is to design distributed rules to manage distributed energy resources in the power network; the second one is to design distributed optimization rules for more general multi-agent systems.

1.1 The smart power network

With the increasing penetration of renewable and distributed energy resources and the rapid deployment of communication, sensing, and computing infrastructures, the power network architecture is in transition from a centralized, vertically integrated structure to one that is more distributed, open, and autonomous. This trend provides tremendous opportunities for improvements in sustainability, efficiency, power quality and reliability. But it also presents daunting technical challenges, particularly those imposed by non-dispatchable and volatile renewable generation, and the large number of active end-points in the future system. The goal of my research is to achieve system-wide efficiency, reliability, and robustness for the future power grid, through

developing foundational theories, innovative algorithms, and novel architecture for scalable, distributed, real-time control and optimization. In this doctorate dissertation, I will present my work in pursuit of this direction through the following aspects:

1.1.1 Demand response: market models with appliance characteristics and the power network structure.

Demand response is increasingly needed to improve power system efficiency and integrate renewable generation. It will not only be applied to reduce peaks and shift load for economic benefits, but will increasingly be evoked to improve stability and reduce operating reserves by adapting elastic loads to intermittent and fluctuating renewable generation. Demand response involves both economic and engineering aspects of the power system, and requires coordinating actions among users and electric appliances while ensuring security, stability, and reliability of the grid.

We first study an abstract market model where a set of households are served by a single load-serving entity (LSE). The LSE may represent a regulated monopoly like most utility companies in the United States today, or a non-profit cooperative that serves a community of end users. We consider households that operate different appliances including air conditioners, washers, lighting, electric vehicles, batteries, etc, each of which provides a certain benefit depending on the pattern or volume of power it consumes. Each household wishes to optimally schedule its power consumption so as to maximize its individual net benefit subject to various consumption constraints. Based on utility maximization, we proposed a dynamic pricing scheme and a distributed approach for the LSE to coordinate users' demand response to benefit the overall system, including reducing the peak load, smoothing the entire demand profile, and saving significant generation costs. This work serves as a good starting point to study the market dynamics and the residential model for demand response.

We further extend our approaches to study demand response over a radial distribution networks with the power flow constraints and the operation constraints. We formulate the distributed load management over a radial network as an optimal power flow (OPF) problem that maximizes the aggregate user utilities and minimizes the supply cost and the power line losses, subject to the power flow constraints and operating

constraints. The OPF problem is general non-convex. This necessitates our work on OPF and its distributed solutions.

1.1.2 Optimal power flow (OPF): convexification and distributed power optimization

The optimal power flow (OPF) problem is a fundamental problem that underlies many power systems operations and planning. It seeks to optimize a certain objective subject to the power flow constraints and the operation constraints. The OPF problem is, in general, non-convex and difficult to solve. Recently, convex optimization tools have been used to relax the OPF problem to a convex problem in order to explore the power network structure for better system operations. Previous work showed that convex relaxation is exact for the radial networks if there are no lower bounds on the power injection. However, this condition does not hold for various applications including demand response and Volt/VAR control. Thus we explore other sufficient conditions without removing the lower bounds on power injection. We provide a series of sufficient conditions to guarantee the exact relaxation of the OPF problem for the radial network when the voltage upper bound is removed or modified by an approximation. These conditions are verified to hold for a wide class of distribution circuits, and the resulting voltage is in the safe operation range.

Convexity does not only facilitate the design of effective pricing schemes for the power market involved in demand response, but also enables the development of tractable, scalable, and distributed algorithms for system operations. We design a locational marginal pricing scheme and distributed algorithms for the utility company to guide users' decisions over a distribution network. Case studies on South California Edison distribution circuits showed that the algorithm converges to the optimum very fast. We further develop a fully decentralized OPF algorithm where the users make their own local decisions based only on local information and local communication with their direct neighbors.

1.1.3 Real-time energy balancing: economic automatic generation control (AGC)

In the distributed control of smart grids, the distributed algorithms derived from optimization tools usually regard certain physical variables, such as branch power flow and frequency, as computable controls that can

be instantaneously updated to arbitrary values, which is not usually the case for power systems. Hence these algorithms cannot be implemented as real-time controls that are required or desired, as amplified by mitigating fluctuations in renewable generation. For real-time control, the algorithm (derived from the optimization model) that governs the update of a physical variable must coincide with the real physical dynamics or the built-in control mechanisms that govern the evolution of that variable, namely that the computation is implicitly carried out by the real physical dynamics of the power network. This would also make local sensing sufficient for distributed control, e.g., distributed load management based on local frequency measurement. However, it imposes hard constraints on algorithm design as those conventional optimization algorithms such as the gradient algorithms are usually not consistent with the physical dynamics.

One way to take into account the impact of a built-in mechanism is to reverse-engineer this mechanism to find out what optimization problem it implicitly solves, and then incorporate the corresponding objective function into the optimization model for the design or control problem. As an initial step, we have studied automatic generation control (AGC). AGC uses deviations in generator speeds and/or frequency as control signals to invoke appropriate valve action in order to regulate the mechanical power generation in response to load changes. The main objective of AGC is to maintain power balance and nominal system frequency; however how to optimize AGC to improve energy efficiency is less studied. We reverse-engineered AGC by showing that the AGC can be formulated as a partial primal-dual gradient algorithm to solve an optimization problem. We extended the resulting optimization problem to include generation cost, and proposed a distributed management scheme that is based only on local measurements and communications and takes into account the impact of AGC. This work provides a good starting point for developing a framework for systematic design of distributed, low-complexity load/generation control mechanisms to achieve system-wide efficiency and robustness.

1.2 Decentralized optimization: a game theoretical approach

The central goal in multiagent systems is to design *local* control laws for the individual agents to ensure that the emergent global behavior is desirable with respect to a given system level objective. These control

laws provide the groundwork for a decision making architecture that possesses several desirable attributes including real-time adaptation and robustness to dynamic uncertainties. However, realizing these benefits requires addressing the underlying complexity associated with a potentially large number of interacting agents and the analytical difficulties of dealing with overlapping and partial information. Furthermore, the design of such control laws is further complicated by restrictions placed on the set of admissible controllers which limit informational and computational capabilities.

Game theory is beginning to emerge as a powerful tool for the design and control of multiagent systems. Utilizing game theory for this purpose requires two steps. The first step is to model the agents as self-interested decision makers in a game theoretic environment. This step involves defining a set of choices and a local objective function for each decision maker. The second step involves specifying a distributed learning algorithm that enables the agents to reach a desirable operating point, e.g., a Nash equilibrium of the designed game. One of the core advantages of game theory is that it provides a hierarchical decomposition between the decomposition of the systemic objective (*game design*) and the specific local decision rules (*distributed learning algorithms*). For example, if the game is designed as a potential game then there is an inherent robustness to decision making rules as a wide class of distributed learning algorithms can achieve convergence to a pure Nash equilibrium under a variety of informational dependencies.

The main contribution of this dissertation is the development of a systematic methodology for the design of local agent objective functions that guarantees the efficiency of the resulting equilibria. In particular, we derived a methodology for designing local agent objective functions that guarantees (i) an equivalence between the resulting game-theoretic equilibria and the system level design objective and (ii) that the resulting game possesses an inherent structure that can be exploited for distributed learning, e.g., potential games. The control design can then be completed by applying any distributed learning algorithm that guarantees convergence to the game-theoretic equilibrium. This hierarchical decomposition between the decomposition of the systemic objective and the specific local decision rules provides the system designer with the tremendous flexibility to meet the design objectives and constraints inherent in a broad class of multiagent systems. Furthermore, in many settings the resulting controllers will be inherently robust to a host of uncertainties including asynchronous clock rates, delays in information, and component failures.

1.3 Structure and contributions of the thesis

The contribution of each chapter is listed below. All of the chapters can be read separately according to the readers' interests and backgrounds.

Part I: Distributed Energy Management in Power Networks (Chapters 2, 3, 4, 5)

Chapter 2 studies a demand response problem where a set of households are served by a single load-serving entity (LSE) and each household operates different appliances. Based on utility maximization, we proposed a dynamic pricing scheme and a distributed approach for the utility company to coordinate users' demand response to benefit the overall system, including reducing the peak load, smoothing the entire demand profile, and saving significant generation costs.

Chapter 3 focuses on the optimal power flow (OPF) problem, which is generally non-convex. We advocate a second-order cone relaxation for OPF using the branch flow model and provide sufficient conditions under which the relaxation is exact. These conditions are demonstrated to hold for a wide class of practical power distribution systems.

Chapter 4 studies the distributed load management over a radial distribution network, by formulating it as an optimal power flow (OPF) problem. We propose two different distributed mechanisms to achieve the optimum. In the first one, there is a load-serving entity to set the price signals in order to coordinate the users' demand response and in the second one the users coordinate their decisions through local communications with neighbors.

Chapter 5 studies the real-time control mechanisms to balance generation and load. We focus on modifying automatic generation control (AGC) to keep energy balanced and also to make energy allocation efficient at the same time.

Part II: Designing Games for Distributed Optimization (Chapters 6, 7,8)

Chapter 6 propose a game design for distributed optimization where the optimization problem has coupled objective function but decoupled constraints. We also provide a learning algorithm and prove its convergence to an equilibrium in the game that we propose to use.

Chapter 7 propose a game design for distributed optimization where the optimization problem has coupled constraints. The novelty of our approach stems from integrating classical optimization techniques, in

particular exterior penalty methods and barrier function methods, into the design of the agents' objective functions.

Chapter 8 proposes a game design for addressing distributed optimization problems with a time-varying communication graph. The key enabler for this result is that the resulting game possesses a property which is invariant to the structure of the communication graph.

Part I

Distributed Energy Management in Power Systems

Chapter 2

Demand Response Using Utility Maximization

[] Demand side management will be a key component of the future smart grid that can help reduce peak load and adapt elastic demand to fluctuating generations. We study an abstract market model where a set of households are served by a single load-serving entity (LSE). Each household operates different appliances including air conditioners, washers, lighting, electric vehicles, batteries, etc, each of which provides a certain benefit depending on the pattern or volume of power it consumes. Based on utility maximization, we proposed a dynamic pricing scheme and a distributed approach for the LSE to coordinate users' demand response to benefit the overall system, including reducing the peak load, smoothing the entire demand profile, and saving significant generation costs.

2.1 Introduction

There is a large literature on various forms of load side management from the classical direct load control to the more recent real-time pricing [1, 2]. Direct load control in particular has been practised for a long time and optimization methods have been proposed to minimize generation cost, e.g., [3–6], maximize utility's profit, e.g., [7], or minimize deviation from users' desired consumptions, e.g., [8, 9], sometimes integrated with unit commitment and economic dispatch e.g. [4, 10]. Almost all demand response programs today target large industrial or commercial users, or, in the case of residential users, a small number of them, for two, among other, important reasons. First, demand side management is invoked rarely, mostly to cope with a large correlated demand spike due to weather or a supply shortfall due to faults, e.g., during the few hottest

days in summer. Second, the lack of ubiquitous two-way communication in the current infrastructure prevents the participation of a large number of diverse users with heterogeneous and time-varying consumption requirements. Both reasons favor a simple and static mechanism involving a few large users that is sufficient to deal with the occasional need for load control, but both reasons are changing.

Renewable sources can fluctuate rapidly and by large amounts. As their penetration continues to grow, the need for regulation services and operating reserves will increase, e.g., [11, 12]. This can be provided by additional peaker units, at a higher cost, or supplemented by real-time demand response [12–16]. We believe that demand response will not only be invoked to shave peaks and shift load for economic benefits, but will increasingly be called upon to improve security and reduce reserves by adapting elastic loads to intermittent and random renewable generation [17]. Indeed, the authors of [12, 18, 19] advocate the creation of a distribution/retail market to encourage greater load side participation as an alternative source for fast reserves. Such an application, however, will require a much faster and more dynamic demand response than practiced today. This will be enabled in the coming decades by the large-scale deployment of a sensing, control, and two-way communication infrastructure, including the flexible AC transmission systems, the GPS-synchronized phasor measurement units, and the advanced metering infrastructure, which is currently underway around the world [20].

Demand response in such a context must allow the participation of a large number of users, and be dynamic and distributed. Dynamic adaptation by hundreds of millions of end users on a sub-second control timescale, each contributing a tiny fraction of the overall traffic, is being practiced everyday on the Internet in the form of congestion control. Even though both the grid and the Internet are massive distributed nonlinear feedback control systems, there are important differences in their engineering, economic, and regulatory structures. Nonetheless the precedence of the Internet lends hope to a much bigger scale and more dynamic and distributed demand response architecture and its benefit to grid operation. Our goal is to design algorithms for such a system.

2.1.1 Summary

Specifically, in this chapter we consider a demand response problem where a set of households are served by a single load-serving entity (LSE). The LSE may represent a regulated monopoly like most utility companies in the United States today, or a non-profit cooperative that serves a community of end users. Its purpose is (possibly regulated) to promote the overall system welfare. The LSE purchases electricity on the wholesale electricity markets (e.g., day-ahead, real-time balancing, and ancillary services) and sells it on the retail market to end users. It provides two important values: it aggregates loads so that the wholesale markets can operate efficiently, and it hides the complexity and uncertainty from the users, in terms of both power reliability and prices.

We will consider households that operate different appliances including PHEVs and batteries and propose a demand response approach based on utility maximization. Each appliance provides a certain benefit depending on the pattern or volume of power it consumes. Each household wishes to optimally schedule its power consumption so as to maximize its individual net benefit subject to various consumption and power flow constraints. We show that there exist time-varying prices that can align individual optimality with social optimality, i.e., under such prices, when the households selfishly optimize their own benefits, they automatically also maximize the social welfare. The LSE can thus use dynamic pricing to coordinate demand responses to the benefit of the overall system. We propose a distributed algorithm for the LSE and the customers to jointly compute this optimal prices and demand schedules. We also present simulation results that illustrate several interesting properties of the proposed scheme, as follows:

1. Different appliances are coordinated indirectly by real-time pricing, so as to flatten the total demand over different time-periods as much as possible.

2. Compared with no demand response or flat-price schemes, real-time pricing is very effective in shaping the demand: it not only greatly reduces the peak load, but also the variation in demand.

3. The integration of the battery helps reap more benefit from demand response: it not only reduces the peak load but further flattens the entire load profile and reduces the demand variation.

4. The real-time pricing scheme can increase the load factor greatly and save a large amount of generation

cost without hurting customers' utility; here again, the battery amplifies this benefit.

5. The cost of the battery (such as its lifetime in terms of charging/discharging cycles) is important: the benefit of demand response increases with lower battery cost.

6. As the number of the households increases, the benefit of our demand response increases but will eventually saturate.

2.1.2 Previous work

There exists a large literature on demand response, see, e.g., [9, 21–29]. We briefly discuss some papers that are directly relevant to our chapter. First there are papers on modeling specific appliances. For instance, [21] and [22] consider the electricity load control with thermal mass in buildings; [23] considers the coordination of charging PHEV with other electric appliances. Then, there are papers on the coordination among different appliances. [24] studies electricity usage for a typical household and proposes a method for customers to schedule their available distributed energy resources to maximize net benefits in a day-ahead market. [25] proposes a residential energy consumption scheduling framework which attempts to achieve a desired trade-off between minimizing the electricity payment and minimizing the waiting time for the operation of each appliance in household in presence of a real-time pricing tariff by doing price prediction based on prior knowledge. While in practice, for different appliances, the household may have a different objective rather than waiting time for the operation of the appliance.

Besides works such as [24, 25] which consider a single household demand response given a pricing scheme, [26] considers a power network where end customers choose their daily schedules of their household appliances/loads by playing games among themselves and the LSE tries to adopt adequate pricing tariffs that differentiate the energy usage in time and level to make the Nash equilibrium minimize the energy costs. However, they assume that customers have full knowledge of generation cost function and in their proposed algorithm they require customers to update their energy consumption scheduling asynchronously, both of which are hard to implement in practice. [27] considers a centralized complex-bid market-clearing mechanism where customers submit price-sensitive bids in the day-ahead market; they did not study the specific electricity consumptions model for the household.

Notations. We use $q_{i,a}(t)$ to denote the power demanded by customer i for appliance a at time t. Then, $q_{i,a} := (q_{i,a}(t), \forall t)$ denotes the vector of power demands over $t = 1, \ldots, T$; $q_i := (q_{i,a}, \forall a \in \mathcal{A}_i)$ denotes the vector of power demands for all appliances in the collection $\mathcal{A}_i$ of customer i; and $q := (q_i, \forall i)$ denotes the vector of power demands from all customers. Similar convention is used for other quantities such as battery charging schedules $r_i(t), r_i, r$.

2.2 System model

Consider a set N of households/customers that are served by a load service entity (LSE). The LSE participates in wholesale markets (day-ahead, real-time balancing, ancillary services) to purchase electricity from generators and then sell it to the N customers in the retail market. Even though wholesale prices can fluctuate rapidly by large amounts, currently most utility companies hide this complexity and volatility from their customers and offer electricity at a flat rate (fixed unit price), perhaps in multiple tiers based on a customer's consumption. Even though the wholesale prices are determined by (scheduled or real-time) demand and supply and by congestion in the transmission network (except for electricity provisioned through long-term bilateral contracts), the retail prices are set statically independent of the real-time load and congestion. Flat-rate pricing has the important advantage of being simple and predictable, but it does not encourage efficient use of electricity. In this chapter, we propose a way to use dynamic pricing in the retail market to coordinate the customers' demand responses to the benefit of individual customers and the overall system. We now present our model, describe how the utility should set their prices dynamically, how a customer should respond, and the properties of the resulting operating point.

We consider a discrete-time model with a finite horizon that models a day. Each day is divided into T timeslots of equal duration, indexed by $t \in \mathcal{T} := \{1, 2, \cdots, T\}$.

2.2.1 Load sevice entity

The LSE serves as an intermediary that participates in multiple wholesale markets, including day-ahead, real-time balancing and ancillary services, to provision enough electricity to meet the demands of the N

customers. The design of the retail prices needs to at least recover the running costs of the the LSE, including the payments it incurs in the various wholesale markets. It is an interesting subject that is beyond the scope of this chapter. For simplicity, we make the important assumption that this design can be summarized by a cost function $C(Q,t)$ that specifies the cost for the LSE to provide Q amount of power to the N customers at time t. The modeling of cost function is an active research issue [27, 29, 30]. Here we assume that the cost function $C(Q,t)$ is convex increasing in Q for each t. The LSE sets the prices $(p(t), t \in \mathcal{T})$ according to an algorithm described below.

2.2.2 Customers

Each customer $i \in N$ operates a set $\mathcal{A}_i$ of appliances such as air conditioner, refrigerator, plug-in hybrid electric vehicle (PHEV), etc. For each appliance $a \in \mathcal{A}_i$ of customer i, we denote by $q_{i,a}(t)$ its power draw at time $t \in \mathcal{T}$, and by $q_{i,a}$ the vector $(q_{i,a}(t), t \in \mathcal{T})$ of power draws over the whole day. An appliance a is characterized by two parameters:

- a utility function $U_{i,a}(q_{i,a})$ that quantifies the utility user i obtains when it consumes $q_{i,a}(t)$ power at each time $t \in \mathcal{T}$; and
- a set of linear inequalities $A^{i,a} q_{i,a} \leq \eta_{i,a}$ on the vector power $q_{i,a}$.

In Section 2.4, we will describe in detail how we model various appliances through appropriate matrices $A^{i,a}$ and vector $\eta_{i,a}$. Note that inelastic load, e.g., minimum refrigerator power, can be modeled by $q_{i,a}(t) \geq \underline{q}_{i,a}$, which says the appliance a of customer i requires a minimum power $\underline{q}_{i,a}$ at all times t. This is a linear inequality constraint and part of $A^{i,a} q_{i,a} \leq \eta_{i,a}$.

2.2.3 Energy storage

In addition to appliances, a customer i may also possess a battery which provides further flexibility for optimization of its consumption across time. We denote by B_i the battery capacity, by $b_i(t)$ the energy level of the battery at time t, and by $r_i(t)$ the power (energy per period) charged to (when $r_i(t) \geq 0$) or discharged from (when $r_i(t) < 0$) the battery at time t. Assume that battery power leakage is negligible. Then we model

the dynamics of the battery energy level by

$$b_i(t) \quad = \quad \sum_{\tau=1}^{t} r_i(\tau) + b_i(0) \ . \tag{2.1}$$

Battery usually has an upper bound on charge rate, denoted by r_i^{max} for customer i, and an upper bound on discharge rate, denoted by $-r_i^{min}$ for customer i. We thus have the following constraints on $b_i(t)$ and $r_i(t)$:

$$0 \leq b_i(t) \leq B_i, \qquad r_i^{min} \leq r_i(t) \leq r_i^{max} \ . \tag{2.2}$$

When the battery is discharged, the discharged power is used by other electric appliances of customer i. It is reasonable to assume that the battery cannot discharge more power than the appliances need, i.e., $-r_i(t) \leq \sum_{a\in\mathcal{A}_i} q_{i,a}(t)$. Moreover, in order to make sure that there is a certain amount of electric energy in the battery at beginning of the next day, we impose a minimum on the energy level at the end of control horizon: $b(T) \geq \gamma_i B_i$, where $\gamma_i \in (0, 1]$.

The cost of operating the battery is modeled by a function $D_i(r_i)$ that depends on the vector of charged/discharged power $r_i := (r_i(t), t \in \mathcal{T})$. This cost, for example, may correspond to the amortized purchase and maintenance cost of the battery over its lifetime, which depends on how fast/much/often it is charged and discharged. The cost function D_i is assumed to be a convex function of the vector r_i.

2.3 Equilibrium and distributed algorithm

2.3.1 Equilibrium

With the battery, at each time t the total power demand of customer i is

$$Q_i(t) \quad := \quad \sum_{a\in\mathcal{A}_i} q_{i,a}(t) + r_i(t) \ . \tag{2.3}$$

We assume that the LSE is regulated so that its objective is not to maximize its profit through selling electricity, but rather to induce customers' consumption in a way that maximizes the social welfare, total customer

utility minus the utility's cost of providing the electricity demanded by all the customers. Hence the LSE aims to solve:

Utility's objective (max welfare):

$$\max_{q,r} \quad \sum_i \left(\sum_{a \in A_i} U_{i,a}(q_{i,a}) - D_i(r_i) \right) - \sum_t C\left(\sum_i Q_i(t) \right) \tag{2.4}$$

$$\text{s. t.} \quad A^{i,a} q_{i,a} \leq \eta_{i,a}, \quad \forall a, i \tag{2.5}$$

$$0 \leq Q_i(t) \leq Q_i^{\max}, \quad \forall t, i \tag{2.6}$$

$$r_i \in \mathcal{R}_i, \quad \forall i \tag{2.7}$$

where $Q_i(t)$ is defined in (2.3), the inequality (2.5) models the various customer appliances (see Section 2.4 for details), the lower inequality of (2.6) says that customer i's battery cannot provide more power than the total amount consumed by all i's appliances, and the upper inequality of (2.6) imposes a bound on the total power drawn by customer i. The constraint (2.7) models the operation of customer i's battery with the feasible set $\mathcal{R}_i$ defined by: for all t, the vectors $r_i \in \mathcal{R}_i$ if and only if

$$0 \leq b_i(t) \leq B_i, \quad b_i(T) \geq \gamma_i B_i \tag{2.8}$$

$$r_i^{min} \leq r_i(t) \leq r_i^{max} \tag{2.9}$$

where $b_i(t)$ is defined in terms of $(r_i(\tau), \tau \leq t)$ in (2.1).

By assumption, the objective function is concave and the feasible set is convex, and hence an optimal point can in principle be computed centrally by the LSE. This, however, will require the LSE to know all the customer utility and cost functions and all the constraints, which is clearly impractical. The strategy is for the LSE to set prices $p := (p(t), t \in \mathcal{T})$ in order to induce the customers to individually choose the right consumptions and charging schedules (q_i, r_i) in response, as follows.

Given the price p, we assume that each customer i chooses the power demand and battery charging schedule $(q_i, r_i) := (q_{i,a}(t), r_i(t), \forall t, \forall a \in \mathcal{A}_i)$ so as to maximize its net benefit, the total utility from operating appliances a at power levels $q_{i,a}$ minus the cost of battery operation and electricity; i.e., each

customer i solves:

Customer i's objective (max own benefit):

$$\begin{aligned} \max_{q_i, r_i} \quad & \sum_{a \in \mathcal{A}_i} U_{i,a}(q_{i,a}) - D_i(r_i) - \sum_t p(t) Q_i(t) \\ \text{s. t.} \quad & (2.5) - (2.7) \ . \end{aligned} \tag{2.10}$$

Note that an optimal solution of customers i depends on the prices $p := (p(t), t \in \mathcal{T})$ set by the LSE. We denote it by $(q_i(p), r_i(p)) := (q_{i,a}(t;p), r_i(t;p), \forall t, \forall a \in \mathcal{A}_i)$; similarly, we denote an optimal total power by $Q_i(p) := (Q_i(t;p))$ defined as in (2.3) but with optimal $q_{i,a}(p)$ and $r_i(p)$.

Definition 2.1. *The prices p and the customer demands $(q, r) := (q_i, r_i, \forall i)$ are in equilibrium if $(q, r) = (q(p), r(p))$, i.e., a solution $(q_i(p), r_i(p))$ to (2.10) with prices p that is optimal to each customer i is also optimal to the utility company, i.e., maximizes the welfare (2.4).*

The following result follows from the welfare theorem. It implies that setting the price to be the marginal cost of power is optimal.

Theorem 2.1. *There exists an equilibrium p^* and $(q_i^*, r_i^*, \forall i)$. Moreover, $p^*(t) = C'(\sum_i Q_i^*(t)) \geq 0$ for each time t.*

Proof. Write the LSE's problem as

$$\begin{aligned} \max_{(q,r) \in X} \quad & \sum_i V_i(q_i, r_i) - \sum_t C\left(\sum_i Q_i(t)\right) \\ \text{s. t.} \quad & Q_i(t) = \sum_{a \in \mathcal{A}_i} q_{i,a}(t) + r_i(t), \quad \forall i, t \end{aligned}$$

where $V_i(q_i, r_i) := \sum_{a \in A_i} U_{i,a}(q_{i,a}) - D_i(r_i)$ and the feasible set X is defined by the constraints (2.5)–(2.9). Clearly, an optimal solution (q^*, r^*) exists. Moreover, there exist Lagrange multipliers $p_i^*(t)$, $\forall i, t$, such that (taking derivative with respect to $Q_i(t)$)

$$p_i^*(t) \quad = \quad C'\left(\sum_i Q_i^*(t)\right) \ \geq \ 0$$

Since the right-hand side is independent of i, the utility company can set the prices as $p^*(t) := p_i^*(t) \geq 0$ for all i. One can check that the KKT condition for the utility's problem are identical to the KKT conditions for the collection of customers' problems. Since both the utility's problem and all the customers' problems are convex, the KKT conditions are both necessary and sufficient for optimality. This proves the theorem. □

2.3.2 Distributed algorithm

Theorem 2.1 motivates a distributed algorithm where the LSE and the customers jointly compute an equilibrium based on a gradient algorithm, where the LSE sets the prices to be the marginal costs of electricity and each customer solves its own maximization problem in response. The model is that at the beginning of each day, the utility company and (the automated control agents of) the customers iteratively compute the electricity prices $p(t)$, consumptions $q_i(t)$, and charging schedules $r_i(t)$, for each period t of the day, in advance. These decisions are then carried out for that day.

At k-th iteration:

- The LSE collects forecasts of total demands $(Q_i(t),\ \forall t)$ from all customers i over a communication network. It sets the prices to the marginal cost

$$p^k(t) = C'\left(\sum_i Q_i^k(t)\right) \tag{2.11}$$

  and broadcasts $(p^k(t), \forall t)$ to all customers over the communication network.

- Each customer i updates its demand q_i^k and charging schedule r_i^k after receiving the updated p^k, according to

$$\begin{aligned} \bar{q}_{i,a}^{k+1}(t) &= q_{i,a}^k(t) + \gamma\left(\frac{\partial U_{i,a}\left(q_i^k\right)}{\partial q_{i,a}^k(t)} - p^k(t)\right) \\ \bar{r}_i^{k+1}(t) &= r_i^k(t) - \gamma\left(\frac{\partial D_i(r_i^k)}{\partial r_i^k(t)} + p^k(t)\right) \\ (q_i^{k+1}, r_i^{k+1}) &= \left[\bar{q}_i^{k+1}, \bar{r}_i^{k+1}\right]^{S_i} \end{aligned} \tag{2.12}$$

  where $\gamma > 0$ is a constant stepsize, and $[\cdot]^{S_i}$ denotes projection onto the set S_i specified by constraints (2.5)-(2.7).

When γ is small enough, the above algorithm converges [31].

2.4 Detailed appliance models

In this section, we describe detailed models of electric appliances commonly found in a household. We separate these appliances into four types, each type characterized by a utility function $U_{i,a}(q_{i,a})$ that models how much customer i values the consumption vector $q_{i,a}$, and a set of constraints on the consumption vector $q_{i,a}$. The description in this section elaborates on the utility functions $U_{i,a}(q_{i,a})$ and the constraint $A^{i,a}q_{i,a} \leq \eta_{i,a}$ in the optimization problems defined in Section 2.3.

2.4.1 Type 1

The first type includes appliances such as air conditioners and refrigerators which control the temperature of customer i's environment.

We denote by $\mathcal{A}_{i,1}$ the set of Type 1 appliances for customer i. For each appliance $a \in \mathcal{A}_{i,1}$, $T^{in}_{i,a}(t)$ and $T^{out}_{i,a}(t)$ denote the temperatures at time t inside and outside the place that the appliance is in charge of, and $\mathcal{T}_{i,a}$ denotes the set of timeslots during which customer i actually cares about the temperature. For instance, for air conditioners, $T^{in}_{i,a}(t)$ is the temperature inside the house, $T^{out}_{i,a}(t)$ is the temperature outside the house, and $\mathcal{T}_{i,a}$ is the set of timeslots when the resident is at home.

Assume that, at each time $t \in \mathcal{T}_{i,a}$, customer i attains a utility $U_{i,a}(T_{i,a}) := U_{i,a}(T^{in}_{i,a}(t), T^{comf}_{i,a})$ when the temperature is $T^{in}_{i,a}(t)$. The utility function is parameterized by a constant $T^{comf}_{i,a}$ which represents the most comfortable temperature for the customer. We assume that $U_{i,a}(T^{in}_{i,a}(t))$ is a continuously differentiable, concave function of $T^{in}_{i,a}(t)$.

The inside temperature evolves according to the following linear dynamics:

$$T^{in}_{i,a}(t) = T^{in}_{i,a}(t-1) + \alpha(T^{out}_{i,a}(t) - T^{in}_{i,a}(t-1)) + \beta q_{i,a}(t) \tag{2.13}$$

where α and β are parameters that specify the thermal characteristics of the appliance and the environment in which it operates. The second term in equation (2.13) models heat transfer. The third term models the

thermal efficiency of the system: $\beta > 0$ if appliance a is a heater and $\beta < 0$ if it is a cooler. Here, we define $T^{in}_{i,a}(0)$ as the temperature $T^{in}_{i,a}(T)$ from the previous day. This formulation models the fact that the current temperature depends on the current power draw as well as the temperature in the previous timeslot. Thus the current power consumption has an effect on future temperatures [9,21,22]. For each customer i and each appliance $a \in \mathcal{A}_{i,1}$, there is a range of temperature that customer i takes as comfortable, denoted by $[T^{comf,min}_{i,a},\ T^{comf,max}_{i,a}]$. Thus we have the following constraint

$$T^{comf,min}_{i,a} \ \leq \ T^{in}_{i,a}(t) \ \leq \ T^{comf,max}_{i,a},\ \forall t \in \mathcal{T}_{i,a} \ . \tag{2.14}$$

We now express the constraints and the argument to the utility functions in terms of the load vector $q_{i,a} := (q_{i,a}(t), \forall t)$. Using equation (2.13), we can write $T^{in}_{i,a}(t)$ in terms of $(q_{i,a}(\tau), \tau = 1, \ldots, t)$:

$$T^{in}_{i,a}(t) \quad = \quad (1-\alpha)^t T^{in}_{i,a}(0) + \sum_{\tau=1}^{t} (1-\alpha)^{t-\tau} \alpha T^{out}_{i,a}(\tau) + \sum_{\tau=1}^{t} (1-\alpha)^{t-\tau} \beta q_{i,a}(\tau) \ .$$

Define $T^t_{i,a} := (1-\alpha)^t T^{in}_{i,a}(0) + \sum_{\tau=1}^{t} (1-\alpha)^{t-\tau} \alpha T^{out}_{i,a}(\tau)$.[1] We can further write $T^{in}_{i,a}(t)$ as

$$T^{in}_{i,a}(t) = T^t_{i,a} + \sum_{\tau=1}^{t} (1-\alpha)^{t-\tau} \beta q_{i,a}(\tau) \ . \tag{2.15}$$

With equation (2.15), the constraint (2.14) becomes a linear constraint on the load vector $q_{i,a} := (q_{i,a}(t), \forall t)$: for any $t \in \mathcal{T}_{i,a}$,

$$T^{comf,min}_{i,a} \leq T^t_{i,a} + \sum_{\tau=1}^{t} (1-\alpha)^{t-\tau} \beta q_{i,a}(\tau) \leq T^{comf,max}_{i,a} \ . \tag{2.16}$$

The overall utility $U_{i,a}(q_{i,a})$ in the form used in (2.4) and (2.10) can then be written in terms of $U_{i,a}(T^{in}_{i,a}(t), T^{comf}_{i,a})$

[1] $T^t_{i,a}$ represents the temperature at time t if the appliance a doesn't exist. It is determined by outside temperature and not controlled by the customer.

as[2]

$$U_{i,a}(q_{i,a}) := \sum_{t \in \mathcal{T}_{i,a}} U_{i,a}\left(T_{i,a}^{t} + \sum_{\tau=1}^{t} (1-\alpha)^{t-\tau} \beta q_{i,a}(\tau), T_{i,a}^{comf}\right) \tag{2.17}$$

which is a concave function of the vector $q_{i,a}$ since $U_{i,a}(T_{i,a}^{in}(t), T_{i,a}^{comf})$ is concave in $T_{i,a}^{in}(t)$.

In addition, there is a maximum power $q_{i,a}^{max}(t)$ that the appliance can bear at each time, thus we have another constraint on the $q_{i,a}$:

$$0 \leq q_{i,a}(t) \leq q_{i,a}^{max}(t), \ \forall t \ .$$

2.4.2 Type 2

The second category includes appliances such as PHEV, dish washer, and washing machine. For these appliances, a customer only cares about whether the task is completed before a certain time. This means that the cumulative power consumption by such an appliance must exceed a threshold by the deadline [23–25].

We denote $\mathcal{A}_{i,2}$ as the set of Type 2 appliances. For each $a \in \mathcal{A}_{i,2}$, $\mathcal{T}_{i,a}$ is the set of times that the appliance can work. For instance, for PHEV, $\mathcal{T}_{i,a}$ is the set of times that the vehicle can be charged. For each customer i and $a \in \mathcal{A}_{i,2}$, we have the following constraints on the load vector $q_{i,a}$:

$$\begin{gathered} q_{i,a}^{min}(t) \ \leq \ q_{i,a}(t) \ \leq \ q_{i,a}^{max}(t), \ \ \forall t \in \mathcal{T}_{i,a}, \\ q_{i,a}(t) \ = \ 0, \ \ \forall t \in \mathcal{T} \backslash \mathcal{T}_{i,a} \\ \bar{Q}_{i,a}^{min} \ \leq \ \textstyle\sum_{t \in \mathcal{T}_{i,a}} q_{i,a}(t) \ \leq \ \bar{Q}_{i,a}^{max} \end{gathered}$$

where $q_{i,a}^{min}(t)$ and $q_{i,a}^{max}(t)$ are the minimum and maximum power load that the appliance can consume at time t, and $\bar{Q}_{i,a}^{min}$ and $\bar{Q}_{i,a}^{max}$ are the minimum and maximum total power draw that the appliance requires. If

[2] We abuse notation to use $U_{i,a}$ to denote two different functions; the meaning should be clear from the context.

we set $q_{i,a}^{min}(t) = q_{i,a}^{max}(t) = 0$ for $t \in \mathcal{T} \backslash \mathcal{T}_{i,a}$, we can rewrite these constraints as

$$\begin{aligned} & q_{i,a}^{min}(t) \leq q_{i,a}(t) \leq q_{i,a}^{max}(t), \ \forall t \in \mathcal{T} \\ & \bar{Q}_{i,a}^{min} \leq \textstyle\sum_{t \in \mathcal{T}_{i,a}} q_{i,a}(t) \leq \bar{Q}_{i,a}^{max} \ . \end{aligned} \tag{2.18}$$

The overall utility that customer i obtains from a Type-2 appliance a depends on the total power consumption by a over the whole day. Hence the utility function in the form used in Section 2.3 is: $U_{i,a}(q_{i,a}) := U_{i,a}\left(\sum_t q_{i,a}(t)\right)$. We assume that the utility function is a continuously differentiable, concave function of $\sum_t q_{i,a}(t)$.

2.4.3 Type 3

The third category includes appliances such as lighting that must be on for a certain period of time. A customer cares about how much light they can get at each time t. We denote by $\mathcal{A}_{i,3}$ the set of Type-3 appliances and by $\mathcal{T}_{i,a}$ the set of times that the appliance should work. For each customer i and $a \in \mathcal{A}_{i,3}$, we have the following constraints on the load vector $q_{i,a}$:

$$q_{i,a}^{min}(t) \leq q_{i,a}(t) \leq q_{i,a}^{max}(t), \ \forall t \in \mathcal{T}_{i,a}. \tag{2.19}$$

At each time $t \in \mathcal{T}_{i,a}$, we assume that customer i attains a utility $U_{i,a}(q_{i,a}(t), t)$ from consuming power $q_{i,a}(t)$ on appliance a. The overall utility is then $U_{i,a}(q_{i,a}) := \sum_t U_{i,a}(q_{i,a}(t), t)$. Again, we assume $U_{i,a}$ is a continuously differentiable, concave function.

2.4.4 Type 4

The fourth category includes appliances such as TV, video games, and computers that a customer uses for entertainment. For those appliances, the customer cares about two things: how much power they use at each time they want to use the appliance, and how much total power they consume over the entire day.

We denote by $\mathcal{A}_{i,4}$ the set of Type-4 appliances and by $\mathcal{T}_{i,a}$ the set of times that customer i can use the appliance. For instance, for TV, $\mathcal{T}_{i,a}$ is the set of times that the customer is able to watch TV. For each

customer i and $a \in \mathcal{A}_{i,4}$, we have the following constraints on the load vector $q_{i,a}$:

$$\begin{aligned} & q_{i,a}^{min}(t) \leq q_{i,a}(t) \leq q_{i,a}^{max}(t),\ \forall t \in \mathcal{T}_{i,a} \\ & \bar{Q}_{i,a}^{min} \leq \textstyle\sum_{t \in \mathcal{T}_{i,a}} q_{i,a}(t) \leq \bar{Q}_{i,a}^{max} \end{aligned} \tag{2.20}$$

where $q_{i,a}^{min}(t)$ and $q_{i,a}^{max}(t)$ are the minimum and maximum power that the appliance can consume at each time t; $\bar{Q}_{i,a}^{min}$ and $\bar{Q}_{i,a}^{max}$ are the minimum and maximum total power that the customer demands for the appliance. For example, a customer may have a favorite TV program that he wants to watch everyday. With DVR, the customer can watch this program at any time. However the total power demand from TV should at least be able to cover the favorite program.

Assume that customer i attains a utility $U_{i,a}(q_{i,a}(t), t)$ from consuming power $q_{i,a}(t)$ on appliance $a \in \mathcal{A}_{i,4}$ at time t. The time dependent utility function models the fact that the resident would get different benefits from consuming the same amount of power at different times. Take watching the favorite TV program as an example. Though the resident is able to watch it at any time, he may enjoy the program at different levels at different times.

2.5 Numerical Experiments

In this section, we provide numerical examples to complement the analysis in the previous sections.

2.5.1 Simulation setup

We consider a simple system with 8 households in one neighborhood that join in the demand response system. The households are divided into two types evenly. For the households of the first type (indexed by $i = 1, 2, 3, 4$), there are residents staying at home for the whole day; for the households of the second type (indexed by $i = 5, 6, 7, 8$), there is no person staying at home during the day time (8am-6pm). A day starts at 8am, i.e., $t \in \mathcal{T}$ corresponds to the hour $[7 + t \pmod{24}, 8 + t \pmod{24}]$. Each household is assumed to have 6 appliances: air conditioner, PHEV, washing machine, lighting, entertainment,[3] and electric battery.

[3]Here we aggregate different entertainment devices such as TV and PC effectively as one "entertainment" device.

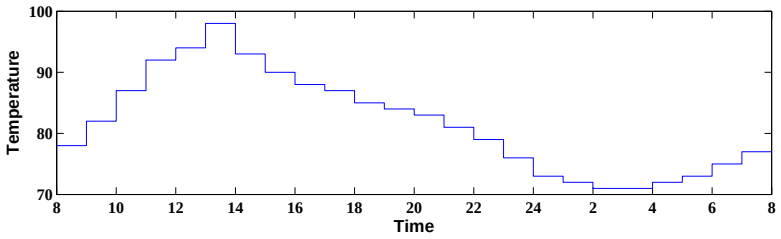


Figure 2.1. Outside Temperature over a day.

The basic parameters of each appliance used in simulation are shown as follows.

1. Air conditioner: This appliance belongs to Type 1. The outside temperature is shown in Figure 2.1. It captures a typical summer day in Southern California. For each resident, we assume that the comfortable temperature range is $[70\text{F}, 79\text{F}]$, and the most comfortable temperature is randomly chosen from $[73\text{F}, 77\text{F}]$. The thermal parameters $\alpha = 0.9$ and β is chosen randomly from $[-0.011, -0.008]$. For each household's air conditioner, we assume that $q^{max} = 4000\text{wh}$ and $q^{min} = 0\text{wh}$, and the utility function takes the form of $U_{i,a}(T_i(t)) := c_{i,a} - b_{i,a}(T_{i,a}(t) - T_{i,a}^{com})^2$, where $b_{i,a}$ and $c_{i,a}$ are positive constants. We further assume that the residents will turn off the air conditioner when they go to sleep.[4] The households of the first type care about the inside temperature through the whole day, and the other households care about the inside temperature during the time $\mathcal{T}_{i,a} = \{18, \cdots, 24, 1, \cdots, 7\}$.

2. PHEV: This appliance belongs to Type 2. We assume that the available charging time, $\mathcal{T}_{i,a} = \{18, \cdots, 24, 1, \cdots, 7\}$, is the same for all houses. The storage capacity is chosen randomly from $[5500\text{wh}, 6000\text{wh}]$; and the minimum total charging requirement is chosen randomly from $[4800\text{wh}, 5100\text{wh}]$. The minimum and maximum charging rates are 0w and 2000w. The utility function takes the form of $U_{i,a}(Q) = b_{i,a}Q + c_{i,a}$, where $b_{i,a}$ and $c_{i,a}$ are positive constants.

3. Washing machine: This appliance belongs to Type 2. For the households of the first type, the available working time is the whole day; for the other households, the available working time is $\mathcal{T}_{i,a} = \{18, \cdots, 24, 1, \cdots, 7\}$. The minimum and maximum total power demands are chosen from $[1400\text{wh}, 1600\text{wh}]$ and $[2000\text{wh}, 2500\text{wh}]$ respectively. The minimum and maximum working rate are 0w and 1500w re-

[4]Notice that the outside temperature during 23pm-8am in Southern California is comfortable. It is common that customers turn off the air conditioner during the mid-night.

spectively. The utility function takes the form of $U_{i,a}(Q) = Q + c_{i,a}$, where $c_{i,a}$ is a positive constant.

4. Lighting: This appliance belongs to Type 3. $\mathcal{T}_{i,a} = \{18, \cdots, 23\}$, and the minimum and maximum working power requirements are 200w and 800w respectively. The utility function takes the form of $U_{i,a}(q_{i,a}(t)) = c_{i,a} - (b_{i,a} + \frac{q_{i,a}(t)}{\bar{q}})^{-1.5}$, where $b_{i,a}$ and $c_{i,a}$ are positive constants.

5. Entertainment: This appliance belongs to Type 4. For the households of the first type, $\mathcal{T}_{i,a} = \{12, \cdots, 23\}$, $Q_i^{max} = 3500$wh, and $Q_i^{min} = 1200$wh; for the other households, $\mathcal{T}_{i,a} = \{18, \cdots, 24\}$, $Q_i^{max} = 2000$wh, and $Q_i^{min} = 500$wh. The minimum and maximum working rate are 0w and 400w respectively. The utility function takes the form of $U_{i,a}(q_{i,a}(t)) = c_{i,a} - (b_{i,a} + \frac{q_{i,a}(t)}{\bar{q}})^{-1.5}$, where $b_{i,a}$ and $c_{i,a}$ are positive constants.

6. Battery: The storage capacity is chosen randomly from $[5500\text{wh}, 6500\text{wh}]$ and the maximum charging/discharging rates are both 1800w. We set $\gamma_i = 0.5$, and the cost function takes the following form:

$$D_i\left(r_i\right) = \left(\eta_1 \sum_{t\in\mathcal{T}} (r_i(t))^2 - \eta_2 \sum_{t=1}^{T-1} r_i(t) r_{i(t+1)} + \eta_3 \sum_{t\in\mathcal{T}} \left(\min(b_i(t) - \delta B_i, 0)\right)^2 + c_{i,b}\right)$$

where η_1, η_2, η_3, δ and $c_{i,b}$ are positive constants. The first term captures the damaging effect of fast charging and discharging; the second term penalizes charging/discharging cycles;[5] the third term captures the fact that deep discharge can damage the battery. We set $\delta = 0.2$.[6]

On the supply side, we assume that the electricity cost function is a smooth piecewise quadratic function [32], i.e.,

$$C(Q) = \begin{cases} c_1 Q^2 + b_1 Q + a_1; & 0 \le Q \le Q_1 \\ c_2 Q^2 + b_2 Q + a_2; & Q_1 < Q \le Q_2 \\ \vdots & \vdots \\ c_m Q^2 + b_m Q + a_m; & Q_{m-1} < Q \end{cases}$$

[5] If $r(t)$ and $r(t+1)$ have different signs, then there will be a cost. As long as η_2 is smaller than η_1, the cost function is a positive convex function. The second item can also be seen as a correction term to the first term.

[6] We assume that the batteries are lead-acid type batteries rather than NiCd batteries.

where $c_m > c_{m-1} > \ldots \geq c_1 > 0$.

2.5.2 Real-time pricing demand response

Let us first see the performance of our proposed demand response scheme with real-time pricing, without and with battery.

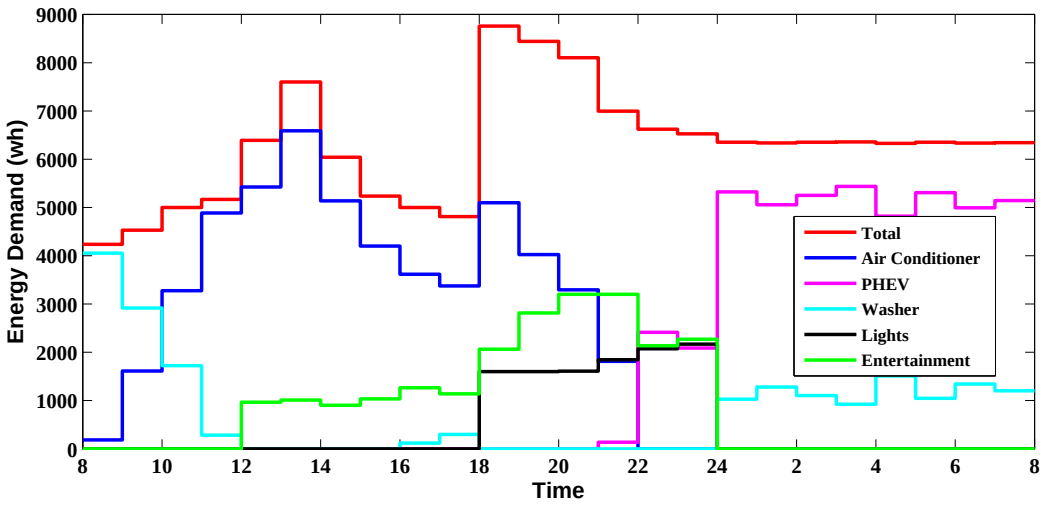


Figure 2.2. Total electricity demand under the real-time pricing demand response scheme without battery.

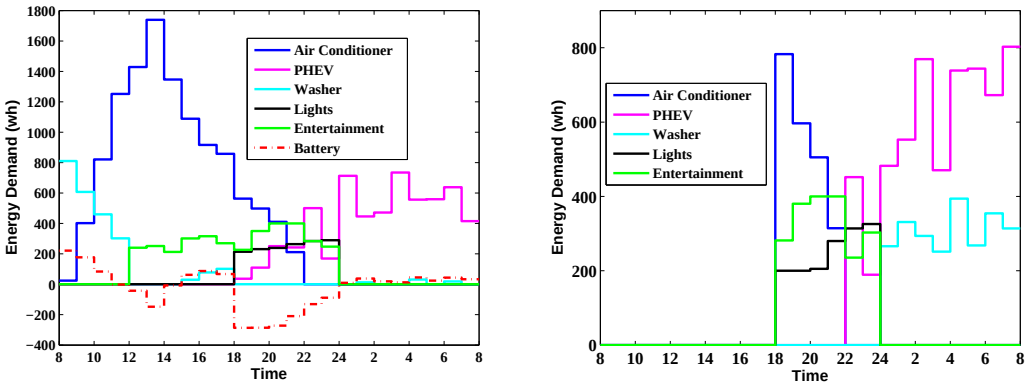


Figure 2.3. Electricity demand response for two typical households of different types without battery. The left panel shows the electric energy allocation for the household of the first type. The right panel shows the electric energy allocation for the household of the second type.

Figure 2.2 shows the total electricity demand under the real-time pricing demand response scheme without battery; Figure 2.3 shows the corresponding electricity allocation for two typical households of different types. We see that different appliances are coordinated indirectly by real-time pricing, so as to flatten the

total power demand at different times as much as possible.

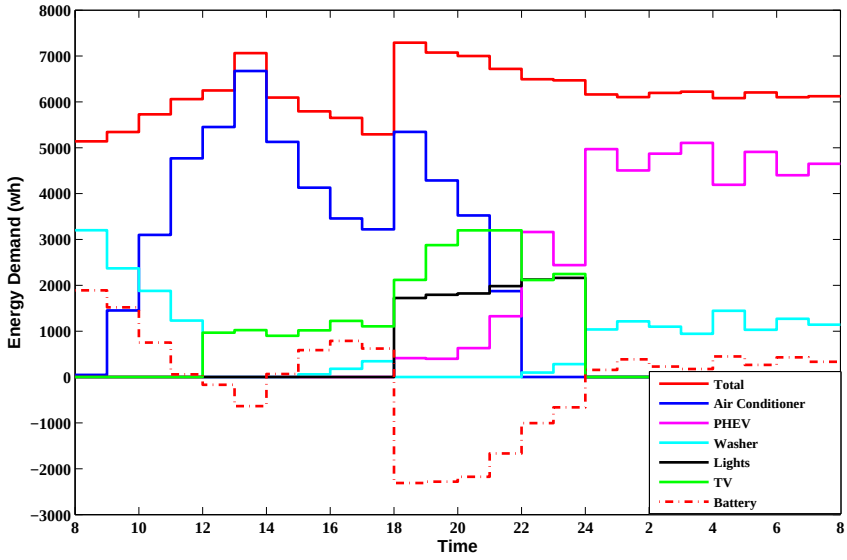


Figure 2.4. Total electricity demand under the real-time pricing demand response scheme with battery.

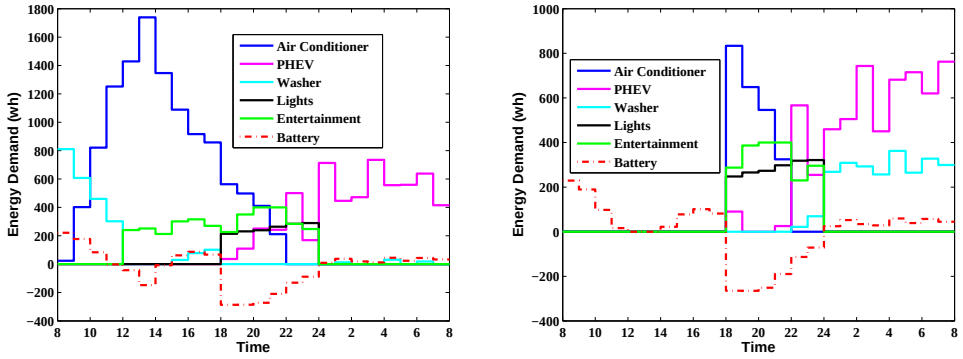


Figure 2.5. Electricity demand response for two typical households of different types with battery. The left panel shows the electric energy allocation for the household of the first type. The right panel shows the electric energy allocation for the household of the second type.

Figure 2.4 shows the total electricity demand under the real-time pricing demand response scheme with battery; Figure 2.5 shows the corresponding electricity allocation for two typical households of different types. Those figures show the value of the battery for demand response: it not only reduces the peak load but also helps to further flatten the total power demand at different times.

Figure 2.6 shows room temperature for two typical households of different types under the real-time pricing demand response scheme, without and with battery. We can see that the temperatures are around

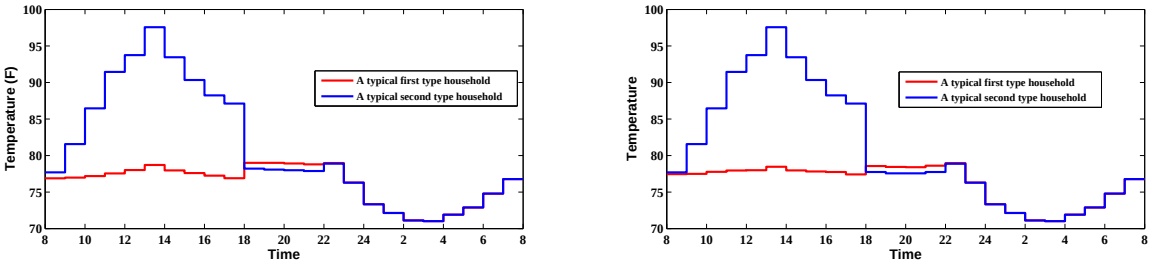


Figure 2.6. Room Temperature for two households of different types: the left panel shows the room temperature for the households with real-time pricing demand response without battery; the right panel shows the room temperature for the hoseholds with real-time pricing demand response with battery.

the comfortable temperature in both cases. The battery is able to keep the temperature closer to the most comfortable temperature.

2.5.3 Comparisons among different demand response schemes

In order to evaluate the performance of our proposed demand response scheme, we consider three other schemes. In the first scheme the customer is not responsive to any price or cost, but just wants to live a comfortable lifestyle; in the second and third ones, the customer responds to a certain flat price.

1. **No demand response:** The customers just allocate their energy usage according to their own preference without paying any attention to the price, i.e., they just optimize their utility without caring about their payment. For example, the customer sets the air conditioner to keep the temperature to the most comfortable level all the time; charges PHEV, washes clothes and watches TV at the favorite times. The electricity demand over a day under this scheme is shown by the blue plot in Figure 2.7.

2. **Flat price scheme 1:** In this scheme, the customer is charged a flat price p, such that

$$p = \frac{(1+\Delta)\sum_{t\in\mathcal{T}} C(Q(t),t)}{\sum_{t\in\mathcal{T}} Q(t)}$$

with $\{Q(t)\}_{t\in\mathcal{T}}$ the best response to such a price from the customers. To find such a price, we run iterations between the LSE and customers. At each iteration $k = 1, 2, \cdots$, the LSE sets the price as $p_k = \frac{(1+\Delta)\sum_{t\in\mathcal{T}} C(Q_k(t),t)}{\sum_{t\in\mathcal{T}} Q_k(t)}$ and then the customers will shape their demand in response to such a flat

price. Eventually, p_k will converge to a fixed point, which is the flat price we need.[7] The electricity demand over a day under this scheme is shown by the magenta plot in Figure 2.7.

3. **Flat price scheme 2:** In this scheme we use the information obtained from our proposed real-time pricing demand response scheme to set a flat price p. We collect the price $\{p(t)\}_{t\in\mathcal{T}}$ and total power demand $\{Q(t)\}_{t\in\mathcal{T}}$ information under the real time pricing scheme and then set the flat price as $p = \frac{\sum_{t\in\mathcal{T}} p(t)Q(t)}{\sum_{t\in\mathcal{T}} Q(t)}$. The electricity demand over a day under this scheme is shown by the black plot in Figure 2.7.

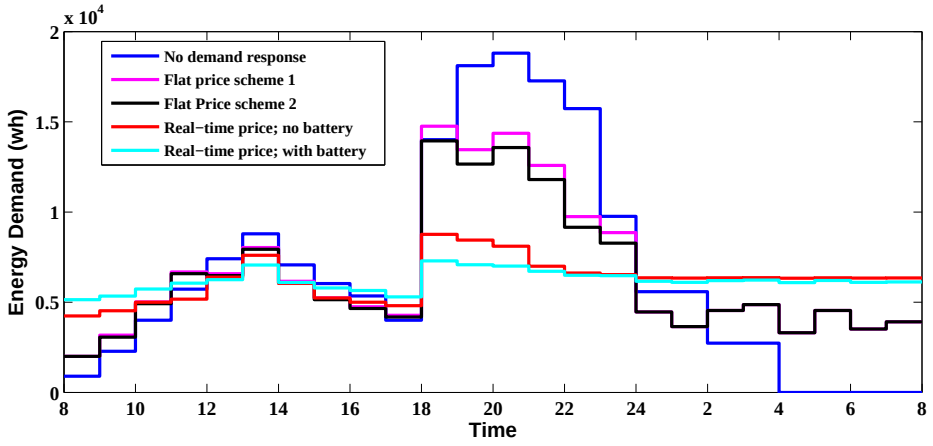


Figure 2.7. Electricity demand response under different schemes.

Figure 2.7 also shows the electricity demand response under the real-time pricing scheme with and without battery. We see that the real-time pricing demand response scheme is very effective in shaping the demand: not only is the peak load reduced greatly, but also the variation in power demand decreases greatly; with the integration of the battery, the peak load and the variation in power demand will be reduced further.

Table 2.1 summarizes the differences among the three pricing schemes. We see that the real-time pricing scheme can increase the load factor greatly and save a large amount of generation cost without hurting customers' utility. The integration of the battery can further increase the load factor and reap larger savings in generation cost.

[7] In general, such a price may not exist and the iterative procedure described may not converge.

Table 2.1. Demand response without Battery.

	No Demand Response	Flat Pricing (Scheme 1)	Flat Pricing (Scheme 2)	Real-time Pricing; no Battery	Real-Time Pricing; with Battery
Load Factor	0.3587	0.4495	0.4577	0.7146	0.8496
Peak Demand	18.8 kwh	14.7 kwh	13 kwh	8.76 kwh	7.29 kwh
Total Demand	162 kwh	158 kwh	153 kwh	150 kwh	148 kwh
Generation Cost	\$64.41	\$45.49	\$41.80	\$32.82	\$31.50
Total Payment	\$137.40[a]	\$ 54.59	\$58.56	\$57.42	\$55.69
Customers' Utility	\$212.41	\$201.72	\$200.14	\$198.82	\$198.82[b]
Customers' Net Utility[c]	\$75.01	\$147.14	\$141.57	\$141.40	\$143.13
Social Welfare	\$148.00	\$156.24	\$158.33	\$166.00	\$167.32

[a]The price at each time slot is set as the real-time marginal generation cost.
[b]When there is a battery, a customer's utility is defined as the benefits the customer gets from electric appliances minus the battery cost.
[c]Customers' net utility is defined as customers' utility minus payment.

2.5.4 Battery with different cost

One of the challenges in the integration of the battery is its economic (in)viability because of high battery cost. In order to study the impact of battery cost on demand response, we consider three scenarios with high, mild, and low cost, by choosing different scaling factors (10, 1 and 0.1) for the battery cost in the objective function.

Figure 2.8 shows the electricity demand under the real-time pricing scheme with batteries of different costs.

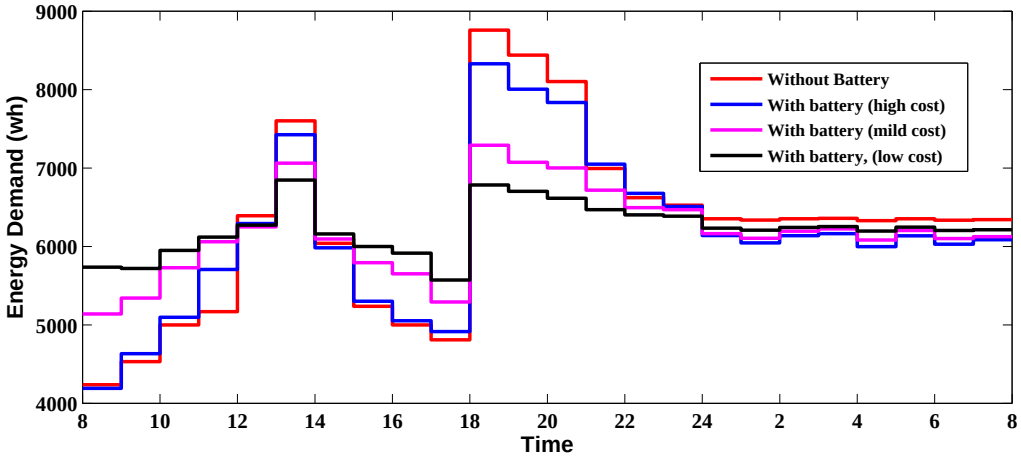


Figure 2.8. Electricity demand response with battery at different costs.

Table 2.2 summarizes the differences among those different scenarios. We see that the economic viability of the battery is important, and the more economically viable battery will reap more benefits from demand

Table 2.2. Demand response with Battery.

	No Battery	Battery (high-cost)	Battery (mild-cost)	Battery (low-cost)
Load Factor	0.7146	0.7390	0.8496	0.9095
Peak Demand	8.76 kwh	8.33 kwh	7.29 kwh	6.84 kwh
Total Demand	150 kwh	148 kwh	148 kwh	149 kwh
Generation Cost	\$32.82	\$31.72	\$31.50	\$31.70
Total Payment	\$57.42	\$56.35	\$55.69	\$55.99
Customers' Utility[a]	\$198.82	\$198.55	\$198.82	\$199.42
Customers' Net Utility[b]	\$141.40	\$142.92	\$143.13	\$143.43
Social Welfare	\$166.00	\$166.84	\$167.32	\$167.69

[a] A customer' utility is defined as the benefits the customer gets from electric appliances minus the battery cost.
[b] A customer' utility is defined as the customer's utility minus the payment.

response.

2.5.5 Performance scaling with different numbers of households

In order to study the effect of the system size on the performance of our demand response scheme, we simulate systems with the number of customers being $N = 2, 4, 6, \cdots, 24$. Figure 2.9 summarizes three interesting characteristic factors for the demand response systems with different numbers of households. We see that as the number of households increases, the load factor will first increase till a maximum value and then decrease a bit and finally level off, but the peak load and total demand at each household will decrease and finally level off. This shows that as the number of the households increases, our demand response scheme will reap more benefits but the gain will eventually saturate.

2.6 Conclusion

We have studied optimal demand response based on utility maximization in power networks. We consider households that operate different appliances including PHEVs and batteries and propose a demand response approach based on utility maximization. Each appliance provides a certain benefit depending on the pattern or volume of power it consumes. Each household wishes to optimally schedule its power consumption so as to maximize its individual net benefit subject to various consumption and power flow constraints. We show that there exist time-varying prices that can align individual optimality with social optimality, i.e., under

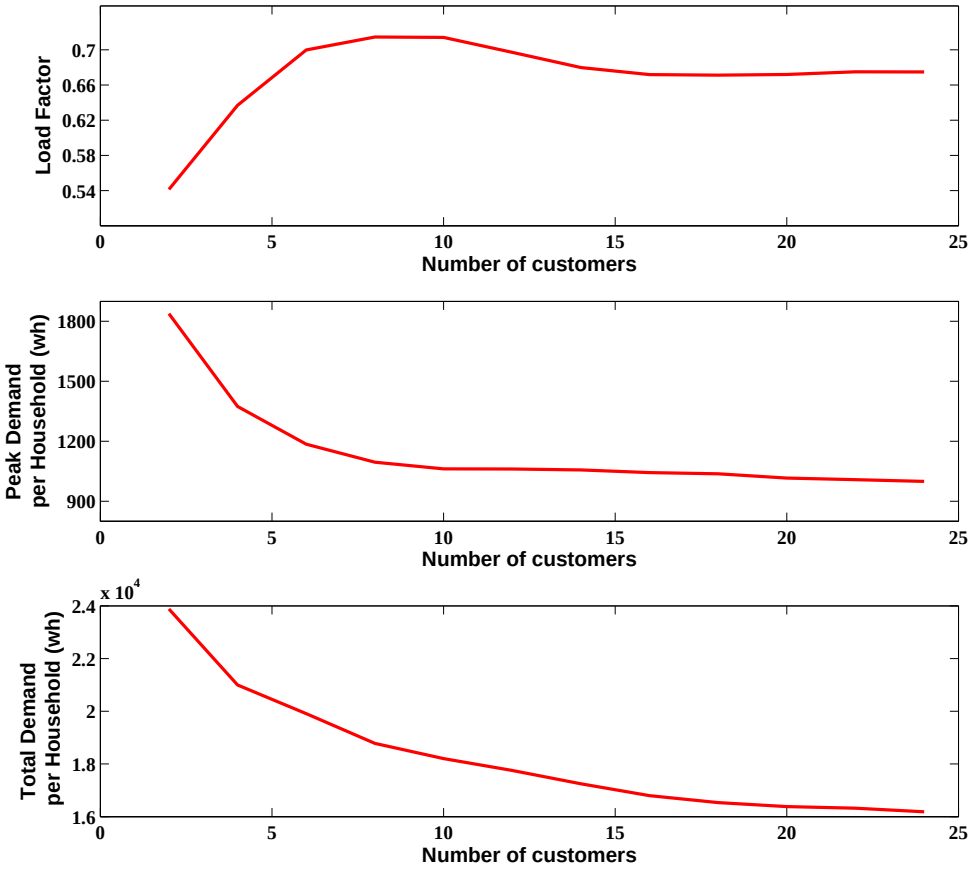


Figure 2.9. Electricity demand response without battery for different power networks with different numbers of customers.

such prices, when the households selfishly optimize their own benefits, they automatically also maximize the social welfare. The LSE can thus use dynamic pricing to coordinate demand responses to the benefit of the overall system. We propose a distributed algorithm for the LSE and the customers to jointly compute this optimal prices and demand schedules. Finally, we present simulation results that illustrate several interesting properties of the proposed scheme.

Chapter 3

Optimal Power Flow

[] In the previous chapter, we only consider demand response that balances aggregate load and supply, and abstract away the underlying power network. We will consider demand response in a radial distribution network in the next chapter by formulating it as an optimal power problem. In this chapter, we will focus on the optimal power flow problem, which is generally nonconvex. We advocate a second-order cone relaxation for OPF using the branch flow model and provide sufficient conditions under which the relaxation is exact. These conditions are demonstrated to hold for a wide class of practical power distribution systems.

3.1 Introduction

In the work of [33, 34], we advocate the use of branch flow models for the design and operation of power systems, including optimal power flow, demand response, and Volt/VAR control. In contrast to bus injection models which focus on nodal variables such as bus current and power injections, branch flow models focus on currents and power flows on individual branches [35, 36]. They have been used mainly for modeling distribution circuits which tend to be radial, but have received far less attention. The optimal power flow (OPF) problem seeks to minimize a certain cost function, such as power loss and generation cost, subject to physical constraints including Kirchoff's laws, and thermal constraints, as well as voltage regulation constraints. There has been a great deal of research on OPF since Carpentier's first formulation in 1962 [37]. OPF is generally nonconvex and NP hard, and a large number of optimization algorithms and relaxations have been proposed; see, e.g., [38–42]. Recently, a semidefinite relaxation (SDR) of OPF is proposed in [43] and a

sufficient condition is derived in [44] under which the SDR is exact. This condition is shown to essentially hold in various IEEE test systems. While this line of research has generated a lot of interest, limitations of the SDR have also been studied in [45] using 3, 5, and 7-bus system. Moreover, if SDR fails to provide exact relaxations, the solutions produced by the SDR are physically meaningless in those cases. Remarkably, it turns out that if the network is radial, then the sufficient condition of [44] always holds, provided that the bounds on the power flows satisfy a simple pattern [46–48]. This is important as almost all distribution systems are radial networks.

Indeed, for radial networks, different convex relaxations have also been studied using branch flow models. The model considered in this chapter is first proposed in [35, 36] for the optimal placement and sizing of switched capacitors in distribution circuits for Volt/VAR control. Recasting the model as a set of linear constraints together with a set of quadratic equality constraints, references [49] [33] propose a second-order-cone (SOC) convex relaxation, and prove that the relaxation is exact for radial networks, when there are no upper bounds on the loads. See also [50] for an SOC relaxation of a linear approximation of the branch flow model in [35, 36], and [51–53] for other branch flow models.

Ignoring upper bounds on the load may be unrealistic, e.g., in the context of demand response. In a previous paper [34], we prove that the SOC relaxation is exact for radial networks, provided there are no upper bounds on the voltage magnitudes and several other sufficient conditions hold. Those sufficient conditions however place strong requirements on the impedance of the distribution lines and on the load and generation patterns in the radial network. In this chapter, we propose less restrictive sufficient conditions under which the SOC relaxation is exact. As examples, we show that these conditions hold in two distribution circuits of the Southern California Edison (SCE), with high penetration of photovoltaic (PV) generation. Roughly speaking, these sufficient conditions hold in many real distribution systems where $v \sim 1$ p.u., $p, q < 1$ p.u. , $r, x << 1$ p.u., and $\frac{r}{x}$ is bounded. Here, v, p, q are the bus voltage, real power consumption, and reactive power consumption, and r, x are the resistance and reactance of the distribution lines. Moreover, we provide upper bounds on the voltage magnitudes for the SOC relaxation solutions. This would facilitate the voltage regulation in distribution systems.

The paper is organized as follows. We first present the branch flow model in Section 3.2. We then provide

in Section 3.3 sufficient conditions under which the SOC relaxation is exact for radial networks when there are no upper bounds on bus voltage magnitudes. Finally, in Section 3.4, we illustrate these sufficient conditions using two real-world distribution circuits.

3.2 Problem formulation

3.2.1 Branch flow model for radial networks

Table 3.1. Notations.

V_i, v_i	complex voltage on bus i with $v_i = \|V_i\|^2$
$s_i = p_i + \mathbf{i}q_i$	complex net load on bus i
I_{ij}, ℓ_{ij}	complex current from buses i to j with $\ell_{ij} = \|I_{ij}\|^2$
$S_{ij} = P_{ij} + \mathbf{i}Q_{ij}$	complex power flowing out from buses i to bus j
$z_{ij} = r_{ij} + \mathbf{i}x_{ij}$	impedance on line (i, j)

Consider a radial distribution circuit that consists of a set N of buses and a set E of distribution lines connecting these buses. We index the buses in N by $i = 0, 1, \ldots, n$, and denote a line in E by the pair (i, j) of buses it connects. Bus 0 represents the substation and other buses in N represent branch buses. For each line $(i, j) \in E$, let I_{ij} be the complex current flowing from buses i to j, $z_{ij} = r_{ij} + \mathbf{i}x_{ij}$ the impedance on line (i, j), and $S_{ij} = P_{ij} + \mathbf{i}Q_{ij}$ the complex power flowing from buses i to bus j. On each bus $i \in N$, let V_i be the complex voltage and s_i be the complex net load, i.e., the consumption minus generation. As customary, we assume that the complex voltage V_0 on the substation bus is given.

The branch flow model was first proposed in [35, 36] to model power flows in a steady state in a radial distribution circuit. We introduce here an abridged version of the branch flow model; see, e.g., [33, 34] for more details.

$$p_j = P_{ij} - r_{ij}\ell_{ij} - \sum_{k:(j,k)\in E} P_{jk},\ j = 1, \ldots, n \tag{3.1}$$

$$q_j = Q_{ij} - x_{ij}\ell_{ij} - \sum_{k:(j,k)\in E} Q_{jk},\ j = 1, \ldots, n \tag{3.2}$$

$$v_j = v_i - 2(r_{ij}P_{ij} + x_{ij}Q_{ij}) + (r_{ij}^2 + x_{ij}^2)\ell_{ij}, (i, j) \in E \tag{3.3}$$

$$\ell_{ij} = \frac{P_{ij}^2 + Q_{ij}^2}{v_i}, \qquad (i, j) \in E, \tag{3.4}$$

where $\ell_{ij} := |I_{ij}|^2$, $v_i := |V_i|^2$, and p_i and q_i are the real and reactive net loads at node i. Equations (3.1)–(3.4) define a system of equations in the variables $(P, Q, \ell, v) := (P_{ij}, Q_{ij}, \ell_{ij}, (i,j) \in E,\ v_i, i = 1, \ldots, n)$, which do not include phase angles of voltages and currents. Given a (P, Q, ℓ, v), these phase angles can be uniquely determined for radial networks. This is not the case for mesh networks; see [33] for exact conditions under which phase angles can be recovered for (an extension of the model here for) mesh networks.

3.2.2 Optimal power flow

Consider the problem of minimizing a cost function over the network where the optimization variables are $p := (p_1, \ldots, p_n)$, $q := (q_1, \ldots, q_n)$, as well as (P, Q, ℓ, v). Let

$$p_i := p_i^c - p_i^g, \ \ q_i := q_i^c - q_i^g,$$

where p_i^c and q_i^c are the real and reactive power consumption at node i, and p_i^g and q_i^g are the real and reactive power generation at node i. In addition to power flow equations (3.1)–(3.4), we impose the following constraints on power consumption and generation:

$$\underline{p}_i^c \le p_i^c \le \overline{p}_i^c, \quad \underline{q}_i^c \le q_i^c \le \overline{q}_i^c, \quad i = 1, \ldots, n. \tag{3.5}$$

$$\underline{p}_i^g \le p_i^g \le \overline{p}_i^g, \quad \underline{q}_i^g \le q_i^g \le \overline{q}_i^g, \quad i = 1, \ldots, n. \tag{3.6}$$

$$f_i^c(p_i^c, q_i^c) \le 0, \quad f_i^g(p_i^g, q_i^g) \le 0, \quad i = 1, \ldots, n. \tag{3.7}$$

Here, equation (3.7) models additional constraints on (p_i^c, q_i^c) and (p_i^g, q_i^g). For example, for PV generators, $(p_i^g)^2 + (q_i^g)^2 \le C^2$ where C is the capacity of the PV generation [54]. We assume f_i^c, f_i^g are convex for all $i = 1, \cdots, n$.

Finally, the voltage magnitudes must be maintained above certain thresholds:

$$\underline{v}_i \ \ \le \ \ v_i, \qquad i = 1, \ldots, n. \tag{3.8}$$

Here we do not impose upper bounds on the voltage magnitudes. However, we derive below upper bounds

on the *optimal* voltage magnitudes.

The objective of the optimal power flow problem is to minimize the power generation costs $C_i(p_i^g)$, the power losses $r_{i,j}\ell_{i,j}$, and maximize the user utilities $f_i(p_i^c)$: [1]

OPF:

$$\begin{aligned} \min_{P,Q,\ell,v,p,q} \quad & \sum_{i=1}^{n} C_i(p_i^g) - \sum_{i=1}^{n} f_i(p_i^c) + \sum_{(i,j)\in E} r_{i,j}\ell_{i,j} \\ \text{s.t.} \quad & (3.1)-(3.4),\ (3.5)-(3.8). \end{aligned}$$

OPF is NP hard in general, due to the quadratic equality constraint (3.4).

3.3 Exact relaxation

3.3.1 Second-order cone relaxation

Following [33, 34, 49], we relax the quadratic equalities in (3.4) into inequalities and consider the following convex relaxation of OPF.

ROPF:

$$\begin{aligned} \min_{P,Q,l,v,p,q} \quad & \sum_{i=1}^{n} C_i(p_i^g) - \sum_{i=0}^{n} f_i(p_i^c) + \sum_{(i,j)\in E} r_{i,j}\ell_{i,j} \\ \text{s.t.} \quad & (3.1)-(3.3),\ (3.5)-(3.8) \\ & \ell_{ij} \geq \frac{P_{ij}^2 + Q_{ij}^2}{v_i}, \quad (i,j) \in E. \end{aligned} \tag{3.9}$$

Obviously, ROPF provides a lower bound on OPF. It was shown in [33, 49] that this relaxation is exact when there are no upper bounds on the real and reactive power consumptions in (3.5) but with upper bounds on the voltage magnitudes in (3.8).

[1] We can also include in the objective function of the cost $C_0\left(\sum_{(0,j)\in E} P_{0,j}\right)$ on the total power fed into the radial network. This additional term does not change the results of the paper.

The main result of this chapter is a variety of sufficient conditions for exact relaxation when there are no upper bounds on the voltage magnitudes. Given a solution of the relaxed problem ROPF, one can always check if equality is attained in (3.4). If it is, then the relaxed solution is optimal for the original problem OPF as well. Otherwise, it is not feasible for OPF. Our goal is to develop sufficient conditions for exact relaxation that can be checked without having to solve ROPF first.

3.3.2 Sufficient condition for exact relaxation

We start by developing our results on a simple network, a one-line distribution circuit (main feeder). Then we will extend the results to general radial networks.

3.3.2.1 Line networks

For a one-line network, we can abbreviate r_{ij}, x_{ij}, P_{ij}, Q_{ij}, and l_{ij} by r_i, x_i, P_i, Q_i and l_i respectively, as shown in Figure 3.1. Rewrite the OPF problem in terms of the simplified notations as:

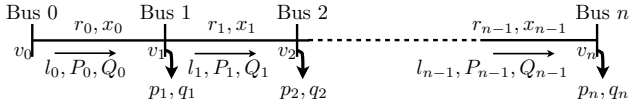


Figure 3.1. A one-line distribution network.

LOPF:

$$\min_{P,Q,\ell,v,p,q} \quad \sum_{i=1}^{n} C_i(p_i^g) - \sum_{i=1}^{n} f_i(p_i^c) + \sum_{i=0}^{n-1} r_i \ell_i$$

$$\text{s.t.} \quad \frac{P_i^2 + Q_i^2}{v_i} = \ell_i, \quad i = 0, \cdots, n-1 \tag{3.10}$$

$$P_i = P_{i+1} + r_i \ell_i + p_{i+1}^c - p_{i+1}^g, i = 0, \cdots, n-1 \tag{3.11}$$

$$Q_i = Q_{i+1} + x_i \ell_i + q_{i+1}^c - q_{i+1}^g, i = 0, \cdots, n-1 \tag{3.12}$$

$$v_i - v_{i+1} = 2(r_i P_i + x_i Q_i) - (r_i^2 + x_i^2)\ell_i, i = 0, \cdots, n-1 \tag{3.13}$$

$$(3.5) - (3.8).$$

The above optimization problem can be relaxed to the following second-order cone program:

RLOPF

$$\begin{aligned} \min_{P,Q,l,v,p,q} \quad & \sum_{i=1}^{n} C_i(p_i^g) - \sum_{i=1}^{n} f_i(p_i^c) + \sum_{i=0}^{n-1} r_i \ell_i \\ s.t. \quad & (3.5)-(3.8), (3.11)-(3.13) \\ & \frac{P_i^2 + Q_i^2}{v_i} \leq \ell_i, \quad i = 0, \cdots, n-1. \end{aligned} \tag{3.14}$$

The next lemma provides a sufficient condition guaranteeing that RLOPF is an exact relaxation of OPF. For each bus $i, k \in N \backslash \{0\}$, define

$$R_k := \sum_{j=0}^{k-1} r_j, \;\; X_k := \sum_{j=0}^{k-1} x_j, \;\; R_{i,k} := \sum_{j=i}^{k-1} r_j, \;\; X_{i,k} := \sum_{j=i}^{k-1} x_j$$

as the cumulative resistance and reactance from the feeder or bus i to bus k. Also define $[a]^+ = \max(a, 0)$.

Lemma 3.1. *Any optimal solution (P, Q, ℓ, v, p, q) of RLOPF is also optimal for LOPF, provided that for each $k \in N \backslash \{0\}$ the following condition holds: if $\frac{r_k}{x_k} - \frac{R_k}{X_k} \geq 0$, then*

$$v_i + 2P_i \left(\frac{r_k}{x_k} X_k - R_{i,k} \right) + 2Q_i X_i > 0; \forall i < k \tag{3.15}$$

otherwise,

$$v_i + 2P_i R_i + 2Q_i \left(\frac{x_k}{r_k} R_k - X_{i,k} \right) > 0, \forall i < k \tag{3.16}$$

Moreover, for each node $i \in N \backslash \{0\}$, the voltage is upper-bounded by:[2]

$$v_i \leq v_0 - 2 \sum_{k=0}^{i-1} \left(r_k (P_k - r_k \ell_k) + x_k (Q_k - x_k \ell_k) \right).$$

[2]Note that $P_k - r_k \ell_k$ and $Q_k - x_k \ell_k$ are the real and reactive power received by bus $k+1$ from bus k.

Proof. Introducing dual variables for Problem LOPF, denoted as

$$\alpha = (\lambda_i, \theta_i, w_i, \xi_i, \gamma_i^c, \gamma_i^g, \bar{\zeta}_i^c, \underline{\zeta}_i^c, \bar{\eta}_i^c, \underline{\eta}_i^c, \bar{\zeta}_i^g, \underline{\zeta}_i^g, \bar{\eta}_i^g, \underline{\eta}_i^g, \mu_i),$$

where $\mu_i \geq 0$, $\xi_i \leq 0$. Let x denote the primal variables. The Lagrangian dual function is given as:

$$\begin{aligned}
L(x,\alpha) \quad = \quad & -\sum_{i=0}^{n} f_i(p_i^c) + \sum_{i=1}^{n} C_i(p_i^g) + \sum_{i=0}^{n-1} r_i\ell_i + \sum_{i=0}^{n-1} \lambda_i(P_i - P_{i+1} - r_i\ell_i - p_{i+1}) \\
& + \sum_{i=0}^{n-1} \theta_i(Q_i - Q_{i+1} - x_i\ell_i - q_{i+1}) + \sum_{i=0}^{n-1} \mu_i(\frac{P_i^2 + Q_i^2}{v_i} - \ell_i) \\
& + \sum_{i=0}^{n-1} w_i(v_i - v_{i+1} - 2(r_iP_i + x_iQ_i) + (r_i^2 + x_i^2)\ell_i) + \sum_{i=1}^{n} (\gamma_i^c f_i^c(p_i^c, q_i^c) + \gamma_i^g f_i^g(p_i^g, q_i^g)) \\
& + \sum_{i=1}^{n} (\xi_i v_i + (\bar{\zeta}_i^c - \underline{\zeta}_i^c)p_i^c + (\bar{\eta}_i^c - \underline{\eta}_i^c)q_i^c) + \sum_{i=1}^{n} ((\bar{\zeta}_i^g - \underline{\zeta}_i^g)p_i^g + (\bar{\eta}_i^g - \underline{\eta}_i^g)q_i^g) \\
& + \sum_{i=1}^{n} (-\xi_i\underline{v}_i - \bar{p}_i^c\bar{\zeta}_i^c + \underline{p}_i^c\underline{\zeta}_i^c - \bar{q}_i^c\bar{\eta}_i^c) + \sum_{i=1}^{n} (\underline{q}_i^g\underline{\eta}_i^g - \bar{p}_i^g\bar{\zeta}_i^g + \underline{p}_i^g\underline{\zeta}_i^g - \bar{q}_i^g\bar{\eta}_i^g + \underline{q}_i^g\underline{\eta}_i^g).
\end{aligned}$$

If (x^*, α^*) are the primal-dual optimal, then $\frac{\partial L(x^*,\alpha^*)}{x} = 0$. In the following, we will drop the '$*$' for simplicity. Define $\beta_i = \frac{\mu_i}{v_i}$ for each $i = 0, \cdots, n-1$. $\frac{\partial L(x^*,\alpha^*)}{x} = 0$ gives that for all $i = 0, \cdots, n-1$,

$$\frac{\partial L}{P_i} = -\lambda_{i-1} + \lambda_i + 2\beta_i P_i - 2w_i r_i = 0, \tag{3.17}$$

$$\frac{\partial L}{Q_i} = -\theta_{i-1} + \theta_i + 2\beta_i Q_i - 2w_i x_i = 0, \tag{3.18}$$

$$\frac{\partial L}{\ell_i} = r_i - \lambda_i r_i - \theta_i x_i - \beta_i v_i + w_i(r_i^2 + x_i^2) = 0, \tag{3.19}$$

and for all $i = 1, \cdots, n$,

$$\frac{\partial L}{v_i} \quad = \quad -\mu_i \frac{P_i^2 + Q_i^2}{v_i^2} - w_{i-1} + w_i + \xi_i = 0, \tag{3.20}$$

where we introduce dummy variables $\lambda_{-1} = \theta_{-1} = \beta_n = \mu_n = l_n = w_n = 0$.

By (3.20), it is straightforward to show that $w_i \leq 0$, for $i = 0, \cdots, n$. Note that for $i = n$, we have $-w_{n-1} + \xi_n = 0$, which implies that $w_{n-1} = \xi_n \leq 0$. Then by induction, we can show that $w_i \leq 0$, for $i = 0, \cdots, n-1$.

The rest of the proof will show that $\beta_i > 0$ for all $i \in N$. Then by complementary slackness, we know that all equality (3.10) holds for each $i \in N$, implying that exact relaxation holds.

Suppose there exists such $i \in N$ that $\beta_i = 0$. Let $k := \min\{i \in N : \beta_i > 0\}$. Suppose $k = 0$. By (3.17,3.18,3.19), we have:

$$\lambda_0 - 2w_0 r_0 = 0 \tag{3.21}$$

$$\theta_0 - 2w_0 x_0 = 0 \tag{3.22}$$

$$r_0 - \lambda_0 r_0 - \theta_0 x_0 + w_0(r_0^2 + x_0^2) = 0. \tag{3.23}$$

Substituting (3.21,3.22) into (3.23), we have:

$$r_0 - w_0(r_0^2 + x_0^2) = 0.$$

The LHS is strictly positive since $r_0 > 0$ and $w_0 \leq 0$. Therefore by contradiction, $k > 0$.

Suppose $k > 0$. Write λ_l and θ_l for each $l \leq k$ in terms of w and β. Summing up (3.17) from $i = 0$ to $i = l$, we have:

$$\lambda_l = 2\left(\sum_{i=0}^{l} w_i r_i - \sum_{i=0}^{l} \beta_i P_i\right), l = 0, \cdots, k. \tag{3.24}$$

Similarly, summing up (3.18) from $i = 0$ to $i = l$:

$$\theta_l = 2(\sum_{i=0}^{l} w_i x_i - \sum_{i=0}^{l} \beta_i Q_i), l = 0, \cdots, k. \tag{3.25}$$

Substituting (3.24,3.25) into (3.19) for $l \leq k$:

$$\beta_l v_l = r_l + w_l(r_l^2 + x_l^2) - 2r_l\left(\sum_{i=0}^{l} w_i r_i - \sum_{i=0}^{l} \beta_i P_i\right) - 2x_l\left(\sum_{i=0}^{l} w_i x_i - \sum_{i=0}^{l} \beta_i Q_i\right). \tag{3.26}$$

Summing up (3.26) from $l = 0$ to $l = k-1$ gives:

$$\begin{aligned}\sum_{l=0}^{k-1}\beta_l v_l &= R_k + \sum_{l=0}^{k-1} w_l(r_l^2 + x_l^2) - 2\sum_{l=0}^{k-1}\sum_{i=0}^{l}(w_i r_i r_l - \beta_i P_i r_l + w_i x_i x_l - \beta_i Q_i x_l) \\ &= R_k + \sum_{l=0}^{k-1} w_l(r_l^2 + x_l^2) - 2\sum_{i=0}^{k-1}\sum_{l=i}^{k-1}(w_i r_i r_l - \beta_i P_i r_l + w_i x_i x_l - \beta_i Q_i x_l) \\ &= R_k - 2\sum_{i=0}^{k-1}(w_i r_i R_{i,k} - \beta_i P_i R_{i,k}) - 2\sum_{i=0}^{k-1}(w_i x_i X_{i,k} - \beta_i Q_i X_{i,k}) + \sum_{l=0}^{k-1} w_l\left(r_l^2 + x_l^2\right). \quad (3.27)\end{aligned}$$

where $R_{i,k} := \sum_{l=i}^{k-1} r_l$ and $X_{i,k} := \sum_{l=i}^{k-1} x_l$.

Also, by (3.26) with $l = k$, we have:

$$0 \quad = \quad r_k - 2r_k\sum_{i=0}^{k-1} w_i r_i - 2x_k\sum_{i=0}^{k-1} w_i x_i - w_k(r_k^2 + x_k^2) + 2r_k(\sum_{i=0}^{k-1}\beta_i P_i) + 2x_k(\sum_{i=0}^{k-1}\beta_i Q_i)) \quad (3.28)$$

If $\frac{r_k}{x_k} - \frac{R_k}{X_k} \geq 0$, applying the following operation on (3.27) and (3.28), $\frac{(3.28)}{x_k} - \frac{(3.27)}{X_k}$, and using the fact that $w_i \leq 0$ for all $i \in N$,

$$\begin{aligned}&\frac{r_k}{x_k} - \frac{R_k}{X_k} - 2\sum_{i=0}^{k-1} w_i r_i\left(\frac{r_k}{x_k} - \frac{R_{i,k}}{X_k}\right) - 2\sum_{i=0}^{k-1} w_i x_i\left(1 - \frac{X_{i,k}}{X_k}\right) \quad (3.29)\\ \leq \quad &-\sum_{i=0}^{k-1}\frac{\beta_i}{X_k}\left(v_i + 2P_i\left(\frac{r_k}{x_k}X_k - R_{i,k}\right) + 2Q_i X_i\right).\end{aligned}$$

Note that under the condition specified in Section 3.3, the RHS of inequality (3.29) is negative; but the LHS is non-negative. Therefore we have a contradiction.

Otherwise, if $\frac{r_k}{x_k} - \frac{R_k}{X_k} \leq 0$, applying the following operation on (3.27) and (3.28), $\frac{(3.28)}{r_k} - \frac{(3.27)}{R_k}$, and following similar arguments, we will get a contradiction as well. We can therefore claim the sufficient condition in the Lemma.

In the rest of the proof we will bound the voltage on each node $i \in N\backslash\{0\}$. Equation (3.13) implies that for each $k \in N$,

$$\begin{aligned}v_{k+1} \quad &= \quad v_k - 2(r_k P_k + x_k Q_k) + (r_k^2 + x_k^2)\ell_k \\ &\leq \quad v_k - 2(r_k P_k + x_k Q_k) + 2(r_k^2 + x_k^2)\ell_k \\ &= \quad v_k - 2(r_k(P_k - r_k\ell_k) + x_k(Q_k - x_k\ell_k)\ .\end{aligned}$$

Summing up this inequality from $k = 0$ to $k = i - 1$, we have

$$v_i \leq v_0 - 2\sum_{k=0}^{i-1}(r_k(P_k - r_k\ell_k) + x_k(Q_k - x_k\ell_k).$$

□

The condition in Lemma 3.1 is not checkable before solving RLOPF as it involves a solution (P, Q, ℓ, v, p, q) of RLOPF. We now provide a checkable condition by bounding v_i, P_i, Q_i in terms of system parameters $\underline{p}_i, \bar{p}_i, \underline{q}_i, \bar{q}_i, \underline{v}_i$. Define

$$\underline{P}_i^{nom} \triangleq \sum_{j=i+1}^{n}(\underline{p}_j^c - \bar{p}_j^g);\ \underline{Q}_i^{nom} \triangleq \sum_{j=i+1}^{n}(\underline{q}_j^c - \bar{q}_j^g).$$

Using (3.11–3.12) we can iteratively derive that for any $i \in N$,

$$P_i \geq P_i - r_i\ell_i \geq \underline{P}_i^{nom},$$

and

$$Q_i \geq Q_i - r_i\ell_i \geq \underline{Q}_i^{nom}.$$

Combining the above two inequalities with Lemma 3.1 gives the following result.

Theorem 3.2. *Any optimal solution of RLOPF is also optimal for LOPF, provided that for each $k \in N\backslash\{0\}$ the following condition holds: if $\frac{r_k}{x_k} - \frac{R_k}{X_k} \geq 0$,*

$$\underline{v}_i + 2\underline{P}_i^{nom}\left(\frac{r_k}{x_k}X_k - R_{i,k}\right) + 2\underline{Q}_i^{nom}X_i > 0; \forall i < k; \tag{3.30}$$

otherwise,

$$\underline{v}_i + 2\underline{P}_i^{nom}R_i + 2\underline{Q}_i^{nom}\left(\frac{x_k}{r_k}R_k - X_{i,k}\right) > 0, \forall i < k\ . \tag{3.31}$$

Moreover, for each node $i \in N\backslash\{0\}$, the voltage is upper-bounded by:

$$v_i \leq v_0 - 2\sum_{k=0}^{i-1}\left(r_k \underline{P}_k^{nom} + x_k \underline{Q}_k^{nom}\right).$$

Since $\underline{v}_i > 0$ for each $i \in N$, we have the following special case. If $\underline{P}_i^{nom} > 0$ and $\underline{Q}_i^{nom} > 0$ for all $i \in N\backslash\{0\}$, then the right-hand sides of (3.30,3.31) are always non-positive, which implies that (**??**) the sufficient condition in Theorem 3.2 is always satisfied. Hence the relaxation is exact provided that both the real and reactive powers do not flow backward. This condition in the special case is more stringent than (3.30,3.31) and usually does not hold in practice. The sufficient condition (3.30,3.31) depends only on how $\underline{v}_i$ compare with the products of resistances (reactances) and real (reactive) powers. In practice, $|V| \sim 1$ p.u., $r, x << 1$ p.u., $\frac{r}{x} \sim [0.1, 10]$, and $p, q < 1$ p.u.. As we show in Section 3.4, condition (3.30,3.31) usually holds when the system parameters are in these ranges.

3.3.2.2 General radial networks

We now extend Lemma 3.1 and Theorem 3.2 to general radial distribution circuits. Given a radial network:

- For each node i, k, denote the unique path from i to node k by

$$\mathcal{P}_{i,k} \triangleq \{(j_1, j_2) : (j_1, j_2) \in E \text{ is on the path from node } i \text{ to node } k\}.$$

- Define the cumulative resistance and reactance from root i to node k as $R_{i,k} \triangleq \sum_{(j_1,j_2)\in\mathcal{P}_{i,k}} r_{j_1,j_2}$ and $X_{i,k} \triangleq \sum_{(j_1,j_2)\in\mathcal{P}_{i,k}} x_{j_1,j_2}$.

It is straightforward to extend Lemma 3.1 to the case of general radial networks.

Lemma 3.3. *Any optimal solution (P, Q, ℓ, v, p, q) of ROPF is also optimal for OPF, provided that for each $(k, l) \in E$ the following condition holds: if $\frac{r_{k,l}}{x_{k,l}} - \frac{R_{0,k}}{X_{0,k}} \geq 0$, then*

$$v_i + 2P_{i,j}\left(\frac{r_{k,l}}{x_{k,l}}X_{0,k} - R_{i,k}\right) + 2Q_{i,j}X_{0,i} > 0; \forall (i,j) \in \mathcal{P}_k; \tag{3.32}$$

otherwise,

$$v_i + 2P_{i,j}R_{0,i} + 2Q_{i,j}\left(\frac{x_{k,l}}{r_{k,l}}R_{0,k} - X_{i,k}\right) > 0, \forall (i,j) \in \mathcal{P}_k. \tag{3.33}$$

Moreover, for each node $i \in N\backslash\{0\}$, the voltage is upper-bounded by:

$$v_i \leq v_0 - 2\sum_{(j,k)\in\mathcal{P}_i}\left(r_{j,k}(P_{j,k} - r_{j,k}\ell_{j,k}) + x_{j,k}(Q_{j,k} - x_{i,j}\ell_{j,k})\right).$$

Proof. The proof for Lemma 3.1 can be easily extended to a radial network. We can first prove $w_{i,j} \leq 0$ for each $(i,j) \in E$ by induction (from leaves to the feeder.). To prove $\beta_{i,j} > 0$, just focus on each lateral and use the same arguments as those for a line distribution network to prove that $\beta_{i,j} > 0$ on each lateral. □

Similarly, this lemma involves a solution $v_i, P_{i,j}, Q_{i,j}$ of ROPF. For a sufficient condition that does not require solving ROPF first, define

$$\underline{P}_j^{nom} \triangleq \sum_{i\in D(j)}^{n}(\underline{p}_i^c - \bar{p}_i^g);\ \ \underline{Q}_j^{nom} \triangleq \sum_{i\in D(j)}^{n}(\underline{q}_i^c - \bar{q}_i^g),$$

for each $j = 1, \cdots, n$. Here $D(j)$ is the set of all the descendants of j including j itself.[3] Note that for any $(i,j) \in E$,

$$P_{i,j} \geq P_{i,j} - r_{i,j}\ell_{i,j} \geq \underline{P}_j^{nom}, Q_{i,j} \geq Q_{i,j} - x_{i,j}\ell_{i,j} \geq \underline{Q}_j^{nom}.$$

Lemma 3.3 then implies the following extension of Theorem 3.2.

Theorem 3.4. *Any optimal solution of ROPF is also optimal for OPF, provided that for each $(k,l) \in E$ the*

[3]A rigorous definition of $D(j)$ is: $D(j) \triangleq \{l \in N$: there exist a sequence of nodes, $j_0, j_1, \cdots, j_m,$ such that $j_0 = j, j_m = l,$ and $(j_i, j_{i+1}) \in E, \forall i = 0, \ldots, m-1,$ where $m \geq 0\}$.

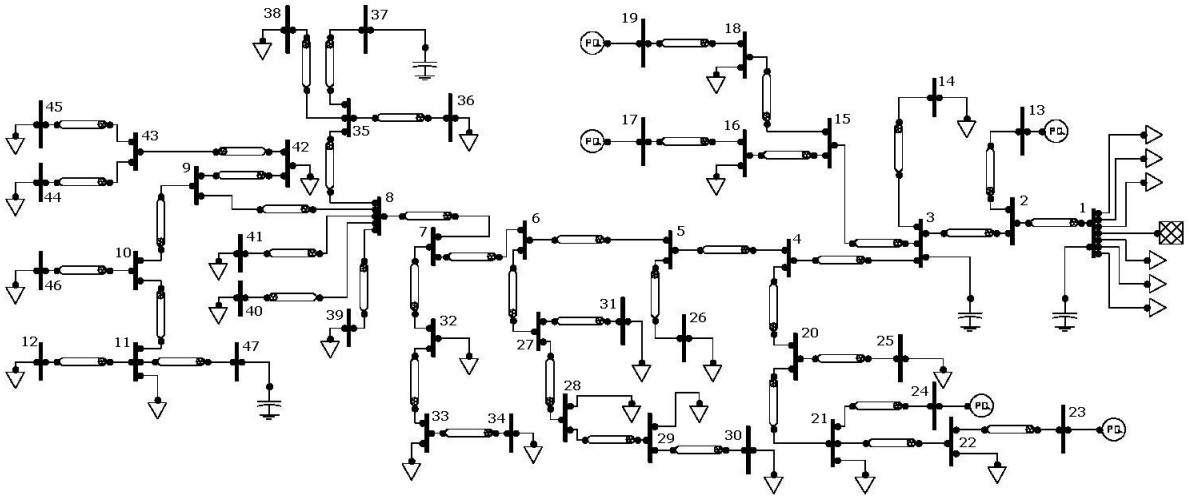


Figure 3.2. Schematic diagram of a 47-bus SCE distribution systems.

following condition holds: if $\frac{r_{k,l}}{x_{k,l}} - \frac{R_{0,k}}{X_{0,k}} \geq 0$*, then*

$$\underline{v}_i + 2\underline{P}_j^{nom}\left(\frac{r_{k,l}}{x_{k,l}}X_{0,k} - R_{i,k}\right) + 2\underline{Q}_j^{nom}X_{0,i} > 0; \forall (i,j) \in \mathcal{P}_k; \tag{3.34}$$

otherwise,

$$\underline{v}_i + 2\underline{P}_j^{nom}R_{0,i} + 2\underline{Q}_j^{nom}\left(\frac{x_{k,l}}{r_{k,l}}R_{0,k} - X_{i,k}\right) > 0, \forall (i,j) \in \mathcal{P}_k. \tag{3.35}$$

Moreover, for each node $i \in N\backslash\{0\}$*, the voltage is upper-bounded by:*

$$v_i \leq v_0 - 2\sum_{(j,k)\in\mathcal{P}_i}\left(r_{j,k}\underline{P}_k^{nom} + x_{j,k}\underline{Q}_k^{nom}\right).$$

Since $|V| \sim 1$ p.u., $r, x << 1$ p.u., and $p, q < 1$ p.u. in practice, the condition in Theorem 3.4 holds for both a 47-bus distribution circuit and a 56-bus distribution circuit of Southern California Edison (SCE), as show in Section 3.4.

Table 3.2. Line impedances, peak spot load KVA, Capacitors and PV generation's nameplate ratings for the distribution circuit in Figure 3.2.

Network Data																	
Line Data				Line Data				Line Data				Load Data		Load Data		PV Generators	
From Bus.	To Bus.	R (Ω)	X (Ω)	From Bus.	To Bus.	R (Ω)	X (Ω)	From Bus.	To Bus.	R (Ω)	X (Ω)	Bus No.	Peak MVA	Bus No.	Peak MVA	Bus No.	Nameplate Capacity
1	2	0.259	0.808	8	41	0.107	0.031	21	22	0.198	0.046	1	30	34	0.2		
2	13	0	0	8	35	0.076	0.015	22	23	0	0	11	0.67	36	0.27	13	1.5MW
2	3	0.031	0.092	8	9	0.031	0.031	27	31	0.046	0.015	12	0.45	38	0.45	17	0.4MW
3	4	0.046	0.092	9	10	0.015	0.015	27	28	0.107	0.031	14	0.89	39	1.34	19	1.5 MW
3	14	0.092	0.031	9	42	0.153	0.046	28	29	0.107	0.031	16	0.07	40	0.13	23	1 MW
3	15	0.214	0.046	10	11	0.107	0.076	29	30	0.061	0.015	18	0.67	41	0.67	24	2 MW
4	20	0.336	0.061	10	46	0.229	0.122	32	33	0.046	0.015	21	0.45	42	0.13		
4	5	0.107	0.183	11	47	0.031	0.015	33	34	0.031	0	22	2.23	44	0.45	Shunt Capacitors	
5	26	0.061	0.015	11	12	0.076	0.046	35	36	0.076	0.015	25	0.45	45	0.2	Bus No.	Nameplate Capacity
5	6	0.015	0.031	15	18	0.046	0.015	35	37	0.076	0.046	26	0.2	46	0.45		
6	27	0.168	0.061	15	16	0.107	0.015	35	38	0.107	0.015	28	0.13				
6	7	0.031	0.046	16	17	0	0	42	43	0.061	0.015	29	0.13	V_{base} = 12.35kV		1	6000 KVAR
7	32	0.076	0.015	18	19	0	0	43	44	0.061	0.015	30	0.2			3	1200 KVAR
7	8	0.015	0.015	20	21	0.122	0.092	43	45	0.061	0.015	31	0.07			37	1800 KVAR
8	40	0.046	0.015	20	25	0.214	0.046					32	0.13			47	1800 KVAR
8	39	0.244	0.046	21	24	0	0					33	0.27				

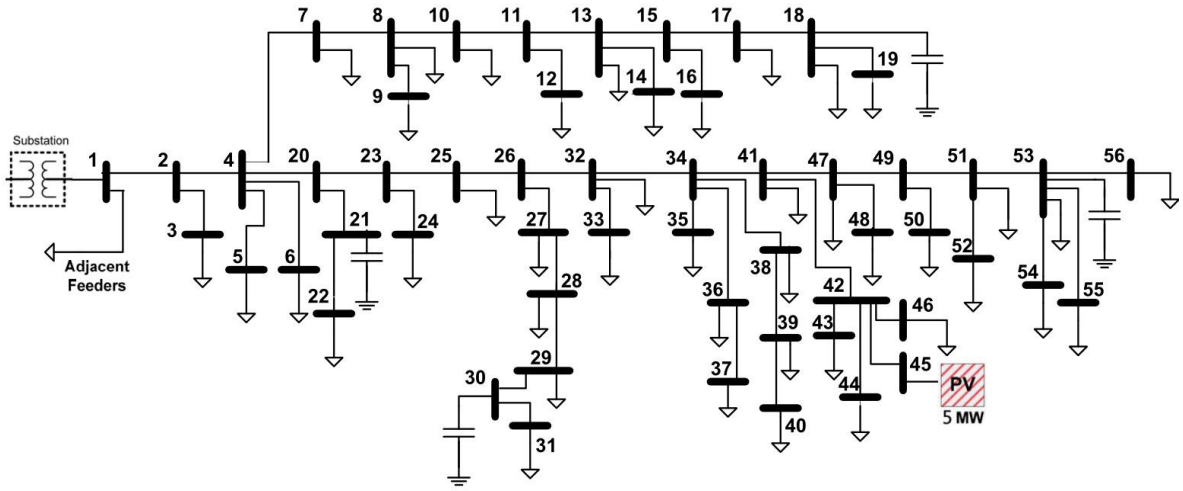


Figure 3.3. Schematic diagram of a 56-bus SCE distribution systems.

Table 3.3. Line impedances, peak spot load kVA, capacitors and PV generation's nameplate ratings for the distribution circuit in Figure 3.3.

Network Data																	
Line Data				Line Data				Line Data				Load Data		Load Data		Load Data	
From Bus.	To Bus.	R (Ω)	X (Ω)	From Bus.	To Bus.	R (Ω)	X (Ω)	From Bus.	To Bus.	R (Ω)	X (Ω)	Bus No.	Peak MVA	Bus No.	Peak MVA	Bus No.	Peak MVA
1	2	0.160	0.388	20	21	0.251	0.096	39	40	2.349	0.964	3	0.057	29	0.044	52	0.315
2	3	0.824	0.315	21	22	1.818	0.695	34	41	0.115	0.278	5	0.121	31	0.053	54	0.061
2	4	0.144	0.349	20	23	0.225	0.542	41	42	0.159	0.384	6	0.049	32	0.223	55	0.055
4	5	1.026	0.421	23	24	0.127	0.028	42	43	0.934	0.383	7	0.053	33	0.123	56	0.130
4	6	0.741	0.466	23	25	0.284	0.687	42	44	0.506	0.163	8	0.047	34	0.067	Shunt Cap	
4	7	0.528	0.468	25	26	0.171	0.414	42	45	0.095	0.195	9	0.068	35	0.094	Bus	Mvar
7	8	0.358	0.314	26	27	0.414	0.386	42	46	1.915	0.769	10	0.048	36	0.097	19	0.6
8	9	2.032	0.798	27	28	0.210	0.196	41	47	0.157	0.379	11	0.067	37	0.281	21	0.6
8	10	0.502	0.441	28	29	0.395	0.369	47	48	1.641	0.670	12	0.094	38	0.117	30	0.6
10	11	0.372	0.327	29	30	0.248	0.232	47	49	0.081	0.196	14	0.057	39	0.131	53	0.6
11	12	1.431	0.999	30	31	0.279	0.260	49	50	1.727	0.709	16	0.053	40	0.030	Photovoltaic	
11	13	0.429	0.377	26	32	0.205	0.495	49	51	0.112	0.270	17	0.057	41	0.046	Bus	Capacity
13	14	0.671	0.257	32	33	0.263	0.073	51	52	0.674	0.275	18	0.112	42	0.054		
13	15	0.457	0.401	32	34	0.071	0.171	51	53	0.070	0.170	19	0.087	43	0.083	45	5MW
15	16	1.008	0.385	34	35	0.625	0.273	53	54	2.041	0.780	22	0.063	44	0.057		
15	17	0.153	0.134	34	36	0.510	0.209	53	55	0.813	0.334	24	0.135	46	0.134	V_{base} = 12kV	
17	18	0.971	0.722	36	37	2.018	0.829	53	56	0.141	0.340	25	0.100	47	0.045		
18	19	1.885	0.721	34	38	1.062	0.406					27	0.048	48	0.196		
4	20	0.138	0.334	38	39	0.610	0.238					28	0.038	50	0.045		

3.4 Case study

In this section we evaluate these conditions for exact relaxation for two distribution circuits of SCE with high penetration of photovoltaic (PV) generation [49, 55]. Figures 3.2 and 3.3 show a 47-bus and a 56-bus distribution circuit respectively and Table 3.2 and 3.3 list the corresponding network data including line impedances, peak MVA demand of loads, and the nameplate capacity of the shunt capacitors and the photovoltaic generations. Note that in the 47-bus circuit, bus 1 indicates the substation, and there are 5 photovoltaic (PV) generators located on buses 13, 17, 19, 23 and 24. In the 56-bus circuit, there is 1 photovoltaic (PV) generator located on bus 45.

3.4.1 Verifying sufficient conditions

We verify that the condition in Theorem 3.4 holds in both circuits. To calculate $\underline{P}^{nom}$ and $\underline{Q}^{nom}$, we only need values for lower bounds of (p_i^c, q_i^c) and upper bounds of (p_i^g, q_i^g):

- For load buses, we set $\underline{p}_i^c$ to be 0 and set $\underline{q}_i^c$ to be the negative of peak MVA value.
- For PV generators, we set $\bar{p}_i^g$ to be the generators' capacities.
- For shunt capacitors, we treat them as reactive power generators and set $\bar{q}_i^g$ to be their shunt capacities.

After checking conditions in Theorem 3.4, we know that as long as the voltage magnitudes are maintained above 85% of the nominal value (which they are in practice) for each bus i, the conditions holds and ROPF is an exact relaxation of OPF.

Remark 3.1. *All the above analysis is worst-case. In reality, $\underline{p}_i^c$ and $\underline{q}_i^c$ tend to be larger than the values we used above, and $\underline{p}_i^g$ and $\underline{q}_i^g$ smaller. This implies larger $(\underline{P}^{nom}, \underline{Q}^{nom})$ and larger values for the left-hand side of inequality (3.34) and of inequality (3.35) than the values we have calculated above. Thus the sufficient condition in Corollary 3.4 is easier to meet in practice.*

Remark 3.2. *The condition in Theorem 3.4 can be used as a rule of thumb for designing distribution circuits that will ensure that ROPF is an exact relaxation of OPF. Specifically, if the distribution lines have smaller resistance and reactance, then condition in Theorem 3.4 is easier to satisfy.*

3.4.2 Simulation

We have also solved Problem ROPF using the CVX toolbox [56]. In the simulation:

- for each load bus, we set $\bar{p}_i^c$ and $\bar{q}_i^c$ as the peak MVA value, and $\underline{p}_i^c$, $\underline{q}_i^c$ as half of the peak MVA value. We use utility functions of the form $-a_i(p_i - \bar{p}_i^c)^2 + b_i$ where a_i, b_i are drawn randomly from $[2.5, 5]$;
- for each PV generator, we set $(\bar{p}_i^g, \bar{q}_i^g)$ as the generator's capacity value, and $\underline{p}_i^g$, $\underline{q}_i^g$ as 0. We use cost functions of the form $a_i p_i^2$ where a_i are drawn randomly from $[2.5, 5]$;
- for each shunt capacitor, we treat them as reactive power generators and set $\bar{p}_i^c = \underline{p}_i^c = \underline{q}_i^c = 0$ and $\bar{q}_i^g$ as their shunt capacities;
- we use a cost $C_0(P_0) := C_0(\sum_{j:(0,j)\in E} P_{0,j})$ on the total power fed into this distribution system of the form $a_0 P_0^2 + b_0 P_0$ with $a_0 = 0.1, b_0 = 0.1$.

After solving ROPF using the CVX toolbox for both the 47-bus system and the 56-bus system, we verify that the solutions of ROPF satisfy the equality constraint (3.4) and are therefore optimal for OPF. This implies that ROPF indeed is an exact relaxation of OPF for both distribution circuits. Moreover, in each case, the maximum voltage magnitude of the optimal solution is 12.35KV which is the voltage magnitude of feeder and much less than the upper bound we provided in Section 3.4.1.

3.5 Conclusion

We have studied the second-order cone relaxation of the optimal power flow problem in radial networks using the branch flow model. We provide sufficient conditions under which the relaxation is exact when there are no upper bounds on the voltage magnitudes. These conditions are verified to hold in two real-world distribution circuits.

Chapter 4

Distributed Load Management Over the Power Network

[] In this chapter, we study distributed load management over a radial distribution network, by formulating it as an optimal power flow (OPF) problem that maximizes the aggregate user utilities and minimizes the supply cost and the power line losses, subject to the power flow constraints and operating constraints. In the previous chapter, we showed that the OPF problem is non-convex and we proposed a convex relaxation that is usually exact for the real-world distribution circuits. Following those results, we propose two different distributed mechanisms to achieve the optimum. In the first one, there is a load-serving entity to set the price signals in order to coordinate the users' demand response and in the second one the users coordinate their decisions through local communications with neighbors. Numerical examples with the real-world distribution circuits are provided to complement our theoretical analysis.

4.1 Introduction

Most of the work on load management considers only the balance between aggregate load and supply, and abstract away the underlying power network and the associated power flow constraints and operating constraints. As a result, the schemes proposed may end up with an electricity consumption/shedding decision that would violate those network and operating constraints. There is some recent work on load management that takes into consideration the physical network constraints and proposes location-based marginal pricing schemes for load management; see, e.g., [57–60]. But they usually use either the DC approximation model

or the bus injection model for the electricity network, which are more suitable for the transmission system.

In this chapter, we study optimal load management in the presence of the network and operating constraints for the radial distribution networks, using the branch flow model introduced in the previous chapter. Specifically, we formulate the load management problem as an AC optimal power flow (OPF) problem whose objective is to maximize the aggregate users' utility and minimize the supply cost and the power line losses, subject to the power flow constraints and operating constraints such as the voltage regulation constraint and power injection constraints. The resulting OPF is in the same form of OPF in the previous chapter. Though it is non-convex, we have proposed a convex relaxation of the optimization problem, and discussed whether the relaxation can be exact and under what conditions. Convexity not only facilitates the design of effective pricing schemes for the power market involved in demand response, but also enables the development of tractable, scalable, and distributed algorithms for system operations.

We then consider two different distributed demand response mechanisms. In the first one, the radial distribution network is served by a single load serving entity (LSE), which coordinates the end users' demand response decisions by setting the right prices. Using the Lagrangian duality decomposition method, we show that there exists an optimal price scheme, under which, if each user maximizes its net utility, the global welfare, i.e., the aggregate utilities minus the power losses, turns out to be maximized. We next develop a distributed algorithm to iteratively calculate the optimal price, where i) the LSE does not need to know users' information such as the utility functions or consumption constraints, and ii) each user makes a demand response decision based only on the price and its own utility function and consumption constrains. This algorithm requires two-way communication between the LSE and each user, and at each iteration, the LSE is required to solve a large OPF problem. In the second mechanism, we develop a fully distributed OPF algorithm for demand response, where the end users make and coordinate their local demand response decisions through local communication with their neighbors. This demand response scheme requires two-way communication only between the end users that are directly connected in the distribution network, and each user only needs to solve a small optimization problem. Both of the two demand response algorithms are based on a well-known distributed algorithm, Predictor Corrector Proximal Multiplier (PCPM) [61]. Provided that the convex relaxation of the OPF problem for demand response is exact, the algorithm is guaranteed to

converge to the global optimum of the OPF problem. Lastly, case studies on Southern California Edison distribution circuits show that the proposed algorithms converge to the global optimal solution.

The rest of the chapter is organized as follows. We first formulate the optimal demand response problem, introduce the PCPM algorithm, and discuss convex relaxation of the optimization problem in Section 4.2. We then study the first demand response scheme in Section 4.3 and the second demand response scheme in Section 4.4. In Section 4.6, we provide numerical examples to complement the theoretical analysis, using a real-word distribution circuit.

4.2 Problem formulation & preliminary

4.2.1 Problem formulation

Consider a radial distribution circuit that consists of a set N of buses and a set E of distribution lines connecting these buses. We index the buses in N by $i = 0, 1, \ldots, n$, and denote a line in E by the pair (i, j) of buses it connects and the index i denotes the bus that is closer to the feeder. Bus 0 denotes the feeder, which has fixed voltage but flexible power injection to balance the loads; each of the other buses $i \in N\backslash\{0\}$ represents an aggregator that can participate in demand response. For convenience we call aggregator i as user i, which actually represents a customer or a group of customers that are connected to bus i and join the demand response system as a single entity.

For each link $(i, j) \in E$, let $z_{ij} = r_{ij} + \mathbf{i}x_{ij}$ be the impedance on line (i, j), and $S_{i,j} = P_{i,j} + \mathrm{i}Q_{i,j}$ and $I_{i,j}$ the complex power and current flowing from bus i to bus j. At each bus $i \in N$, let $s_i = p_i + \mathbf{i}q_i$ be the complex load and V_i the complex voltage. As customary, we assume that the complex voltage V_0 on the feeder is given and fixed. Here we replicate the branch flow model which is provided in the previous chapter:

for each $(i,j) \in E$,

$$\frac{P_{i,j}^2 + Q_{i,j}^2}{v_i} = \ell_{i,j}, \tag{4.1}$$

$$P_{i,j} = \sum_{h:(j,h)\in E} P_{j,h} + r_{i,j}\ell_{i,j} + p_j, \tag{4.2}$$

$$Q_{i,j} = \sum_{h:(j,h)\in E} Q_{j,h} + x_{i,j}\ell_{i,j} + q_j, \tag{4.3}$$

$$v_i - v_j = 2(r_{i,j}P_{i,j} + x_{i,j}Q_{i,j}) - (r_{i,j}^2 + x_{i,j}^2)\ell_{i,j}, \tag{4.4}$$

where $\ell_{i,j} := |I_{i,j}|^2$, $v_i := |V_i|^2$. Each user $i \in N\backslash\{0\}$ achieves certain utility $f_i(p_i)$ when its (real) power consumption is p_i. The utility function $f_i(\cdot)$ is usually assumed to be continuous, nondecreasing, and concave. Furthermore, there are the following operating constraints for each $i \in N\backslash\{0\}$:

$$\underline{v}_i \le v_i \le \bar{v}_i, i = 1, \cdots, n, \tag{4.5}$$

$$\underline{q}_i \le q_i \le \bar{q}_i, i = 1, \cdots, n, \tag{4.6}$$

$$\underline{p}_i \le p_i \le \bar{p}_i, i = 1, \cdots, n. \tag{4.7}$$

The electricity is delivered from the main grid to the radial distribution network through the feeder (i.e., the bus 0). The total (real) power supply P_0 is given by $P_0 := \sum_{j:(0,j)\in E} P_{0,j}$.

We consider a situation where the power supply P_0 is constrained by an upper bound $\bar{P}_0$, i.e.,

$$P_0 = \sum_{j:(0,j)\in E} P_{0,j} \le \bar{P}_0. \tag{4.8}$$

Under such a situation, we would like to design distributed mechanisms to guide each user i to choose a proper load p_i, so as to i) meet the supply constraint (4.8) as well as the power flow constraints and operating constraints listed in (4.1–4.7) and ii) maximize the aggregate user utilities and minimize the power supply costs and power line losses. This demand response problem is formulated as the following optimal power

flow problem (OPF):

$$\textbf{OPF:} \max_{P,Q,l,v,p,q} \quad \sum_{i=1}^{n} f_i(p_i) - C_0(P_0) - \rho \sum_{(i,j)\in E} r_{i,j}\ell_{i,j}$$
$$\text{s.t.} \quad (4.1) - (4.8),$$

where ρ is a trade off parameter.[1] Throughout the chapter, we assume that the feasible set of this problem is nonempty. In the following, we will develop two distributed OPF algorithms for demand response.

4.2.2 A decentralized optimization algorithm: predictor corrector proximal multiplier (PCPM)

In this chapter we focus on using the decentralized algorithm, predictor corrector proximal multiplier (PCPM) [61] to develop distributed algorithms for demand response. Consider the following convex problem:

$$\max_{x\in X, y\in Y} \quad f(x) + g(y) \tag{4.9a}$$
$$\text{s.t.} \quad Ax + By = C \ . \tag{4.9b}$$

Introduce the Lagrangian variable z for constraint (4.9b).

The algorithm PCPM is given as follows:

1. Initially set $k \leftarrow 0$ and randomly choose initial (x^0, y^0, z^0).

2. For each $k \geq 0$, update a virtual variable $\hat{z}^k := z^k + \gamma(Ax^k - By^k - C)$. Here $\gamma > 0$ is a constant parameter.

3. Based on the virtual variable $\hat{z}^k$, update x, y according to:

$$x^{k+1} = \arg\min_{x\in X}\{f(x) + (\hat{z}^k)^T Ax + (1/(2\gamma))||x - x^k||^2\},$$
$$y^{k+1} = \arg\min_{y\in Y}\{g(y) + (\hat{z}^k)^T By + (1/(2\gamma))||y - y^k||^2\}.$$

[1] Here we just consider demand managements at one instance for the simplicity of exposition. The model and the following results in this chapter can be easily extended to demand management over multiple instances. We provide a detailed example in Section 4.5.

4. z is updated according to $z^{k+1} = z^k + \gamma(Ax^{k+1} + By^{k+1} - C)$.

5. $k \leftarrow k + 1$, and go to step 2).

From the algorithm, we see that PCPM is highly decomposable. In terms of convergence, it has been shown in [61] that as long as strong duality holds for the convex problem (4.9), the algorithm will converge to a primal-dual optimal solution (x^*, y^*, z^*) for sufficient small positive γ.

4.2.3 Convexification of problem OPF

OPF is non-convex due to the quadratic equality constraints in (4.1) and thus difficult to solve. Moreover, most decentralized algorithms require convexity to ensure convergence, e.g., PCPM as described in 4.2.2. We therefore consider the following convex relaxation of OPF:

$$\begin{aligned} \textbf{ROPF:} \max_{P,Q,l,v,p,q} \quad & \sum_{i=1}^{n} f_i(p_i) - C_0(P_0) - \rho \sum_{(i,j)\in E} r_{i,j} l_{i,j} \\ s.t. \quad & (4.2) - (4.7) \\ & \frac{P_{i,j}^2 + Q_{i,j}^2}{v_i} \leq l_{i,j},\ (i,j) \in E, \end{aligned} \tag{4.10}$$

where the equality constraints (4.1) are relaxed to the inequality constraints (4.10). ROPF provides a lower bound on OPF. For an optimal solution $X^* := (P^*, Q^*, \ell^*, v^*, p^*, q^*)$ of ROPF, if the equality in (4.10) is attained at X^*, then X^* is also a solution to OPF. We call ROPF an *exact relaxation* of OPF if every solution to ROPF is also a solution to OPF, and vice versa. In the previous chapter we have studied whether and when ROPF is an exact relaxation of OPF for the radial networks. It is shown that the relaxation is exact provided that instead there are no upper bounds on the voltage magnitudes and certain other conditions hold, which are verified to hold for many real-world distribution systems. Moreover, the upper bounds on the voltage magnitudes for the relaxation solution are characterized.

The benefit of convexity is that convexity does not only facilitates the design of efficient pricing schemes for power market and demand response, but it also facilitates the development of tractable, scalable and distributed algorithms for system operations. Hence the conditions for exact relaxation of OPF to ROPF

specified in the previous chapter are important for our demand response design. In the rest of the chapter, we will assume that ROPF is an exact relaxation of OPF and strong duality holds for ROPF. As ROPF is an exact relaxation of OPF, in the rest of the chapter we will just focus on solving the convex optimization problem ROPF.

4.3 Demand management through the LSE

In this section, we consider the setting where the radial distribution network is served by a single load serving entity (LSE), which coordinates the end users' demand response decisions to solve Problem ROPF (OPF) by setting the right prices. Here we consider that the utility functions and constraints (4.6-4.7) are private information of the users, while the LSE has the network information, i.e., power loss $\sum_{(i,j)\in E} r_{i,j}\ell_{i,j}$ and the constraints (4.2-4.5,4.8,4.10). Each user i chooses power consumption according to certain price signal μ_i sent by LSE, and the LSE adapts the price signal $\mu := (\mu_1, \ldots, \mu_n)$ to coordinate users' consumptions. The price signal μ_i can be implemented as the actual price of electricity usage or just a control signal that is used to coordinate users' decisions. Each user $i \in N\backslash\{0\}$ is assumed to choose p_i to maximize its net utility, i.e., user utility minus payment:

$$\textbf{DR-User:} \quad \max_{p_i} \quad f_i(p_i) - \mu_i p_i$$
$$\text{s.t.} \quad \underline{p}_i \le p_i \le \bar{p}_i.$$

Since the reactive power q_i is not directly involved in the net utility of user i, we assume that user i is willing to report the feasible range $[\underline{q}_i, \bar{q}_i]$ for q_i to the LSE.[2] Hence, the LSE has the following information, the power loss $\sum_{(i,j)\in E} r_{i,j}\ell_{i,j}$ and the constraints (4.2-4.6, 4.8, 4.10). Given price μ, the LSE maximizes its

[2]Note that in practice, VAR control is usually carried out by the LSE. So, it is reasonable to assume that the LSE knows the feasible range of reactive power.

net benefit, i.e., the total payment received minus the power loss:

$$\textbf{DR-LSE:} \max_{P,Q,l,v,p,q} \quad \sum_{i=1}^{n} \mu_i p_i - \rho \sum_{(i,j)\in E} r_{i,j}\ell_{i,j}$$
$$\text{s.t.} \quad (4.2)-(4.6), (4.8), (4.10).$$

In the rest of the section, we show how the LSE chooses the price signal μ to coordinate the users' demand response decisions so as to solve Problem ROPF.

Definition 4.1. *The price $\mu^* = (\mu_1^*, \ldots, \mu_n^*)$ and the variable $(P^*, Q^*, \ell^*, v^*, p^*, q^*)$ are in equilibrium if i) p_i^* is an optimal solution of DR-User for each user i given the price μ_i^*, and ii) $(P^*, Q^*, \ell^*, v^*, p^*, q^*)$ is an optimal solution of DR-LSE for the LSE given the price μ^*.*

The above definition implies that if such an equilibrium $(\mu^*; P^*, Q^*, \ell^*, v^*, p^*, q^*)$ exists, μ^* can serve as the price signal for the LSE to guide users' decisions. The following result establishes the existence of the equilibrium $(\mu^*; P^*, Q^*, \ell^*, v^*, p^*, q^*)$ and characterizes its properties. Let $\lambda_{i,j}$, $\underline{\xi}_i$, $\bar{\xi}_i$ denote the corresponding Lagrangian dual variables of ROPF for the constraint (4.2), and the constraint (4.7) respectively.

Theorem 4.1. *There exists at least one equilibrium $(\mu^*; P^*, Q^*, \ell^*, v^*, p^*, q^*)$. Moreover, a tuple $(\mu^*; P^*, Q^*, \ell^*, v^*, p^*, q^*)$ is an equilibrium if and only if $(P^*, Q^*, \ell^*, v^*, p^*, q^*)$ is an optimal solution of ROPF and for each $i > 0$, $\mu_i^* = f_i'(p_i^*) - \xi_i^* = \lambda_{\pi(i),i}^*$, where $\pi(i)$ is the parent of bus i.*

Proof. First note that problems ROPF, DR-User, and DR-LSE are convex problems and a strong duality holds for all of them. The main idea of the proof is to compare the KKT optimality conditions for these convex problems.

Let $\alpha = (\lambda_{i,j}, \theta_{i,j}, \omega_{i,j}, \underline{\gamma}_i, \bar{\gamma}_i, \underline{\eta}_i, \bar{\eta}_i, \underline{\xi}_i, \underline{\xi}_i, \kappa_0, \mu_{i,j})$ be the Lagrangian dual variables of ROPF corresponding to the constraints (4.2–4.10) respectively. Given an optimal primal-dual pair $(P^*, Q^*, \ell^*, v^*, p^*, q^*; \alpha^*)$ of ROPF, $(P^*, Q^*, \ell^*, v^*, p^*, q^*; \alpha^*)$ satisfies the KKT condition of ROPF. This implies that $f_i'(p_i^*) - \underline{\xi}_i^* + \bar{\xi}_i^* = \lambda_{\pi(i),i}$. Let $\mu_i^* = f_i'(p_i^*) - \underline{\xi}_i^* + \bar{\xi}_i^* = \lambda_{\pi(i),i}^*$ for all $i = 1, \cdots, n$. Then the KKT condition for ROPF implies that $(p_i^*, \underline{\xi}_i^*, \bar{\xi}_i^*)$ satisfies the KKT condition for problem DR-User for each $i = 1, \cdots, n$; and $(P^*, Q^*, \ell^*, v^*, p^*, q^*, \beta^*)$ satisfies the KKT condition for DR-LSE where

$\beta^* = (\lambda^*_{i,j}, \theta^*_{i,j}, \omega^*_{i,j}, \kappa^*_0, \underline{\gamma^*}_i, \bar{\gamma^*}_i, \underline{\eta^*}_i, \bar{\eta^*}_i, \mu^*_{i,j})$. Therefore, i) p^*_i is an optimal solution of DR-User for each user i given the price μ^*_i, and ii) $(P^*, Q^*, \ell^*, v^*, p^*, q^*)$ is an optimal solution of DR-LSE for the LSE given the price μ^*.

On the other hand, suppose $(\mu^*; P^*, Q^*, \ell^*, v^*, p^*, q^*)$ is an equilibrium. The KKT conditions of DR-LSE and DR-User imply that there exists a dual variable α^* such that $(P^*, Q^*, \ell^*, v^*, p^*, q^*; \alpha^*)$ satisfies the KKT condition of ROPF. Thus $(P^*, Q^*, \ell^*, v^*, p^*, q^*)$ is an optimal solution of problem ROPF. □

4.3.1 Distributed algorithm

Following the algorithm *predictor corrector proximal multiplier (PCPM)* [61], we propose a distributed learning algorithm to achieve an equilibrium $(\mu^*; P^*, Q^*, \ell^*, v^*, p^*, q^*)$:

1. Initially set $k \leftarrow 0$. The LSE randomly chooses initial price μ^k_i and initial p^k_i for each bus i. Each user i randomly chooses initial $\hat{p}^k_i$ and returns $\hat{p}^k_i$ to the LSE.

2. For each $k \geq 0$, the LSE sends a virtual price signal $\hat{\mu}^k_i := \mu^k_i + \gamma(\hat{p}^k - p^k)$ to each bus i. Here $\gamma > 0$ is a constant parameter.

3. Based on the virtual price $\hat{\mu}^k_i$, each bus $i \in N \backslash \{0\}$ solves the following problem:

$$\begin{aligned} \max_{\hat{p}_i} \quad & f_i(\hat{p}_i) - \hat{\mu}^k_i \hat{p}_i - \frac{1}{2\gamma}||\hat{p}_i - \hat{p}^k_i||^2 \\ \text{s.t.} \quad & \underline{p}_i \leq \hat{p}_i \leq \bar{p}_i. \end{aligned}$$

   The optimal $\hat{p}_i$ is set as $\hat{p}^{k+1}_i$.

4. The LSE solves the following problem:

$$\begin{aligned} \max_{P,Q,l,v,p,q} \quad & (\hat{\mu}^k)^T p - \rho \sum_{(i,j)\in E} r_{i,j}\ell_{i,j} - \frac{1}{2\gamma}||p - p^k||_2^2 \\ \text{s.t.} \quad & (4.2) - (4.6), (4.8), (4.10). \end{aligned}$$

   The optimal p is set as p^{k+1}_i.

5. Each bus i returns $\hat{p}_i^{k+1}$ to the LSE and the LSE updates the price μ as $\mu^{k+1} = \mu^k + \gamma(\hat{p}^{k+1} - p^{k+1})$.

6. $k \leftarrow k+1$, and go to step 2).

For sufficiently small γ, $(\mu^k; P^k, Q^k, \ell^k, v^k, p^k, q^k)$ will converge to an equilibrium, and $\hat{p}^k - p^k$ and $\hat{\mu}^k - \mu^k$ will become zero [61]. Numerical experiments show that this algorithm converges to the optimum of problem ROPF(OPF) very fast.

4.4 A fully decentralized algorithm

In this section, we develop a fully distributed OPF algorithm for demand response, where the end users make and coordinate their local demand response decisions through local communication with their neighbors. Specifically, we assume that each user has certain computation ability to decide a set of local variables of the OPF. The composition of those variables determines the global status of the power flow over the distribution network. We also assume that there is two way communication available between any two users that are directly connected in the distribution network. In the decentralized OPF algorithm, at each iteration each user makes decisions about the local variables, communicates those decisions with neighbors, and then updates their local variables and repeats the process.

Before establishing the algorithm, let us define the local decision variables for each user. Let $\pi(i)$ be the parent of bus i and $\delta(i)$ be the direct children of bus i. The local decision variables for each bus are:

- For bus 0, P_0, v_0, where v_0 is fixed by convention.

- For bus $i > 0$, $P_{\pi(i),i}$, $Q_{\pi(i),i}$, $\ell_{\pi(i),i}$, p_i,q_i,v_i,$\hat{v}_i$. Here $\hat{v}_i$ is bus i's estimation about its parent's voltage $v_{\pi(i)}$. To simplify the notations, we denote $P_{\pi(i),i}$, $Q_{\pi(i),i}$, $\ell_{\pi(i),i}$ as P_i, Q_i, ℓ_i; and $r_{\pi(i),i}$, $x_{\pi(i),i}$ as r_i, x_i.

With the new notations, OPF can be rewritten as:

$$\min_{P,Q,l,v,p,q} \quad \sum_{i=1}^{n} f_i(p_i) - C_0(P_0) - \sum_{i=1}^{n} r_i l_i \tag{4.11a}$$

$$\text{s.t.} \quad P_0 = \sum_{j:(0,j)\in E} P_j, \tag{4.11b}$$

$$P_i = \sum_{j\in\delta(i)} P_j + r_i l_i + p_i, i \in N\backslash\{0\} \tag{4.11c}$$

$$Q_i = \sum_{j\in\delta(i)} Q_j + x_i l_i + q_i, i \in N\backslash\{0\} \tag{4.11d}$$

$$\hat{v}_i = v_{\pi(i)}, i \in N\backslash\{0\} \tag{4.11e}$$

$$P_0 \le \bar{P}, \tag{4.11f}$$

$$\underline{v}_i \le v_i \le \bar{v}_i, i \in N\backslash\{0\} \tag{4.11g}$$

$$\underline{p}_i \le p_i \le \bar{p}_i, i \in N\backslash\{0\} \tag{4.11h}$$

$$\underline{q}_i \le q_i \le \bar{q}_i, i \in N\backslash\{0\} \tag{4.11i}$$

$$\frac{P_i^2 + Q_i^2}{\hat{v}_i} \le l_i, i \in N\backslash\{0\} \tag{4.11j}$$

$$\hat{v}_i - v_i = 2(r_i P_i + x_i Q_i) - (r_i^2 + x_i^2) l_i, i \in N\backslash\{0\}\,. \tag{4.11k}$$

The new formulation has the following properties, which can be utilized for the design of distributed algorithms:

- The objective function (4.11a) is fully decomposable.
- Constraints (4.11b-4.11e) are linear coupled constraints but each constraint only constrains "local" information; namely, each constraint is defined over the local variables of one node and its direct neighbors over the radial network.
- Constraints (4.11f-4.11k) are just local constraints that are defined over bus i's local decision variables.

Then we can apply algorithm *PCPM* to define a decentralized algorithm. We will use PCPM to decouple those linear coupled constraints (4.11b-4.11e). Let the Lagrangian dual variable corresponding to constraint (4.11b) be λ_0 and dual variables corresponding to constraints (4.11c-4.11e) be $\lambda_i, \theta_i, \omega_i$ for each $i \in N\backslash\{0\}$. In the following distributed algorithm, node 0 takes charge of updating λ_0 and node $i \in N\backslash\{0\}$ takes charge

of updating $\lambda_i, \theta_i, \omega_i$. Now let us introduce the distributed demand response algorithm which converges to a global optimal solution of the OPF.

1. Initially set $k \leftarrow 0$. Node 0 randomly chooses P_0^k and λ_0^k and node $i \in N\backslash\{0\}$ randomly chooses P_i^k, Q_i^k, ℓ_i^k, p_i^k,q_i^k,v_i^k,$\hat{v}_i^k$ and the dual variables $\lambda_i^k, \theta_i^k, \omega_i^k$. Each node $i \in N\backslash\{0\}$ sends the primal variables P_i^k, Q_i^k, ℓ_i^k to its parent $\pi(i)$, and each node $i \in N$ except the leaves in the network sends v_i^k to its children. Note that v_0^k is fixed for any k.

2. For each $k \geq 0$, node 0 sends a virtual dual signal $\hat{\lambda}_0^k := \lambda_0^k + \gamma(P_0^k - \sum_{j:(0,j)\in E} P_j^k)$ to its children; and each node $i \in N\backslash\{0\}$ except the leaves in the network sends the following virtual signals to its children:

$$\begin{aligned}
\hat{\lambda}_i^k &= \lambda_i^k + \gamma\left(P_i^k - \left(\sum_{j\in\delta(i)} P_j^k + r_i l_i^k + p_i^k\right)\right), \\
\hat{\theta}_i^k &= \theta_i^k + \gamma\left(Q_i^k - \left(\sum_{j\in\delta(i)} Q_j^k + x_i l_i^k + q_i^k\right)\right);
\end{aligned}$$

and each node $i \in N\backslash\{0\}$ sends the following virtual signals to its parent:

$$\hat{\omega}_i^k = \omega_i^k + \gamma(\hat{v}_i^k - v_{\pi(i)}^k).$$

Here $\gamma > 0$ is a constant parameter.

3. Each node updates its local primal variables according to the following rules.

Case 1: Node 0 solves the following problem:

$$\begin{aligned}
\max_{P_0} \quad & C_0(P_0) + \hat{\lambda}_0^k P_0 + \frac{1}{2\gamma}||P_0 - P_0^k||^2 \\
\text{s.t.} \quad & P_0 \leq \bar{P}.
\end{aligned}$$

The optimal P_0 is set as P_0^{k+1}.

Case 2: Each node i such that $(0,i) \in E$, solves the following problem:

$$\begin{aligned}
\max \quad & f_i(p_i) - r_i\ell_i - \hat{\lambda}_0^k P_i + \hat{\lambda}_i(P_i - r_i\ell_i - p_i) + \hat{\theta}_i(Q_i - x_i\ell_i - q_i) + \hat{\omega}_i\hat{v}_i - \sum_{j:(i,j)\in E} \hat{\omega}_j v_i \\
& + \frac{1}{2\gamma}\left((P_i - P_i^k)^2 + (Q_i - Q_i^k)^2 + (\ell_i - \ell_i^k)^2 + (p_i - p_i^k)^2 + (q_i - q_i^k)^2\right. \\
& \left. + (v_i - v_i^k)^2 + (\hat{v}_i - \hat{v}_i^k)^2\right) \\
\text{over} \quad & P_i, Q_i, \ell_i, p_i, q_i, v_i, \hat{v}_i \\
\text{s.t.} \quad & (4.11k - 4.11j).
\end{aligned}$$

The optimal $P_i, Q_i, \ell_i, p_i, q_i, v_i, \hat{v}_i$ is set as $P_i^{k+1}, Q_i^{k+1}, \ell_i^{k+1}, p_i^{k+1}, q_i^{k+1}, v_i^{k+1}, \hat{v}_i^{k+1}$.

Case 3: Each node i such that $(0,i) \notin E$ solves the following problem:

$$\begin{aligned}
\max \quad & f_i(p_i) - r_i\ell_i - \lambda_{\pi(i)} P_i - \theta_{\pi(i)} Q_i + \hat{\lambda}_i(P_i - r_i\ell_i - p_i) + \hat{\theta}_i(Q_i - x_i\ell_i - q_i) \\
& + \hat{\omega}_i\hat{v}_i - \sum_{j:(i,j)\in E} \hat{\omega}_j v_i + \frac{1}{2\gamma}\left((P_i - P_i^k)^2 + (Q_i - Q_i^k)^2\right. \\
& \left. + (\ell_i - \ell_i^k)^2 + (p_i - p_i^k)^2 + (q_i - q_i^k)^2 + (v_i - v_i^k)^2 + (\hat{v}_i - \hat{v}_i^k)^2\right) \\
\text{over} \quad & P_i, Q_i, \ell_i, p_i, q_i, v_i, \hat{v}_i \\
\text{s.t.} \quad & (4.11k - 4.11j).
\end{aligned}$$

The optimal $P_i, Q_i, \ell_i, p_i, q_i, v_i, \hat{v}_i$ is set as $P_i^{k+1}, Q_i^{k+1}, \ell_i^{k+1}, p_i^{k+1}, q_i^{k+1}, v_i^{k+1}, \hat{v}_i^{k+1}$.

4. Each node $i \in N\backslash\{0\}$ sends the primal variables P_i^{k+1}, Q_i^{k+1}, ℓ_i^{k+1} to its parent $\pi(i)$, and each node $i \in N$ except leaves in the network send v_i^{k+1}. Note that v_0^k is fixed as v^* for any k. Then node 0 updates the dual signal $\lambda_0^{k+1} := \lambda_0^k + \gamma(P_0^{k+1} - \sum_{j:(0,j)\in E} P_j^{k+1})$ to its children, and each node $i \in N\backslash\{0\}$ except the leaves in the network updates the following variables:

$$\begin{aligned}
\lambda_i^{k+1} &= \lambda_i^k + \gamma\left(P_i^{k+1} - \left(\sum_{j\in\delta(i)} P_j^{k+1} + r_i l_i^{k+1} + p_i^{k+1}\right)\right), \\
\theta_i^{k+1} &= \theta_i^k + \gamma\left(Q_i^{k+1} - \left(\sum_{j\in\delta(i)} Q_j^{k+1} + x_i l_i^{k+1} + q_i^{k+1}\right)\right), \\
\omega_i^{k+1} &= \omega_i^k + \gamma(\hat{v}_i^{k+1} - v_{\pi(i)}^{k+1}).
\end{aligned}$$

5. $k \leftarrow k+1$, and go to step 2).

For sufficiently small γ, the algorithm will converge to the optimal solutions. Notice that in the distributed algorithm, each node only needs to determine a few variables by solving a small optimization problem.

4.5 Generalization to demand response over multiple time instants

In the previous sections, we have studied demand management at one instance. The method and results can be easily extended to demand management over multiple instances. The distribution network may need to schedule supply to meet the demand for each time period of the next day, represented as a set of time slots $\mathcal{T} = \{1, 2, \cdots, T\}$, and its objective is to maximize the aggregate user utilities minus the power line losses over the whole period of $\mathcal{T}$. Let $\mathbf{P} = (P(1), \ldots, P(T))$, $\mathbf{Q} = (Q(1), \ldots, Q(T))$, $\mathbf{l} = (\ell(1), \ldots, \ell(T))$,$\mathbf{v} = (v(1), \ldots, v(T))$, $\mathbf{p} = (p(1), \ldots, p(T))$, and $\mathbf{q} = (q(1), \ldots, q(T))$ be the corresponding variables of the power network at different times. Mathematically, the load management problem over multiple instances can be formulated as the following optimization problem:[3]

$$\textbf{MOPF:} \max_{\mathbf{P},\mathbf{Q},\mathbf{l},\mathbf{v},\mathbf{p},\mathbf{q}} \quad \sum_{i=1}^{n} f_i\left(\mathbf{p}_i\right) - \rho \sum_{t\in\mathcal{T},(i,j)\in E} r_{i,j}\ell_{i,j}(t) - \sum_{t\in\mathcal{T}} C_t\left(\sum_{j:(0,j)\in E} P_{0,j}(t)\right)$$

$$\text{s.t.} \quad (4.1)-(4.5), \forall t \in \mathcal{T}$$

$$\underline{q}_i(t) \le q_i(t) \le \bar{q}_i(t), \forall t \in \mathcal{T}, \tag{4.12}$$

$$\underline{p}_i(t) \le p_i(t) \le \bar{p}_i(t), \forall t \in \mathcal{T}, \tag{4.13}$$

$$\sum_{t\in\mathcal{T}} p_i(t) \ge \underline{d}_i, \ \forall i \in N\backslash\{0\}. \tag{4.14}$$

Here $C_t\left(\cdot\right)$ is a cost function of the total real power injected to the network through the feeder at time t. It can be interpreted as the cost in power provisioning for the LSE. The cost function $C_t(\cdot)$ is assumed to be nondecreasing and convex. Compared with OPF, MOPF has the following differences:

1. Instead of constraining the power supply P_0 at the feeder as in (4.8), MOPF allows greater elasticity in

[3]Note that the requirements and constraints on demand (4.12-4.14) and the utility function f_i can be modeled in a more complicated form if we consider every appliance for each user; see Chapter 2 for the detailed user models.

power supply subject to a cost $C_t\left(P_0(t)\right)$;

2. The utility function $f_i(p_i)$ in OPF for one instance is replaced by the utility function $f_i(\mathbf{p}_i) = f_i(p_i(1), \cdots, p_i(T))$ which characterizes user i's utility over the whole period of $\mathcal{T}$ given the demand profile $(p_i(1), \cdots, p_i(T))$;

3. The constraints (4.6), (4.7) that bound power consumption for each user i become time-dependent constraints (4.12), (4.13).

4. There is a constraint (4.14) on the total real power consumption over the whole period of $\mathcal{T}$, corresponding to a minimum demand that is required to power basic daily routines for user i.

The convex relaxation of MOPF is given as follows:

$$\textbf{RMOPF:} \max_{\mathbf{P},\mathbf{Q},\mathbf{l},\mathbf{v},\mathbf{p},\mathbf{q}} \quad \sum_{i=1}^{n} f_i\left(\mathbf{p}_i\right) - \rho \sum_{t\in\mathcal{T},(i,j)\in E} r_{i,j}\ell_{i,j}(t) - \sum_{t\in\mathcal{T}} C_t\left(\sum_{j:(0,j)\in E} P_{0,j}(t)\right)$$
$$\text{s.t.} \quad (4.2)-(4.5), (4.10-4.14).$$

Provided that the sufficient conditions for exact relaxation in the previous chapter are satisfied, RMOPF is also an exact relaxation of MOPF. Similarly, in the rest of this chapter, we assume that RMOPF is an exact relaxation and strong duality holds for RMOPF.

As Problem MOPF(RMOPF) and Problem OPF(ROPF) are almost in the same form except the additional local constraint (4.14), all the results in Section 4.3 and Section 4.4 can be readily extended to MOPF (RMOPF). For the space limit, we just show an example of extending the methods in Section 4.3 to MOPF (RMOPF). In this demand response setting, the utility functions f_i and the constraints (4.12-4.14) are private information of the users, while the LSE has the network information. Each user i chooses power consumption according to certain price signal $\{\mu_i(t)\}_{t\in\mathcal{T}}$ sent by LSE, and the LSE adjusts the price signal $\{\mu_i(t)\}_{i=1,\cdots,n}^{t=1,\cdots,T}$ to coordinate the users' consumption decisions. We have the following distributed learning algorithm to achieve the optimum of RMOPF(MOPF):

1. Initially set $k \leftarrow 0$. The LSE randomly chooses initial price $\mu_i^k(t)$ and inital $p_i^k(t)$ for each bus i at each time $t \in \mathcal{T}$. Each user i randomly chooses initial $\hat{p}_i^k(t)$ for each time $t \in \mathcal{T}$ with $\sum_{t\in\mathcal{T}} p_i^k(t) \geq \underline{d}_i$ and

returns $\hat{p}_i^k(t)$ to the LSE.

2. For each $k \geq 0$, the LSE sends a virtual price signal $\hat{\mu}_i^k(t) := \mu_i^k(t) + \gamma(\hat{p}_i^k(t) - p_i^k(t))$ to bus i. Here $\gamma \geq 0$ is a constant parameter.

3. Based on the virtual price $\hat{\mu}_i^k(t)$, each bus $i \in N \backslash \{0\}$ solves the following problem:

$$\begin{aligned} \max_{\hat{\mathbf{p}}_i} \quad & f_i(\hat{\mathbf{p}}_i) - \sum_{t\in\mathcal{T}} \hat{\mu}_i^k(t)\hat{p}_i(t) - \frac{1}{2\gamma}\sum_{t\in\mathcal{T}} ||\hat{p}_i(t) - \hat{p}_i^k(t)||^2 \\ \text{s.t.} \quad & \underline{p}_i(t) \leq \hat{p}_i(t) \leq \bar{p}_i(t), \forall t \in \mathcal{T} \\ & \sum_{t\in\mathcal{T}} \hat{p}_i(t) \geq \underline{d}_i. \end{aligned}$$

   The optimal $\hat{p}_i(t)$ is set as $\hat{p}_i^{k+1}(t)$.

4. For each time $t \in \mathcal{T}$, the LSE solves the following problem:

$$\begin{aligned} \max \quad & (\hat{\mu}^k(t))^T p(t) - \rho \sum_{(i,j)\in E} r_{i,j}\ell_{i,j}(t) - C_t\left(\sum_{j:(0,j)\in E} P_{0,j}(t)\right) - \frac{1}{2\gamma}||p(t) - p^k(t)||_2^2 \\ \text{over:} \quad & P(t), Q(t), \ell(t), v(t), p(t), q(t) \\ \text{s.t.} \quad & (4.2 - 4.5), (4.10), (4.12). \end{aligned}$$

   The optimal $p_i(t)$ is set as $p_i^{k+1}(t)$.

5. Each user i returns $\hat{p}_i^k(t)$ to the LSE and the LSE updates the price μ as $\mu^{k+1} = \mu^k + \gamma(\hat{p}^k(t) - p^k(t))$.

6. $k \leftarrow k + 1$, and go to step 2).

4.6 Case study

This section provides numerical examples to complement the analysis in previous sections. We apply the algorithm developed in Section 4.4 to a practical distribution circuit of Southern California Edison (SCE) with 56 buses, as shown in Fig. 3.3. The corresponding network data including the line impedances, the peak MVA demand of loads, and the nameplate capacity of the shunt capacitors and the photovoltaic generations are given in Table. 3.3. Note that there is a photovoltaic (PV) generator located at bus 45. Since the focus of

this chapter is to study demand response in power networks, so in the simulation we remove the PV generator. Previous chapter has shown that this 56-bus circuit satisfies the sufficient conditions for the exact relaxation of OPF to ROPF. Therefore, we can apply the proposed algorithm for the demand response in this circuit. In the simulation, the user utility function $f_i(p_i)$ is set to the quadratic form $f_i(p_i) = -a_i(p_i - \bar{p}_i)^2 + a_i(\bar{p}_i)^2$ where a_i is randomly drawn from $[2, 5]$. For each bus i, set $\bar{p}_i$ and $\bar{q}_i$ to the peak demand and $\underline{p}_i$ to the half of the peak demand. If there is no shunt capacitor attached to bus i, we set $\underline{q}_i$ to the half of the peak demand as well, and if there is a shunt capacitor attached, we set $\underline{q}_i$ to the negative of the nameplate capacity. We set $\gamma = 0.01$, and $\bar{P}_0 = 2.5$MVA.

4.6.1 Load management with an LSE

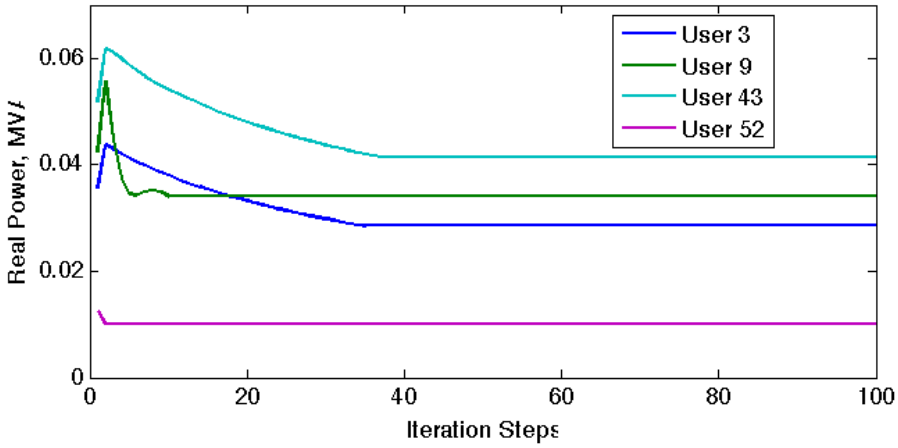


Figure 4.1. Dynamics of the distributed demand response algorithm: Bus i's calculated $\hat{p}_i$.

Figures 4.1 and 4.2 show the dynamics of the distributed algorithm proposed in Section 4.3.1. We see that the algorithm converges very fast for this distribution system. We also solve problem ROPF by using CVX toolbox [56], which implements a centralized algorithm, and verify that it gives the same solution as our distributed algorithm. We further verify that the optimal solution of ROPF is a feasible point of OPF, i.e., ROPF is an exact relaxation of OPF.

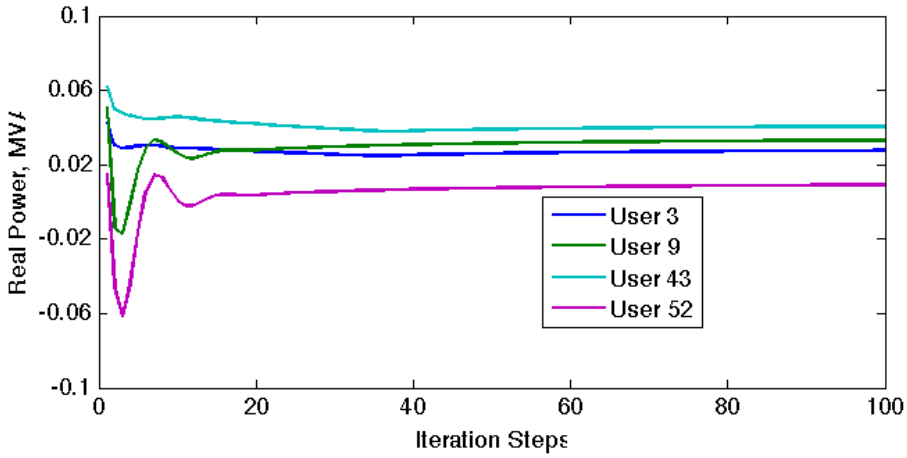


Figure 4.2. Dynamics of the distributed demand response algorithm: LSE's calculated p_i for each bus i.

4.6.2 Fully decentralized load management

Fig. 4.3 shows the dynamics of the distributed algorithm proposed in Section 4.4. We see that the algorithm converges to the optimal solution fast for this distribution system. Notice that since at each iteration step, each node only needs to solve a small optimization problem and the algorithm is highly parallel, the total running time is very fast.

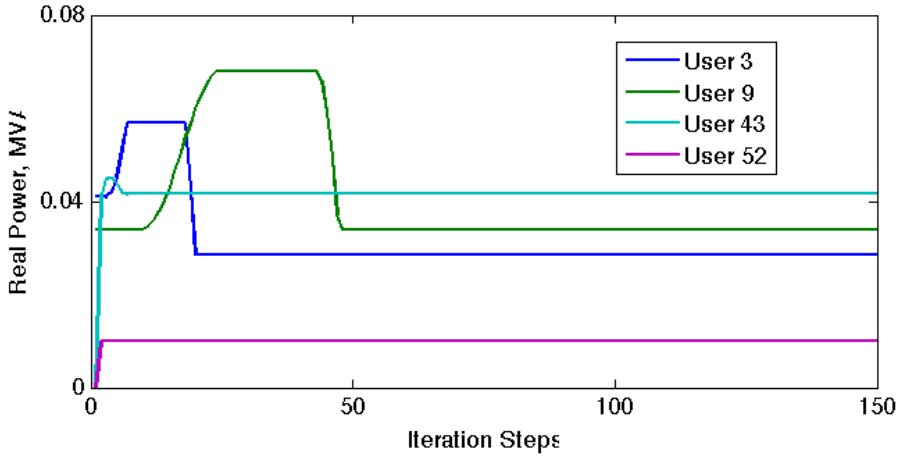


Figure 4.3. Dynamics of the distributed demand response algorithm: Bus i's decision $\hat{p}_i$.

4.7 Conclusion

In this chapter, we have studied demand response in the radial distribution network with power flow constraints and operating constraints, by formulating it as an optimal power flow problem. We discuss the exact convex relaxation of the OPF problem, based on which we propose two fully distributed algorithms. In the first one, the LSE set the prices to coordinate users' demand response decisions and in the second one, the end users make and coordinate their local demand response decisions through local communication with their direct neighbors in the distribution network. Numerical examples show that the proposed algorithm converges fast for the real-world distribution circuits.

Chapter 5

Economic Automatic Generation Control

[]

The distributed algorithms derived in the previous chapters regard certain physical variables such as branch power flow as computable controls that can be instantaneously updated to arbitrary values, which is not usually the case for power systems. Hence these algorithms cannot be implemented as real-time controls that are required or desired, as amplified by mitigating fluctuations in renewable generation. In this chapter we study the real-time control mechanism–automatic generation control (AGC). We will show how to modify AGC to keep energy balanced and also to make energy allocation efficient at the same time.

5.1 Introduction

An interconnected electricity system can be described as a collection of subsystems, each of which is called a control area. Within each control area the mechanical power input to the synchronous generators is automatically regulated by automatic generation control (AGC). AGC uses deviations in generator speeds and/or frequency as control signals to invoke appropriate valve actions automatically in response to load changes. The main objectives of the conventional AGC in response to load changes is to (i) maintain system nominal frequency, and (ii) let each area absorbs its own load changes so as to maintain the scheduled net interchanges between control areas [62, 63]. The scheduled interchanges are determined at a much slower time scale than the AGC by individual generating companies considering economic dispatch among different generators.

Since the traditional loads (which are mainly passive) change slowly and are predictable with high accuracy, the conventional AGC does not incur much efficiency loss by requiring supply-demand balance within each control area after the load changes. However due to the penetration of renewable energy resources as well as demand response in the future power grid, the aggregate net loads, e.g., traditional passive loads plus electric vehicle loads minus renewable generations, can fluctuate fast and by a large amount. Therefore the conventional AGC can become much less economically efficient. We thus propose to relax the aforementioned second objective of the conventional AGC. We develop a novel modification of the conventional AGC to (i) maintain nominal frequency and (ii) allow revised power dispatch between different control areas to balance supply and demand within the whole interconnected electricity system instead of within each control area to achieve greater economic efficiency. We call this modified AGC the economic AGC.

We take a reverse and forward engineering approach to develop the economic AGC.[1] We first reverse-engineer the conventional AGC by showing that the power system dynamics with the conventional AGC can be interpreted as a partial primal-dual gradient algorithm to solve a certain optimization problem. We then engineer the optimization problem to include general generation costs and general power flow balance (which will guarantee supply-demand balance within the whole interconnected electricity system), and propose a distributed generation control scheme that is integrated into the AGC. Combined with [64] on distributed load control, this work lends the promise to develop a modeling framework and solution approach for systematic design of distributed, low-complexity generation and load control to achieve system-wide efficiency and robustness.

There has been a large amount of work on AGC in the last few decades, including, e.g., stability and optimum parameter setting [65], optimal or adaptive controller design [66–68], decentralized control [69, 70], and multilevel or multi timescale control [71, 72]; see also [63] and the references therein for a thorough and up-to-date review on AGC. Most of these work focuses on improving the control performance of AGC, such as stability and transient dynamics, but not on improving the economic efficiency. References [73–75] introduce approaches for AGC that also support an economic dispatch feature which operates at a slower time scale and interacts with AGC frequency stabilization function. References [74, 75] bring in the notion

[1] A similar approach has been used to design a decentralized optimal load control in our previous work [64].

of minimal regulation which reschedules the entire system generation and minimizes generation cost with respect to system-wide performance. Our work aims to improve the economic efficiency of AGC in response to the load changes as well; the difference is that instead of using different hierarchical control to improve AGC, we incorporate economic dispatch automatically and dynamically into the AGC. Moreover, our control is decentralized, where each control area can update its generation based only on local information and communications with neighboring areas.

This chapter is organized as follows. In Section 5.2, we introduce a dynamic power network model with AGC and the objective of the economic AGC. In Section 5.3, we reverse-engineer the conventional AGC and in Section 5.4, we design an economic AGC scheme from the insight obtained by the reverse engineering. In Section 5.5, we simulate and compare the convention AGC and the economic AGC. We conclude the chapter in Section 5.6.

5.2 System model

5.2.1 Dynamic network model with AGC

Consider a power transmission network, denoted by a graph $(\mathcal{N}, \mathcal{E})$, with a set $\mathcal{N} = \{1, \cdots, n\}$ of buses and a set $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$ of transmission lines connecting the buses. Here each bus may denote an aggregated bus or a control area. We make the following assumptions:

- The lines $(i, j) \in \mathcal{E}$ are lossless and characterized by their reactance x_{ij};
- The voltage magnitudes $|V_i|$ of buses $i \in \mathcal{N}$ are constants;
- Reactive power injections at the buses and reactive power flows on the lines are ignored.

We assume that $(\mathcal{N}, \mathcal{E})$ is connected and directed, with an arbitrary orientation such that if $(i, j) \in \mathcal{E}$, then $(j, i) \notin \mathcal{E}$. We use $i : i \to j$ and $k : j \to k$ respectively to denote the set of buses i such that $(i, j) \in \mathcal{E}$ and the set of buses j such that $(j, k) \in \mathcal{E}$. We study generation control when where there is a step change in net loads from their nominal (operating) points, which may result from a change in demand or in non-dispatchable renewable generation. To simplify notation, all the variables in this chapter represent deviations

from their nominal (operating) values.

Frequency Dynamics: For each bus j, let ω_j denote the frequency, P_j^M the mechanical power input, and P_j^L the total load. For a link (i, j), let P_{ij} denote the transmitted power form bus i to bus j. The frequency dynamics at bus j is given by the swing equation:

$$\dot{\omega}_j = -\frac{1}{M_j}\left(D_j\omega_j - P_j^M + P_j^L + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij}\right), \tag{5.1}$$

where M_j is the generator inertia and D_j is the damping constant at bus j.

Branch Flow Dynamics: Assume that the frequency deviation ω_j is small for each bus $j \in \mathcal{N}$. Then the deviations P_{ij} from the nominal branch flows follow the dynamics:

$$\dot{P}_{ij} = B_{ij}(\omega_i - \omega_j), \tag{5.2}$$

where

$$B_{ij} := \frac{|V_i||V_j|}{x_{ij}}\cos(\theta_i^0 - \theta_j^0)$$

is a constant determined by the nominal bus voltages and the line reactance. Here θ_i^0 is the nominal voltage phase angle of bus $i \in \mathcal{N}$. The detailed derivation is given in [64].

Turbine-Governor Control: For each generator, we consider a governor-turbine control model, where a speed governor senses a speed deviation and/or a power change command and converts it into appropriate valve action, and then a turbine converts the change in the valve position into the change in mechanical power output. The governor-turbine control is usually modeled as a two-state dynamic system. One state corresponds to the speed governor and the other state corresponds to the turbine. Since the time constant of the governor is much smaller than the turbine for most systems, we simplify the governor-turbine control model from two states to a single state P_j^M:

$$\dot{P}_j^M = -\frac{1}{T_j}\left(P_j^M - P_j^C + \frac{1}{R_j}\omega_j\right), \tag{5.3}$$

where P_j^C is the power change command and T_j and R_j are constant parameters. See [62] for a detailed introduction of governor-turbine control.

ACE-based control: In the conventional AGC, power change command P_j^C is adjusted automatically by the tie-line bias control which drives the area control errors (ACEs) to zero. For a bus j, the ACE is defined as:

$$\text{ACE}_j = B_j \omega_j + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij}.$$

The adjustment of power change command is given as follows:

$$\dot{P}_j^C = -K_j \left(B_j \omega_j + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij} \right), \tag{5.4}$$

where both B_j and K_j are positive constant parameters. In this chapter, we also call this AGC the ACE-based AGC.

In summary, the dynamic model with power control over a transmission network is given by equations (5.1)-(5.4). If the system is stable given certain load changes, then by simple analysis we can show that the conventional AGC drives the system to a new steady state where the load change in each control area is absorbed within each area, i.e., $P_j^M = P_j^L$ for all $j \in \mathcal{N}$, and the frequency is returned to the nominal value, i.e., $\omega_j = 0$ for all $j \in \mathcal{N}$; as shown in Proposition 5.1 in Section 5.3.

5.2.2 Optimal generation control

Due to the proliferation of renewable energy resources such as solar and wind in the future power grid, the aggregate net loads will fluctuate much faster and by large amounts. The ACE-based AGC that requires each control area to absorb its own load changes may be economically inefficient. Therefore, we proposed to modify the ACE-based AGC to (i) maintain the nominal frequency and (ii) drive the mechanical power

output $P_j^M, j \in \mathcal{N}$ to the optimum of the following economic dispatch problem:[2]

$$\begin{aligned} \min \quad & \sum_{j\in\mathcal{N}} C_j(P_j^M) && \text{(5.5a)} \\ \text{s.t.} \quad & P_j^M = P_j^L + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij},\ j \in \mathcal{N} && \text{(5.5b)} \\ \text{over} \quad & P_j^M, P_{ij}, j \in \mathcal{N}, (i,j) \in \mathcal{E}, \end{aligned}$$

where each generator at j incurs certain cost $C_j(P_j^M)$ when its power generation is P_j^M. Equation (5.5b) imposes power flow balanced at each bus. The cost function $C_j(\cdot)$ is assumed to be continuous, convex. We call this modified AGC as the economic AGC. In the following sections, we will show how to reverse and forward engineer the ACE-based AGC to design an economic AGC scheme.

5.3 Reverse engineering of ACE-based AGC

In this section, we reverse-engineer the dynamic model with the ACE-based AGC (5.1)-(5.4). We show that the equilibrium points of (5.1)-(5.4) are the optima of a properly defined optimization problem and furthermore the dynamics (5.1)-(5.4) can be interpreted as a partial primal-dual gradient algorithm to solve this optimization problem. The reverse-engineering suggests a way to modify the ACE-based AGC to design an economic AGC scheme.

We first characterize the equilibrium points of the power system dynamics with AGC (5.1)-(5.4). Let $\omega = \{\omega_j, j \in \mathcal{N}\}$, $P^M = \{P_j^M, j \in \mathcal{N}\}$, $P^C = \{P_j^C, j \in \mathcal{N}\}$, and $P = \{P_{i,j}, (i,j) \in \mathcal{E}\}$.

Proposition 5.1. *(ω, P^M, P^C, P) is an equilibrium point of the system (5.1)-(5.4) if and only if $\omega_j = 0$, $P_j^C = P_j^M = P_j^L$, and $\sum_{i:i\to j} P_{ij} = \sum_{k:j\to k} P_{jk}$ for all $j \in \mathcal{N}$.*

Proof. At a fixed point,

$$\dot{P}_{ij} = B_{ij}(\omega_i - \omega_j) = 0.$$

[2]Because all the variables denote the deviations in this chapter, it may be not straightforward to interpret this economic dispatch problem, e.g., how this problem is connected with the slower timescale economic dispatch problem using the absolute value of each variable instead of the deviated value? This problem can be seen as revising energy dispatch, because of the load changes, over the nominal values that are determined by the slower time-scale economic dispatch problem that is usually operated by ISOs or generating companies.

Therefore $\omega_i = \omega_j$ for all $i, j \in \mathcal{N}$, given that the transmission network is connected. Moreover,

$$ACE_j = B_j\omega_j + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij} = 0.$$

Thus $\sum_{j\in\mathcal{N}} \text{ACE}_j = \sum_{j\in\mathcal{N}} B_j\omega_j = \omega_i \sum_{j\in\mathcal{N}} B_j = 0$, so $\omega_i = 0$ for all $i \in \mathcal{N}$. The rest of the proof is straightforward. We omit it due to space limit. □

Consider the following optimization problem:

OGC-1

$$\min \ \sum_{j\in\mathcal{N}} C_j(P_j^M) + \sum_{j\in\mathcal{N}} \frac{D_j}{2}\,|\omega_j|^2 \tag{5.6a}$$

$$\text{s.t. } P_j^M = P_j^L + D_j\omega_j + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij} \tag{5.6b}$$

$$P_j^M = P_j^L \tag{5.6c}$$

$$\text{over } \ \omega_j, P_j^M, P_{ij},\ j \in \mathcal{N}, (i,j) \in \mathcal{E},$$

where equation (5.6c) requires that each control area absorbs its own load changes. The following result is straightforward.

Lemma 5.2. *(ω^*, P^{M^*}, P^*) is an optimum of OGC-1 if and only if $\omega_j^* = 0$, $P_j^{M^*} = P_j^L$, and $\sum_{k:j\to k} P_{jk}^* = \sum_{i:i\to j} P_{ij}^*$ for all $j \in \mathcal{N}$.*

Note that problem OGC-1 appears simple, as we can easily identify its optima if we know all the information on the objective function and the constraints. However, in practice these information is unknown. Moreover, even if we know an optimum, we cannot just set the system to the optimum. As the power network is a physical system, we have to find a way that respects the power system dynamics to steer the system to the optimum. Though the cost function $C_j(P_j^M)$ does not play any role in determining the optimum of OGC-1, it will become clear later that the choice of the cost function does have important implication to the algorithm design and the system dynamics.

We now show that the dynamic system (5.1)-(5.4) is actually a partial primal-dual gradient algorithm for

solving OGC-1 with $C_j(P_j^M) = \frac{\beta_j}{2}(P_j^M)^2$ where $\beta_j > 0$:

Introducing Lagrangian multipliers λ_j and μ_j for the constraints in OGC-1, we obtain the following Lagrangian function:

$$\begin{aligned} L &= \sum_{j\in\mathcal{N}} \frac{\beta_j}{2}(P_j^M)^2 + \sum_{j\in\mathcal{N}} \frac{D_j}{2}|\omega_j|^2 \\ &\quad + \sum_{j\in\mathcal{N}} \lambda_j \left(P_j^M - P_j^L - D_j\omega_j - \sum_{k:j\to k} P_{jk} + \sum_{i:i\to j} P_{ij} \right) \\ &\quad + \sum_{j\in\mathcal{N}} \mu_j \left(P_j^M - P_j^L \right). \end{aligned}$$

Based on the above Lagrangian function, we can write down a partial primal-dual subgradient algorithm of OGC-1 as follows:

$$\omega_j = \lambda_j \tag{5.7a}$$

$$\dot{P}_{ij} = \epsilon_{P_{ij}}(\lambda_i - \lambda_j) \tag{5.7b}$$

$$\dot{P}_j^M = -\epsilon_{P_j}(\beta_j P_j^M + \lambda_j + \mu_j) \tag{5.7c}$$

$$\dot{\lambda}_j = \epsilon_{\lambda_j} \left(P_j^M - P_j^L - D_j\omega_j - \sum_{k:j\to k} P_{jk} + \sum_{i:i\to j} P_{ij} \right) \tag{5.7d}$$

$$\dot{\mu_j} = \epsilon_{\mu_j} \left(P_j^M - P_j^L \right), \tag{5.7e}$$

where $\epsilon_{P_{ij}}, \epsilon_{P_j}, \epsilon_{\lambda_j}$ and ϵ_{μ_j} are positive stepsizes. Note that equation (5.7a) solves $\max_{w_j} \frac{D_j}{2}w_j^2 - \lambda_j D_j w_j$ rather than follows the primal gradient algorithm with respect to w_j; hence the algorithm (5.7) is called a "partial" primal-dual gradient algorithm. See the Appendix for a description of the general form of partial primal-dual gradient algorithm and its convergence.

Let $\epsilon_{\lambda_j} = \frac{1}{M_j}$ for all $j \in \mathcal{N}$. By applying linear transformation from (λ_j, μ_j) to (ω_j, P_j^C):

$$\begin{aligned} \omega_j &= \lambda_j \\ P_j^C &= K_j M_j \left(\lambda_j - \frac{1}{\epsilon_{\mu_j} M_j}\mu_j \right), \end{aligned}$$

the partial primal-dual gradient algorithm (5.7) becomes:

$$\dot{\omega}_j = -\frac{1}{M_j}\left(D_j\omega_j - P_j^M + P_j^L + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij}\right) \tag{5.8a}$$

$$\dot{P}_{ij} = \epsilon_{P_{ij}}(\omega_i - \omega_j) \tag{5.8b}$$

$$\dot{P}_j^M = -\epsilon_{P_j}\beta_j\left(P_j^M - \frac{\epsilon_{\mu_j}}{K_j\beta_j}P_j^C + \frac{1+\epsilon_{\mu_j}M_j}{\beta_j}\omega_j\right) \tag{5.8c}$$

$$\dot{P}_j^C = -K_j\left(D_j\omega_j + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij}\right). \tag{5.8d}$$

If we set $\epsilon_{P_{ij}} = B_{ij}$, $\epsilon_{\mu_j} = \frac{R_jK_j}{1-R_jK_jM_j}$, $\beta_j = \frac{R_j}{1-R_jK_jM_j}$, and $\epsilon_{P_j} = \frac{1}{\beta_jT_j}$, then the partial primal-dual algorithm (5.8) is exactly the power system dynamics with AGC (5.1)-(5.4) if $B_j = D_j,\ j \in \mathcal{N}$. Note that the assumption of $B_j = D_j$ looks restrictive. However, B_j is a design parameter, so we can set it to D_j. Algorithm (5.8) provides a tractable and easy way to choose parameters for the ACE-based AGC in order to guarantee its convergence.

Theorem 5.3. *If $1 > R_jK_jM_j$ for all $j \in \mathcal{N}$, with the above chosen ϵ_{λ_j}, ϵ_{μ_j}, $\epsilon_{P_{ij}}$ and ϵ_{Pj}, the partial primal-dual gradient algorithm (5.8) (i.e., the system dynamics (5.1)-(5.4)) converges to a fixed point $(\omega^*, P^*, P^{M*}, P^{C*})$ where (ω^*, P^*, P^{M*}) is an optimum of problem OGC-1 and $P^{C*} = P^{M*}$.*

Proof. See the Appendix. □

Remarks: We have made an equivalence transformation in the above: from algorithm (5.7) to algorithm (5.8). The reason for doing these transformation is to derive an algorithm that admits physical interpretation and can thus be implemented as the system dynamics. In particular, P_j^L is unknown and hence μ_j can not be directly observed or estimated, while P_j^C can be estimated/calculated based on the observable variables ω_j and P_{ij}. As the control should be based on observable or estimable variables, the power system implements algorithm (5.8) instead of (5.7) for the ACE-based AGC.

The above reverse-engineering, i.e., the power system dynamics with AGC as the partial primal-dual gradient algorithm solving an optimization problem, provides a modeling framework and systematic approach to design new AGC mechanisms that achieve different (and improved) objectives by engineering the associated

optimization problem. The new AGC mechanisms have different dynamic properties (such as responsiveness) and incur different implementation complexity by choosing different optimizing algorithms to solve the optimization problem. In the next section, we will engineer problem OGC-1 to design an AGC scheme that achieves economic efficiency.

5.4 Economic AGC by forward engineering

We have seen that the power system dynamics with the ACE-based AGC (5.1)-(5.4) is a partial primal-dual gradient algorithm solving a cost minimization problem OGC-1 with a "restrictive" constraint $P_j^M = P_j^L$ that requires supply-demand balance within each control area. As mentioned before, this constraint may render the system economically inefficient. Based on the insight obtained from the reverse-engineering of the conventional AGC, we relax this constraint and propose an AGC scheme that (i) keeps the frequency deviation to 0, i.e., $\omega_j = 0$ for all $j \in \mathcal{N}$, and (ii) achieves economic efficiency, i.e., the mechanical power generation solves the economic dispatch problem (5.5).

Consider the following optimization problem:

OGC-2

$$\min \ \sum_{j\in\mathcal{N}} C_j(P_j^M) + \sum_{j\in\mathcal{N}} \frac{D_j}{2}\,|\omega_j|^2 \tag{5.9a}$$

$$\text{s.t.} \ \ P_j^M = P_j^L + D_j\omega_j + \sum_{k:j\to k} P_{jk} - \sum_{i:i\to j} P_{ij} \tag{5.9b}$$

$$P_j^M = P_j^L + \sum_{k:j\to k} \gamma_{jk} - \sum_{i:i\to j} \gamma_{ij} \tag{5.9c}$$

$$\text{over} \ \ \omega_j, P_j^M, P_{ij}, \gamma_{ij},\ j\in\mathcal{N}, (i,j)\in\mathcal{E},$$

where γ_{ij} are auxiliary variables introduced to facilitate the algorithm design. As will become clear later, the reason to include constraint (5.9c) is in order to keep $\omega_j = 0$ for all $j \in \mathcal{N}$ and to derive an implementable control algorithm, similar to equations (5.3)-(5.4).

Lemma 5.4. *Let $(\omega^*, P^{M*}, P^*, \gamma^*)$ be an optimum of OGC-2, then $\omega_j^* = 0$ for all $j \in \mathcal{N}$ and P^{M*} is the optimal solution of the economic dispatch problem (5.5).*

Proof. First, note that at the optimum, $\omega_i^* = \omega_j^*$ for all $(i,j) \in \mathcal{N}$. Second, combining (5.9b) and (5.9c) gives

$$D_j\omega_j + \sum_{k:j\to k}(P_{jk} - \gamma_{jk}) - \sum_{i:i\to j}(P_{ij} - \gamma_{ij}) = 0$$

for all $j \in \mathcal{N}$. Following similar arguments as in Proposition 5.1, we have $\omega_i^* = 0$ for all $i \in \mathcal{N}$. Therefore the constraint (5.9c) is redundant and can be removed. So, problem OGC-2 reduces to the economic dispatch problem (5.5). □

Following the same procedure as in Section 5.3, we can derive the following partial prime-dual algorithm solving OGC-2:

$$\omega_j = \lambda_j \tag{5.10a}$$

$$\dot{P}_{i,j} = \epsilon_{P_{ij}}(\lambda_i - \lambda_j) \tag{5.10b}$$

$$\dot{P}_j^M = -\epsilon_{P_j}(C_j'(P_j^M) + \lambda_j + \mu_j) \tag{5.10c}$$

$$\dot{\gamma}_{ij} = \epsilon_{\gamma_{ij}}(\mu_i - \mu_j) \tag{5.10d}$$

$$\dot{\lambda}_j = \epsilon_{\lambda_j}\left(P_j^M - P_j^L - D_j\omega_j - \sum_{k:j\to k} P_{jk} + \sum_{i:i\to j} P_{ij}\right) \tag{5.10e}$$

$$\dot{\mu_j} = \epsilon_{\mu_j}\left(P_j^M - P_j^L - \sum_{k:j\to k} \gamma_{jk} + \sum_{i:i\to j} \gamma_{ij}\right), \tag{5.10f}$$

Let $\epsilon_{\lambda_j} = \frac{1}{M_j}$, $\epsilon_{P_{ij}} = B_{ij}$, $\epsilon_{\mu_j} = \frac{R_jK_j}{1-R_jK_jM_j}$ and $\epsilon_{P_j} = \frac{1-R_jK_jM_j}{T_jR_j}$ as in Section 5.3. By using linear transformation $\omega_j = \lambda_j$ and $P_j^C = K_jM_j\left(\lambda_j - \frac{1}{\epsilon_{\mu_j}M_j}\mu_j\right)$, the partial primal-dual gradient algorithm

(5.10) becomes:

$$\dot{\omega}_j = -\frac{1}{M_j}\left(D_j\omega_j - P_{M_j} + P_{L_j} + \sum_{k:j\to k} P_{j,k} - \sum_{i:i\to j} P_{i,j}\right) \tag{5.11a}$$

$$\dot{P}_{ij} = B_{ij}(\omega_i - \omega_j) \tag{5.11b}$$

$$\dot{P}_j^M = -\frac{1}{T_j}\left(\frac{1-R_jK_jM_j}{R_j}C_j'(P_j^M) - P_j^C + \frac{1}{R_j}\omega_j\right) \tag{5.11c}$$

$$\dot{P}_j^C = -K_j\left(D_j\omega_j + \sum_{k:j\to k}(P_{jk}-\gamma_{jk}) - \sum_{i:i\to j}(P_{ij}-\gamma_{ij})\right) \tag{5.11d}$$

$$\dot{\gamma}_{ij} = \epsilon_{\gamma_{ij}}\left(\left(M_i\omega_i - \frac{P_i^C}{K_i}\right)\epsilon_{\mu_i} - \left(M_j\omega_j - \frac{P_j^C}{K_j}\right)\epsilon_{\mu_j}\right). \tag{5.11e}$$

Compared with algorithm (5.8) (i.e., the power system dynamics with the conventional AGC), the difference in algorithm (5.11) is the new variables γ_{ij} and the marginal cost $C_j'(\cdot)$ in the generation control (5.11c). Note that γ_{ij} can be calculated based on the observable/measurable variables. So, the above algorithm is implementable. Also, when we choose a specific cost function $C_j(P_j^M) = \frac{R_j}{2(1-R_jK_jM_j)}(P_j^M)^2$, equation (5.11c) recovers equation (5.3) in the conventional AGC.

Similarly, we have the following result.

Theorem 5.5. *The algorithm (5.11) converges to a fixed point* $(\omega^*, P^*, P^{M^*}, P^{C^*}, \gamma^*)$ *where* $(\omega^*, P^*, P^{M^*}, \gamma^*)$ *is an optimum of problem OGC-2 and* $P_j^{C*} = \frac{1-R_jK_jM_j}{R_j}C_j'(P_j^{M^*})$.

Proof. See the Appendix. □

With Lemma 5.4 and Theorem 5.5, we can implement algorithm (5.11c)-(5.11e) as an economic AGC for the power system. It has only a slight modification to the conventional AGC, and keeps the decentralized structure of AGC.

Remarks: We can actually derive a simpler and yet implementable algorithm without introducing variable $\gamma_{ij}, (i,j) \in \mathcal{E}$. However, in order to have minimal modification to the existing conventional AGC and also keep the resulting control decentralized, we choose to derive the algorithm (5.11).

5.5 Case study

Consider a small 4-area interconnected system, as shown in Figure 5.1. The values of the generator and transmission line parameters are shown in Table 5.2 and 5.1. For each area, the generation cost takes on the form of $C_i(P_{M_i}) = a_i P_{M_i}^2$ where a is randomly drawn from $[1, 2]$.

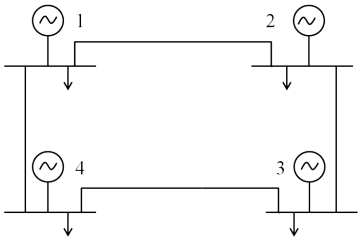


Figure 5.1. A 4-area interconnected system.

Table 5.1. Generator Parameters.

Area, j	M_j	D_j	$\|V_j\|$	T_j	R_j	K_j	B_j
1	3	4	1.045	4	0.05	2	4
2	2.5	3.5	0.98	4	0.05	2	3.5
3	4	3	1.033	4	0.05	2	3
4	3.5	4.5	0.997	4	0.05	2	4.5

Table 5.2. Line Parameters.

line	1-2	2-3	3-4	4-1
r	0.004	0.005	0.006	0.0028
x	0.0386	0.0294	0.0596	0.0474

In the model used for simulation, we relax some of the assumptions made in the previous analysis. For each transmission line we consider non-zero line resistance and do not assume small differences between phase angle deviations, which means that the power flow model is in the form of

$$P_{i,j} = \frac{|V_i||V_j|}{x_{i,j}^2 + r_{i,j}^2} \left(x_{i,j} \sin\theta_{ij} - r_{i,j} \cos\theta_{ij} \right). \tag{5.12}$$

Simulations results show that our proposed AGC scheme works well even in these non-ideal, practical systems.

At time $t = 10$s, a step change of load occurs at area 4 where $P_4^L = 0.3$ pu. Figure 5.2 shows the dynamics of the frequencies and mechanical power outputs for the 4 areas using ACE-based AGC (5.1)–

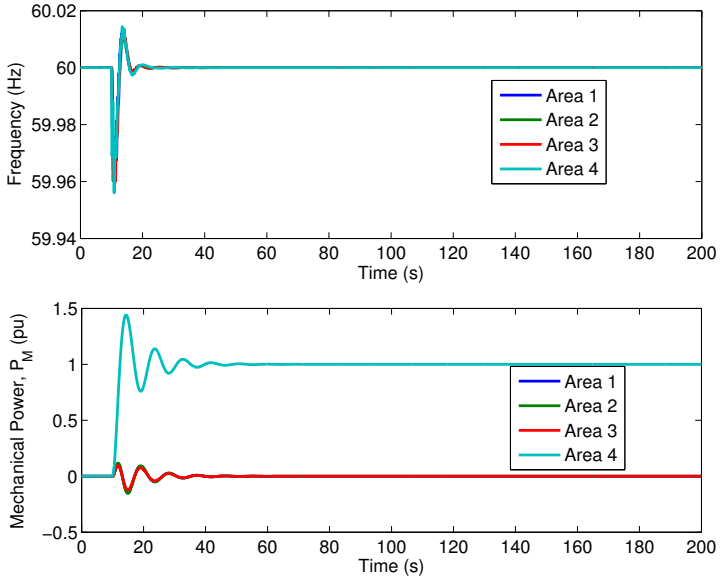


Figure 5.2. The ACE-based AGC.

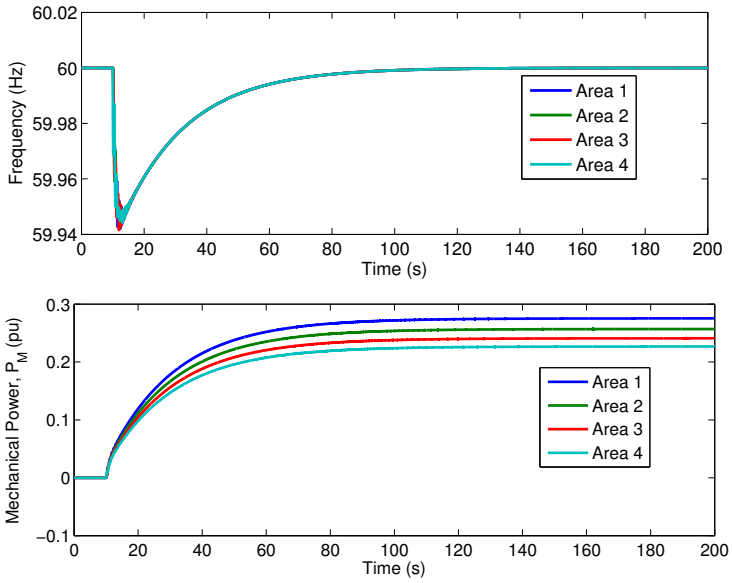


Figure 5.3. The economic AGC.

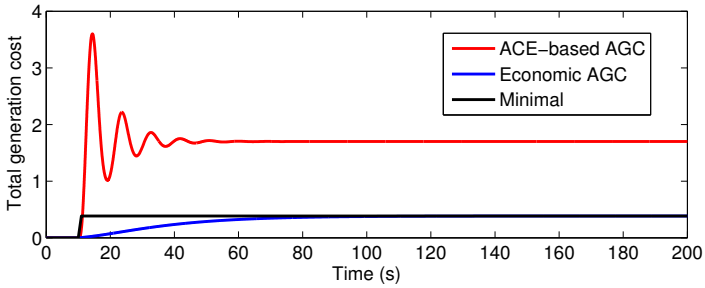


Figure 5.4. The generation cost.

(5.4). Figure 5.3 shows the dynamics of the frequencies and mechanical power outputs for the 4 areas using the economic AGC (5.11). Figure 5.4 compares the total generation costs using the ACE-based AGC and the economic AGC with the minimal generation cost of the economic dispatch problem (5.5). We see that the economic AGC does not only track the optimal value of the economic dispatch problem but also smooths out the frequency dynamics.

5.6 Conclusion

We reverse-engineer the conventional AGC, and based on the insight obtained from the reverse engineering, we design a decentralized generation control scheme that is integrated into the AGC and achieves economic efficiency. Combined with the previous work [64] on distributed load control, this work lends the promise to develop a modeling framework and solution approach for systematic design of distributed, low-complexity generation and load control to achieve system-wide efficiency and robustness.

5.7 Appendix: A partial primal-dual gradient algorithm

Consider the following optimization problem:

$$\begin{aligned} \min_{x,y} \quad & f(x) + g(y) \\ \text{s.t.} \quad & Ax + By = C, \end{aligned} \tag{5.13}$$

where $f(x)$ is a strict convex function of x, $g(y)$ is a convex function of y, and both f, g are differentiable. Notice that $g(y)$ can be a constant function.

The Lagrangian function of this optimization problem is given by:

$$L(x, y, \lambda) = f(x) + g(y) + \lambda^T (Ax + By - C).$$

Assume that the constraint is feasible and an optimal solution exists, then the strong duality holds. Moreover,

the primal-dual optimal solution (x^*, y^*, λ^*) is a saddle point of $L(x, y, \lambda)$ and vice versa.

The partial primal-dual gradient algorithm is given as follows:

Algorithm-1:

$$\begin{aligned} x(t) &= \min_x \left\{ f(x) + \lambda^T A x \right\} \\ \dot{y} &= -\Xi_y (\frac{\partial g(y)}{\partial y} + B^T \lambda) \\ \dot{\lambda} &= \Xi_\lambda (Ax + By - C) \end{aligned}$$

where $\Xi_y = diag(\epsilon_{y_i})$ and $\Xi_\lambda = diag(\epsilon_{\lambda_j})$.

In the following we will study the convergence of this algorithm.

Define

$$q(\lambda) \triangleq \min_x \left\{ f(x) + \lambda^T A x \right\}$$

and

$$\hat{L}(y, \lambda) \triangleq q(\lambda) + g(y) + \lambda^T (By - C).$$

The following proposition demonstrate some properties of $q(\lambda)$ and $\hat{L}(y, \lambda)$.

Proposition 5.6. *$q(\lambda)$ is a concave function and its gradient is given as $\frac{\partial q(\lambda)}{\partial \lambda} = Ax$. If ker$(A^T) = 0$, then $q(\lambda)$ is a strict concave function and the optimal dual λ^* is unique. As a consequence, given any y, there is a unique maximizer for $\max_\lambda \hat{L}(y, \lambda)$.*

Proof. This proposition follows directly from Proposition 6.1.1 in [76]. □

In order to use the result of this draft for automatic generation control, need to prove ker$(A^T) = 0$. This can be done.

Moreover, we have the following connections between $L(x, y, \lambda)$ and $\hat{L}(y, \lambda)$.

Lemma 5.7. *If (x^*, y^*, λ^*) is a saddle point of L, then (y^*, λ^*) is a saddle point of $\hat{L}$ and $x^* = argmin_x \left\{ f(x) + (\lambda^*)^T A x \right\}$. Moreover, if (y^*, λ^*) is a saddle point of $\hat{L}$, then (x^*, y^*, λ^*) is a saddle point of L where $x^* = argmin_x \left\{ f(x) + (\lambda^*)^T A x \right\}$.*

Proof. A simple proof is to write done and compare first order conditions of saddles points for both L and $\hat{L}$. Notice that convexity of f, g, and concavity of q implies that those first order conditions are necessary and sufficient conditions for saddle points. □

Now let study the convergence of **Algorithm-1**. With $\hat{L}(y, \lambda)$, **Algorithm-1** can be written as follows:

$$\dot{y} = -\Xi_y \left(\frac{\partial \hat{L}(y, \lambda)}{\partial y} \right) \tag{5.14}$$

$$\dot{\lambda} = \Xi_\lambda \left(\frac{\partial \hat{L}(y, \lambda)}{\partial \lambda} \right) \tag{5.15}$$

Let (y^*, λ^*) be a saddle point of $\hat{L}(y, \lambda)$. Define a nonegative function as:

$$\begin{aligned} U(y, \lambda) &= \frac{1}{2} \begin{bmatrix} y - y^* \\ \lambda - \lambda^* \end{bmatrix}^T \begin{bmatrix} \Xi_y^{-1} & \\ & \Xi_\lambda^{-1} \end{bmatrix} \begin{bmatrix} y - y^* \\ \lambda - \lambda^* \end{bmatrix} \\ &= \sum_{i=1}^{n} \frac{1}{2\epsilon_{y_i}} (y_i - y_i^*)^2 + \sum_{i=1}^{m} \frac{1}{2\epsilon_{\lambda_i}} (\lambda_i - \lambda_i^*)^2 \end{aligned} \tag{5.16}$$

Notice that $U \geq 0$ for any (y, λ). The derivative of U along the trajectory defined in (5.14,5.15) is given as:

$$\begin{aligned} \frac{\partial U}{\partial t} &= -\frac{\partial \hat{L}(y, \lambda)}{\partial y}^T (y - y^*) + \frac{\partial \hat{L}(y, \lambda)}{\partial \lambda} (\lambda - \lambda^*) \\ &\leq -\hat{L}(y, \lambda) + \hat{L}(y^*, \lambda) + \hat{L}(y, \lambda) - \hat{L}(y, \lambda^*) \\ &= \hat{L}(y^*, \lambda) - \hat{L}(y^*, \lambda^*) + \hat{L}(y^*, \lambda^*) - \hat{L}(y, \lambda^*) \\ &\leq 0 \end{aligned} \tag{5.17}$$

where the first equality comes from (5.14,5.15,5.16), the first inequality follows from the strictly concavity of $\hat{L}$ in λ and convexity of $\hat{L}$ in y and last inequality comes from that (y^*, λ^*) is a saddle point of $\hat{L}$. Therefore U is actually a Lyapunov function of (5.14,5.15). For simplicity, we will denote (y, λ) as z.

Lemma 5.8. $\frac{\partial U(z)}{\partial t} \leq 0$ *for all* z, *and* $\left\{ \hat{z} : \frac{\partial U(\hat{z})}{\partial t} = 0 \right\} = \left\{ \hat{z} : \hat{\lambda} = \lambda^*, \hat{L}(\hat{y}, \lambda^*) = \hat{L}(y^*, \lambda^*) \right\}$.

Proof. (5.17) has shown that $\frac{\partial U(z)}{\partial t} \leq 0$. To ensure $\frac{\partial U(\hat{z})}{\partial t} = 0$, the last inequality in (5.17) tells that

$\hat{L}(y^*, \hat{\lambda}) = \hat{L}(y^*, \lambda^*) = \hat{L}(\hat{y}, \lambda^*)$ which implies that $\hat{\lambda} = \lambda^*$ because $\hat{L}$ is strictly concave in λ and (y^*, λ^*) is a saddle point. Thus we can conclude the lemma. □

Now we can have our first result regards to the convergence:

Lemma 5.9. *Given any two saddle points (y_1^*, λ_1^*), (y_2^*, λ_2^*), we have $\lambda_1^* = \lambda_2^*$, and $\hat{L}(y_1^*, \lambda_1^*) = \hat{L}(y_2^*, \lambda_2^*)$. Any solution $(y(t), \lambda(t))$ of (5.14,5.15) for $t \geq 0$ asymptotically approaches to a nonempty, compact subset of the set of saddle points, denoted as $\mathcal{Z}^*$.*

Proof. (5.16) tells that $U(z) \geq 0$ for any z, and (5.17) tells that $U(z(t))$ is decreasing with time t and $U(z(t)) \leq U(z(0))$ for any $t \geq 0$. Because of the structure of $U(z)$ in (5.16), $z(t) = (y(t), \lambda(t))$ is bounded for $t \geq 0$. By Lyapunov convergence theory [77] , $z(t) = (y(t), \lambda(t))$ converges to a nonempty invariant compact subset of $\left\{\hat{z} : \frac{\partial U(\hat{z})}{\partial t} = 0\right\} = \left\{\hat{z} : \hat{\lambda} = \lambda^*, \hat{L}(\hat{y}, \lambda^*) = \hat{L}(y^*, \lambda^*)\right\}$. To ensure the subset is invariant, we have $\dot{\lambda} = \frac{\partial \hat{L}(\hat{Z})}{\partial \lambda} = 0$ which implies that such $\hat{z}$ is a saddle point of $\hat{L}$. □

Theorem 5.10. *Any solution $(y(t), \lambda(t))$ of (5.14,5.15) for $t \geq 0$ asymptotically converges to a saddle point (y^*, λ^*). The saddle point (y^*, λ^*) may depend on the initial point $(y(0), \lambda(0))$.*

Proof. The proof of Lemma 5.9 show that $\{z(t)\}_{t \geq 0}$ is a bounded sequences, therefore, we know that there exists a subsequence $\{z(t_j) = (y(t_j), \lambda(t_j))\}$ converges to a point $z^\infty = (y^\infty, \lambda^\infty)$. This implies that:

$$\lim_{t_j \to \infty} \sum_{i=1}^{n} \frac{1}{2\epsilon_{y_i}} (y_i(t_j) - y_i^\infty)^2 + \sum_{i=1}^{m} \frac{1}{2\epsilon_{\lambda_i}} (\lambda_i(t_j) - \lambda_i^\infty)^2 = 0. \tag{5.18}$$

As shown in Lemma 5.9, $z^\infty = (y^\infty, \lambda^\infty)$ is a saddle point of $\hat{L}$. Therefore Lemma 5.8,5.9 tells that:

$$\begin{aligned} & \lim_{t \to \infty} U(y(t) - y^\infty, \lambda(t) - \lambda^\infty) \\ = & \lim_{t \to \infty} \sum_{i=1}^{n} \frac{1}{2\epsilon_{y_i}} (y_i(t) - y_i^\infty)^2 + \sum_{i=1}^{m} \frac{1}{2\epsilon_{\lambda_i}} (\lambda_i(t) - \lambda_i^\infty)^2 \\ = & \quad u \end{aligned} \tag{5.19}$$

for some constant u. Since $\{z(t_j) = (y(t_j), \lambda(t_j))\}$ is a subsequence of $\{z(t)\}$, (5.18) tells that $u = 0$. Therefore, we can conclude that $(y(t), \lambda(t))$ converges to $(y^\infty, \lambda^\infty)$. □

Part II

Designing Games for Distributed Optimization

Chapter 6

Optimization Problem with Coupled Objective Function

[] The central goal in multiagent systems is to design *local* control laws for the individual agents to ensure that the emergent global behavior is desirable with respect to a given system level objective. Ideally, a system designer seeks to satisfy this goal while conditioning each agent's control law on the least amount of information possible. The second part of this thesis focuses on achieving this goal using the field of game theory. In particular, we derive a systematic methodology for designing local agent objective functions that guarantees (i) an equivalence between the resulting Nash equilibria and the optimizers of the system level objective and (ii) that the resulting game possesses an inherent structure that can be exploited in distributed learning, e.g., potential games. The control design can then be completed utilizing any distributed learning algorithm that guarantees convergence to a Nash equilibrium for the attained game structure. Furthermore, in many settings the resulting controllers will be inherently robust to a host of uncertainties including asynchronous clock rates, delays in information, and component failures.

In this chapter, we will focus on the cases in which the system level objective can be formulated as an optimization problem with coupled objective functions but decoupled constraints. But in the next chapter we will extend our approaches to the problems with coupled constraints. The communication graphs in these two chapters are time invariant; however, Chapter 8 will further extend the results to mult-agent systems with time-varying communication graphs.

6.1 Introduction

The central goal in multiagent systems is to design *local* control laws for the individual agents to ensure that the emergent global behavior is desirable with respect to a given system level objective, e.g., [78–83]. These control laws provide the groundwork for a decision making architecture that possesses several desirable attributes including real-time adaptation and robustness to dynamic uncertainties. However, realizing these benefits requires addressing the underlying complexity associated with a potentially large number of interacting agents and the analytical difficulties of dealing with overlapping and partial information. Furthermore, the design of such control laws is further complicated by restrictions placed on the set of admissible controllers which limit informational and computational capabilities.

Game theory is beginning to emerge as a powerful tool for the design and control of multiagent systems [82–86]. Utilizing game theory for this purpose requires two steps. The first step is to model the agents as self-interested decision makers in a game theoretic environment. This step involves defining a set of choices and a local objective function for each decision maker. The second step involves specifying a distributed learning algorithm that enables the agents to reach a desirable operating point, e.g., a Nash equilibrium of the designed game.

One of the core advantages of game theory is that it provides a hierarchical decomposition between the distribution of the optimization problem (*game design*) and the specific local decision rules (*distributed learning algorithms*) [87]. For example, if the game is designed as a potential game [88] then there is an inherent robustness to decision making rules as a wide class of distributed learning algorithms can achieve convergence to a pure Nash equilibrium under a variety of informational dependencies [89–92]. Several recent papers focus on utilizing this decomposition in distributed control by developing methodologies for designing games, in particular agent utility functions, that adhere to this potential game structure [82, 85, 87, 93]. However, these methodologies typically provide no guarantees on the locality of the agent utility functions or the efficiency of the resulting pure Nash equilibria. Furthermore, the theoretical limits of what such approaches can achieve are poorly understood.

The goal of this chapter is to establish a methodology for the design of *local* agent objective functions that lead to desirable system-wide behavior. We define the locality of an objective function by the underlying

interdependence, i.e., the set of agents that impact this objective function. For convention, we refer to this set of agents as the neighbor set. Accordingly, an objective function (A) is more local than an objective function (B) if the neighbor set of (A) is strictly smaller than the neighbor set of (B). The existing utility design methodologies, i.e., the wonderful life utility [82, 85] and the Shapley value utility [94, 95], prescribe procedures for deriving agent objective functions from a given system level objective function. While both procedures guarantee that the resulting game is a potential game, the degree of locality in the agent objective functions is an artifact of the methodology and the underlying structure of the system level objective. Hence, these methodologies do not necessarily yield agent objective functions with the desired locality.

The main contribution of this chapter is the development of a systematic methodology for the design of local agent objective functions that guarantees the efficiency of the resulting equilibria. In particular, in Theorem 6.3 we prove that our proposed methodology ensures (i) that there is an equivalence between the equilibria of the resulting game and the optimizers of the system level objective and (ii) that the resulting game is a *state based potential game* as introduced in [96].[1] A state based potential game is an extension of a potential game where there is an underlying state space introduced into the game structure. Our design utilizes these state variables as a coordinating entity to decouple the system level objective into agent specific objectives of the desired interdependence.

Our second result focuses on learning in state based potential games with continuous action sets. Much like potential games, state based potential games possess an underlying structure that can be exploited in distributed learning. Accordingly, in this chapter we prove that the learning algorithm gradient play, introduced in [97, 98] in the context of strategic form games, converges to an equilibrium in any state based potential game (see Theorem 6.4). Moreover, we provide a characterization of the convergence rate of gradient play for state based potential games (see Theorem 6.5). This work is complimentary to the results in [96] which provide similar results for state based potential games with finite action sets.

The design of multiagent systems parallels the theme of distributed optimization which can be thought of as a concatenation between a designed game and a distributed learning algorithm. One core difference

[1] It is important to highlight that [96] focuses predominantly on learning in state based potential games with finite action sets. The design of agent utility functions to ensure the efficiency of the resulting equilibria, which is the focus of this manuscript, is not addressed in [96].

between these two domains is the fact that multiagent systems frequently place restrictions on the set of admissible controllers. In terms of distributed optimization, this places a restriction on the set of admissible distributed algorithms. Accordingly, the applicability of some common approaches to distributed optimization, e.g, subgradient methods [99–104], consensus based methods [78, 79, 105, 106], or two-step consensus based approaches [86, 107, 108], may be limited by the structure of the system level objective.

There is also a family of distributed algorithms that are similar in spirit to the algorithms presented in this chapter. In particular, the algorithms presented in [103] and [108] introduce a communication protocol between the agents with the purpose of providing the agents with sufficient degrees of information so that the agents can estimate their gradient to the system level objective. While the proposed algorithms provide the desired asymptotic guarantees, the robustness to variations in clock rates, delays in information, and component failures is currently uncharacterized. Furthermore, the complexity regarding the analysis of these algorithms could make providing such a characterization challenging. In contrast to [103] and [108], our focus is on a methodological decomposition of the system level objective into local agent objective functions. Through this decomposition, we can take advantage of existing results in the field of learning in games to derive distributed algorithms that are robust to delays in information and heterogeneous clock rates. This follows directly from [90] and [96] which prove that any *reasonable* distributed learning algorithm will converge to a pure Nash equilibrium in any (finite) potential game or (finite) state based potential game.

This chapter focuses on establishing a systematic approach for distributed optimization. Accordingly, we focus predominantly on a general class of optimization problem with the realization that many problem instantiations relevant to multiagent systems can be represented within this problem formulation. Examples include collaborative sensing in a distributed PTZ camera network and the design of local control strategies for mobile sensor networks [109, 110]. For concreteness, in Section 6.5.2 we formally describe a distributed routing problem and illustrate how the proposed methodology can lead to desirable system behavior even when the agents possess incomplete information regarding the network behavior.

The rest of the chapter is organized as follows. Section 6.2 formally describes our model and provides a brief background about game theory and the framework of the state based game. Section 6.3 proposes a state based game design to solve the distributed optimization problem. Section 6.4 establishes a distributed

learning algorithm to enable agents to reach the desired operating points. Section 6.5 provides illustrative examples and Section 6.6 concludes the chapter.

6.2 Problem setup and preliminaries

We consider a multiagent system consisting of n agents denoted by the set $N = \{1, \cdots, n\}$. Each agent $i \in N$ is endowed with a set of decisions (or values) denoted by $\mathcal{V}_i$ which is a nonempty convex subset of $\mathbb{R}$. We denote a joint value by the tuple $v = (v_1, \cdots, v_n) \in \mathcal{V} = \prod_i \mathcal{V}_i$ where $\mathcal{V}$ is referred to as the set of joint values. Lastly, there is a global cost function of the form $\phi : \mathbb{R}^N \to \mathbb{R}$ that a system designer seeks to minimize. More formally, the optimization problem takes on the form:[2]

$$\begin{aligned} \min_v \quad & \phi(v_1, v_2, \ldots, v_n) \\ \text{s.t.} \quad & v_i \in \mathcal{V}_i, \forall i \in N. \end{aligned} \tag{6.1}$$

We assume throughout that ϕ is differentiable convex and that a solution of this optimization problem is guaranteed to exist.[3]

The focus of this chapter is to establish an interaction framework where each agent $i \in N$ chooses its value independently in response to local information. The information available to each agent is represented by an undirected and connected communication graph $\mathcal{G} = \{N, \mathcal{E}\}$ with nodes N and edges $\mathcal{E}$. Define the neighbors of agent i as $N_i = \{j \in N : (i, j) \in \mathcal{E}\}$ and we adopt the convention that $i \in N_i$ for each i. This interaction framework produces a sequence of values $v(0), v(1), v(2), \ldots$, where at each iteration $t \in \{0, 1, \ldots\}$ each agent i chooses a value independently according to a local control law of the form:

$$v_i(t) = F_i\left(\{\text{Information about agent } j\}_{j \in N_i}\right) \tag{6.2}$$

which designates how each agent processes available information to formulate a decision at each iteration.

[2]For ease of exposition we let $\mathcal{V}_i \subseteq \mathbb{R}$, which is just one dimension. In general, $\mathcal{V}_i$ can be any convex subset of $\mathbb{R}^{d_i}$ for any dimension $d_i \geq 1$. The results in this chapter immediately extend to the cases where $d_i > 1$ and $d_i \neq d_j$ for $i \neq j$. Furthermore, this work focuses on problems with decoupled constraints on agents' actions, i.e., $v_i \in \mathcal{V}_i$. The forthcoming methodologies can also incorporate coupled constraints using the approach demonstrated in the next chapter.

[3]There are many sufficient conditions to guarantee the existence of the optimal solution, e.g., compactness of $\mathcal{V}$ or coercivity of ϕ.

The goal in this setting is to design the local controllers $\{F_i(\cdot)\}_{i\in N}$ such that the collective behavior converges to a joint value v^* that solves the optimization problem in (6.1).

6.2.1 An illustrative example

We begin by presenting a simple example to motivate the theoretical developments in this chapter. Consider the following instance of (6.1) where

$$\phi(v_1, v_2, v_3) = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}^T \begin{bmatrix} 2 & 1 & 1 \\ 1 & 3 & 1 \\ 1 & 1 & 4 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} + [1\ 1\ 1] \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \tag{6.3}$$

and $\mathcal{V}_i = \mathbb{R}$ for all agents $N = \{1, 2, 3\}$. Here, the goal is to derive local agent control laws of the form (6.2) that converge to the minimizer of the cost function in (6.3) while adhering to the communication graph $1 \leftrightarrow 2 \leftrightarrow 3$. Note that this communication graph implies that the control policy of agent 1 is not able to depend on the true value of agent 3.

6.2.1.1 Gradient methods

Gradient methods represent a popular algorithm for solving nonlinear optimization problems [76]. A gradient method for the optimization problem in (6.3) takes on the form

$$v_i(t+1) = v_i(t) - \epsilon \frac{\partial \phi}{\partial v_i}, \tag{6.4}$$

where $\frac{\partial \phi}{\partial v_1} = 4v_1 + 2v_2 + 2v_3 + 1$; $\frac{\partial \phi}{\partial v_2} = 2v_1 + 6v_2 + 2v_3 + 1$; $\frac{\partial \phi}{\partial v_3} = 2v_1 + 2v_2 + 8v_3 + 1$ and ϵ is a positive step size. Note that both agents 1 and 3 require global information to calculate their gradients which is not admissible in our setting.

6.2.1.2 A game theoretic approach

Since ϕ in (6.3) does not possess a locally decomposable structure, the resulting gradient descent algorithms were not of the desired locality. A game theoretic approach introduces an intermediate step to the control design where each agent is assigned an objective function of the form $J_i : \prod_{j \in N_i} \mathcal{V}_j \to \mathbb{R}$. Here the goal is to embed the information admissibility constraints directly into the agents' objective function. For example, if we design agent objective functions of the form:

$$J_1 : \mathcal{V}_1 \times \mathcal{V}_2 \to \mathbb{R}$$

$$J_2 : \mathcal{V}_1 \times \mathcal{V}_2 \times \mathcal{V}_3 \to \mathbb{R}$$

$$J_3 : \mathcal{V}_2 \times \mathcal{V}_3 \to \mathbb{R}$$

and each agent follows a gradient-based approach to their local objectives, i.e., for any agent $i \in N$,

$$v_i(t+1) = v_i(t) - \epsilon \frac{\partial J_i}{\partial v_i},$$

then the resulting agents' control policies will satisfy our locality constraints. However, the convergence properties of such an algorithm are not as straightforward as the gradient algorithm given in (6.4), which leads to the work of this chapter.

6.2.2 Preliminaries: potential games

A strategic form game is characterized by a set of agents $N = \{1, \ldots, n\}$ where each agent $i \in N$ has an action set $\mathcal{A}_i$ and a cost function $J_i : \mathcal{A} \to \mathbb{R}$ where $\mathcal{A} = \prod_{i \in N} \mathcal{A}_i$ denotes the set of joint actions. For an action profile $a = (a_1, ..., a_n)$, let a_{-i} denote the action profile of agents other than agent i, i.e., $a_{-i} = (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n)$.

One class of games that plays a prominent role in engineering multiagent systems is that of potential games [88].

Definition 6.1. *(Potential Games) A game $\{N, \{\mathcal{A}_i\}, \{J_i\}\}$ is called an (exact) potential game if there exists*

a global function $\Phi : \mathcal{A} \to \mathbb{R}$ *such that for every agent* $i \in N$, $a_{-i} \in \mathcal{A}_{-i}$ *and* $a_i', a_i'' \in \mathcal{A}_i$,

$$J_i(a_i', a_{-i}) - J_i(a_i'', a_{-i}) = \Phi(a_i', a_{-i}) - \Phi(a_i'', a_{-i}).$$

There are three main properties regarding potential games which makes them an attractive paradigm for distributed engineering systems. First, in a potential game a pure Nash equilibrium, i.e., an action profile $a^* \in \mathcal{A}$ such that

$$J_i(a_i^*, a_{-i}^*) = \min_{a_i \in \mathcal{A}_i} J_i(a_i, a_{-i}^*), \forall i \in N,$$

is guaranteed to exist. Second, there are several available distributed learning algorithms with proven asymptotic guarantees that could be utilized for the control design [88–92]. Lastly, learning pure Nash equilibria in potential games is inherently robust [90]. That is, any "reasonable" learning algorithm where players seek to optimize their individual objective function will converge to a pure Nash equilibrium in potential games [90]. Hence, issues such as heterogeneous clock rates and informational delays are not problematic to learning pure Nash equilibria in such games.

6.2.3 Preliminaries: state based potential games

State based games, a simplification of stochastic games [111], represent an extension to strategic form games where an underlying state space is introduced to the game theoretic environment [96]. The class of state based games considered in this chapter consists of the following elements:

1. an agent set N,
2. a state space X,
3. a state dependent action set, $\mathcal{A}_i(x)$, for each agent $i \in N$ and state $x \in X$,
4. a state dependent cost function of the form $J_i(x, a) \in \mathbb{R}$, for each agent $i \in N$, $x \in X$, and $a \in \mathcal{A}(x) = \prod_{i \in N} \mathcal{A}_i(x)$, and
5. a deterministic state transition function $f(x, a) \in X$ for $x \in X$ and $a \in \mathcal{A}(x)$.

Furthermore, we focus on state based games where for any $x \in X$ there exists a null action $\mathbf{0} \in \mathcal{A}(x)$ such that $x = f(x, \mathbf{0})$. This implies that the state will remain unchanged if all of the agents take the null action. We will frequently denote a state based game by $G = \{N, X, \mathcal{A}, J, f\}$, where $\mathcal{A} = \bigcup_{x \in X} \mathcal{A}(x)$.

Repeated play of a state based game produces a sequence of action profiles $a(0), a(1), \cdots$, and a sequence of states $x(0), x(1), \ldots$, where $a(t) \in \mathcal{A}$ is referred to as the action profile at time t and $x(t) \in X$ is referred to as the state at time t. At any time $t \geq 0$, each agent $i \in N$ selects an action $a_i(t) \in \mathcal{A}_i(x(t))$ according to some specified decision rule which depends on the current state $x(t)$. The state $x(t)$ and the joint action profile $a(t) = (a_1(t), \ldots, a_n(t)) \in \mathcal{A}(x(t))$ determine each agent's one stage cost $J_i(x(t), a(t))$ at time t. After all agents select their respective action, the ensuring state $x(t+1)$ is chosen according to the deterministic state transition function $x(t+1) = f(x(t), a(t))$ and the process is repeated.

In this chapter we focus on the class of games termed state based potential games which represents an extension of potential games to the framework of state based games.

Definition 6.2. *(State Based Potential Game) A (deterministic) state based game G with a null action $\mathbf{0}$ is a (deterministic) state based potential game if there exists a potential function $\Phi : X \times \mathcal{A} \to \mathbb{R}$ satisfying the following two properties for every state $x \in X$:*

(D-1): For every agent $i \in N$, action profile $a \in \mathcal{A}(x)$ and action $a_i' \in \mathcal{A}_i(x)$

$$J_i(x, a_i', a_{-i}) - J_i(x, a) = \Phi(x, a_i', a_{-i}) - \Phi(x, a).$$

(D-2): For every action profile $a \in \mathcal{A}(x)$ and the ensuing state $\tilde{x} = f(x, a)$, the potential function satisfies

$$\Phi(x, a) = \Phi(\tilde{x}, \mathbf{0}).$$

The first condition states that each agent's cost function is aligned with the potential function in the same fashion as in potential games (Definition 6.1). The second condition relates to the evolution on the potential

function along the state trajectory.[4] We focus on the class of state based potential games as dynamics can be derived that converge to the following class of equilibria (see Theorem 6.4).

Definition 6.3. *(Stationary State Nash Equilibrium) A state action pair $[x^*, a^*]$ is a stationary state Nash equilibrium if*

(D-1): For any agent $i \in N$,

$$a_i^* \in \arg \min_{a_i \in \mathcal{A}_i(x^*)} J_i(x^*, a_i, a_{-i}^*).$$

(D-2): The state x^* is a fixed point of the state transition function, i.e., $x^* = f(x^*, a^*)$.

Note that in the case of a single state, i.e., $X = 1$, the definition of Stationary State Nash Equilibrium is precisely that of a Nash equilibrium since Condition (D-2) is satisfied trivially. The following proposition proves the existence of a stationary state Nash equilibrium in any state based potential game.

Proposition 6.1. *Let G be a state based potential game with potential function Φ and a null action $\mathbf{0}$. If $x^* \in \operatorname{argmin}_{x \in X} \Phi(x, \mathbf{0})$, then $[x^*, \mathbf{0}]$ is a stationary state Nash equilibrium. Moreover, for any $a \in \mathcal{A}(x^*)$ such that $x^* = f(x^*, a)$, $[x^*, a]$ is also a stationary state Nash equilibrium.*

Proof. In order to prove that $[x^*, \mathbf{0}]$ is a stationary state Nash equilibrium we only need to show that $\mathbf{0} \in \operatorname{argmin}_{a \in \mathcal{A}(x^*)} \Phi(x^*, a)$ because $x = f(x, \mathbf{0})$ for any $x \in X$ and Φ is a potential function of the game G. Let $a^* \in \operatorname{argmin}_{a \in \mathcal{A}(x^*)} \Phi(x^*, a)$. Thus $\Phi(x^*, \mathbf{0}) \geq \Phi(x^*, a^*)$. However since $x^* \in \operatorname{argmin}_{x \in X} \Phi(x, \mathbf{0})$, we have that $\Phi(x^*, a^*) = \Phi(\tilde{x}^*, \mathbf{0}) \geq \Phi(x^*, \mathbf{0})$ where $\tilde{x}^* = f(x^*, a^*)$. Therefore we have $\Phi(x^*, \mathbf{0}) = \Phi(x^*, a^*) = \min_{a \in \mathcal{A}(x^*)} \Phi(x^*, a)$. Hence $[x^*, \mathbf{0}]$ is a stationary state Nash equilibrium. For any a such that $x^* = f(x^*, a)$, we have $\Phi(x^*, a) = \Phi(x^*, \mathbf{0}) = \min_{a \in \mathcal{A}(x^*)} \Phi(x^*, a)$ implying that $[x^*, a]$ is also a stationary state Nash equilibrium. □

[4] The definition of state based games differs slightly from [96] as we focus on state dependent actions sets and games where there exist null actions.

6.3 State based game design

In this section we introduce a state based game design for the distributed optimization problem in (6.1). The goal of our design is to establish a state based game formulation that satisfies the following four properties:

(i) The state represents a compilation of local state variables, i.e., the state x can be represented as $x = (x_1, \ldots, x_n)$ where each x_i represents the state of agent i. Furthermore, the state transition f should also rely only on local information.

(ii) The objective function for each agent i is local and of the form $J_i(\{x_j, a_j\}_{j \in N_i}) \in \mathbb{R}$.

(iii) The resulting game is a state based potential game.

(iv) The stationary state Nash equilibria are optimal in the sense that they represent solutions to the optimization problem in (6.1), i.e., $v_i = v_i^*$.[5]

6.3.1 A state based game design for distributed optimization

We now introduce the specifics of our designed game.

State Space: The starting point of our design is an underlying state space X where each state $x \in X$ is defined as a tuple $x = (v, e)$, where

- $v = (v_1, \ldots, v_n) \in \mathbb{R}^n$ is the profile of values and
- $e = (e_1, \ldots, e_n)$ is the profile of estimation terms where $e_i = (e_i^1, \cdots, e_i^n) \in \mathbb{R}^n$ is agent i's estimation for the joint action profile $v = (v_1, \ldots, v_n)$. The term e_i^k captures agent i's estimate of agent k's actual value v_k.

The estimation terms are introduced as a means to relax the degree of information available to each agent. More specifically, each agent is aware of its own estimation as opposed to the true value profile which may

[5] There is a significant body of work in the field of algorithmic game theory that focuses on analyzing the inefficiency of Nash equilibria [112]. A common measure for this inefficiency, termed price of anarchy, is the worst case ratio between the system level performance of a Nash equilibrium and the optimal systems level performance. The vast literature in this area is predominantly analytical where the price of anarchy is characterized for situations where both the system level objective function and the agent cost functions are given. This work, on the other hand, focuses on the the counterpart of this analytical direction. In particular, is it possible to design local agent cost functions such that the price of anarchy is 1 for given a system level objective function? For the class of optimization problems considered in this manuscript, we provide a systematic methodology for accomplishing this task.

in fact be different, i.e., e_i^k need not equal v_k.

Action Sets: Each agent i is assigned an action set $\mathcal{A}_i$ that permits agents to change their value and change their estimation through communication with neighboring agents. Specifically, an action for agent i is defined as a tuple $a_i = (\hat{v}_i, \hat{e}_i)$ where $\hat{v}_i \in \mathbb{R}$ indicates a change in the agent's value v_i and $\hat{e}_i = (\hat{e}_i^1, \cdots, \hat{e}_i^n)$ indicates a change in the agent's estimation terms e_i. We represent each of the estimation terms $\hat{e}_i^k$ by the tuple $\hat{e}_i^k = \{\hat{e}_{i\to j}^k\}_{j\in N_i\setminus\{i\}}$ where $\hat{e}_{i\to j}^k \in \mathbb{R}$ represents the estimation value that agent i passes to agent j regarding the value of agent k.

State Dynamics: Define that $\hat{e}_{i\leftarrow\text{in}}^k = \sum_{j\in N_i\setminus\{i\}} \hat{e}_{j\to i}^k$ and $\hat{e}_{i\to\text{out}}^k = \sum_{j\in N_i\setminus\{i\}} \hat{e}_{i\to j}^k$ denote the total estimation passed to and from agent i regarding the value of the k-th agent respectively. We represent the state transition function $f(x,a)$ by a set of local state transition functions $\{f_i^v(x,a)\}_{i\in N}$ and $\left\{f_{i,k}^e(x,a)\right\}_{i,k\in N}$. For a state $x = (v,e)$ and an action $a = (\hat{v},\hat{e})$, the ensuing state $\tilde{x} = (\tilde{v},\tilde{e})$ is given by

$$\begin{aligned} \tilde{v}_i = f_i^v(x,a) &= v_i + \hat{v}_i \\ \tilde{e}_i^k = f_{i,k}^e(x,a) &= e_i^k + n\delta_i^k\hat{v}_i + \hat{e}_{i\leftarrow\text{in}}^k - \hat{e}_{i\to\text{out}}^k \end{aligned} \tag{6.5}$$

where δ_i^k is an indicator function, i.e., $\delta_i^i = 1$ and $\delta_i^k = 0$ for all $k \neq i$. Since the optimization problem in (6.1) imposes the requirement that $v_i \in \mathcal{V}_i$, we condition agents' available actions on the current state. That is, the available action set for agent i given state $x = (v,e)$ is defined as

$$\mathcal{A}_i(x) = \{(\hat{v}_i,\hat{e}_i) : v_i + \hat{v}_i \in \mathcal{V}_i\}. \tag{6.6}$$

Invariance associated with state dynamics: Let $v(0) = (v_1(0), ..., v_n(0))$ be the initial values of the agents. Define the initial estimation terms $e(0)$ to satisfy $\sum_{i\in N} e_i^k(0) = n\cdot v_k(0)$, for each agent $k \in N$; hence, the initial estimation values are contingent on the initial values. Note that satisfying this condition is trivial as we can set $e_i^i(0) = n\cdot v_i(0)$ and $e_i^j(0) = 0$ for all agents $i,j \in N$ where $i \neq j$. Define the initial state as $x(0) = [v(0), e(0)]$. It is straightforward to show that for any action trajectory $a(0), a(1), \cdots$, the resulting state trajectory $x(t) = (v(t), e(t)) = f(x(t-1), a(t-1))$ satisfies the following equalities for all times

$t \geq 1$ and agents $k \in N$

$$\sum_{i=1}^{n} e_i^k(t) = n \cdot v_k(t) \ . \tag{6.7}$$

Agent Cost Functions: The cost functions possess two distinct components and take on the form

$$J_i(x, a) = J_i^{\phi}(x, a) + \alpha \cdot J_i^{e}(x, a) \tag{6.8}$$

where $J_i^{\phi}(\cdot)$ represents the component centered on the objective function ϕ, $J_i^{e}(\cdot)$ represents the component centered on the disagreement of estimation based terms e, and α is a positive constant representing the tradeoff between the two components.[6] We define each of these components as

$$\begin{aligned} J_i^{\phi}(x, a) &= \textstyle\sum_{j \in N_i} \phi(\tilde{e}_j^1, \tilde{e}_j^2, ..., \tilde{e}_j^n) \\ J_i^{e}(x, a) &= \textstyle\sum_{j \in N_i} \sum_{k \in N} \left[\tilde{e}_i^k - \tilde{e}_j^k\right]^2 \end{aligned} \tag{6.9}$$

where $\tilde{x} = (\tilde{v}, \tilde{e}) = f(x, a)$ represents the ensuing state. The null action $\mathbf{0}$ is characterized by

$$\hat{v}_i = 0, \ \ \hat{e}_{i \to j}^k = 0, \forall i, j, k \in N.$$

Since $x = f(x, \mathbf{0})$, the agents' cost functions satisfy $J_i(x, a) = J_i(\tilde{x}, \mathbf{0})$.

6.3.2 Analytical properties of the designed game

In this section we derive two analytical properties of the designed state based game. The first property establishes that the designed game is a state based potential game.

Theorem 6.2. *The state based game depicted in Section 6.3.1 is a state based potential game with potential function*

$$\Phi(x, a) = \Phi^{\phi}(x, a) + \alpha \cdot \Phi^{e}(x, a) \tag{6.10}$$

[6] We will show that for any positive α, the results demonstrated in this chapter hold. However, choosing the right α is important for the learning algorithm implementation, e.g., the convergence rate of the learning algorithm.

where

$$\begin{aligned} \Phi^{\phi}(x,a) &= \textstyle\sum_{i\in N}\phi(\tilde{e}_i^1,\tilde{e}_i^2,...,\tilde{e}_i^n) \\ \Phi^{e}(x,a) &= \tfrac{1}{2}\textstyle\sum_{i\in N}\sum_{j\in N_i}\sum_{k\in N}\left[\tilde{e}_i^k-\tilde{e}_j^k\right]^2 \end{aligned} \tag{6.11}$$

and $\tilde{x}=(\tilde{v},\tilde{e})=f(x,a)$ *represents the ensuing state.*

Proof. It is straightforward to verify that Conditions (D-1)-(D-2) of state based potential games in Definition 6.2 are satisfied using the state based potential function defined in (6.10). □

The following theorem demonstrates that *all* equilibria of our designed game are solutions to the optimization problem in (6.1).

Theorem 6.3. *Let G be the state based game depicted in Section 6.3.1. Suppose that ϕ is a differentiable convex function, the communication graph $\mathcal{G}$ is connected and undirected, and at least **one** of the following conditions is satisfied:*

(i) The communication graph $\mathcal{G}$ is non-bipartite;[7]

(ii) The communication graph $\mathcal{G}$ contains an odd number of nodes, i.e., the number of agents is odd;

(iii) The communication graph $\mathcal{G}$ contains at least two agents which have a different number of neighbors, i.e., $|N_i|\neq|N_j|$ for some agents $i,j\in N$;

(iv) For each agent $i\in N$ the actions set $\mathcal{V}_i$ is open.

Then the state action pair $[x,a]=[(v,e),(\hat{v},\hat{e})]$ is a stationary state Nash equilibrium if and only if the following conditions are satisfied:

(a) The estimation profile e satisfies that $e_i^k=v_k$, $\forall i,k\in N$;

(b) The value profile v is an optimal solution for problem (6.1);

(c) The change in value profile satisfies $\hat{v}=\mathbf{0}$;

(d) The change in estimation profile satisfies that for all agents $i,k\in N$, $\hat{e}_{i\leftarrow\text{in}}^k=\hat{e}_{i\rightarrow\text{out}}^k$.

[7] A bipartite graph is a graph that does not contain any odd-length cycles.

The above theorem proves that the resulting equilibria of our state based game coincide with the optimal solutions to the optimization problem in (6.1) under relatively minor conditions on the communication graph. Hence, our design provides a systematic methodology for distributing an optimization problem under virtually any desired degree of locality in the agents' objective functions. A natural question arises as to whether the results in Theorem 6.2 and 6.3 could have been attained using the framework of strategic form games. In Appendix we prove that it is impossible to accomplish such a task.

6.4 Gradient play

In this section we prove that the learning algorithm *gradient play*, studied previously in [97] and [98] for strategic form games, converges to a stationary state Nash equilibrium in state based potential games. Since the designed game depicted in Section 6.3.1 is a state based potential game, the algorithm gradient play can be utilized to design control laws of the form (6.2) that guarantee convergence to the optimal solution of (6.1).

6.4.1 Gradient play for state based potential games

Given a state based potential game $G = \{N, \mathcal{A}, X, J, f\}$, suppose that $\mathcal{A}_i(x)$ is a closed convex set for all $i \in N$ and $x \in X$. Let $x(t)$ represent the state at time t. According to the learning algorithm gradient play, each agent $i \in N$ selects an action $a_i(t) \in \mathcal{A}_i\left(x(t)\right)$ according to

$$a_i(t) \quad = \quad \left[-\epsilon_i \cdot \left.\frac{\partial J_i(x(t), a)}{\partial a_i}\right|_{a=\mathbf{0}}\right]^+ \tag{6.12}$$

where $[\cdot]^+$ represents the projection onto the closed convex set $\mathcal{A}_i\left(x(t)\right)$ and ϵ_i is the step size which is a positive constant. Note that the agents' step sizes can be heterogeneous.

Before establishing the convergence results, we make the following assumptions for the state based potential game G:

A-1: $\Phi(x, \mathbf{0})$ is continuously differentiable and bounded below on x and $\Phi(x, a)$ is convex and differentiable on variable a.

A-2: $\nabla_a \Phi(x,a)$ is a Lipschitz function on variable a, i.e., there exists a constant L such that for any $x \in X$ and for any $a, a' \in \mathcal{A}(x)$, $||\nabla_a \Phi(x,a) - \nabla_a \Phi(x,a')||_2 \leq L||a - a'||_2$ where $\nabla_a \Phi(x,a) = (\frac{\partial \Phi}{\partial a_1}, \ldots, \frac{\partial \Phi}{\partial a_n})$.

Theorem 6.4. *Let G be a state based potential game with a potential function $\Phi(x,a)$ that satisfies Assumption (A-1,2). If the step size ϵ_i is smaller than $2/L$ for all $i \in N$, then the state action pair $[x(t), a(t)]$ of the gradient play process in (6.12) asymptotically converges to a stationary state Nash equilibrium of the form $[x, \mathbf{0}]$.*

Proof. From the definition of the state based potential game, we have $\Phi(x(t+1), \mathbf{0}) = \Phi(x(t), a(t))$ where $x(t+1) = f(x(t), a(t))$. We will first prove that $\Phi(x(t+1), \mathbf{0})$ is monotonically decreasing during the gradient play process provided that the step size is sufficiently small. The gradient play process in (6.12) can be expressed using the state based potential function as

$$a_i(t) = \left[-\epsilon \cdot \left. \frac{\partial J_i(x(t), a)}{\partial a_i} \right|_{a=\mathbf{0}} \right]^+ = \left[-\epsilon \cdot \left. \frac{\partial \Phi(x(t), a)}{\partial a_i} \right|_{a=\mathbf{0}} \right]^+ \tag{6.13}$$

Therefore, we have

$$\begin{aligned} \Phi(x(t+1), \mathbf{0}) - \Phi(x(t), \mathbf{0}) &= \Phi(x(t), a(t)) - \Phi(x(t), \mathbf{0}) \\ &\leq a(t)^T \left. \frac{\partial \Phi(x(t), a)}{\partial a} \right|_{a=\mathbf{0}}^T + \frac{L}{2} \|a(t)\|_2^2 \end{aligned}$$

where the second inequality is based on Proposition A.24 in [76]. By the Projection Theorem (Proposition 2.1.3 in [76]), we know that

$$\left(-\epsilon_i \cdot \left. \frac{\partial \Phi(x(t), a)}{\partial a_i} \right|_{a=\mathbf{0}} - a_i(t) \right)^T \cdot (-a_i(t)) \leq 0$$

which is equivalent to

$$a_i(t)^T \cdot \left. \frac{\partial \Phi(x(t), a)}{\partial a_i} \right|_{a=\mathbf{0}} \leq -\frac{1}{\epsilon_i} a_i(t)^T a_i(t).$$

If ϵ_i is smaller than $\frac{2}{L}$ for all $i \in N$, we have that

$$\Phi(x(t+1), \mathbf{0}) - \Phi(x(t), \mathbf{0}) \leq \sum_i \left(\frac{L}{2} - \frac{1}{\epsilon_i} \right) \|a_i(t)\|_2^2 \leq 0$$

and the equality holds in the second inequality if and only if $a(t) = \mathbf{0}$. Therefore, $\Phi(x(t), \mathbf{0})$ is monotonically decreasing along the trajectory $x(t)$. Since $\Phi(x(t), \mathbf{0})$ is bounded below, $\Phi(x(t), \mathbf{0})$ keeps decreasing until it reaches a fixed point, which means $a(t) = \mathbf{0}$. By Lemma 6.12 in Appendix, we know that such a fixed point is a stationary state Nash equlibrium. Hence $[x(t), a(t)]$ converges to a stationary state Nash equilibrium in the form of $[x, \mathbf{0}]$. □

First note that the asymptotic guarantees given in Theorem 6.4 hold for heterogeneous step sizes. This implies that the agents can take actions synchronously or asynchronously without altering the asymptotic guarantees. Second, the rate of convergence of gradient play depends on the structure of the potential function Φ, the state transition function f, and the stepsize ϵ_i. Larger step sizes ϵ_i generally lead to faster convergence but can also lead to instability. The bound on the stepsize ϵ_i in Theorem 6.4 is conservative as larger stepsize can usually be used without losing stability. Moreover, the stepsizes can vary with time as long as some additional mild conditions are satisfied.[8]

The following theorem establishes the convergence rate of the gradient play algorithm for state based potential games. For ease of exposition, we let $\epsilon_i = \epsilon_j = \epsilon$ for all the agents $i, j \in N$ and $\mathcal{A}_i(x) = \mathbb{R}^{d_x}$ for some dimension d_x, which means that the gradient play algorithm in (6.12) takes on the form: $a_i(t) = -\epsilon \cdot \frac{\partial J_i(x(t),a)}{\partial a_i}\Big|_{a=\mathbf{0}}$. Additionally, we make the following assumptions.

A-3 : The state transition rule is linear, namely that $\tilde{x} = f(x, a) = x + Ba$. Thus $\Phi(x, a) = \Phi(x + Ba, \mathbf{0})$ for all $a \in \mathcal{A}(x)$.

A-4 : There exit constants $M, m > 0$ such that for any $[x, a] \in X \times \mathcal{A}$,

$$\frac{m}{2}||a||^2 \leq \Phi(x, a) - \Phi(x, 0) - a^T \cdot \nabla_a \Phi|_{(x,0)} \leq \frac{M}{2}||a||^2.$$

[8] This is similar with the gradient methods in optimization literature [113].

Note that if $\Phi(x,a)$ is a strict convex function on variable a, one choice for M, m is that

$$\begin{aligned} M &= \max_{[x,a]\in X\times\mathcal{A}} \left(\sigma_{\max}\nabla_a^2\Phi(x,a)\right); \\ m &= \min_{[x,a]\in X\times\mathcal{A}} \left(\sigma_{\min}\nabla_a^2\Phi(x,a)\right). \end{aligned}$$

Here $\nabla_a^2\Phi(x,a)$ denotes the Hessian matrix of Φ on variable a and σ denotes the singular values of this matrix.

Theorem 6.5. *Let G be a state based potential game that satisfies Assumptions (A-1,3,4). If the step size ϵ is smaller than $2/M$, then the state action pair $[x(t),a(t)]$ of the gradient play process asymptotically converges to a stationary state Nash equilibrium of the form $[x^*,\mathbf{0}]$. Moreover, $\Phi(x(t),a(t))$ is monotonically non-increasing and for all $t>1$,*

$$\Phi(x(t),a(t)) - \Phi(x^*,\mathbf{0}) \leq \theta\cdot\left(\Phi(x(t-1),a(t-1) - \Phi(x^*,\mathbf{0})\right)$$

where $\theta = \left(1-2m(\epsilon - \frac{M}{2}\epsilon^2)\right)$.

Proof. Please see Appendix. □

6.4.2 Gradient play for our designed game

Suppose that $\mathcal{V}_i$ is a closed convex set for all $i\in N$. The gradient play algorithm applied to the game depicted in Section 6.3.1 takes on the following form. At each time $t\geq 0$, given the state $x(t)=(v(t),e(t))$, each agent i selects an action $a_i=(\hat{v}_i,\hat{e}_i)$ according to

$$\begin{aligned} \hat{v}_i(t) &= \left[-\epsilon_i^v\cdot\left.\frac{\partial J_i\left(x(t),a\right)}{\partial\hat{v}_i}\right|_{a=0}\right]^+ \qquad (6.14)\\ &= \left[-\epsilon_i^v(n\ \phi_i|_{e_i(t)} + 2n\alpha\sum_{j\in N_i}(e_i^i(t)-e_j^i(t)))\right]^+ \\ \hat{e}_{i\to j}^k(t) &= -\epsilon_{i\to j}^{k,e}\cdot\left.\frac{\partial J_i\left(x(t),a\right)}{\partial\hat{e}_{i\to j}^k}\right|_{a=0} \\ &= \epsilon_{i\to j}^{k,e}\cdot\left(\ \phi_k|_{e_i(t)} - \phi_k|_{e_j(t)} + 2\alpha\left(e_i^k(t)-e_j^k(t)\right) + 2\alpha\sum_{l\in N_i}\cdot\left(e_i^k(t)-e_l^k(t)\right)\ \right) \qquad (6.15)\end{aligned}$$

where $[\cdot]^+$ represents the projection onto the closed convex set $\mathcal{A}_i^{\hat{v}}(x) = \{\hat{v}_i : v_i + \hat{v}_i \in \mathcal{V}_i\}$; and ϵ_i^v and $\left\{\epsilon_{i\to j}^{k,e}\right\}_{j\in N_i}$ are the stepsizes which are positive constants.

If $\phi(v)$ in (6.1) is a bounded differentiable convex function, it is straightforward to verify that the designed state based potential game satisfies Assumptions (A-1,2). Therefore, if the step sizes are sufficiently small, Theorem 6.4 ensures that the gradient play algorithm (6.14,6.15) will converge to a stationary state Nash equilibrium in the form of $[(v, e), \mathbf{0}]$, where v is the optimal solution of (6.1). Moreover, notice that the station transition rule given in (6.5) is linear; hence Theorem 6.5 guarantees a linear convergence rate.

6.5 Illustrations

In this section we illustrate the theoretical developments in this chapter on two independent problems. The first problem rigorously explores our state based game design on the motivational example given in Section 6.2.1. The second problem focuses on distributed routing with information constraints.

6.5.1 A simple example

Following the state based game design rule given in Section 6.3.1, each agent $i \in \{1, 2, 3\}$ in the example in Section 6.2.1 is assigned a local state variable of the form $x_i = (v_i, e_i^1, e_i^2, e_i^3)$ where e_i^k is agent i's estimate of agent k's value v_k. Agent i's action a_i is of the form $a_i = (\hat{v}_i, \hat{e}_i^1, \hat{e}_i^2, \hat{e}_i^3)$ where $\hat{e}_i^k = \left\{\hat{e}_{i\to j}^k\right\}_{j\in N_i}$ for $k = 1, 2, 3$. The state transition rule and local cost function are defined in (6.5) and (6.8) respectively.

For concreteness, consider agent 1 as an example.

- A state associated with agent 1 is of the form $x_1 = (v_1, e_1^1, e_1^2, e_1^2)$.
- An action associated with agent 1 is of the form $a_1 = (\hat{v}_1, \hat{e}_{1\to 2}^1, \hat{e}_{1\to 2}^2, \hat{e}_{1\to 2}^3)$.

- The state transition rule is of the form $[\tilde{v}, \tilde{e}] = f\left([v, e], [\hat{v}, \hat{e}]\right)$ where

$$\begin{aligned}
\tilde{v}_1 &= v_1 + \hat{v}_1, \\
\tilde{e}_1^1 &= e_1^1 + \hat{v}_1 - \hat{e}_{1\to 2}^1 + \hat{e}_{2\to 1}^1 \\
\tilde{e}_1^2 &= e_1^2 - \hat{e}_{1\to 2}^2 + \hat{e}_{2\to 1}^2 \\
\tilde{e}_1^3 &= e_1^3 - \hat{e}_{1\to 2}^3 + \hat{e}_{2\to 1}^3.
\end{aligned}$$

- The local cost function of agent 1 is of the form

$$J_1\left([v, e], [\hat{v}, \hat{e}]\right) = \phi(\tilde{e}_1^1, \tilde{e}_1^2, \tilde{e}_1^3) + \frac{\alpha}{2} \sum_{k=1,2,3} \left(\tilde{e}_1^k - \tilde{e}_2^k\right)^2 \quad .$$

Figure 6.1 shows simulation results associated with this example. The top figure includes the following: (i) the red curve shows the dynamics of ϕ using a centralized gradient method, (ii) the blue curve shows the dynamics of ϕ using our proposed state based game design with gradient play where agents take actions synchronously with a homogeneous step size $\epsilon = 0.02$, and (iii) the black curve shows the dynamics of ϕ using our proposed state based game design with gradient play where agents take actions asynchronously with heterogeneous step sizes, $\epsilon_1 = 0.01$, $\epsilon_2 = 0.02$, and $\epsilon_3 = 0.015$. In the asynchronous simulation, each agent took an action with probability 0.9 or took the null action $\mathbf{0}$ with probability 0.1. Lastly, we set $\alpha = 1$ for the above simulation. These simulations demonstrate that our state based game design can efficiently solve the optimization problem under the presented informational constraints. Furthermore, the agents achieve the correct estimate of the true value v as highlighted in the bottom figure. Note that the bottom figure only highlights the estimation errors for agent 1 as the plots for agents 2 and 3 are similar.

6.5.2 Distributed routing problem

In this section we focus on a simple distributed routing problem with a single source, a single destination, and a disjoint set of routes $\mathcal{R} = \{r_1, ..., r_m\}$. There exists a set of agents $N = \{1, ..., n\}$ each seeking to send an amount of traffic, represented by $Q_i \geq 0$, from the source to the destination. The action set $\mathcal{V}_i$ for

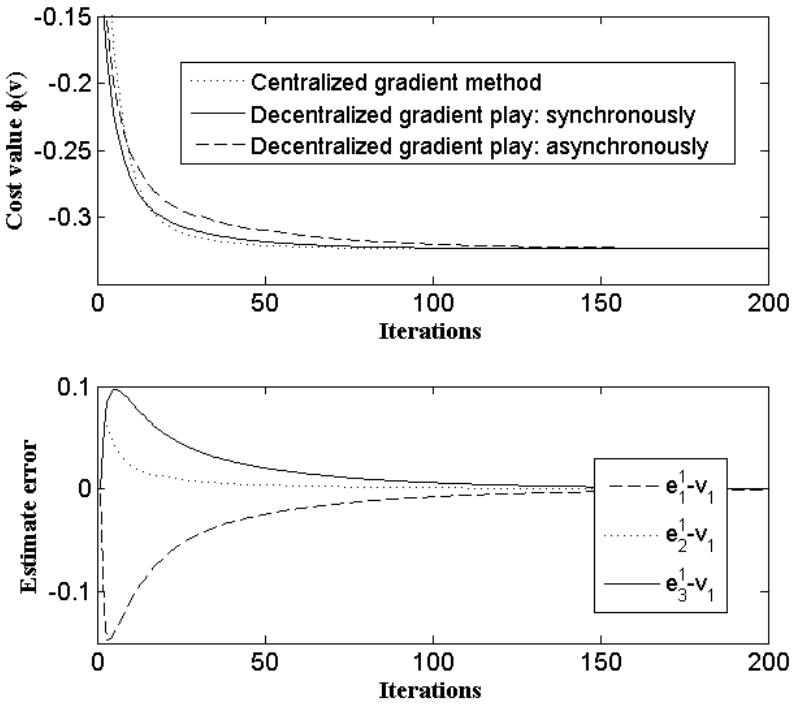


Figure 6.1. Simulation results for the optimization problem in (6.2.1). The top figure shows the evolution of the system cost $\phi(v)$ using (i) centralized gradient algorithm, (ii) our proposed state based game design with gradient play, homogeneous step sizes, and synchronous updates (blue), and (iii) our proposed state based game design with gradient play, heterogeneous step sizes, and asynchronously updates (black). The bottom figure shows the evolution of agent 1's estimation errors, i.e., $e_1^1 - v_1$, $e_1^2 - v_2$, and $e_1^3 - v_3$, during the gradient play algorithm with homogeneous step sizes and synchronous updates.

each agent is defined as

$$\left\{ v_i = (v_i^{r_1}, \ldots, v_i^{r_m}) : 0 \leq v_i^r \leq 1, \forall r \in \mathcal{R}; \sum_{r \in \mathcal{R}} v_i^r = 1 \right\} \tag{6.16}$$

where v_i^r represents that percentage of traffic that agent i designates to route r. Alternatively, the amount of traffic that agent i designates to route r is $v_i^r Q_i$. Lastly, for each route $r \in \mathcal{R}$, there is an associated "congestion function" of the form: $c_r : [0, +\infty) \to \mathbb{R}$ that reflects the cost of using the route as a function of the amount of traffic on that route.[9] For a given routing decision $v \in \mathcal{V}$, the total congestion in the network

[9]This type of congestion function is referred to as anonymous in the sense that all agents contribute equally to traffic. Non-anonymous congestion functions could also be used for this example.

takes the form

$$\phi(v) = \sum_{r \in \mathcal{R}} f_r \cdot c_r(f_r)$$

where $f_r = \sum_{i \in N} v_i^r Q_i$. The goal is to establish a local control law for each agent that converges to the allocation which minimizes the total congestion, i.e., $v^* \in \arg\min_{v \in \mathcal{V}} \phi(v)$. One possibility for a distributed algorithm is to utilize a gradient descent algorithm where each agent adjusts traffic flows according to

$$\frac{\partial \phi}{\partial v_i^r} = Q_i \cdot \left(c_r' \left(\sum_{i \in N} Q_i v_i^r \right) + c_r \left(\sum_{i \in N} Q_i v_i^r \right) \right)$$

where $c_r'(\cdot)$ represents the gradient of the congestion function. Note that implementing this algorithm requires each agent to have complete information regarding the decision of all other agents. In the case of non-anonymous congestion functions this informational restriction would be even more pronounced.

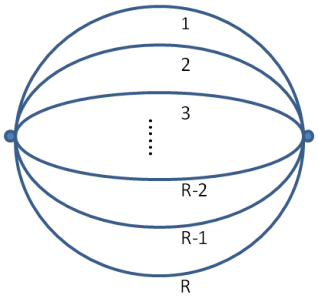


Figure 6.2. Distributed Traffic Routing.

Using the methodology developed in this chapter, we can localize the information available to each agent by allowing them only to have estimates of other agents' flow patterns. Consider the above routing problem with 10 agents and the following communication graph

$$1 \leftrightarrow 2 \leftrightarrow 3 \leftrightarrow \cdots \leftrightarrow 10.$$

Now, each agent is only aware of the traffic patterns for at most two of the other agents and maintaining and responding to estimates of the other agents' traffic patterns. Suppose we have 5 routes where each route $r \in \mathcal{R}$ has a quadratic congestion function of the form $c_r(k) = a_r k^2 - b_r k + c_r$ where $k \geq 0$ is the amount of traffic, and a_r, b_r, and c_r are positive and randomly chosen coefficients. Set the tradeoff parameter α to

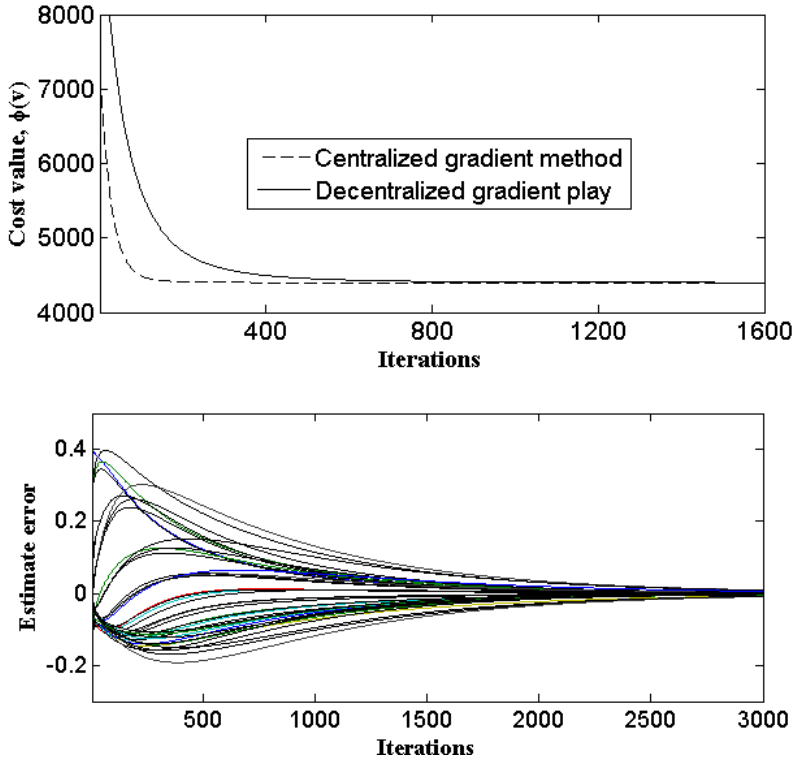


Figure 6.3. Simulation results: The upper figure shows the evolution of the system cost ϕ using the centralized gradient decent algorithm (red) and our proposed algorithm (black). The bottom figure shows the evolution of agent 1's estimation error, i.e., $e_1^{k,r} - v_k^r$ for each route $r \in \mathcal{R}$ and each agent $k \in N$.

be 900. Figure 6.3 illustrates the results of the gradient play algorithm presented in Section 6.4 coupled with our game design in Section 6.3. Note that our algorithm does not perform as well as the centralized gradient descent algorithm in transient. This is expected since the informational availability to the agents is much lower. However, the convergence time is comparable which is surprising.

6.6 Conclusion

This work presents an approach to distributed optimization using the framework of state based potential games. In particular, we provide a systematic methodology for localizing the agents' objective function while ensuring that the resulting equilibria are optimal with regards to the system level objective function. Furthermore, we proved that the learning algorithm gradient play guarantees convergence to a stationary state Nash equilibria in any state based potential game. By considering a game theoretic approach to distributed optimization, as opposed to the more traditional algorithmic approaches, we were able to attain immediate

robustness to variation in clock rates and step sizes as highlighted in Sections 6.3 and 6.4. There are several open and interesting questions that this chapter promotes. One in particular is regarding the communication requirements on the agents. In our design, each agent possessed n additional state variables as estimates for the n components of the value profile v. Could similar guarantees be attained with fewer variables? What happens if we transition from a fixed to time varying communication topology? Lastly, how does this approach extend to alternative classes of system level objective functions? We will cover some of those questions in the following chapters.

6.7 Appendix

6.7.1 An impossibility result for game design

This section addresses the question as to whether the results in Theorem 6.2 and 6.3 could have been attained using the framework of strategic form games. More specifically, is it possible to design agent objective functions that achieve the following four objectives:

- Each agent's cost function relies solely on local information as defined by the communication graph. Moreover, agents' cost functions should possess a degree of scalability with regards to the size of the system and the topology of the communication graph.
- All Nash equilibria of the resulting game represent solutions to the optimization problem (6.1);
- The resulting game possesses an underlying structure that can be exploited by distributed learning algorithms, e.g., potential games.

Accomplishing these objectives would ensure that the agents' control policies resulting from the designed game plus a suitable learning algorithm would be of the local form in (6.2).

In the following we demonstrate that achieving these objectives using the framework of strategic form games is impossible in general. To show this we focus on the following optimization problem

$$\begin{aligned} \min_v \quad & \left(\sum_{i\in N} v_i\right)^2 \\ \text{s.t.} \quad & v_i \in [c_i, d_i] \subset \mathbb{R}. \end{aligned} \tag{6.17}$$

To make the control laws $\{F_i(\cdot)\}_{i\in N}$ scalable as to the agent set and the communication graph $\mathcal{G}$, we require that the underlying control design must be invariant to the agents' indices. This implies that if two agents (i, j) have the same number of neighbors, i.e., $|N_i| = |N_j|$, and for each agent k in N_i there is an agent h in N_j such that $v_k = v_h$ and $[c_k, d_k] = [c_h, d_h]$, and vice versa, then the control policies of agent i, j should be the same, i.e., $F_i\left(\{v_k, c_k, d_k\}_{k\in N_i}\right) = F_j\left(\{v_k, c_k, d_k\}_{k\in N_j}\right)$.

Accordingly, we formulate the optimization problem as a game where the agent set is N, the action set of each agent is the set $\mathcal{A}_i = [c_i, d_i]$, and each agent is assigned a cost function of the form $J_i : \prod_{j\in N_i} \mathcal{V}_j \to \mathbb{R}$. To facilitate the design of scalable agent control policies, we focus on the design of agent cost functions of the form:

$$J_i(v) = J\left(\{v_j, c_j, d_j\}_{j\in N_i}\right) \tag{6.18}$$

where the function $J(\cdot)$ is invariant to specific indices assigned to agents. Notice that this design of $J(\cdot)$ leads to a well defined game irrespective of the agent set N, constraint sets $[c_i, d_i]$ or the structure of the communication graph $\{N_i\}_{i\in N}$. The following proposition demonstrates that it is impossible to design $J(\cdot)$ such that for any game induced by a constraint profile $[c, d]$ and communication graph $\mathcal{G}$ all resulting Nash equilibria solve the optimization problem in (6.3).

Proposition 6.6. *There does not exist a single $J(\cdot)$ such that for any game induced by a connected communication graph $\mathcal{G}$, a constraint profile $[c, d]$, and agents' cost functions of the form (6.18), the Nash equilibria of the induced game represent solutions to the optimization problem in (6.17).*

Proof. Suppose that there exists a single $J(\cdot)$ that satisfies the proposition. We will now construct a counterexample to show that this is impossible. Consider two optimization problems of the form (6.17) with a

single communication graph given by

$$1 \leftrightarrow 2 \leftrightarrow 3 \leftrightarrow 4 \leftrightarrow 5 \leftrightarrow 6.$$

Here, we have $N = \{1, 2, 3, 4, 5, 6\}$ and $\mathcal{E} = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{5, 6\}\}$. In the first optimization problem the constraint profile is: $[c_1, d_1] = [c_6, d_6] = [-1, -\frac{21}{22}]$, $[c_2, d_2] = [c_3, d_3] = [c_4, d_4] = [\frac{6}{11}, \frac{7}{11}]$, and $[c_5, d_5] = [0, 0]$. In the second optimization problem, the constraint profile is:$[c_1, d_1] = [c_6, d_6] = [-1, -\frac{21}{22}]$ and $[c_2, d_2] = [c_3, d_3] = [c_4, d_4] = [c_5, d_5] = [\frac{6}{11}, \frac{7}{11}]$. We call the settings for the two optimization problems as setting (a) and (b) respectively. Under those constraints, the optimal solution for setting (a) is $v^a = (v_1^a, v_2^a, v_3^a, v_4^a, v_5^a, v_6^a) = (-\frac{21}{22}, \frac{7}{11}, \frac{7}{11}, \frac{7}{11}, 0, -\frac{21}{22})$ and the optimal solutions for setting (b) is $v^b = (v_1^b, v_2^b, v_3^b, v_4^b, v_5^b, v_6^b) = (-1, \frac{6}{11}, \frac{6}{11}, \frac{6}{11}, \frac{6}{11}, -1)$.

We start by defining agent cost functions of the form (6.18) which ensures that v^a is a Nash equilibrium for setting (a). This implies that for any agent $i \in N$, we have

$$J\left(\{v_i^a, c_i, d_i\}, \{v_j^a, c_j, d_j\}_{j \in N_i \setminus \{i\}}\right) \leq J\left(\{v_i, c_i, d_i\}, \{v_j^a, c_j, d_j\}_{j \in N_i \setminus \{i\}}\right) \tag{6.19}$$

for any $v_i \in \mathcal{V}_i$. By writing down the Nash equilibrium condition in (6.19) for setting (b), it is straightforward to see that agents $1, 2, 3, 4, 5, 6$ in setting (b) have the same structure form of the cost function as agents $1, 2, 3, 3, 2, 1$ in setting (a) respectively. For example, agent 4 in setting (a) has an identical cost function to agent 3 in setting (b). Since v^a represents a Nash equilibrium for setting (a) then no agent $i \in \{1, \ldots, 6\}$ has a unilateral incentive to deviate from v^a. As agents $1, 2, 3, 4, 5, 6$ in (b) can be mapped precisely to agents $1, 2, 3, 3, 2, 1$ in (a), $v^* = (v_1^*, v_2^*, v_3^*, v_4^*, v_5^*, v_6^*) = (-\frac{21}{22}, \frac{7}{11}, \frac{7}{11}, \frac{7}{11}, \frac{7}{11}, -\frac{21}{22})$ is a Nash equilibrium of setting (b) since no agent $i \in \{1, \ldots, 6\}$ has a unilateral incentive to deviate from v^*. The impossibility comes from the fact that v^* is not an optimal solution to setting (b). □

6.7.2 Proof of Theorem 6.3

Since the designed state based game is a state based potential game, we can apply Proposition 6.1 to prove the sufficient condition of the theorem. The proof involves two steps: (i) If x^* satisfies Condition (a)-(b)

listed in the theorem, then $x^* \in \operatorname{argmin}_{x \in X} \phi(x, \mathbf{0})$; (ii) if a^* satisfies Condition (c)-(d) in the theorem, then $x = f(x, a)$ for all $x \in X$. Therefore it is straightforward to prove that if a state action pair $[x, a]$ satisfies Conditions (a)-(d) listed in the theorem, then $[x, a]$ is a stationary state Nash equilibrium.

Let us prove the necessary condition of Theorem 6.3. Suppose $[x, a]$ is a stationary state Nash equilibrium. First notice that to ensure $[x, a]$ satisfies Condition (D-2) of Definition 6.3, i.e. $x = f(x, a)$, the action profile $a = (\hat{v}, \hat{e})$ should satisfy Condition (c)-(d) of this theorem. To prove Condition (a)-(b), we will use a series of lemmas to prove that under one of Cases (i)-(iv) of this theorem, if a station action pair $[x, a]$ satisfies Condition (D-1) of Definition 6.3, i.e. $a_i \in \operatorname{argmin}_{\check{a}} J_i(x, \check{a}_i, a_{-i})$ for all $i \in N$, then the ensuing state $\tilde{x} = f(x, a)$ satisfies the following conditions:

1. *Estimation alignment:* The ensuing estimation terms are aligned with the ensuing value profile, i.e., for all agents $i, k \in N$ we have $\tilde{e}_i^k = \tilde{v}_k$ where $(\tilde{v}, \tilde{e}) = f(x, a)$. (Lemma 6.7 for Case (i)–(ii), Lemma 6.8 for Case (iii) and Lemma 6.10 for Case (iv).)

2. *Optimality alignment:* The ensuing value profile $\tilde{v}$ is an optimal solution to (6.1). (Lemma 6.9 for Case (i)–(iii) and Lemma 6.10 for Case (iv).)

Combining this with the fact that $\tilde{x} = f(x, a) = x$, we can conclude that under one of Cases (i)-(iv) of this theorem if $[x, a]$ is a state based Nash equilibrium, then Condition (a)-(d) must be satisfied.

In the subsequent lemmas we consistently express the ensuing state for a state action pair $[x, a] = [(v, e), (\hat{v}, \hat{e})]$ as $(\tilde{v}, \tilde{e}) = f(x, a)$.

Lemma 6.7. *Suppose that the communication graph $\mathcal{G}$ satisfies either Condition (i) or (ii) of Theorem 6.3. If $[x, a]$ satisfies $a_i \in \operatorname{argmin}_{\check{a} \in \mathcal{A}_i(x)} J_i(x, \check{a}, a_{-i})$ for all $i \in N$, then all agents have correct estimates of the value profile. That is, for all agents $i, k \in N$ we have $\tilde{e}_i^k = \tilde{v}_k$.*

Proof. If $a_i \in \operatorname{argmin}_{\check{a}_i = (\check{v}_i, \check{e}_i) \in \mathcal{A}_i(x)} J_i(x, \check{a}_i, a_{-i})$ for all $i \in N$, then

$$\left. \frac{\partial J_i(x, \check{a}_i, a_{-i})}{\check{e}_{i,l}^k} \right|_{a_i} = 0, \forall i, k \in N, l \in N_i \backslash \{i\}$$

which is equivalent to

$$\phi_k|_{\tilde{e}_i} + 2\alpha \sum_{j \in N_i} \left(\tilde{e}_i^k - \tilde{e}_j^k\right) = \phi_k|_{\tilde{e}_l} - 2\alpha \left(\tilde{e}_i^k - \tilde{e}_l^k\right) \tag{6.20}$$

where $\phi_k|_{\tilde{e}_i}$ represents the derivative of ϕ relative to $\tilde{e}_i^k$ for the profile $\tilde{e}_i$, i.e., $\phi_k|_{\tilde{e}_i} = \frac{\partial \phi(\tilde{e}_i)}{\partial \tilde{e}_i^k}$. Consider any two connected agents $i, j \in N$, i.e., $(i, j) \in \mathcal{E}$. The equality in (6.20) translates to

$$\begin{aligned} \phi_k|_{\tilde{e}_i} + 2\alpha \textstyle\sum_{l \in N_i} \left(\tilde{e}_i^k - \tilde{e}_l^k\right) &= \phi_k|_{\tilde{e}_j} - 2\alpha \left(\tilde{e}_i^k - \tilde{e}_j^k\right) \\ \phi_k|_{\tilde{e}_j} + 2\alpha \textstyle\sum_{l \in N_j} \left(\tilde{e}_j^k - \tilde{e}_l^k\right) &= \phi_k|_{\tilde{e}_i} - 2\alpha \left(\tilde{e}_j^k - \tilde{e}_i^k\right). \end{aligned}$$

Adding these two equalities gives us

$$\textstyle\sum_{l \in N_i} (\tilde{e}_i^k - \tilde{e}_l^k) = -\sum_{l \in N_j} (\tilde{e}_j^k - \tilde{e}_l^k) \tag{6.21}$$

for all agents $i, j, k \in N$ such that $(i, j) \in N_j$. Since our communication graph is connected, the equality condition in (6.21) tells us that the possible values for the summation terms $\sum_{l \in N_i} (\tilde{e}_i^k - \tilde{e}_l^k)$ for each agent $i \in N$ can be at most one of two possible values that differ purely with respect to sign, i.e., for any agent $i \in N$ we have

$$\textstyle\sum_{l \in N_i} (\tilde{e}_i^k - \tilde{e}_l^k) \in \left\{e_{\text{diff}}^k, -e_{\text{diff}}^k\right\} \tag{6.22}$$

where $e_{\text{diff}}^k \in \mathbb{R}$ is a constant. We can utilize the underlying topology of the communication graph coupled with (6.22) to demonstrate that $e_{\text{diff}}^k = 0$.

1. If there exists a cycle in the communication graph with an odd number of nodes, applying equality (6.21), we can get that $e_{\text{diff}}^k = -e_{\text{diff}}^k$, which tells us that $e_{\text{diff}}^k = 0$.

2. Since the communication graph is undirected we know that $\sum_{i \in N} \sum_{l \in N_i} (\tilde{e}_i^k - \tilde{e}_l^k) = 0$. If the number of agents n is odd, condition (6.22) tells us that $\sum_{i \in N} \sum_{l \in N_i} (\tilde{e}_i^k - \tilde{e}_l^k) = h \cdot e_{\text{diff}}^k$ where h is a nonzero integer. Hence $e_{\text{diff}}^k = 0$.

In summary, if the total number of agents is odd or there exists a cycle in the communication graph with an

odd number of nodes we have that for all $i, k \in N$, $\sum_{l \in N_i}(\tilde{e}_i^k - \tilde{e}_l^k) = 0$. Since the communication graph is connected and undirected, it is straightforward to show that for all agents $i, j \in N$, $\tilde{e}_i^k = \tilde{e}_j^k, \forall k \in N$. Here the main idea of this proof is to write $\sum_{l \in N_i}(\tilde{e}_i^k - \tilde{e}_l^k) = 0, \forall i \in N$ in a matrix form for each $k \in N$. The rank of this matrix is $n-1$ resulting from the fact that the communication graph is *connected* and *undirected* hence proving the result. Combining this with the equality (6.7), we get that for all agents $i, k \in N$, $\tilde{e}_i^k = v_k$. $\square$

Remark 6.1. *It is important to note that alternative graph structures may very well provide the same guarantees.*

Lemma 6.8. *Suppose that the objective function ϕ and communication graph $\mathcal{G}$ satisfy Condition (iii) of Theorem 6.3. If $[x, a]$ satisfies $a_i \in \operatorname{argmin}_{\check{a}_i \in \mathcal{A}_i(x)} J_i(x, \check{a}_i, a_{-i})$ for all $i \in N$, then all agent have correct estimates of the value profile. That is, for all agents $i, k \in N$ we have $\tilde{e}_i^k = \tilde{v}_k$.*

Proof. In the proof of the last lemma, we have proved that if $a_i \in \operatorname{argmin}_{\check{a}_i} J_i(x, \check{a}_i, a_{-i})$, then equation (6.20) should be satisfied. Consider any agent $i \in N$, and any pair of agents $j_1, j_2 \in N_i$. Equation (6.20) tells us that

$$\begin{aligned} \phi_k|_{\tilde{e}_i} + 2\alpha \textstyle\sum_{j \in N_i} \left(\tilde{e}_i^k - \tilde{e}_j^k\right) &= \phi_k|_{\tilde{e}_{j_1}} - 2\alpha \left(\tilde{e}_i^k - \tilde{e}_{j_1}^k\right) \\ \phi_k|_{\tilde{e}_i} + 2\alpha \textstyle\sum_{j \in N_i} \left(\tilde{e}_i^k - \tilde{e}_j^k\right) &= \phi_k|_{\tilde{e}_{j_2}} - 2\alpha \left(\tilde{e}_i^k - \tilde{e}_{j_2}^k\right). \end{aligned} \tag{6.23}$$

Combining the two equations, we have the following equality

$$\phi_k|_{\tilde{e}_{j_1}} - \phi_k|_{\tilde{e}_{j_2}} - 2\alpha \left(\tilde{e}_{j_2}^k - \tilde{e}_{j_1}^k\right) = 0.$$

Note that agents j_1 and j_2 are not necessarily connected but are rather siblings as both agents are connected to agent i. Therefore, the above analysis can be repeated to show that for any siblings $j_1, j_2 \in N$, we have the equality

$$\phi_k|_{\tilde{e}_{j_1}} - \phi_k|_{\tilde{e}_{j_2}} = 2\alpha \left(\tilde{e}_{j_2}^k - \tilde{e}_{j_1}^k\right), \tag{6.24}$$

for all agents $k \in N$. Applying Lemma 6.11 in the appendix, condition (6.24) coupled with the fact that ϕ is

a convex function implies that for any siblings $j_1, j_2 \in N$,

$$\tilde{e}_{j_1} = \tilde{e}_{j_2}. \tag{6.25}$$

Since the communication graph is connected and undirected, Equality (6.25) guarantees that there exist at most two different estimation values which we denote by $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$, i.e.,

$$\tilde{e}_i \in \{x, y\}, \forall i \in N. \tag{6.26}$$

Now applying equality (6.22), for each $i \in N$, we have that either $e^k_{\text{diff}} = 2n_i(x_k - y_k)$ or $e^k_{\text{diff}} = -2n_i(x_k - y_k)$, where $n_i = |N_i| - 1 > 0$. If there exist two agents having different numbers of neighbors, we can derive that $x = y$, i.e. $\tilde{e}_i = \tilde{e}_j, \forall i, j \in N$. Following the same arguments as the previous proof, we have that $\tilde{e}^k_i = v_k, \forall i, k \in N$. □

Lemma 6.9. *Suppose that at least one of conditions (i)–(iii) of Theorem 6.3 is satisfied. If* $[x, a]$ *satisfies* $a_i \in \operatorname{argmin}_{\check{a}_i \in \mathcal{A}_i(x)} J_i(x, \check{a}_i, a_{-i})$ *for all* $i \in N$*, then* $\tilde{v}$ *is an optimal solution to (6.1).*

Proof. If $a_k \in \operatorname{argmin}_{\check{a}_i} J_k(x, \check{a}_k, a_{-k})$, where $\check{a}_k = (\check{v}_k, \check{e}_k)$,we have

$$\left.\frac{\partial J_k(x, \check{a}_k, a_{-k})}{\check{v}_k}\right|_{a_k} \cdot (\hat{v}'_k - \hat{v}_k) \geq 0, \forall \hat{v}'_k \in \mathcal{A}^{\hat{v}}_i(x)$$

which is equivalent to

$$\left[n\, \phi_k|_{\tilde{e}} + 2n_k \sum_{j \in N_k} (\tilde{e}^k_k - \tilde{e}^k_j) \right] \cdot (\hat{v}'_k - \hat{v}_k) \geq 0 \ . \tag{6.27}$$

We have shown in Lemma 6.7 and Lemma 6.8 that if $[x, a] = [(v, e), (\hat{v}, \hat{e})]$ satisfies $a_i \in \operatorname{argmin}_{\check{a}_i} J_i(x, \check{a}_i, a_{-i})$, then $\tilde{e}^k_i = v_k, \forall i, k \in N$. Therefore, equation (6.27) tells that

$$\phi_k|_{(\tilde{v})} \cdot (\tilde{v}'_k - \tilde{v}_k) \geq 0, \forall \tilde{v}'_k \in \mathcal{V}_k. \tag{6.28}$$

This implies that $\tilde{v}$ is an optimal profile for the optimization problem (6.1) given that ϕ is convex over $\mathcal{V}$. $\square$

Lemma 6.10. *Suppose that condition (iv) of Theorem 6.3 is satisfied. If $[x, a]$ satisfies $a_i \in \operatorname{argmin}_{\check{a}_i \in \mathcal{A}_i(x)} J_i(x, \check{a}_i, a_{-i})$ for all $i \in N$, then $\tilde{e}_i^k = \tilde{v}_k$ for all $i, j \in N$, and $\tilde{v}$ is an optimal profile for the optimization problem (6.1).*

Proof. In the proof of Lemma 6.7 and Lemma 6.8, we have shown that if $[x, a]$ satisfies $a_i \in \operatorname{argmin}_{\check{a}_i} J_i(x, \check{a}_i, a_{-i})$, equations (6.20) and (6.27) should satisfy. Since $\mathcal{V}_k$ is open, equation (6.27) is equivalent to

$$\phi_k|_{\tilde{e}_k} + 2 \sum_{j \in N_k} \left(\tilde{e}_k^k - \tilde{e}_j^k \right) = 0, \ \forall k \in N. \tag{6.29}$$

Substituting this equation into equation (6.20), we have

$$\phi_k|_{\tilde{e}_l} + 2\tilde{e}_l^k = 2\tilde{e}_k^k, \ \forall l \in \mathcal{N}_k, k \in \mathcal{N}. \tag{6.30}$$

Since ϕ is a convex function, we already have equality (6.26) as shown in the proof of Lemma 6.8. We will show that $x = y$. Suppose that $x \neq y$. For each $i \in N$, either $\tilde{e}_i = x$ or $\tilde{e}_i = y$. Suppose that $\tilde{e}_i = x$. Then for all $j \in N_i$, $\tilde{e}_j = y$; otherwise if $\tilde{e}_j = x$ for some $j \in N_i$, equation (6.25) implies that $\tilde{e}_j = x, \forall j \in N$, i.e. $x = y$. Equation (6.29) tells us that

$$\phi_k|_x = 2n_k(y_k - x_k)$$

where $n_k = |N_k| - 1$. Equation (6.30) tells us that

$$\phi_k|_y = 2(x_k - y_k)$$

If $\tilde{e}_k = y$, similarly we will have:

$$\begin{aligned} \phi_k|_y &= 2n_k(x_k - y_k) \\ \phi_k|_x &= 2(y_k - x_k) \ . \end{aligned}$$

In both cases, we have $\phi_k|_x - \phi_k|_y = 2(n_k + 1)(y_k - x_k)$. Applying Lemma 6.11, we know that $x = y$. Now we can conclude that $\tilde{e}_i = \tilde{e}_j$ and hence $\tilde{e}_i^k = v_k, \forall i, k \in N$. Substituting those equalities into equation

(6.29), we have:

$$\phi_k|_{(\tilde{v}_1,\ldots,\tilde{v}_n)} = 0, \forall k \in N$$

which implies that $\tilde{v}$ is an optimal point of the optimization problem (6.1) given that ϕ is an convex function and $\mathcal{V}$ is open. □

Lemma 6.11. *Given a continuously differentiable convex function $\phi(x_1, x_2, \ldots, x_n)$ and two vectors $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$, if for all $k = 1, 2, \ldots n$, we have $\phi_k|_x - \phi_k|_y = \alpha_k(y_k - x_k)$ where $\alpha_k > 0$, then $x = y$.*

Proof. Since ϕ is a convex function, we have

$$\begin{aligned} \phi(x) &\geq \phi(y) + (x-y)^T \nabla\phi|_y, \\ \phi(y) &\geq \phi(x) + (y-x)^T \nabla\phi|_x. \end{aligned}$$

Adding up the two inequalities, we have

$$0 \geq (x-y)^T(\nabla\phi|_y - \nabla\phi|_x).$$

Since $\phi_k|_x - \phi_k|_y = \alpha(y_k - x_k)$ for all k, we have

$$0 \geq \sum_k \alpha_k (x_k - y_k)^2 \geq 0.$$

Therefore $x = y$. □

6.7.3 A Lemma for gradient play

Lemma 6.12. *Let G be a state based potential game and the potential function $\Phi(x, a)$ a differentiable convex function on variable a. Suppose all agents are using the gradient play algorithm and the state at time t is $x(t) = [v(t), e(t)]$. The action profile at time t is the null action, i.e., $a(t) = \mathbf{0}$, if and only if the state action pair $[x(t), \mathbf{0}]$ is a stationary state Nash equilibrium of the state based game G.*

Proof. Since $f(x(t), \mathbf{0}) = x(t)$, by Definition 6.3 we know that $[x(t), \mathbf{0}]$ is a stationary state Nash equilibrium if and only if $0 \in \operatorname{argmin}_{a_i \in \mathcal{A}_i(x(t))} J_i(x(t), a_i, 0)$ for all $i \in N$. This is equivalent to

$$\left(\frac{\partial J_i(x(t), a)}{\partial a_i}\bigg|_{a=\mathbf{0}} \right) \cdot a_i \geq 0$$

for all $i \in N$ and $a_i \in \mathcal{A}_i(x(t))$. By Projection Theorem, this inequality is equivalent to the fact that the projection of $-\epsilon_i \left.\frac{\partial J_i(x(t),a)}{\partial a_i}\right|_{a=\mathbf{0}}$ onto $\mathcal{A}_i(x(t))$ is 0, i.e.

$$a_i(t) = \left[-\epsilon \cdot \left. \frac{\partial J_i(x(t), a)}{\partial a_i} \right|_{a=\mathbf{0}} \right]^+ = 0, \forall i \in N.$$

□

6.7.4 Proof of Theorem 6.5

1. From Assumption (A-4), we have

$$\begin{aligned} \Phi(x(t+1), \mathbf{0}) &= \Phi(x(t), a(t)) \\ &\leq \Phi(x(t), \mathbf{0}) + a(t)^T \nabla_a \Phi(x(t), \mathbf{0}) + \tfrac{M}{2} ||a(t)||_2^2 \\ &\leq \Phi(x(t), \mathbf{0}) - (\epsilon - \tfrac{M}{2}\epsilon^2) ||\nabla_a \Phi(x(t), \mathbf{0})||_2^2 \ . \end{aligned} \tag{6.31}$$

   Therefore if $\epsilon < \frac{2}{M}$, $\Phi(x(t+1), \mathbf{0}) \leq \Phi(x(t), \mathbf{0})$.

2. Assumption (A-4) also implies the following inequality:

$$\begin{aligned} \Phi(x + Ba, \mathbf{0}) &= \Phi(x, a) \\ &\geq \Phi(x, \mathbf{0}) + a^T \cdot \nabla_a \Phi(x, 0) + \tfrac{m}{2} ||a||^2 \\ &\geq \min_a \left(\Phi(x, \mathbf{0}) + a^T \cdot \nabla_a \Phi(x, 0) + \tfrac{m}{2} ||a||^2 \right) \\ &= \Phi(x, \mathbf{0}) - \tfrac{1}{2m} ||\nabla_a \Phi(x, \mathbf{0})||_2^2 \ . \end{aligned}$$

Since the state transition rule is $x(t+1) = f(x(t), a(t)) = x(t) + Ba(t)$, we have:

$$\begin{aligned} \Phi(x(T), \mathbf{0}) &= \Phi(x(T-1) + Ba(T-1), \mathbf{0}) \\ &= \dots \\ &= \Phi(x(t) + B\sum_{\tau=t}^{T-1} a(t), \mathbf{0}) \\ &\geq \Phi(x(t), \mathbf{0}) - \tfrac{1}{2m}||\nabla_a \Phi(x(t), \mathbf{0})||_2^2 \end{aligned} \tag{6.32}$$

for any $T > t \geq 0$. If we pick $t = 0$, we know that $\{\Phi(x(T), \mathbf{0})\}_{T \geq 0}$ is bounded below. As we showed in the proof for Theorem 6.4, we know that $(x(t), a(t))$ will asymptotically converge to a stationary state Nash equilibrium $[x^*, \mathbf{0}]$ and $\Phi(x(t), \mathbf{0}) \geq \Phi(x^*, \mathbf{0})$ for any $t \geq 0$.

3. Since $x(T) = x(t) + B\sum_{\tau=t}^{T-1} a(\tau)$ and $\lim_{T\to\infty} x(T) = x^*$, we know that $\lim_{T\to\infty} x(t) + B\sum_{\tau=t}^{T-1} a(\tau) = x^*$. Combining with Inequality (6.32), we have:

$$\Phi(x^*, \mathbf{0}) - \Phi(x(t), \mathbf{0}) \geq -\frac{1}{2m}||\nabla_a \Phi(x(t), \mathbf{0})||_2^2 \tag{6.33}$$

for any $t \geq 0$. Substituting this into Inequality (6.31), we have

$$\Phi(x(t+1), \mathbf{0}) \leq \Phi(x(t), \mathbf{0}) - 2m(\epsilon - \frac{M}{2}\epsilon^2)\left(\Phi(x(t), \mathbf{0}) - \Phi(x^*, \mathbf{0})\right)$$

which gives the following inequality:

$$\Phi(x(t+1), \mathbf{0}) - \Phi(x^*, \mathbf{0}) \leq \theta\left(\Phi(x(t), \mathbf{0}) - \Phi(x^*, \mathbf{0})\right) \tag{6.34}$$

where $\theta = \left(1 - 2m(\epsilon - \frac{M}{2}\epsilon^2)\right)$. Therefore we can conclude the statement in this theorem.

□

Chapter 7

Optimization Problem with Coupled Constraints

[]

Several multiagent systems exemplify the need for establishing distributed control laws which ensure that the resulting agents' behavior satisfies a given coupled constraint. This chapter focuses on the design of such distributed control laws through the game theoretic framework given in the previous chapter. In particular, this chapter provides a systematic methodology for the design of local agent objective functions which guarantee that all resulting Nash equilibria optimize the system level objective while also satisfying the given coupled constraints.

7.1 Introduction

In many multi-agent systems, the desired collective behavior must also satisfy a given coupled constraint on the agents' behavior [62,79,81–83,114]. One example is the problem of TCP control where the users' sending rates need to satisfy link capacity constraints [114]. An alternative example is the problem of economic dispatch in an electricity power system where the total power generation needs to satisfy the total power demands [62]. Regardless of the specific application domain, these coupled constraints bring additional complexity to the control algorithm design.

There are two main research directions aimed at designing distributed control algorithms to satisfy performance criteria involving coupled constraints. The first direction seeks to design algorithms which ensure that the coupled constraint is always satisfied, e.g., the well-studied consensus algorithm [79, 101, 115, 116].

While theoretically appealing, such algorithms lack a robustness to environmental uncertainties, noisy measurements, and inconsistent clock rates amongst the agents. The second direction seeks to design algorithms which ensure that the asymptotic behavior satisfies the coupled constraints, e.g., dual decomposition [86, 108, 114, 117, 118] and subgradient methods [119, 120]. Such algorithms often require a two-time scale solution approach by introducing intermediate state variables, such as pricing terms or communication variables, to help coordinate behavior. Depending on the application domain, these approaches may be prohibitive either by the informational dependence on the pricing terms or the rigidity of the update algorithm.

The approaches highlighted above can be viewed as advancements to the field of distributed optimization where the emphasis is on the design of distributed control laws for multiagent systems. Here, the goal is to establish *local* agent control policies, i.e., control policies that depend solely on information regarding a limited subset of neighboring agents, which guarantee that the agents' asymptotic behavior reaches a desired performance regardless of the initial state of the agents. While the design of such networked control systems parallels the theme of distributed optimization, one of the central issues associated with the applicability of distributed optimization algorithms for multiagent coordination is robustness. That is, how are the asymptotic guarantees associated with the aforementioned algorithms affected when there are delays in information, asynchronous clock rates, dynamically changing agent capabilities, or component failures? Unfortunately, the robustness of these algorithms to such issues is typically not characterized, e.g., [108, 119, 120].

In contrast to these algorithmic approaches, we focuses on dealing with the underlying control design through a complimentary direction which involves assigning each agent a local objective function. These objective functions permit each agent to evaluate the desirability of all available actions for any given information pertaining to the environment and behavior of the other agents. The resulting agents' control laws can then be attained by prescribing a protocol for how each agent should process available information to formulate a decision.

The main contribution of this chapter is the development of a systematic methodology for the design of agent objective functions such that (i) the agents' objective functions depend only on local information, (ii) *all* resulting Nash equilibria satisfy the desired performance criterion which embodies coupled constraints, and (iii) the resulting game is a close variant of a *potential game*. The novelty of our approach stems from in-

tegrating classical optimization techniques, in particular exterior penalty methods and barrier function methods, into the design of the agents' objective functions as shown in Sections 7.3.1 and 7.3.2 respectively. Both methodologies ensure that all three objectives are satisfied. The core difference between the two approaches is that the barrier function method can also be used to ensure that the constraint is satisfied dynamically in addition to asymptotically. This work builds on our previous work in Chapter 6 which establishes a similar methodology for the design of agent objective function. However, Chapter 6 did not focus on performance criteria with coupled constraints.

7.2 Problem formulation

We consider a multiagent system consisting of n agents denoted by the set $N = \{1, \cdots, n\}$. Each agent $i \in N$ is endowed with a set of possible decisions (or values), denoted by $\mathcal{V}_i$, which we assume is a convex subset of $\mathbb{R}^{d_i}$, i.e., $\mathcal{V}_i \subseteq \mathbb{R}^{d_i}$ for some integer $d_i \geq 1$. We denote a joint decision by the tuple $v = (v_1, ..., v_n) \in \mathcal{V} = \prod_{i \in N} \mathcal{V}_i$ where $\mathcal{V}$ is referred to as the set of joint decisions. The goal of this chapter is to establish a methodology for attaining a distributed solution to the following optimization problem

$$\begin{array}{ll} \min_{v_i \in \mathcal{V}_i, i \in N} & \phi(v) = \sum_{i \in N} C_i(v_i) \\ \text{s.t.} & \sum_{i=1}^{n} A_i^k v_i - B^k \leq 0, \quad k \in M \end{array} \tag{7.1}$$

where $C_i : \mathcal{V}_i \to \mathbb{R}$ represents a local cost function for agent i, which is assumed to be a differentiable convex function, and the linear inequalities $\{\sum_{i=1}^{n} A_i^k v_i - B^k \leq 0\}_{k \in M}$ where $M = \{1, ..., m\}$ characterizes the coupled constraints on the agents' decisions. The distributed algorithm will produce a sequence of decision profile $v(1)$, $v(2)$, ..., where the decision of each agent $i \in N$ at each iteration $t \in \{1, 2, \dots\}$ is selected according to a control law of the form

$$v_i(t) = \Pi_i \left(\{\text{Information about agent } j \text{ at time } t\}_{j \in N_i} \right), \tag{7.2}$$

where $N_i \subseteq N$ identifies the neighbor set (or information set) of agent i. The neighbor sets $\{N_i\}_{i \in N}$, which we will refer to as the *communication graph*, capture the locality of the distributed algorithm. By convention,

we assume that $i \in N_i$ for each $i \in N$. In Section 7.4 we provide a detailed example pertaining to the problem of economic dispatch and demand management in electricity power systems which is precisely of this form.[1]

7.3 A methodology for objective function design

In this section we present two methodologies for local objective design which incorporate penalty functions and barrier functions into the design of agent objective functions. The core difference between the two approaches is that barrier functions can be used to ensure that the coupled constraint is satisfied dynamically in addition to asymptotically.

7.3.1 Design using exterior penalty functions

Our first design methodology integrates *exterior penalty functions* into the agents' cost functions. The forthcoming design will embody the following four properties:

(i) The state represents a compilation of local state variables, i.e., the state x can be represented as $x = (x_1, \ldots, x_n)$ where each x_i represents the state of agent i. Furthermore, the state transition depends only on local information.

(ii) The objective function for each agent i is local and of the form $J_i(\{x_j, a_j\}_{j \in N_i}) \in \mathbb{R}$.

(iii) The resulting game is a state based potential game.

(iv) The stationary state Nash equilibria are optimal in the sense that they represent solutions to the optimization problem in (7.1).

[1] We now provide a few remarks regarding the optimization problem presented in (7.1). First, we do not explicitly highlight the equality constraint, $\sum_{i=1}^n A_i^k v_i - B^k = 0$, since this can be handled by two inequalities of the form $\sum_{i=1}^n A_i^k v_i - B^k \leq 0$ and $-\left(\sum_{i=1}^n A_i^k v_i - B^k\right) \leq 0$. Second, for ease of exposition we focus purely on the case $d_i = 1$ for all $i \in N$. However, the forthcoming results also hold for both higher dimensions, i.e, $\mathcal{V}_i \subseteq \mathbb{R}^{d_i}$ where $d_i > 1$, and heterogeneous dimensions, i.e., d_i need not equal d_j for $i \neq j$. Lastly, since the focus of this chapter is about decoupling coupled constraints, we focus purely on the case when the objective function ϕ is decomposable but the constraints are coupled. By combining the design presented in the last chapter, which focuses on optimization problems with coupled objective functions but decoupled constraints, we can also deal with coupled objective functions in (7.1) in a similar fashion.

State Space: The starting point of our design is an underlying state space X where each state $x \in X$ is defined as a tuple $x = (v, e)$, where $v = (v_1, \ldots, v_n) \in \mathbb{R}^n$ is the profile of values and $e = \left\{e_i^k\right\}_{k \in M, i \in N}$ is the profile of estimation terms. The term e_i^k represents agent i's estimate for the k-th constraint, i.e., $e_i^k \sim \sum_{j=1}^n A_j^k v_j - B^k$. Note that each agent possesses an estimation term for each constraint $k \in M$.

Actions: Each agent i is assigned a state dependent action set $\mathcal{A}_i(x)$ that permits the agent to change its value and constraint estimation through communication with neighboring agents. Specifically, an action a_i is defined as a tuple $a_i = \left(\hat{v}_i, \left\{\hat{e}_i^1, ..., \hat{e}_i^m\right\}\right)$ where $\hat{v}_i$ indicates a change in the agent's value and $\hat{e}_i^k$ indicates a change in the agent's estimate of the k-th constraint. Here, the change in estimation terms for agent i pertaining to constraint k is represented by a tuple $\hat{e}_i^k = \left\{\hat{e}_{i\to j}^k\right\}_{j \in N_i}$. The term $\hat{e}_{i\to j}^k$ indicates the estimation value that agent i exchanges (or passes) to agent $j \in N_i$ regarding the k-th constraint.

State Dynamics: For any state $x = (v, e)$ and action $a = (\hat{v}, \hat{e})$, the state transition function $f(x, a)$ is of the form

$$\begin{aligned} \tilde{v}_i &= v_i + \hat{v}_i, \\ \tilde{e}_i &= \left\{e_i^k + A_i^k \hat{v}_i + \hat{e}_{i\leftarrow \text{in}}^k - \hat{e}_{i\to \text{out}}^k\right\}_{k \in M}, \end{aligned} \tag{7.3}$$

where $(\tilde{v}, \tilde{e}) = f(x, a)$ is the ensuing state and $\hat{e}_{i\leftarrow \text{in}}^k = \sum_{j \in N_i} \hat{e}_{j\to i}^k$ and $\hat{e}_{i\to \text{out}}^k = \sum_{j \in N_i} \hat{e}_{i\to j}^k$. The admissible action set of agent i given a state x is

$$\mathcal{A}_i^{\text{PF}}(x) = \{(\hat{v}, \hat{e}) : v_i + \hat{v}_i \in \mathcal{V}_i\} \tag{7.4}$$

and the null action, $\mathbf{0}$, takes on the form $\hat{v}_i = 0$ and $\hat{e}_{i\to j}^k = 0$ for all $i, k \in N$ and $j \in N_i$. Note that $\mathbf{0} \in A_i(x)$ for any $i \in N, x \in X$.

Invariance Property: If the initial estimation terms $e(0)$ satisfy $\sum_{i \in N} e_i^k(0) = \sum_{i \in N} A_i^k v_i(0) - B^k$, $\forall k \in M$, then for any sequence of actions $a(0), a(1), \ldots$, the resulting state trajectory generated according to

process $x(t+1) = f(x(t), a(t))$ satisfies

$$\sum_{i \in N} e_i^k(t) = \sum_{i \in N} A_i^k v_i(t) - B^k \tag{7.5}$$

for all constraints $k \in M$ and $t \geq 0$. Hence, for any constraint $k \in M$ we have know that

$$\sum_{i \in N} e_i^k(t) \leq 0 \Leftrightarrow \sum_{i \in N} A_i^k v_i(t) - B^k \leq 0 \tag{7.6}$$

as the estimation terms encode information pertaining to constraint violations. Note that if the initial value profile $v(0)$ satisfies the constraints $k \in M$, then assigning $e_i^k(0) = A_i^k v_i(0) - (1/n)B^k$ ensures that $e(0)$ satisfy the above condition. We will assume throughout that the initial value and estimation profiles satisfies these initial conditions.

Agent Cost Functions: For any state $x \in X$ and admissible action profile $a \in \prod_{i \in N} \mathcal{A}_i(x)$, the cost function of agent i is defined as

$$J_i^{\text{PF}}(x, a) = C_i(\tilde{v}_i) + \mu \sum_{j \in N_i} \sum_{k=1}^{m} \left[\max\left(0, \tilde{e}_j^k\right)\right]^2 \tag{7.7}$$

where $(\tilde{v}, \tilde{e}) = f(x, a)$ is the ensuing state and $\mu > 0$ is a trade-off parameter. The first term captures agent i's local cost function while the second term introduces a penalty on inconsistencies in estimation terms between neighboring agents.

We now provide the main result of this chapter.

Theorem 7.1. *Model the constrained optimization problem in (7.1) as a state based game with a fixed trade-off parameter $\mu > 0$ as depicted in Section 7.3.1. The state based game is a state based potential game with potential function*

$$\Phi^{\text{PF}}(x, a) = \phi(\tilde{v}) + \mu \sum_{i \in N} \sum_{k=1}^{m} \left[\max\left(0, \tilde{e}_i^k\right)\right]^2 \tag{7.8}$$

where $(\tilde{v}, \tilde{e}) = f(x, a)$ represents the ensuing state. Furthermore, if the objective function $\phi : \mathcal{V} \to \mathbb{R}$ is

convex and differentiable and the communication graph is undirected and connected, then a state action pair $[x, a] = [(v, e), (\hat{v}, \hat{e})]$ *is a stationary state Nash equilibrium if and only if the following four conditions are satisfied:*

(i) The value profile v *is an optimal point of the uncoupled constrained optimization problem*

$$\min_{v \in \mathcal{V}} \phi(v) + \frac{\mu}{n} \sum_{k \in M} \left[\max \left(0, \sum_{i \in N} A_i^k v_i - B^k \right) \right]^2 . \tag{7.9}$$

(ii) The estimation profile e *satisfies that for all* $i \in N$, $k \in M$,

$$\max \left(0, e_i^k\right) = \frac{1}{n} \max \left(0, \sum_{i \in N} A_i^k v_i - B^k \right),$$

(iii) The change in value profile satisfies $\hat{v}_i = 0$ *for all agents* $i \in N$.

(iv) The net change in estimation profile is 0*, i.e.,* $\hat{e}^k_{i \leftarrow \text{in}} - \hat{e}^k_{i \rightarrow \text{out}} = 0$ *for all agents* $i \in N$ *and constraints* $k \in M$.

This characterization proves the equivalence between the stationary state Nash equilibria of the designed game and the solutions to the uncoupled constrained optimization problem in (7.9). Therefore, as $\mu \to \infty$ all equilibria of our designed game are solutions to the coupled constrained optimization problem in (7.1) [113].

Proof. It is straightforward to show that the potential function in (7.8) satisfies the conditions of state based potential games given in Definition 6.2. Hence, we will focus purely on the presented characterization. Throughout, we will use the notation $(\tilde{v}, \tilde{e})$ to represent the ensuing state for a state action pair $[x, a]$, i.e., $(\tilde{v}, \tilde{e}) = f(x, a)$.

($\Leftarrow$) We start by proving that if a state action pair $[x, a]$ satisfies conditions (i)–(iv) then $[x, a]$ is a stationary state Nash equilibrium. First, we know that if a satisfies conditions (iii)–(iv) then $x = f(x, a)$. Hence, we only need to prove that $a \in \text{argmin}_{\check{a} \in \mathcal{A}(x)} \Phi(x, \check{a})$. Let $\check{a} := (\check{v}, \check{e}) \in \mathcal{A}(x)$. Since $\Phi(x, \check{a}) = \Phi(x, (\check{v}, \check{e}))$ is convex over $(\check{v}, \check{e})$, the necessary and sufficient conditions for $a = (\hat{v}, \hat{e})$ to be an optimal solution of the

optimization problem $\min_{\check{a}\in\mathcal{A}(x)} \Phi(x,\check{a})$ are

$$\left.\frac{\partial\Phi(x,\check{a})}{\partial\check{e}^k_{i\to j}}\right|_a = 0,\ \forall i\in N,\ j\in N_i,\ k\in M, \tag{7.10}$$

$$\left.\frac{\partial\Phi(x,\check{a})}{\partial\check{v}_i}\right|_a \cdot(\hat{v}'_i-\hat{v}_i)\geq 0,\ \forall i\in N,\ \hat{v}'_i\in\mathcal{A}^v_i(x). \tag{7.11}$$

Since $\Phi(x,a)=\Phi(\tilde{x},\mathbf{0})=\Phi(f(x,a),\mathbf{0})$, we have $\frac{\partial\Phi(x,a)}{\partial a}=\frac{\partial\Phi(f(x,a),\mathbf{0})}{\partial a}$. Therefore (7.10) and (7.11) simplify to

$$\max\left(0,\tilde{e}^k_j\right)-\max\left(0,\tilde{e}^k_i\right)=0,\ \forall i\in N,\ j\in N_i,\ k\in M, \tag{7.12}$$

$$\left[\left.\frac{\partial\phi}{\partial v_i}\right|_{\tilde{v}}+2\mu\sum_{k\in M}A^k_i\max\left(0,\tilde{e}^k_i\right)\right]\cdot(\tilde{v}'_i-\tilde{v}_i)\geq 0,\ \forall i\in N,\ \tilde{v}'_i\in\mathcal{V}_i. \tag{7.13}$$

To complete the proof of this direction we will actually prove the following stronger statement: if the ensuing state $\tilde{x}$ of a state action pair $[x,a]$ satisfies conditions (i)–(ii), then $a\in\text{argmin}_{\check{a}\in\mathcal{A}(x)}\Phi(x,\check{a})$. For such a state action pair $[x,a]$, it is straightforward to show that $\tilde{x}$ satisfies the following conditions:

$$\max\left(0,\tilde{e}^k_i\right)=\max\left(0,\tilde{e}^k_j\right)=\frac{1}{n}\max\left(0,\sum_{i=1}^n A^k_i\tilde{v}_i-B^k\right),\ \forall i,j\in N,\ k\in M, \tag{7.14}$$

$$\left[\left.\frac{\partial\phi}{\partial v_i}\right|_{\tilde{v}}+\frac{2\mu}{n}\sum_{k\in M}A^k_i\max\left(0,\sum_{i=1}^n A^k_i\tilde{v}_i-B^k\right)\right]\cdot(\tilde{v}'_i-\tilde{v}_i)\geq 0,\ \forall i\in N,\ \tilde{v}'_i\in\mathcal{V}_i. \tag{7.15}$$

Equation (7.14) is from condition (i) and equality (7.5). Equation (7.15) is the optimal condition of the optimization problem $\min_{v\in\mathcal{V}}\phi(v)+\frac{\mu}{n}\alpha(v)$. Substituting (7.14) into (7.15) proves that $\tilde{x}$ satisfies the two optimality conditions in (7.12) and (7.13). Hence, $a\in\text{argmin}_{\check{a}\in\mathcal{A}(x)}\Phi(x,\check{a})$. Therefore, we can conclude that such $[x,a]$ is a stationary state Nash equilibrium.

($\Rightarrow$) Now we prove the other direction of this theorem. First, notice that if $[x,a]$ is a stationary state Nash equilibrium, then the action profile $a=(\hat{v},\hat{e})$ must satisfy conditions (iii)–(iv). Otherwise, $x=(v,e)\neq f(x,a)$. Secondly if $[x,a]$ is a stationary state Nash equilibrium, $J_i(x,a_i,a_{-i})=\min_{\check{a}_i\in\mathcal{A}_i(x)}J_i(x,\check{a}_i,a_{-i})$

for each $i \in N$. Since $J_i(x, \check{a}_i, a_{-i})$ is a convex function on $\check{a}_i := (\check{v}_i, \check{e}_i) \in \mathcal{A}_i(x)$, we know that

$$\left.\frac{\partial J_i(x, \check{a}_i, a_{-i})}{\partial \check{e}_i}\right|_a = 0, \ \forall i \in N, k \in M, \tag{7.16}$$

$$\left[\left.\frac{\partial J_i(x, \check{a}_i, a_{-i})}{\partial \check{v}_i}\right|_a\right] \cdot (\hat{v}_i' - \hat{v}_i) \geq 0, \ \forall i \in N, \hat{v}_i' \in \mathcal{A}_i^v(x), \tag{7.17}$$

which is equivalent to

$$2\mu \left(\max\left(0, \tilde{e}_l^k\right) - \max\left(0, \tilde{e}_i^k\right)\right) = 0, \ \forall i \in N, j \in N_i, k \in M \tag{7.18}$$

$$\left[\left.\frac{\partial C_i}{\partial v_i}\right|_{\tilde{v}} + 2\mu \sum_{k \in M} A_i^k \max\left(0, \tilde{e}_i^k\right)\right] \cdot (\tilde{v}_i' - \tilde{v}_i) \geq 0, \ \forall i \in N, \ \tilde{v}_i' \in \mathcal{V}_i. \tag{7.19}$$

Equation (7.18) implies that $\max\left(0, \tilde{e}_i^k\right) = \max\left(0, \tilde{e}_j^k\right)$ for all agents $i, j \in N$ and constraints $k \in M$ since the communication graph is connected. Applying the equality in (7.5), we have that for all agents $i \in N$ and constraints $k \in M$, $\max\left(0, \tilde{e}_i^k\right) = \frac{1}{n} \max\left(0, \sum_{i=1}^n A_i^k \tilde{v}_i - B^k\right)$. Substituting this equality into (7.19) gives us

$$\left[\left.\frac{\partial \phi}{\partial v_i}\right|_{\tilde{v}} + \frac{2\mu}{n} \sum_{k \in M} A_i^k \max\left(0, \sum_{i=1}^n A_i^k \tilde{v}_i - B^k\right)\right] \cdot (\tilde{v}_i' - \tilde{v}_i) \geq 0 \tag{7.20}$$

for all $\tilde{v}_i' \in \mathcal{V}_i$. Hence, $\tilde{v}$ is the optimal solution to $\phi(v) + \frac{\mu}{n}\alpha(v)$. Combining this with the fact that $x = \tilde{x} = f(x, a)$, we can conclude that $x = (v, e)$ satisfies Conditions (i)-(iv). □

7.3.2 Design using barrier functions

In this section we introduce our second design which integrates *barrier functions*, as opposed to exterior penalty functions, into the design of the agents' cost functions. The key difference between the two approaches lies in the feasibility of both the intermediate and asymptotic solutions. In particular, barrier functions can be employed to ensure that both the intermediate and asymptotic solutions are in the interior feasible set. Accordingly we assume that the interior feasible set of problem (7.1) is nonempty when implementing barrier function methods. Note that this implies that equality constraints are not permissible.

State Space, Actions, State Dynamics: These three parts are identical to those in Section 7.3.1.

Admissible Action Sets: Let $x = (v, e)$ represent a *strictly feasible* state where the value profile v satisfies

$\sum_{i=1}^{n} A_i^k v_i < B^k$, $v_i \in \mathcal{V}_i$ and the estimation profile e satisfies $e_i^k < 0$ for each $i \in N$ and $k \in M$. Define the admissible action set for each agent $i \in N$ as

$$\mathcal{A}_i^{BF}(x) = \left\{(\hat{v}_i, \hat{e}_i) : v_i + \hat{v}_i \in \mathcal{V}_i, e_i^k + A_i^k \hat{v}_i - \hat{e}_{i\to\text{out}}^k < 0, \hat{e}_{i\to j}^k \leq 0, \forall k \in M\right\}. \tag{7.21}$$

It can be checked that if the initial state $x(0)$ is strictly feasible and the initial estimation term $e(0)$ satisfies $\sum_{i\in N} e_i^k(0) = \sum_{i\in N} A_i^k v_i(0) - B^k$, then the resulting state trajectory generated according to process $x(t+1) = f(x(t), a(t))$ where $a(t) \in \prod_{i\in N} \mathcal{A}_i(x(t))$ for all $t \geq 0$ is also strictly feasible.

Agents' Cost Functions: For any state $x \in X$ and admissible action profile $a \in \prod_{i\in N} \mathcal{A}_i(x)$, the cost function of agent i is defined as

$$J_i^{BF}(x, a) = C_i(\tilde{v}_i) - \mu \sum_{j\in N_i} \sum_{k=1}^{m} \log\left(-\tilde{e}_j^k\right) \tag{7.22}$$

where $(\tilde{v}, \tilde{e}) = f(x, a)$ is the ensuing state and $\mu > 0$ is a trade-off parameter. Note that the sole difference between (7.22) and (7.7) rests on the penalty associated with being close to constraint violations.

The above methodology using barrier functions yields a game that possesses the same analytical properties of the designed game using exterior penalty function as given in Theorem 7.1 with two exceptions. First, the potential function of the state based game is now of the form

$$\Phi^{BF}(x, a) = \phi(\tilde{v}) - \mu \sum_{i\in N} \sum_{k=1}^{m} \log\left(-\tilde{e}_i^k\right). \tag{7.23}$$

Second, (7.9) is replaced with a new optimization problem of the form

$$\begin{array}{ll} \min_{v\in\mathcal{V}} & \phi(v) - n\mu \sum_{k\in M} \log\left(B^k - \sum_{i\in N} A_i^k v_i\right) \\ \text{s.t.} & \sum_{i=1}^{m} A_i^k v_i - B^k < 0, k \in M. \end{array} \tag{7.24}$$

We omit a formal statement of the theorem in addition to the proof for brevity as it is virtually identical to that of Theorem 7.1. Note that as $\mu \to 0$, all equilibria of our designed game are solutions to the constrained optimization problem in (7.1) [113].

7.4 An illustrative example

Consider an economic dispatch problem in electricity power systems, introduced in [62], with N generators and a demand requirement $D \geq 0$. Each generator is capable of generating an amount of power $v_i \in \mathcal{V}_i = [\underline{v}_i, \bar{v}_i]$, where $\underline{v}_i$ and $\bar{v}_i$ denote the minimum and maximum generation levels respectively, subject to a cost $C_i(v_i)$. The system level objective is to meet the demand level D while minimizing the sum of the costs incurred by the generators. More specifically, the system level objective is of the form

$$\begin{array}{ll} \min_{v_i \in [\underline{v}_i, \bar{v}_i]} & \phi(v) = \sum_{i \in N} C_i(v_i) \\ \text{s.t.} & \sum_{i \in N} v_i \geq D. \end{array} \tag{7.25}$$

One of the central challenges associated with attaining generation levels $v \in \mathcal{V}$ to optimize (7.25) is that each individual generator selects its own generation level in response to incomplete information regarding the system as a whole.[2]

Consider a simple economic dispatch problem where $N = \{1, 2, 3, 4\}$, generation capabilities $\mathcal{V}_i = [0, 5]$ for all $i \in N$, cost functions $C_i(v_i) = v_i^2 + v_i + 10$ for $i \in \{1, 2\}$ and $C_i(v_i) = 0.5v_i^2 + v_i + 10$ for $i \in \{3, 4\}$, a demand $D = 12$, and a communication graph of the form $1 \leftrightarrow 2 \leftrightarrow 3 \leftrightarrow 4$. It is straightforward to verify that the optimal generation levels are $(2, 2, 4, 4)$. The methodologies developed in this chapter can be used to attain a distributed solution to this economic dispatch problem that satisfies the communication graph. The following highlights the specifics of our design while focusing on generator 2 for the penalty function method given in Section 7.3.1:

- State: $x_2 = (v_2, e_2)$;
- Action: $a_2 = (\hat{v}_2, \hat{e}_{2\to 1}, \hat{e}_{2\to 3})$;
- Admissible action set: $\mathcal{A}_2^{\mathrm{PF}}(x_2) = \{(\hat{v}_2, \hat{e}_2) : v_2 + \hat{v}_2 \in [\underline{v}_2, \bar{v}_2]\}$;
- State dynamics $(\tilde{v}, \tilde{e}) = f(x, a)$: $\tilde{v}_2 = v_2 + \hat{v}_2$ and $\tilde{e}_2 = e_2 - \hat{v}_2 + \hat{e}_{2\leftarrow \mathrm{in}} - \hat{e}_{2\to \mathrm{out}}$;

[2] Dual algorithms are commonly employed for attaining distributed solutions to constrained optimization problems such as the presented economic dispatch problem. A dual algorithm requires a two-time scale setting where a centralized authority sets appropriate pricing terms, through the use of Lagrangian dual variables, to coordinate behavior. However, in this chapter, we study algorithms where no such centralized authority exists.

- Cost functions: $J_2^{\mathrm{PF}}(x,a) = C_2(\tilde{v}_2) + \mu \sum_{j\in N_2} \sum_{k\in N} \left[\max\left(0, \tilde{e}_j\right)\right]^2$.

The specifics for the alternative generators could be derived in a similar fashion. Likewise, integrating barrier functions as opposed to penalty functions would incorporate substituting $\mathcal{A}_i^{\mathrm{BF}}$ for $\mathcal{A}_i^{\mathrm{PF}}$ as defined in (7.21) and J_i^{BF} for J_i^{PF} as defined in (7.22).

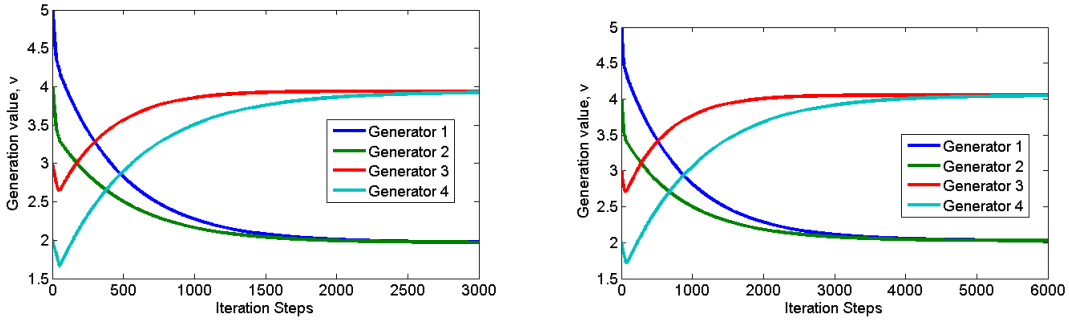
Figure 7.1 shows simulation results for both the penalty function method and barrier function method when employing the learning algorithm gradient play and initializing the generation levels at $v(0) = (5,4,3,2)$. The learning algorithm gradient play, given in the previous chapter, guarantees convergence to a stationary state Nash equilibrium in any state based potential game. Hence gradient play can be utilized to complete the control design. The gradient play algorithm takes on the following forms:

Penalty Function Method	**Barrier Function Method**
$\hat{v}_i(t) = \left[-\epsilon \cdot \frac{\partial J_i(x(t),a)}{\partial \hat{v}_i}\Big\vert_{a=\mathbf{0}}\right]^+_{\mathcal{A}_i^{\mathrm{PF}}(x(t))}$	$\hat{v}_i(t) = \beta(t)\left(-\epsilon \frac{\partial J_i(x(t),a)}{\partial \hat{v}_i}\Big\vert_{a=\mathbf{0}}\right)$
$\hat{e}^k_{i\to j}(t) = -\epsilon \cdot \frac{\partial J_i(x(t),a)}{\partial \hat{e}^k_{i\to j}}\Big\vert_{a=\mathbf{0}}$	$\hat{e}^k_{i\to j}(t) = \beta(t)\min\left(0, -\epsilon \frac{\partial J_i(x(t),a)}{\partial \hat{e}^k_{i\to j}}\Big\vert_{a=\mathbf{0}}\right)$

where $\epsilon = 0.002$ is the step size, $[\cdot]^+$ represents the projection onto the represented closed convex set, and $\beta(t) = \left(\frac{1}{2}\right)^{l(t)}$ where $l(t)$ is the smallest nonnegative integer l such that $(\hat{v}_i(t), \hat{e}_{i\to j}(t)) \in \mathcal{A}_i^{\mathrm{BF}}(x(t))$. Note that computing such gradients only requires each agent $i \in N$ to have access to the state of neighboring agents $j \in N_i$, i.e., $\{x_j(t)\}_{j\in N_i}$. For both approaches, the generation levels quickly converge close to the optimal generation levels; however, the barrier function method takes longer to converge. This is expected since the barrier function approach also ensures that the demand is satisfied dynamically and asymptotically, which is not guaranteed by the penalty function approach.

7.5 Conclusion

This chapter focuses on the general question of how to design local agent objective functions for distributed engineering systems with coupled constraints. By combining the design presented in the last chapter, which focuses on optimization problems with coupled objective functions but decoupled constraints, we can also deal with optimization problems with coupled objective functions and coupled constraints in a similar fashion.



(a) Exterior Penalty Functions

(b) Barrier Functions

Figure 7.1. Simulation results for the economic dispatch problem. Subfigure 7.1(a) shows the simulation results when using gradient play applied to the state based game with exterior penalty functions using a tradeoff parameter $\mu = 60$. The simulation demonstrates that the profile of generation levels quickly approaches $(1.97, 1.97, 3.93, 3.93)$ which is close to optimal. However, the generation levels do necessarily satisfy the demand. Subfigure 7.1(b) shows the simulation results when using gradient play applied to the state based game with barrier functions using a tradeoff parameter $\mu = 0.2$. The simulation demonstrates that the profile of generation levels quickly approaches $(2.03, 2.03, 4.02, 4.02)$ which is close to optimal. Furthermore, the generation levels always exceed the demand in this setting.

Chapter 8

Distributed Optimization with a Time Varying Communication Graph

[] The previous two chapters used the framework of state based games to identify a systematic methodology for engineering a decision making architecture, where agents make independent decisions in response to local information while ensuring that the emergent global behavior is desirable with respect to a given system level objective. A drawback of the previous approach is the dependence on a time-invariant and connected communication graph. These conditions are not practical for a wide variety of multi-agent systems. In this chapter we propose a new game theoretical approach for addressing distributed optimization problems that permits relaxations in the structure of the communication graph.

8.1 Introduction

In Chapters 6-7 we identified a systematic methodology for the design of local agent objective functions that satisfies virtually any degree of locality while ensuring that all resulting Nash equilibria represent optimal solutions to a global optimization problem. That design paralleled the theme of distributed optimization algorithm design which can be considered as a concatenation of a designed game and a distributed learning algorithm.

However, a drawback of our previous proposed game design methodology is the dependence on a connected, undirected, and time-invariant communication graph. These conditions are not practical for a wide variety of multi-agent systems. In this chapter we propose a new methodology for addressing this task that permits relaxations in the structure of the communication graph while still ensuring the efficiency of the re-

sulting equilibria. The communication graph is allowed to be time-varying and even unconnected at frequent times.

The key enabler for this result is the same as in Chapter 6, i.e. the addition of local state variables to the game environment. These state variables are utilized as a coordinating entity to decouple the system level objective into agent specific objectives of the desired interdependence. The difference between this work and our previous work lies in the design of local objective functions. Here, the resulting game is a state based potential game with a state based potential function possessing a property which is invariant to the structure of the communication graph. This is in contrast to the design in Chapter 6 where the state based potential function is dependent on the structure communication graph. This invariant property of the state based potential function allows our proposed methodology to distributively solve the global optimization problem under almost any practical setting for the time-varying communication graph. Therefore, there is no specific time-varying rule modeled in this chapter for the communication graph. Our results show that as long as the communication graph is sufficiently connected over time, the distributed algorithm we propose will converge to the optimal solution of the global optimization problem. More rigorous arguments will follow in the later part of the chapter.

The structure of the chapter is as follows. Section 8.2 introduces the problem setup. Section 8.3 provides a state based game design and analyzes the properties of the designed game. Section 8.4 proposes a distributed learning algorithm to reach the equilibria in the designed state based game which represents the optimal solution for the global optimization problem. Lastly Section 8.5 provides a simple example to illustrate our methodology and Section 8.6 concludes the chapter.

8.2 Preliminaries

8.2.1 Problem setup

We consider a multiagent system consisting of n agents denoted by the set $N := \{1, \cdots, n\}$. Each agent $i \in N$ is endowed with a set of possible decisions (or values) denoted by V_i which is a convex subset of $\mathbb{R}^{d_i}$,

i.e. $V_i \subset \mathbb{R}^{d_i}$.[1] We denote a joint decision by the tuple $(v_1, \cdots, v_n) \in V := \prod_{i \in N} V_i$ where V is referred to as the set of joint decisions. There is a global objective of the form $\phi : \mathbb{R}^n \to \mathbb{R}$ that a system designer seeks to minimize. We assume throughout that the objective function ϕ is differentiable convex unless otherwise noted. More formally, the optimization problem takes on the form:

$$\begin{aligned} \min_{v_i} \quad & \phi(v_1, v_2, \ldots, v_n) \\ s.t. \quad & v_i \in V_i, \forall i \in N. \end{aligned} \tag{8.1}$$

The goal of this chapter is to establish a distributed interaction framework for this optimization problem where each agent $i \in N$ makes its decision independently in response to local information. The agents' decisions interact with each other through local communication which is defined by a communication graph. The difference between the problem considered in this chapter and the one in Chapter 6 is that we now allow the communication graph between agents to be time varying. We represent the information available to each agent at time $t \in \{0, 1, ...\}$ by an undirected communication graph $\mathcal{G}(t) = \{N, \mathcal{E}(t)\}$ with nodes N and edges $\mathcal{E}(t)$. By convention, we let $(i, i) \in \mathcal{E}(t)$ for all $i \in N$ and $t \geq 0$. Define the neighbors of agent i at time t as $N_i(t) := \{j \in N : (i, j) \in \mathcal{E}(t)\}$. The distributed learning framework produces a sequence of decision $v(0), v(1), v(2), \ldots$ where at each iteration $t \in \{0, 1, \ldots\}$ the decision of each agent i is chosen independently according to a local control law of the following form

$$v_i(t) = F_i\left(\{v_j(t-1)\}_{j \in N_i(t-1)}\right).$$

Our goal is to design the local controllers $\{F_i(\cdot)\}_{i \in N}$ within the desired information constraints such that the collective behavior converges to a joint decision v^* that solves the optimization problem in (8.1).

[1] For ease of exposition, we let $d_i = 1$ for all $i \in N$. The results in this paper hold for cases where $d_i > 1$. Moreover, d_i can be different from d_j if $i \neq j$.

8.3 State based game design

8.3.1 A state based game design

State Space: The starting point of our design is an underlying state space X where each state $x \in X$ is defined as a tuple $x = (v, e, \mathcal{G})$ with the following elements:

- $v = (v_1, \ldots, v_n) \in \mathbb{R}^n$ is the profile of values.
- $e = (e_1, \ldots, e_n)$ is the profile of estimation terms where $e_i = (e_i^1, \cdots, e_i^n) \in \mathbb{R}^n$ is player i's estimation for the joint action profile v. The term e_i^k captures player i's estimate of player k's actual value v_k. The estimation terms are introduced as a means to relax the degree of information available to each agent.
- $\mathcal{G}$ is the undirected communication graph. We represent the communication graph as $\mathcal{G} = (N_1, N_2, \ldots, N_n)$ where N_i is the neighbor sets of agent i.

Action Sets: Each agent i is assigned an action set $\mathcal{A}_i$ that permits agents to change their value and change their estimation through communication with neighboring agents. Specifically, an action for agent i is defined as a tuple $a_i = (\hat{v}_i, \hat{e}_i)$ where

- $\hat{v}_i \in \mathbb{R}$ indicates a change in the agent's value v_i, and
- $\hat{e}_i := \{\hat{e}_{i\to j}^k\}_{j\in N}^{k\in N}$ indicates a change in the agent's estimation terms e_i where $\hat{e}_{i\to j}^k \in \mathbb{R}$ represents the estimation value that player i passes to player j regarding to the value of player k.

Since a player is only allowed to communicate with its neighbors, the admissible actions for $\hat{e}_i$ given the state x is

$$\mathcal{A}_i^e(x) := \left\{ \{\hat{e}_{i\to j}^k\}_{j\in N}^{k\in N} : \hat{e}_{i\to j}^k = 0, \forall j \notin N_i, k \in N \right\}.$$

Here 0 means that player i does not pass any estimation to player j.

State Transition Rules: We now describe how the state involves.

- The evolution of the value profile v is captured by a time-invariant, deterministic, and local state transition rule of the form:

$$P_t^v(x,a) = P^v(x,a) = \{v_i + \hat{v}_i\}_{i\in N}. \tag{8.2}$$

- The evolution of the estimation profile e is also captured by a time-invariant, deterministic, and local state transition rules of the form:

$$P_t^e(x,a) = P^e(x,a) = \{e_i^k + n\delta_i^k \hat{v}_i + \check{e}_i^k\}_{i,k\in N} \tag{8.3}$$

  where $\check{e}_i^k \triangleq \sum_{j\in N_i} \hat{e}_{j\to i}^k - \sum_{j\in N_i} \hat{e}_{i\to j}^k$ and δ_i^k is an indicator function, i.e., $\delta_i^i = 1$ and $\delta_i^k = 0$ for all $k \neq i$.

- The state transition for the communication graph $\mathcal{G}$ is given as $P_t^{\mathcal{G}} : X \times \mathcal{A} \to \Delta(\bar{\mathcal{G}})$ at each time t. Here $\bar{\mathcal{G}}$ denotes the set of all undirected communication graph and $\Delta(\bar{\mathcal{G}})$ denotes the set of probability distributions over this set. In practice different applications would have different evolution rules $P_t^{\mathcal{G}}$. To ensure the generality of our model, we do not assign any specific evolution rule for $P_t^{\mathcal{G}}$ and later we will show that as long as the undirected $\mathcal{G}(t)$ is connected sufficiently over the time, our approach can solve the optimization problem (8.1). Notice that since the state transition rule $P_t^{\mathcal{G}}$ is allowed to be time-dependent/variant, the evolution rule $P_t^{\mathcal{G}}$ can also model the situation where the graph transition is determined/affected by exogenous disturbances.

Notice that each agent i can update its own local state (v_i, e_i) using local state and action information through Equation (8.2,8.3). Since the optimization problem in (8.1) imposes the requirement that $v_i \in V_i$, we condition the available actions for $\hat{v}_i$ to an agent i on the current state $x = (v, e)$ as:

$$\mathcal{A}_i^v(x) := \{\hat{v}_i : v_i + \hat{v}_i \in V_i\}. \tag{8.4}$$

The admissible action set is defined as $\mathcal{A}_i(x) \triangleq \mathcal{A}_i^v(x) \times \mathcal{A}_i^e(x)$.

Invariance Property of State Dynamics: Let $v(0) = (v_1(0), ..., v_n(0))$ be the initial values of the agents. Define the initial estimation terms $e(0)$ to satisfy $\sum_{i \in N} e_i^k(0) = n \cdot v_k(0)$ for each agent $k \in N$; hence, the initial estimation values are contingent on the initial values. Note that satisfying this condition is trivial as we can set $e_i^i(0) = n \cdot v_i(0)$ and $e_i^j(0) = 0$ for all agents $i, j \in N$ where $i \neq j$. Define the initial state as $x(0) = (v(0), e(0), \mathcal{G}(0))$. It is straightforward to show that for any action trajectory $a(0), a(1), \cdots$,, the resulting state realization $x(t+1) \sim P_t(x(t), a(t))$ satisfies the following equalities for all times $t \geq 1$ and agents $k \in N$:

$$\sum_{i=1}^{n} e_i^k(t) = n \cdot v_k(t). \tag{8.5}$$

Agent Cost Functions: The introduced cost functions possess two distinct components and take on the form

$$J_i(x, a) = J_i^{\phi}(x, a) + \alpha \cdot J_i^{e}(x, a) \tag{8.6}$$

where $J_i^{\phi}(\cdot)$ represents the component centered on the objective function ϕ; $J_i^{e}(\cdot)$ represents the component centered on the state x; and α is a positive constant representing the trade-off between the two components.[2] We define each of these components as follows: for any state $x \in X$ and admissible action profile $a \in \prod_{i \in N} \mathcal{A}_i(x)$ we define

$$\begin{aligned} J_i^{\phi}(x, a) &= \textstyle\sum_{j \in N_i} \phi(\tilde{e}_j^1, \tilde{e}_j^2, \ldots, \tilde{e}_j^n) \\ J_i^{x}(x, a) &= \textstyle\sum_{j \in N_i} \sum_{k \in N} \left(\tilde{e}_j^k\right)^2 - n\left(\tilde{v}_i\right)^2 \end{aligned} \tag{8.7}$$

where $\tilde{v} = P^v(x, a)$ and $\tilde{e} = P^e(x, a)$. The local cost function in (8.7) is the main difference between the design in the chapter and the design in Chapter 6. The rest of the chapter shows that the new local cost function design allows us to deal with time-varying communication graphs.

8.3.2 Analytical properties of the designed game

Before analyzing the properties of the designed game, we introduce one core equilibrium concept that we will use in this chapter. Define a state set $\bar{X}(x^0, a^0)$ as the set of all possible ensuing states from the state

[2] We will show that as long as α is positive, all the results demonstrated in this chapter hold. However, choosing the right α is important for the learning algorithm implementation.

action pair $[x^0, a^0]$:

$$\bar{X}(x^0, a^0) \triangleq \left\{x = (v, e, \mathcal{G}) : v = P^v(x^0, a^0), e = P^e(x^0, a^0), \mathcal{G} \text{ is an undirected graph}\right\}.$$

Definition 8.1. *(Stationary Nash Equilibrium) A state action pair $[x^*, a^*]$ is a stationary Nash equilibrium if*

(D-1): for any $x \in \bar{X}(x^*, a^*)$:

$$a_i^* \in \mathcal{A}_i(x) \text{ and } a_i^* \in \operatorname{argmin}_{a_i \in \mathcal{A}_i(x)} J_i(x, a_i, a_{-i}^*).$$

(D-2): $x^* \in \bar{X}(x^*, a^*)$.

The first condition is similar to the Nash equilibrium concept and the second condition requires that the state components v and e are stationary. As the structure of the graph transition rule $P_t^{\mathcal{G}}$ can be very general, in the definition of $\bar{X}[x^*, a^*]$ we include all the undirected graphs as possible ensuing communication graphs. The two conditions imply that stationary Nash equilibria represent fixed points of the better reply process for state based games under any communication graph transition rule $P_t^{\mathcal{G}}$. That is, if a state action pair at time t, i.e., $[x(t), a(t)]$ is a stationary Nash equilibrium, then $a(\tau) = a(t)$ for all time $\tau \geq t$ if all players adhere to a better reply process. The following theorem demonstrates that *all* stationary Nash equilibria of our designed game are solutions to the optimization problem (8.1).

Theorem 8.1. *Model the optimization problem in (8.1) as a state based game G as depicted in Section 8.3.1 with any positive constant α. Then a state action pair $[x, a] := [(v, e, \mathcal{G}), (\hat{v}, \hat{e})]$ is a stationary Nash equilibrium in game G if and only if the following conditions are satisfied:*

(i) The value profile v is optimal for problem (8.1);

(ii) The estimation profile e satisfies that $e_i^k = v_k$, $\forall i, k \in N$;

(iii) The change in value satisfies $\hat{v}_i = 0$, $\forall i \in N$;

(iv) The change in estimation satisfies $\hat{e}_{i \to j}^k = 0$, $\forall i, j, k \in N$.

Proof. Firstly we prove that the two conditions in Definition 8.1 of a stationary Nash equilibrium, (i) $a_i \in \mathcal{A}_i(x')$ for any $x' \in \bar{X}(x,a)$ and (ii) $x \in \bar{X}(x,a)$, are equivalent to Condition (iii) and (iv) in Theorem 8.1, i.e. action a is a null action $\mathbf{0}$. For one direction, it is straightforward to show that if the action a is a null action $\mathbf{0}$, then $a_i \in \mathcal{A}_i(x')$ for any $x' \in \bar{X}(x,a)$ and $x \in \bar{X}(x,a)$. For the other direction, it is also can be shown that if the action $\hat{v} \neq 0$ then $x \notin \bar{X}(x,a)$ and if the action $\hat{e} \neq 0$, $a_i \notin \mathcal{A}_i(x')$ for some x'.

Now notice that $a = \mathbf{0}$ ensures that the ensuing value profile $P^v(x,a) = v$ and the ensuing estimate profile $P^e(x,a) = e$, which means that

$$\bar{X}(x,\mathbf{0}) = \{(v,e,\mathcal{G}') : \mathcal{G}' \text{ is an undirected graph}\}.$$

Therefore the rest of the proof only need to show that the condition in Definition 8.1,

$$0 \in \operatorname{argmin}_{a_i' \in \mathcal{A}_i(x')} J_i(x', a_i', 0)$$

for any $x' \in \bar{X}(x,\mathbf{0})$, is equivalent to the fact that (v,e) satisfies Condition (i) and (ii) in Theorem 8.1. Given a state $x' \in \bar{X}(x,\mathbf{0})$, the condition $0 \in \operatorname{argmin}_{a_i' \in \mathcal{A}_i(x')} J_i(x', a_i', 0)$ is equivalent to:

$$\begin{aligned}
&\left[\frac{\partial J_i(x,a_i',a_{-i}=0)}{\partial \hat{v}_i}\Big|_{a_i'=0}\right] \cdot (\hat{v}_i' - 0) \geq 0, \forall i \in N, \hat{v}_i' \in \mathcal{A}_i^v(x) \\
&\frac{\partial J_i(x,a_i',a_{-i}=0)}{\partial \hat{e}_i^k}\Big|_{a_i'=0} = 0, \forall i,k \in N.
\end{aligned}$$

The two equations are equivalent to

$$\left[\phi_i|_{e_i} + 2\alpha \cdot n e_i^i - 2\alpha \cdot n v_i\right] \cdot (v_i' - v_i) \geq 0, \forall i \in N, \hat{v}_i' \in \mathcal{A}_i^v(x). \tag{8.8}$$

$$\phi_k|_{e_i} - \phi_k|_{e_j} - \left(2\alpha(e_j^k - e_i^k)\right) = 0, \forall i,k \in N,\ j \in N_i. \tag{8.9}$$

Therefore, the rest of the proof only needs to show that two Equations (8.9, 8.8) are equivalent to Condition (i) and (ii) in this theorem.

($\Leftarrow$) If (v, e) satisfies conditions (i) and (ii), we have:

$$[\phi_i|_v] \cdot (v_i' - v_i) \geq 0, \quad \forall i \in N,\ v_i' \in \mathcal{V}_i, \tag{8.10}$$

$$e_i^k = v_k, \qquad \forall i, k \in N. \tag{8.11}$$

Equation (8.11) tells us that equation (8.9) is satisfied. Subsitituting equation (8.11) into equation (8.10), we know that equation (8.8) is satisfied. Therefore, both Equation (8.8) and Equation (8.9) are satisfied.

($\Rightarrow$) Now we prove the other direction. Suppose (v, e) satisfy Equation (8.9, 8.8). Focus on equation (8.9) first. Applying Lemma 6.11, equation (8.9) coupled with the fact that ϕ is a convex function implies that for any pair $i \in N, j \in N_i$,$\tilde{e}_i = \tilde{e}_j$.

Given a connected and undirected graph $\mathcal{G}$, we know that $e_i = e_j$ for all $i, j \in N$. Applying equality (8.5), we have $e_i^k = v_k, \forall i, k \in N$, i.e. (v, e) satisfies Condition (i) listed in the theorem. Substituting this equality into equation (8.8), we have

$$[\phi_i|_v] \cdot (\tilde{v}_i' - v_i) \geq 0, \forall i \in N, \hat{v}_i' \in \mathcal{A}_i^v(x) \tag{8.12}$$

Since ϕ is a convex function, this tells us that v is an optimal solution for problem (8.1). □

□

The above theorem demonstrates that the resulting equilibria of our state based game coincide with the optimal solutions to the optimization problem in (8.1). Moreover, from this theorem, it is straightforward to derive the following corollary:

Corollary 8.2. *If a station action pair* $[x^*, a^*] \triangleq [(v^*, e^*, \mathcal{G}^*), a^*]$ *is a stationary Nash equilibrium, then any station action pair* $[(v^*, e^*, \mathcal{G}), a^*]$ *is a stationary Nash equilibrium for any undirected graph* $\mathcal{G}$.

The following theorem demonstrates that the designed game possesses a property similar with potential games that facilitates the design of learning rules to reach such a stationary Nash equilibrium.

Theorem 8.3. *Model the optimization problem in (8.1) as a state based game G as depicted in Section 8.3.1*

with any positive constant α. The following function $\Phi : X \times \mathcal{A} \to \mathbb{R}$

$$\Phi(x,a) = \Phi^{\phi}(x,a) + \alpha \cdot \Phi^{x}(x,a) \tag{8.13}$$

where

$$\Phi^{\phi}(x,a) = \sum_{i\in N} \phi(\tilde{e}_i^1, \tilde{e}_i^2, ..., \tilde{e}_i^n) \tag{8.14}$$

$$\Phi^{x}(x,a) = \sum_{i\in N}\sum_{k\in N} \left(\tilde{e}_i^k\right)^2 - n \cdot \sum_{i\in N} \left(\tilde{v}_i\right)^2 \tag{8.15}$$

$$\tilde{v} = P^v(x,a) \text{ and } \tilde{e} = P^e(x,a),$$

satisfies the following two properties:

1. *For every state action pair $[x,a]$ any player $i \in N$ any action $a_i' \in \mathcal{A}_i(x)$,*

$$J_i(x, a_i', a_{-i}) - J_i(x,a) = \Phi(x, a_i', a_{-i}) - \Phi(x,a).$$

2. *For every state action pair $[x,a]$ and any $\tilde{x} \in \bar{X}(x,a)$, we have $\Phi(x,a) = \Phi(\tilde{x}, \mathbf{0})$ where $\mathbf{0}$ is a null action given as $\hat{v}_i = 0, \hat{e}_{i\to j}^k = 0$ for any $i,k \in N$.*

Moreover, $\Phi(x,a)$ is a convex function over $a = (\hat{v}, \hat{e})$.

Proof. It is straightforward to verify that $\Phi(x,a)$ defined in Equation (8.13) satisfies Properties 1) and 2) in Theorem 8.3. So we only need to prove that $\Phi(x,a)$ is a convex function over $a = (\hat{v}, \hat{e})$. Substituting equality (8.5) into $\Phi^x(x,a)$ which is defined in equation (8.13),

$$\begin{aligned}
\Phi^x(x,a) &= \sum_{k\in N}\sum_{i\in N} \left(\tilde{e}_i^k\right)^2 - n \sum_{k\in N} \left(\frac{\sum_{i\in N} \tilde{e}_i^k}{n} \right)^2 \\
&= \sum_{k\in N} \left(\sum_{i\in N} \left(\tilde{e}_i^k\right)^2 - \frac{1}{n} \left(\sum_{i\in N} \tilde{e}_i^k \right)^2 \right) \\
&= \frac{1}{n} \sum_{k\in N} \left(\sum_{i,j\in N, j<i} \left(\tilde{e}_i^k - \tilde{e}_j^k\right)^2 \right) \quad .
\end{aligned}$$

Therefore $\Phi^x(x,a)$ is a convex function of $\tilde{e}$. Since Φ^ϕ is also a convex function of $\tilde{e}$, $\Phi(x,a)$ is a convex function of $\tilde{e}$ as well. Thus $\Phi(x,\hat{a})$ is a convex function over $a=(\hat{v},\hat{e})$ for $\tilde{e}$ is a linear function of $(\hat{v},\hat{e})$. $\square$

Property 1) and 2) of this theorem demonstrate that the function $\Phi(x,a)$ satisfies the properties of a state based potential function defined in Chapter 6. Thus we call this game a state based potential game and $\Phi(x,a)$ a state based potential function.

Notice that $\Phi(x,\mathbf{0})$ is independent of the communication graph $\mathcal{G}$. Therefore, even though the communication graph $\mathcal{G}$ is time varying, Theorem 8.3 establishes that our state based game design possesses an underlying structure that facilitates the design of distributed algorithms to reach stationary Nash equilibria. In the next section, we provide a distributed learning algorithm to reach those stationary Nash equilibria that were characterized in Theorem 8.1.

8.4 Gradient play

Since the state based potential function $\Phi(x,a)$ is a convex function over $a=(\hat{v},\hat{e})$, we can apply gradient play algorithm in Chapter 6 to develop a distributed learning algorithm for the state based game depicted in section 8.3. In this section, we assume that $\mathcal{V}_i$ is a closed convex set for all $i\in N$. The gradient play algorithm is given as follows:

1. Each agent i initially randomly chooses a value $v_i(0)$ and set $e_i^i=nv_i(0)$ and $e_i^k(0)=0$ for all $k\neq i$. Set t=0;

2. At each time $t\geq 0$ each agent i selects an action $a_i(t)\triangleq(\hat{v}_i(t),\hat{e}_i(t))$ given the state $x(t)=(v(t),e(t),\mathcal{G}(t))$ according to:

$$\hat{v}_i(t)=\left[-\epsilon\cdot\left.\frac{\partial J_i\left(x(t),a\right)}{\partial\hat{v}_i}\right|_{a=0}\right]^+=\left[-\epsilon(n\left.\phi_i\right|_{e_i(t)}+2n\alpha(e_i^i(t)-v_i(t))\right]^+ \tag{8.16}$$

$$\hat{e}_{i\to j}^k(t)=-\epsilon\left.\frac{\partial J_i\left(x(t),a\right)}{\partial\hat{e}_{i\to j}^k}\right|_{a=0}=\epsilon\left(\left.\phi_k\right|_{e_i(t)}-\left.\phi_k\right|_{e_j(t)}+2\alpha\left(e_i^k(t)-e_j^k(t)\right)\right) \tag{8.17}$$

where $[\cdot]^+$ represents the projection onto the closed convex set $\mathcal{A}_i^{\hat{v}}(x)$; and ϵ is the stepsize which is a positive constant. Notice that each agent i can select its own action using local information since $J_i(\cdot)$

only depends on local information.

3. Each agent i updates the local state $(v_i(t+1), e_i(t+1))$ according to Equation (8.2) and (8.2) using its own local information. The communication graph $\mathcal{G}(t)$ is realized according to $P_t^{\mathcal{G}}$.

4. Increase t by 1 and return to step 2.

The following theorem establishes the convergences of the gradient play.

Theorem 8.4. *Suppose there exists an integer $k > 0$ such that the undirected communication graph $G(t)$ is connected for at least one time step $t \in [nk, nk+k-1]$ for all $n \geq 0$. If the step-size is sufficiently small, and the sequence $(v(1), e(1)), (v(2), x(2)), \cdots$ produced by the gradient play algorithm is contained in a compact subset of $\mathbb{R}^{2n}$, then $[v(t), e(t), a(t)]$ in the gradient play algorithm asymptotically converges to $[(v^*, e^*), \mathbf{0}]$ where $[(v^*, e^*, \mathcal{G}), \mathbf{0}]$ is a stationary Nash equilibrium with any graph $\mathcal{G}$.*

Proof. Notice that $\Phi(x(t), \mathbf{0})$ is independent of $\mathcal{G}(t)$; therefore we can write $\Phi(x(t), \mathbf{0})$ as $\Phi(v(t), e(t), \mathbf{0})$. Then we can show that $\Phi(x(t), \mathbf{0})$ is monotonically decreasing along the gradient play algorithm. The proof of the convergence follows exactly the same as the proof for Theorem 4 in Chapter 6. We omit the details here. □

In combination with Theorem 8.1, Theorem 8.4 demonstrates that the gradient play algorithm provides a distributed learning algorithm to solve the optimization problem in (8.1).

Remark 8.1. *The theorem requires a strong condition on the undirected communication graph $\mathcal{G}(t)$, i.e. if it is connected frequently enough, the results can be extended to more general cases. For example, it can be shown that if there exists a finite $k \geq 0$ such that $\cup_{t=\tau k}^{\tau k+k-1} \mathcal{G}(t) \triangleq (N, \cup_{t=\tau k}^{\tau k+k-1} \mathcal{E}(t))$ is connected for all $\tau \geq 0$, then the gradient play algorithm will converge to the stationary Nash equilibrium. As a non-rigorous statement, as long as the union of $\mathcal{G}(t)$ over a finite time horizon is connected frequently enough, the gradient play algorithm will converge to a stationary Nash equilibrium.*

8.5 Illustrations

We will use a simple abstract example to illustrate the problem and the method. Consider the following optimization problem:

$$\begin{aligned} \min_{v_1,\ldots,v_5} \quad & v^T P v + q^T v \\ \text{s.t.} \quad & v_i \in [0, i] \subset \mathbb{R} \end{aligned}$$

where $q^T = -[9\ 9\ 9\ 9\ 9]$ and

$$P = \begin{bmatrix} 6 & 1 & 1 & 1 & -1 \\ 1 & 7 & 1 & -1 & 2 \\ 1 & 1 & 8 & 2 & -2 \\ 1 & -1 & 2 & 9 & 3 \\ -1 & 2 & -2 & 3 & 9 \end{bmatrix}$$

The goal is to establish a local control law for each agent i that converges to the optimal value v_i^*. One possibility for a distributed algorithm is to utilize a gradient descent algorithm where each agent adjusts its own value according to $\frac{\partial \phi}{\partial v_i} = 2\sum_{j=1}^{5} P(i,j)v(j) + q(i)v(i)$. As P is a full-itemized matrix, implementing this algorithm requires each agent to have complete information regarding the decision of all other agents.

Using the method developed in this chapter, we localize the information available to each agent by allowing them to have estimates of other agents' decision value. We simulate the gradient play algorithm with a time varying communication graph. In the simulation, at each time $t \geq 0$, each communication link (i, j) is drawn randomly with a certain probability. Figure 8.1 illustrates the results of the gradient play algorithm. The top figure in Figure 8.1 shows the evolution of the cost $\phi(v)$ using the true gradient decent algorithm (red) and our proposed gradient play algorithm (blue). The figure shows that the convergence rate is comparable to the centralized gradient descent algorithm. Also we can notice that $\phi(v(t))$ for our distributed algorithm is not monotonically decreasing. This is reasonable since the gradient play only guarantees the potential function $\Phi(x(t), \mathbf{0})$ monotonically decreasing. This is confirmed in the middle figure of Figure 8.1 which shows the evolution of the state based potential function $\Phi(x(t), \mathbf{0})$. The bottom figure shows the evolution of agent i's estimation error as to agent 1's true value, i.e., $e_i^1 - v_1$. Note that the error converges to 0 illustrating that

the agent's estimate converges to the right values as proved in Theorem 8.1 and 8.4.

8.6 Conclusion

We utilize the framework of state based potential games to develop a systematic methodology for distributed optimization with a time-varying communication graph. This work, along with previous work in Chapters 6-7, demonstrates that the framework of state based potential games leads to a value hierarchical decomposition that can be an extremely powerful for the design and control of multiagent systems.

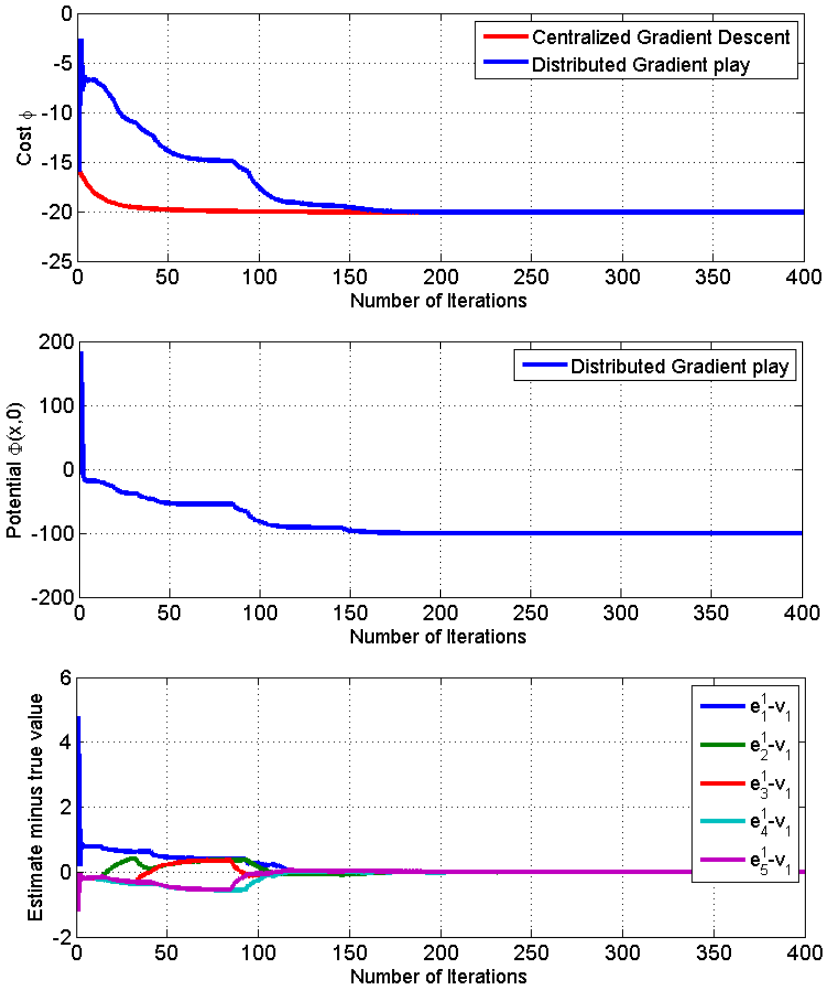


Figure 8.1. Simulation results: The top figure shows the evolution of the system cost using the true gradient descent algorithm (red) and our proposed algorithm (black). The middle figure shows the evolution of the state based potential function $\Phi(x(t), \mathbf{0})$. The bottom figure shows the evolution of agent i's estimation error as to agent 1's true value, i.e., $e_i^1 - v_1$. Note that the error converges to 0 illustrating that the agent's estimate converges to the right values as proved in Theorem 8.1 and 8.4.

Bibliography

[1] C. W. Gellings and J. H. Chamberlin. *Demand-Side Management: Concepts and Methods*. The Fairmont Press, 1988.

[2] M. H. Albadi and E. F. El-Saadany. Demand response in electricity markets: An overview. In *Proceedings of the IEEE Power Engineering Society General Meeting*, June 2007.

[3] A. I. Cohen and C. C. Wang. An optimization method for load management scheduling. *IEEE Transactions on Power Systems*, 3(2):612–618, May 1988.

[4] Y. Y. Hsu and C. C. Su. Dispatch of direct load control using dynamic programming. *IEEE Transactions on Power Systems*, 6(3):1056–1061, August 1991.

[5] D. C. Wei and N. Chen. Air conditioner direct load control by multi-pass dynamic programming. *IEEE Transactions on Power Systems*, 10(1):307–313, February 1995.

[6] J. Chen, F. N. Lee, A. M. Breipohl, and R. Adapa. Scheduling direct load control to minimize system operation cost. *IEEE Transactions on Power Systems*, 10(4):1994–2001, November 1995.

[7] K. H. Ng and G. B. Sheble. Direct load control – a profit-based load management using linear programming. *IEEE Transactions on Power Systems*, 13(2):688–695, May 1998.

[8] W.-C. Chu, B.-K. Chen, and C.-K. Fu. Scheduling of direct load control to minimize load reduction for a utility suffering from generation shortage. *IEEE Transactions on Power Systems*, 8(4):1525–1530, November 1993.

[9] B. Ramanathan and V. Vittal. A framework for evaluation of advanced direct load control with minimum disruption. *Power Systems, IEEE Transactions on*, 23(4):1681–1688, 2008.

[10] M. D. Ilic, L. Xie, and J.-Y. Joo. Efficient coordination of wind power and price-responsive demand part I: Theoretical foundations; part II: Case studies. *IEEE Transactions on Power Systems*, 99, 2011.

[11] Y. V. Makarov, C. Loutan, J. Ma, and P. de Mello. Operational impacts of wind generation on California power systems. *IEEE Transactions on Power Systems*, 24(2):1039–1050, May 2009.

[12] M. C. Caramanis and J. M. Foster. Coupling of day ahead and real-time power markets for energy and reserves incorporating local distribution network costs and congestion. In *Proceedings of the 48th Annual Allerton Conference*, September – October 2010.

[13] D. Kirschen. Demand-side view of electricity market. *IEEE Transactions on Power Systems*, 18(2):520–527, May 2003.

[14] J. C. Smith, M. R. Milligan, E. A. DeMeo, and B. Parsons. Utility wind integration and operating impact: State of the art. *IEEE Transactions on Power Systems*, 22(3):900–908, August 2007.

[15] N. Ruiz, I. Cobelo, and J. Oyarzabal. A direct load control model for virtual power plant management. *IEEE Transactions on Power Systems*, 24(2):959–966, May 2009.

[16] P. P. Varaiya, F. F. Wu, and J. W. Bialek. Smart operation of smart grid: Risk-limiting dispatch. *Proceedings of the IEEE*, 99(1):40 –57, January 2011.

[17] Department of Energy. Benefits of demand response in electricity markets and recommendations for achieving them. Technical report, February 2006.

[18] S. Borenstein. Time-varying retail electricity prices: Theory and practice. In Griffin and Puller, editors, *Electricity Deregulation: Choices and Challenges*. University of Chicago Press, 2005.

[19] C. Triki and A. Violi. Dynamic pricing of electricity in retail markets. *Quarterly Journal of Operations Research*, 7(1):21–36, March 2009.

[20] M. D. Ilic. Dynamic monitoring and decision systems for enabling sustainable energy services. *Proceedings of the IEEE*, 99(1):58–79, January 2011.

[21] P. Xu, P. Haves, M.A. Piette, and L. Zagreus. Demand shifting with thermal mass in large commercial buildings: Field tests, simulation and audits. 2005.

[22] J.E. Braun. Load control using building thermal mass. *Journal of solar energy engineering*, 125:292, 2003.

[23] K. Clement-Nyns, E. Haesen, and J. Driesen. The impact of charging plug-in hybrid electric vehicles on a residential distribution grid. *Power Systems, IEEE Transactions on*, 25(1):371–380, 2010.

[24] M. Pedrasa, T. Spooner, and I. MacGill. Coordinated scheduling of residential distributed energy resources to optimize smart home energy services. *IEEE Transactions on Smart Grid*, 1(2):134–143, 2010.

[25] A. Mohsenian-Rad and A. Leon-Garcia. Optimal residential load control with price prediction in real-time electricity pricing environments. *IEEE Transactions on Smart Grid*, 1(2):120–133, 2010.

[26] A. Mohsenian-Rad, V. 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 Transactions on Smart Grid*, To be Submitted, 2010.

[27] C.L. Su and D. Kirschen. Quantifying the effect of demand response on electricity markets. *Power Systems, IEEE Transactions on*, 24(3):1199–1207, 2009.

[28] M. Fahrioglu and F.L. Alvarado. Using utility information to calibrate customer demand management behavior models. *Power Systems, IEEE Transactions on*, 16(2):317–322, 2002.

[29] J. Zhang, J.D. Fuller, and S. Elhedhli. A stochastic programming model for a day-ahead electricity market with real-time reserve shortage pricing. *Power Systems, IEEE Transactions on*, 25(2):703–713, 2010.

[30] N. Amjady and H. Nasiri-Rad. Nonconvex economic dispatch with AC constraints by a new real coded genetic algorithm. *Power Systems, IEEE Transactions on*, 24(3):1489–1502, 2009.

[31] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*.

[32] CE Lin and GL Viviani. Hierarchical economic dispatch for piecewise quadratic cost functions. *Power Apparatus and Systems, IEEE Transactions on*, (6):1170–1175, 2007.

[33] Masoud Farivar and Steven Low. Branch flow model: Relaxations and convexification. *arXiv:1204.4865v2*, 2012.

[34] L. Gan, N. Li, U. Topcu, and S. Low. Branch flow model for radial networks: convex relaxation. In *Proceedings of the 51st IEEE Conference on Decision and Control*, 2012.

[35] M. E. Baran and F. F Wu. Optimal Capacitor Placement on radial distribution systems. *IEEE Trans. Power Delivery*, 4(1):725–734, 1989.

[36] M. E Baran and F. F Wu. Optimal Sizing of Capacitors Placed on A Radial Distribution System. *IEEE Trans. Power Delivery*, 4(1):735–743, 1989.

[37] J. Carpentier. Contribution to the economic dispatch problem. *Bulletin de la Societe Francoise des Electriciens*, 3(8):431–447, 1962. In French.

[38] J. A. Momoh. *Electric Power System Applications of Optimization*. Power Engineering. Markel Dekker Inc.: New York, USA, 2001.

[39] M. Huneault and F. D. Galiana. A survey of the optimal power flow literature. *IEEE Trans. on Power Systems*, 6(2):762–770, 1991.

[40] J. A. Momoh, M. E. El-Hawary, and R. Adapa. A review of selected optimal power flow literature to 1993. Part I: Nonlinear and quadratic programming approaches. *IEEE Trans. on Power Systems*, 14(1):96–104, 1999.

[41] J. A. Momoh, M. E. El-Hawary, and R. Adapa. A review of selected optimal power flow literature to 1993. Part II: Newton, linear programming and interior point methods. *IEEE Trans. on Power Systems*, 14(1):105 – 111, 1999.

[42] K. S. Pandya and S. K. Joshi. A survey of optimal power flow methods. *J. of Theoretical and Applied Information Technology*, 4(5):450–458, 2008.

[43] X. Bai, H. Wei, K. Fujisawa, and Y. Wang. Semidefinite programming for optimal power flow problems. *Int'l J. of Electrical Power & Energy Systems*, 30(6-7):383–392, 2008.

[44] J. Lavaei and S. Low. Zero duality gap in optimal power flow problem. *IEEE Trans. on Power Systems*, 27(1):92–107, 2012.

[45] B. Lesieutre, D. Molzahn, A. Borden, and C. L. DeMarco. Examining the limits of the application of semidefinite programming to power flow problems. In *Proc. Allerton Conference*, 2011.

[46] S. Bose, D. Gayme, S. H. Low, and K. M. Chandy. Optimal power flow over tree networks. In *Proc. Allerton Conf. on Comm., Ctrl. and Computing*, October 2011.

[47] B. Zhang and D. Tse. Geometry of feasible injection region of power networks. *Arxiv preprint arXiv:1107.1467*, 2011.

[48] S. Sojoudi and J. Lavaei. Network topologies guaranteeing zero duality gap for optimal power flow problem. 2012.

[49] Masoud Farivar, Christopher R. Clarke, Steven H. Low, and K. Mani Chandy. Inverter var control for distribution systems with renewables. In *Proceedings of IEEE SmartGridComm Conference*, October 2011.

[50] Joshua Adam Taylor. *Conic Optimization of Electric Power Systems*. PhD thesis, MIT, June 2011.

[51] R. Cespedes. New method for the analysis of distribution networks. *IEEE Trans. Power Del.*, 5(1):391–396, January 1990.

[52] A. G. Expósito and E. R. Ramos. Reliable load flow technique for radial distribution networks. *IEEE Trans. Power Syst.*, 14(13):1063–1069, August 1999.

[53] R.A. Jabr. Radial Distribution Load Flow Using Conic Programming. *IEEE Trans. on Power Systems*, 21(3):1458–1459, 2006.

[54] K. Turitsyn, P. Sulc, S. Backhaus, and M. Chertkov. Distributed control of reactive power flow in a radial distribution circuit with high photovoltaic penetration. In *Proc. of IEEE PES General Meeting*, pages 1 –6, July 2010.

[55] Masoud Farivar, Russell Neal, Christopher Clarke, and Steven Low. Optimal inverter var control in distribution systems with high pv penetration. In *IEEE Power and Energy Society General Meeting*, San Diego, CA, July 2012.

[56] M. Grant, S. Boyd, and Y. Ye. Cvx user guide. Technical report, Technical Report, Available at: `http://cvxr.com/cvx/cvx_usrguide.pdf`, 2009.

[57] G. Bautista, M.F. Anjos, and A. Vannelli. Formulation of oligopolistic competition in ac power networks: An nlp approach. *Power Systems, IEEE Transactions on*, 22(1):105–115, 2007.

[58] J. Lavaei and S. Sojoudi. Competitive equilibria in electricity markets with nonlinearities. In *American Control Conference*, 2012.

[59] E. Litvinov. Design and operation of the locational marginal prices-based electricity markets. *Generation, Transmission & Distribution, IET*, 4(2):315–323, 2010.

[60] S.S. Oren, P.T. Spiller, P. Varaiya, and F. Wu. Nodal prices and transmission rights: A critical appraisal. *The Electricity Journal*, 8(3):24–35, 1995.

[61] G. Chen and M. Teboulle. A proximal-based decomposition method for convex minimization problems. *Mathematical Programming*, 64(1):81–101, 1994.

[62] A. Bergen and V. Vittal. *Power Systems Analysis*. Prentice Hall, 2 edition, 1999.

[63] P Kumar, Dwarka P Kothari, et al. Recent philosophies of automatic generation control strategies in power systems. *Power Systems, IEEE Transactions on*, 20(1):346–357, 2005.

[64] C. Zhao, U. Topcu, N. Li, and S. Low. Power system dynamics as primal-dual algorithm for optimal load control. In *arXiv:1305.0585*, 2012.

[65] J Nanda and BL Kaul. Automatic generation control of an interconnected power system. In *Proceedings of the Institution of Electrical Engineers*, volume 125, pages 385–390, 1978.

[66] O. I. Elgerd and C. Fosha. The megawatt frequency control problem: A new approach via optimal control theory. *IEEE Transactions on Power Apparatus and Systems*, 89(4):563–577, 1970.

[67] M Aldeen and H Trinh. Load-frequency control of interconnected power systems via constrained feedback control schemes. *Computers & electrical engineering*, 20(1):71–88, 1994.

[68] C-T Pan and C-M Liaw. An adaptive controller for power system load-frequency control. *Power Systems, IEEE Transactions on*, 4(1):122–128, 1989.

[69] Milan S Ćalović. Automatic generation control: Decentralized area-wise optimal solution. *Electric power systems research*, 7(2):115–139, 1984.

[70] M Zribi, M Al-Rashed, and M Alrifai. Adaptive decentralized load frequency control of multi-area power systems. *International Journal of Electrical Power & Energy Systems*, 27(8):575–583, 2005.

[71] NN Bengiamin and WC Chan. Multilevel load-frequency control of interconnected power systems. *Electrical Engineers, Proceedings of the Institution of*, 125(6):521–526, 1978.

[72] Umit Ozguner. Near-optimal control of composite systems: The multi time-scale approach. *Automatic Control, IEEE Transactions on*, 24(4):652–655, 1979.

[73] D Brian Eidson and Marija D Ilic. Advanced generation control with economic dispatch. In *Decision and Control, Proceedings of the 34th IEEE Conference on*, volume 4, pages 3450–3458, 1995.

[74] MD Ilic and Chien-Ning Yu. Minimal system regulation and its value in a changing industry. In *Control Applications, Proceedings of the 1996 IEEE International Conference on*, pages 442–449, 1996.

[75] C. Yu. *Real power and frequency control of large electric power systems under open access*. PhD thesis, MIT, 1996.

[76] D.P. Bertsekas. *Nonlinear programming, 2nd edition*. Athena Scientific Belmont, MA, 2008.

[77] H. K. Khalil, editor. *Nonlinear Systems, 3rd Edition*. Prentice Hall, 2002.

[78] J. Tsitsiklis and M. Athans. Convergence and asymptotic agreement in distributed decision problems. *IEEE Transactions on Automatic Control*, 29(1):42–50, 1984.

[79] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[80] A. Tang, J. Wang, S. H. Low, and M. Chiang. Equilibrium of heterogeneous congestion control: Existence and uniqueness. *IEEE/ACM Transactions on Networking*, 15(4):824–837, 2007.

[81] V. Mhatre, K. Papagiannaki, and F. Baccelli. Interference mitigation through power control in high density 802.11. In *Proceedings of IEEE INFOCOM*, 2007.

[82] E. Campos-Nanez, A. Garcia, and C. Li. A game-theoretic approach to efficient power management in sensor networks. *Operations Research*, 56(3):552, 2008.

[83] G. Scutari, D. P. Palomar, J. Pang, and F. Facchinei. Flexible design of cognitive radio wireless systems. *IEEE Signal Processing Magazine*, 26(5):107–123, 2009.

[84] L. Chen, S. H. Low, and J. C. Doyle. Random access game and medium access control design. *IEEE/ACM Transactions on Networking*, (4):1303–1316, 2010.

[85] J. R. Marden and M. Effros. The price of selfiness in network coding. *IEEE Transactions on Information Theory*, 58(4):2349–2361, 2012.

[86] V. Reddy, S. Shakkottai, A. Sprintson, and N. Gautam. Multipath wireless network coding: a population game perspective. In *Proceedings of IEEE INFOCOM*, 2010.

[87] R. Gopalakrishnan, J. R. Marden, and A. Wierman. An architectural view of game theoretic control. *ACM SIGMETRICS Performance Evaluation Review*, 38(3):31–36, 2011.

[88] D. Monderer and L.S. Shapley. Potential games. *Games and Economic Behavior*, 14:124–143, 1996.

[89] D. Fudenberg and D. K. Levine, editors. *The Theory of Learning in Games*. MIT Press, Cambridge, MA, 1998.

[90] H. P. Young. *Strategic Learning and Its Limits*. Oxford University Press, Oxford, UK, 2004.

[91] J.R. Marden, G. Arslan, and J.S. Shamma. Joint strategy fictitious play with inertia for potential games. *IEEE Transactions on Automatic Control*, 54(2):208–220, 2009.

[92] J. R. Marden, H. P. Young, G. Arslan, and J. S. Shamma. Payoff based dynamics for multi-player weakly acyclic games. *SIAM Journal on Control and Optimization*, 48(1):373–396, 2009.

[93] J. R. Marden and A. Wierman. Overcoming limitations of game-theoretic distributed control. In *48th IEEE Conference on Decision and Control*, 2009.

[94] LS Shapley. A value for n-person games. *Contributions to the theory of games*, 2:307–317, 1953.

[95] E. Anshelevich, A. Dasgupta, J. Kleinberg, E. Tardos, T. Wexler, and T. Roughgarden. The price of stability for network design with fair cost allocation. *SIAM Journal on Computing*, 38(4):1602–1623, 2008.

[96] J. R. Marden. State based potential games. *Automatica*, 48(12):3075–3088, 2012.

[97] S. D. Flam. Equilibrium, evolutionary stability and gradient dynamics. *International Game Theory Review*, 4(4):357–370, 2002.

[98] J. Shamma and G. Arslan. Dynamic fictitious play, dynamic gradient play, and distributed convergence to nash equilibria. *IEEE Transactions on Automatic Control*, 50(3):312–327, 2005.

[99] M. V. Solodov. Incremental gradient algorithms with stepsizes bounded away from zero. *Computational Optimization and Applications*, 11(1):23–35, 1998.

[100] D. Blatt, A.O. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2008.

[101] J. Tsitsiklis, D. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31(9):803–812, 1986.

[102] A. Nedic, A. Olshevsky, A. Ozdaglar, and J.N. Tsitsiklis. On distributed averaging algorithms and quantization effects. *IEEE Transactions on Automatic Control*, 54(11):2506–2517, 2009.

[103] I. Lobel and A. Ozdaglar. Distributed subgradient methods for convex optimization over random networks. *IEEE Transactions on Automatic Control*, (99):1–1, 2010.

[104] M. Zhu and S. Martinez. On distributed convex optimization under inequality and equality constraints via primal-dual subgradient methods. *Arxiv preprint arXiv:1001.2612*, 2010.

[105] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[106] Y. Hatano and M. Mesbahi. Agreement over random networks. *IEEE Transactions on Automatic Control*, 50(11):1867–1872, 2005.

[107] IP Androulakis and GV Reklaitis. Approaches to asynchronous decentralized decision making. *Computers and Chemical Engineering*, 23(3):339–354, 1999.

[108] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundation and Trends in Machine Learning*, 3(1):1–122, 2010.

[109] C. Ding, B. Song, A. Morye, J.A. Farrell, and A.K. Roy-Chowdhury. Collaborative sensing in a distributed ptz camera network. *IEEE Transactions on Image Processing*, 21(7):3282–3295, 2012.

[110] M.S. Stankovic, K.H. Johansson, and D.M. Stipanovic. Distributed seeking of nash equilibria with applications to mobile sensor networks. *IEEE Transactions on Automatic Control*, 57(4):904–919, 2012.

[111] L. S. Shapley. Stochastic games. *Proceedings of the National Academy of Science of the United States of America*, 39(10):1095–1100, 1953.

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

[113] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty. *Nonlinear programming: theory and algorithms, 3rd edition*. John Wiley & Sons, Inc., NJ, 2006.

[114] S.H. Low and D.E. Lapsley. Optimization flow control, I. Basic algorithm and convergence. *IEEE/ACM Transactions on Networking*, 7(6):861–874, 1999.

[115] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[116] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Gossip algorithms: Design, analysis and applications. In *Proceedings of IEEE INFOCOM*, 2005.

[117] L. Pavel. An extension of duality to a game-theoretic framework. *Automatica*, 43:226–237, 2007.

[118] A. Rantzer. Dynamic dual decomposition for distributed control. In *Proceedings of American Control Conference*, 2009.

[119] I. Lobel, A. Ozdaglar, and D. Feijer. Distributed multi-agent optimization with state-dependent communication. *Mathematical Programming*, 129(2):255–284, 2011.

[120] M. Zhu and S. Martinez. An approximate dual subgradient algorithm for distributed cooperative nonconvex constrained optimization. *IEEE Transactions on Automatic Control*, To appear, 2012.