11-13-2003

Learning Average Reward Irreducible Stochastic Games: Analysis and Applications

Jun, Li
University of South Florida

LEARNING AVERAGE REWARD IRREDUCIBLE STOCHASTIC GAMES: ANALYSIS AND APPLICATIONS

by

JUN LI

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Industrial Engineering
Department of Industrial and Management Systems Engineering
College of Engineering
University of South Florida

Major Professor: Tapas K. Das, Ph.D.
William A. Miller, Ph.D.
Ali Yalcin, Ph.D.
Sudeep Sarkar, Ph.D.
Kandethody M. Ramachandran, Ph.D.

Date of Approval:
November 13th, 2003

Keywords: Markov decision processes, game theory, reinforcement learning,
stochastic approximation, power market

ACKNOWLEDGMENTS

There are many people who I am deeply indebted to. First and foremost is my advisor, Dr. Tapas Das. He has been instrumental in directing me from my initial explorations on this topic to the final dissertation revisions. He is a positive, open-minded and hard- working professor. I hope that I will be as professional as him in my future career. I also want to thank Dr. Das's family for these years' concern about me.

Dr. Miller and Dr. Weng were the only people I knew before I came to USA. They have helped me adjust to this strange Country and played significant roles of mentor in guiding my studying and living in USF.

I thank my other committee members, Dr. Sarkar, Dr. Yalcin and Dr. Ramachandran, for their valuable questions, insights and guidance. They have served to greatly improve the content and presentations of this work. I also need to thank Dr. Zayas-Castro, Dr.Okogbaa, Dr. Khator, Dr. Fink and Dr. Ismail, I learned a lots from their courses and talks. I also thank professor Goodings for sharing teaching experience with me.

I must thank people in the department of Industrial Management and system engineering. Chris and Gloria are very nice, they treat people equally. Marsha knows everything, even after she left IMSE, I still reach her for help.

The research in this dissertation was primarily inspired by the work of Dr. Gosavi, Kiran and Rajkumar. I thank them for the road they paved and hope my work will inspire other people. I own my gratitude to many researchers and professors who do not know me but very kindly replied my emails, cleared my doubts and gave

informative comments. I also thank Sanket for helping me understand how power markets work.

When you are far away from your family, friends are the ones you can depend on. I am lucky to have so many good Chinese and non Chinese friends in Tampa, who have been helping and teaching me from cooking to computer problems. They are Zhao, Sumit, William, Aidee, John, Su Yu, Wang Cuiwei, Luo Tong, Wang Yang, Athy and her family, Xiaohua Hu's family, Kiran, Vivek, Dengzi, Michael, Raj,Rajesh, Wu Ling, Xuequan Hu, Jadeep and his wife etc. I aslo thank the whole Chinese community in Tampa, which is very supportive and helpful for Chinese students.

Most of all, I would like to thank my family. My parents and sister are far away in China, but they keep providing me mental support. It is their confidence in me that keep me going and growing. I have a wonderful husband, who is also my friend, encourager, task-master, advocate and helper. Without his support and encouragement, this dissertation would certainy have gone unwritten. He deserves more credit than I can put into words.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LEARNING AVERAGE REWARD IRREDUCIBLE STOCHASTIC GAMES: ANALYSIS AND APPLICATIONS

Jun Li

ABSTRACT

A large class of sequential decision making problems under uncertainty with multiple competing decision makers/agents can be modeled as stochastic games. Stochastic games having Markov properties are called Markov games or competitive Markov decision processes. This dissertation presents an approach to solve non cooperative stochastic games, in which each decision maker makes her/his own decision independently and each has an individual payoff function. In stochastic games, the environment is nonstationary and each agent's payoff is affected by joint decisions of all agents, which results in the conflict of interest among the decision makers.

In this research, the theory of Markov decision processes (MDPs) is combined with the game theory to analyze the structure of Nash equilibrium for stochastic games. In particular, the Laurent series expansion technique is used to extend the results of discounted reward stochastic games to average reward stochastic games. As a result, auxiliary matrix games are developed that have equivalent equilibrium points and values to a class of stochastic games that are irreducible and have average reward performance metric.

R-learning is a well known machine learning algorithm that deals with average reward MDPs. The R-learning algorithm is extended to develop a Nash-R reinforcement learning algorithm for obtaining the equivalent auxiliary matrices. A conver-

gence analysis of the Nash-R algorithm is developed from the study of the asymptotic behavior of its two time scale stochastic approximation scheme, and the stability of the associated ordinary differential equations (ODEs). The Nash-R learning algorithm is tested and then benchmarked with MDP based learning methods using a well known grid game.

Subsequently, a real life application of stochastic games in deregulated power market is explored. According to the current literature, Cournot, Bertrand, and Supply Function Equilibrium (SFEs) are the three primary equilibrium models that are used to evaluate the power market designs. SFE is more realistic for pool type power markets. However, for a complicated power system, the convex assumption for optimization problems is violated in most cases, which makes the problems more difficult to solve. The SFE concept in adopted in this research, and the generators' behaviors are modeled as a stochastic game instead of one shot game. The power market is considered to have features such as multi-settlement (bilateral, day-ahead market, spot markets and transmission congestion contracts), and demand elasticity. Such a market consisting of multiple competing suppliers (generators) is modeled as a competitive Markov decision processes and is studied using the Nash-R algorithm.

CHAPTER 1
INTRODUCTION

1.1 Overview

Many industrial decision making problems such as inventory management, supply chain management, and airline yield management are inherently sequential. In these sequential problems, based on the observed system state, a decision maker chooses an action. The action results in two outcomes: an immediate reward, and a new state at the next decision epoch. An action is then chosen for the new state, and thus the system continues. Markov/semi-Markov decision processes (MDPs/SMDPs) [2][3] models are used to study these sequential decision making problems, if their underlying stochastic models are Markov chains.

Stochastic games are extensions of the above decision problems, and consist of multiple competing decision makers (also referred to as agents or players). The collective actions of the players dictate the next system state and also the individual rewards of the decision makers. A stochastic game is dynamic in the sense that the decision environment is nonstationary to each decision maker. The nonstationarity arises from the changes in the behavior of the other decision makers with time. Also, an inherent aspect of a competitive game is the conflict of interest among the players. Hence, making decisions in a competitive game environment is challenging. Game theory provides a framework to analyze the conflicts of all agents' interest [4][5].

Stochastic games were first studied by Shapley [6]. A central element in game theory is *Nash equilibrium* concept [7], which characterizes the rational players' behavior. This dissertation combines the knowledge from the theory of MDPs and the game theory to show how average reward irreducible stochastic games can be represented as equivalent matrix games. Matrix games are well studied for which solution strategies exist in the literature. A two time scale reinforcement learning algorithm is developed for obtaining the equivalent matrix games. A detailed analysis of convergence is developed for the algorithm.

Stochastic games have received some attention in recent years from mathematicians, computer scientists, and engineers [8][9][10]. Many dynamic programming and reinforcement learning based techniques [11][12][13][14][15][16][17][18] have been presented as solution methods for certain classes of games. Based on the optimality criterion, stochastic games can generally be divided into discounted reward stochastic games and average reward stochastic games. Most of the above referred work studied discounted reward games, since the discount factor that applies to many real life applications allows analytical advantages over the average reward criterion. But when decisions are made frequently or the reward criterion do not need to be described in economic terms, the expected average reward criterion may be more appropriate.

The two common classes of games are, purely collaborative games and purely competitive games. In purely collaborative games, all players have the identical payoff functions. Hence, the strategy that is in the best interest of one player is also in the best interest of all other players. In purely competitive games, there are two players and their payoffs sum to zero. This is commonly referred to as the zero-sum game. All games, except the purely competitive games(or, zero-sum games), are called general-sum games or nonzero-sum games. All games, except purely collaborative games, are called noncooperative games, since all players pursue their individual interest, goals

and make decisions independently. In zero-sum games, there exists one unique Nash equilibrium value, and thus it is relatively easy to obtain. The possible existence of multiple equilibrium values makes it harder to obtain the Nash equilibria for nonzero-sum games.

Shapley [6] was the first researcher to study transient zero-sum stochastic games (with a nonzero stopping probability) under the total reward criterion. He proved the existence of the value for such games. Discounted reward stochastic games are defined as games without nonzero stopping probabilities. In such games, where future payoffs are discounted using a discounting factor (β) [8], the payoffs eventually approach zero. This can be construed as the termination of the game. Hence the discounted reward stochastic games can be viewed as transient stochastic games with stopping probability $(1 - \beta)$ independent of the actions taken. For finite state space and bounded rewards, the discounted rewards are continuous in the strategies. So every discounted reward stochastic game possesses at least one equilibrium point. The fact that average rewards may not be continuous in the strategies makes the average reward stochastic games more challenging. However, for some special categories of games, such as irreducible games, the average reward is continuous in the strategies. The focus of this dissertation is on finite state space and bounded reward irreducible stochastic games with the expected average reward optimality criterion in the infinite horizon. For these games a mathematical structure of their Nash equilibrium policies is presented, which is critical to developing learning based solution algorithms.

This dissertation explores a real life application of the learning approach via modeling and solution of stochastic games in deregulated power markets. In the power market, with transmission technology developed, the entire eastern United States and eastern Canada were united in a single synchronized AC power system. By operating at extremely high voltages, this system is able to move power over great distances

with very little loss, often less than three percent in a thousand miles. This made the trade and competition possible. By 1990, encouraged by a general trend toward market deregulation, the de-integration trend in electric markets was underway. However, the power market has been less fortunate with deregulation and experienced many failures. In a turbulent environment such as the restructuring of the U.S. electric industry, new market designs are nearly always conceived and implemented without rigorous modeling and testing. In the existing literature, the three primary equilibrium models applied to evaluate market designs are the Cournot, Bertrand, and Supply Function Equilibrium (SFE) models of imperfect competition [19][20] [21] [22] [23]. The key difference among those models is how each generating firm anticipates that rivals will react to its decision concerning either quantity, price or supply function. SFE is more realistic regarding the competing behavior of firms in electricity markets. In a centralized market-clearing mechanism, such as POOLCO, the ISO requires all generator to offer a supply function bid. Most SFE calculation like other models is based on the Kuhn-Karesh-Tucker (first order) condition, which requires that the optimization problem is convex [24]. But in most real scenarios, this assumption is violated, hence it is hard to solve for mathematical programs. The methodology presented here is computable and is intended to help evaluate various market design alternatives.

1.2 Contribution

This dissertation contributes to three areas: the study of stochastic games, developing new machine learning algorithms for stochastic games, and a novel modeling methodology of power market design.

- The study of stochastic games shows how the theory of Markov decision processes (MDPs) and the game theory can be brought together to develop auxiliary matrix games having equivalent equilibrium points and values to average reward irreducible stochastic games. This development is essential to study stochastic games, since matrix games are well studied and have available computational algorithms.

- The findings above imply that an average reward irreducible stochastic game can be studied by examining the equivalent matrix game. Hence a critical task is to obtain the equivalent matrices. A new Nash-R reinforcement learning algorithm for obtaining the equivalent auxiliary matrices is developed. A convergence analysis of the algorithm is also developed from the study of the asymptotic behavior of its two time scale stochastic approximation scheme and the stability of the associated ordinary differential equations (ODEs). A simple grid-world game is used to test and benchmark the Nash-R algorithm with several MDP based approximation approaches to solve stochastic games.

- The power market consisting of multiple competing suppliers (generators) is modeled as a competitive Markov decision processes. The more realistic SFE concept is adopted and a system wide approach that considers market uncertainties and almost all of the market features such as multi-settlement (bilateral, day-ahead and spot markets), transmission rights, demand elasticity.

1.3 Outline

Chapter 2 presents an outline of the theory of MDPs and an overview of learning algorithms of MDPs, in order to establish notation and the theoretical framework for stochastic games. Chapter 3 is the underlying framework of stochastic game,

which introduces Matrix game first, then discusses the structure of equilibrium points for discounted reward stochastic games obtained from their equivalent auxiliary matrix games. Several model-based and model-free learning algorithms for discounted reward stochastic games are reviewed. Chapter 4 examines the behavior of the transition probability matrices of the average reward stochastic games in order to establish a connection between the discounted reward and the average reward games. For a special class of the average reward games, such a connection can be shown by limiting the discount factor in discounted reward stochastic games to 1. This allows to extend some of the results of discounted reward stochastic games to average reward games. Subsequently, the bias optimal Nash equilibrium for average reward games is defined, and auxiliary matrix games with equilibrium points and values that are equivalent to the average reward games are developed. Chapter 5 presents a reinforcement learning algorithm with its analysis of convergence for obtaining the equilibrium policy from the auxiliary matrix games. the learning algorithm is tested and benchmarked against other learning approaches using a grid-world game as a test bed. Chapter 6 contains a brief introduction of electricity market, such as market features, operations and economics. The three primary equilibrium models (Cournot, Bertrand, SFE) are presented. For each model, some applications in electricity market are reviewed and their advantages and disadvantages are discussed. Chapter 7 presents a power market model based on the pool market design, which is a bilevel optimization problem. The simulation based learning and testing experiments are performed on a sample network. Finally, the numerical results are analyzed to examine the market design and generators' bidding behavior.

CHAPTER 2

SINGLE DECISION MAKER PROBLEMS–MARKOV DECISION PROCESSES

This chapter gives an outline of the theory of MDPs. Infinite-horizon models with the expected total discounted reward and expected average reward optimal criterions are discussed. The optimality equation for average reward MDPs is obtained by limiting the discount factor to 1 in Bellman's optimality equation for discounted reward MDPs and using Laurent series expansion technique, which inspires our study for average reward stochastic game. A brief overview of reinforcement learning technique is presented, and subsequently some well known reinforcement learning algorithms which successfully learn the value functions of discounted reward and average reward MDPs.

2.1 MDPs Framework

In sequential decision making problems, the decision maker's goal is to choose a sequence of actions that maximizes the utility of the system based on a given criterion. Such problems with an underlying Markovian structure can be defined within the framework of MDPs denoted by a tuple $< S, A, T, R >$. The elements of the tuple are as follows.

- S denotes the finite set of states of the environment.
- A denotes the finite set of actions available to the decision maker (agent).

- $P : S \times A \rightarrow \prod(S)$ is the transition function, where $\prod(\cdot)$ indicates the probability distribution. An element $p(s'|s, a)$ of P gives the one step probability of reaching state s', when action a is taken in state s.

- $R : S \times A \rightarrow \Re$ is the reward function that gives the expected immediate reward corresponding to every action in each state. An element $r(s, a)$ of R denotes the expected reward for action a in state s.

While MDPs may appear simple, they encompasses a wide range of applications and has generated a rich mathematical theory. For MDPs, the consequences of the decisions are uncertain but the environment is stationary. A probability distribution of consequences are associated with each alternative action, and this distribution does not change over time once the decision policy is given. For a given policy, a reward is received in each period. The agent's job is to find a policy π, mapping states to actions, that maximizes some measure of the rewards received. The MDPs are either finite-horizon models or infinite-horizon models. The infinite-horizon *discounted reward* model takes the long-run reward of the agent into account, but rewards that are received in the future are geometrically discounted according to discount factor β (where $0 \leq \beta < 1$). The expected discounted reward for an infinite-horizon MDP starting in state $s \in S$ can be given as $V_\beta(s) = E(\sum_{t=0}^{\infty} \beta^t r_t)$, where r_t is the reward received at the t^{th} decision epoch. Another optimality criterion is long-run *average-reward*, which can be given by $g(s) = \lim_{T\rightarrow\infty} E(\frac{1}{T+1} \sum_{t=0}^{T} r_t)$. The average reward g is also referred to as the gain of this system. A policy is a mapping $\pi : S \rightarrow \prod(A)$. A deterministic policy is one that assigns a probability value of 1 to an action in each state. Every MDP has a deterministic stationary optimal policy [2]. For unichain average reward MDPs where there is only one recurrent class of states, and possibly a set of transient states, the gain is identical for all the states. Since in average reward

MDPs the sum of the rewards can be unbounded, a surrogate used is called *bias* h, which is defined for a starting state $s \in S$ as:

$$h(s) = E_s \sum_{t=1}^{\infty} [r_t - g(s)].$$

The bias is interpreted as the expected total difference between the reward and the stationary reward.

The optimality equation for discounted reward MDPs is given as:

$$V_\beta(s) = \max_{a \in A(s)} \left(r(s,a) + \beta \sum_{s'} p(s'|s,a) V_\beta(s') \right). \tag{2.1}$$

The average reward of unichain MDPs can be seen as the limiting discounted reward with discount factor β approaching 1. Using truncated Laurent series expansion of the discounted value function (2.1) (refer to [2] page 314-355 for details), we can write the value function in vector form for the average reward MDP in terms of gain and bias as follows:

$$g + (I - P)h = r. \tag{2.2}$$

From (2.2), the optimality equation for a unichain average reward MDP may be expressed in component notation as

$$0 = \max_{a \in A_s} \left(r(s,a) - g + \sum_{s' \in S} p(s'|s,a) h(s') - h(s) \right).$$

The above equation can be rewritten in Bellman's optimality equation form as

$$h^*(s) + g^* = \max_{a \in A_s} \left(r(s,a) + \sum_{s'} P(s'|s,a) h^*(s') \right). \tag{2.3}$$

Later in the paper, we use the Laurent series expansion technique, similar to the above, to relate discounted reward games with average reward games.

2.2 Value Function Based Reinforcement Learning

If the reward functions and the transition probability functions are known, dynamic programming algorithms [25][26][27] such as value iteration or policy iteration can be used to find the optimal policy. In real life, for many stochastic decision problems, the transition probability matrices and the reward functions are not easily available. A common approach used is to simulate the system and learn the model first. This approach is called model-based learning. The framework of dynamic Bayesian networks (DBNs) can be used to describe a certain class of MDPs in a compact way [28].

A more common approach adopted by the researchers is model-free learning, which is suitable for solving problems with very large state spaces. In recent years, reinforcement learning methods have shown to yield optimal or near-optimal solutions to large MDPs.

2.2.1 Introduction

Sutton and Barto wrote an introductory textbook on reinforcement learning (RL) [29]. Kaelbling et al. also gave a survey about the field of RL from a computer-science perspective [30]. In this section we give a brief introduction based on the above literature.

Reinforcement learning is defined not by characterizing learning algorithms, but by characterizing a learning problem. RL is the problem faced by an agent that must learn behavior through trial-error interactions with a dynamic environment. Figure 2.1 shows a single agent learning problem. *Environment* comprises everything outside

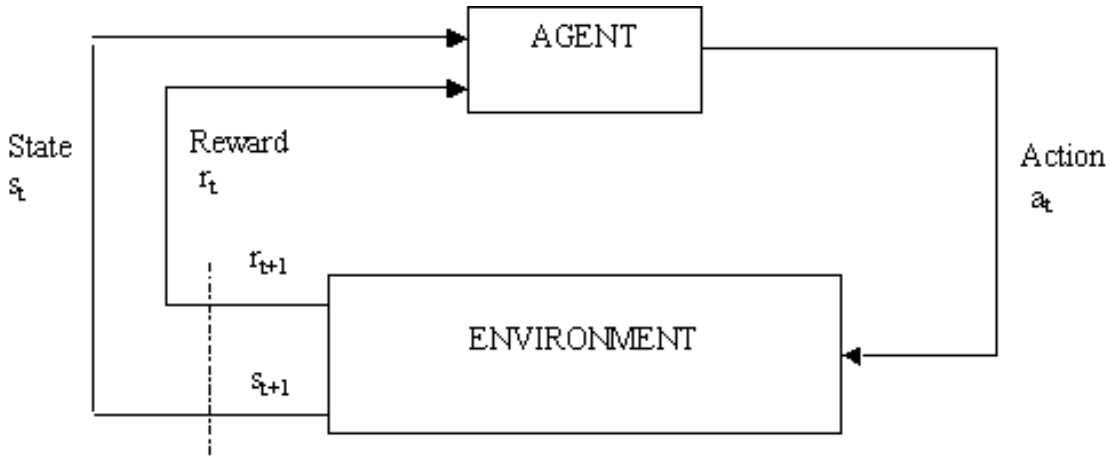

Figure 2.1. The single agent learning framework

the agent. The agent and environment interact at each of a sequence of discrete time steps, $t = 0, 1, 2, \ldots$. At each time step t, the agent receives some representation of the environment's state, $s_t \in S$, and on that basis selects an action, $a_t \in A(s_t)$. One time step later, in part as a consequence of its action, the agent receives a numerical reward, $r_{t+1} \in \Re$, and finds itself in a new state, s_{t+1}. These interactions evolve continually.

The system is *Markovian* if the state transitions are independent of any previous environment states or agent actions. Agent should choose actions that tend to increase the long-run sum of values of the reinforcement signal. It can learn to do this over time by systematic trial and error, guided by a wide variety of algorithms that are the subject of next section.

2.2.2 Q-learning and R-learning Algorithms

We consider algorithms for learning to behave in MDP environments. The optimal behavior is obtained based on the Bellman's optimal value functions (2.1,2.3). One way to find an optimal policy is to find the optimal value function.

One of the most important breakthroughs in reinforcement learning was the development of *Q-learning* by Watkins [31] for discounted reward MDPs. $Q^*(s,a)$ is the expected discounted reinforcement of taking action a in state s, then continuing by choosing actions optimally. So $V_\beta(s) = \max_a Q^*(s,a)$. Denote an experience tuple by $< s, a, r, s' >$, where s is the current state and s' is the next state. The Q-learning rule is

$$Q(s,a) := Q(s,a) + \alpha(r(s,s',a) + \beta \max_{a'} Q(s',a') - Q(s,a),$$

If each action is executed in each state an infinite number of times on an infinite run and α is decayed appropriately, the Q values will converge with probability 1 to Q^* [31][32].

The development of R-learning algorithms for average reward MDPs was not so smooth. Schwartz first presented a R-learning but didn't give convergence analysis. Mahadevan carried a detailed empirical study of R-learning also without convergence analysis. Abounadi [33] was the first to present convergence proof for several R-learning algorithms. Tsitsiklis [34] gave an average cost temporal-difference learning algorithm and its convergence analysis. An algorithm for the more general Semi-Markov decision problems can be found in Das et al.[3]. Gosavi [35][36] provided a convergence analysis for a two time scale algorithm. The two time scale R-learning rules are

$$R_{t+1}(s,a) = (1-\alpha_t)R_t(s,a) + \alpha_t \left(r(s,s',a) - g_t + \max_{a' \in A(s')} R_t(s',a') \right),$$

and

$$g_{t+1} = (1-\alpha'_t)g_t + \alpha'_t \left[\frac{r(s,s',a) + t * g_t}{t+1} \right],$$

where $0 \leq \alpha_t < 1$ is the learning rate to update R which denotes the bias, and $0 \leq \alpha'_t < 1$ is the learning rate to update g which denotes the gain.

Stochastic games are extensions of MDPs to scenarios with multiply players, a game theoretical perspective is used to analyze stochastic games, which is presented in next chapter.

CHAPTER 3

MULTIPLE DECISION MAKER PROBLEMS–STOCHASTIC GAMES

When two or more decision makers get involved in decision making, their interests are coupled because the transition probabilities are coupled and/or their rewards are coupled. Stochastic games having Markov properties are also called Markov games or competitive Markov decision processes(CMDP), which are an extension of MDP to a competitive scenario. Considered in this dissertation are finite state/action *noncooperative games*, where all decision makers make their decision independently and competitively to maximize their individual payoff criterion. But before going into stochastic games, game theory is introduced first. The focus here is on matrix games, which are also called normal form games in game theory.

3.1 Matrix Games

Game theory is a formal way to analyze interaction among a group of rational agents who behave strategically, hence it can be regarded as a multi-agent decision problem. There are cooperative games and noncooperative games. The agents can form coalitions in the cooperative games while choose actions independently in the noncooperative games. Two common forms are used to represent noncooperative games: normal form and extensive form. Normal form games usually are used to represent cases where agents choose actions simultaneously, while extensive-form games are used for sequential moves. Normal form games can be called matrix games, which

can be defined by a tuple $< n, A^1, \ldots, A^n, R^1, \ldots, R^n >$. The elements of the tuple are as follows.

- n denotes the number of players.
- A^i denotes the set of actions available to player i.
- $R^i : A^1 \times \ldots \times A^n \rightarrow \Re$ is the payoff function for player i, where an element $r^i(a^1, \ldots, a^n)$ of R^i is the payoff to agent i when the agents choose actions a^1 through a^n. It is called matrix game, since R^i for all i, can be written as an n-dimensional matrices.

Battle of the Sexes is a matrix game shown in Table 3.1. The idea of this game is that a couple wants to spend the evening together. The wife wants to go to the Opera (O), while the husband wants to go to a football (F) game. Each get at least some utility from going together to at least one of the venues, but each wants to go their favorite one (the husband is denoted as player 1,the column player). The game can be represented by the matrix of payoffs (bimatrix game).

The players select actions from the set of available actions with the goal of maximizing their payoffs which depends on the actions chosen by all players. The concept of *Nash equilibrium* is used to describe the strategy as being the most rational behavior by the players acting to maximize their payoffs. So for a bimatrix game (i.e., a game with two players), a pure strategy Nash equilibrium is an action profile (a^{1*}, a^{2*}), for which $r^1(a^{1*}, a^{2*}) \geq r^1(a^1, a^{2*})$, $\forall a^1 \in A^1$, and $r^2(a^{1*}, a^{2*}) \geq r^2(a^{1*}, a^2)$, $\forall a^2 \in A^2$. The equilibrium values denoted by $Val[\cdot]$ for players 1 and 2 with payoff matrices R^1 and R^2 respectively are obtained as $Val[R^1] = r^1(a^{1*}, a^{2*})$ and $Val[R^2] = r^2(a^{1*}, a^{2*})$.

So in the battle of sexes, (F,F) and (O,O) are the pure Nash equilirium. The appealing feature of the Nash equilibrium is that any unilateral deviation from it by

Table 3.1. The matrix game for *Battle of Sexes*

	F	O
F	2, 1	0,0
O	0,0	1,2

Table 3.2. The matrix game for *Rock, Paper, Scissors*

	rock	paper	scissors
rock	0, 0	-1,1	1,-1
paper	1, -1	0,0	-1,1
scissors	-1, 1	1,-1	0,0

any player is not worthwhile. A mixed strategy Nash equilibrium for bimatrix games is a pair of vectors (π^{1*}, π^{2*}), the probability distributions over actions, for which $\pi^{1*}R^1\pi^{2*} \geq \pi^1 R^1 \pi^{2*}, \forall \pi^1$, and $\pi^{1*}R^2\pi^{2*} \geq \pi^{1*}R^2\pi^2, \forall \pi^2$. For mixed Nash equilibrium strategy, the value is obtained as $Val[R^1] = \pi^{1*}R^1\pi^{2*}$ and $Val[R^2] = \pi^{1*}R^2\pi^{2*}$.

Another example is *Rock, Paper, Scissors* (Table 3.2) in which deterministic policy can be consistently defeated. It can be shown that a mixed strategy Nash equilibrium {(1/3, 1/3, 1/3), (1/3, 1/3, 1/3)} exists in this game. A matrix game may not have a pure strategy Nash equilibrium, but it always has a mixed strategy Nash equilibrium (Nash 1951 [7]).

In a two-player game, if $r^1(a^1, a^2) + r^2(a^1, a^2) = 0$, the game is called a zero-sum game such as *Rock, Paper, Scissors* game, which is also referred to as a purely competitive game. All other games are called nonzero-sum games or general sum games such as *Battle of the Sexes*. There exist linear programming methods to solve for Nash equilibrium of zero-sum matrix games, and quadratic programming methods to solve for Nash equilibrium of nonzero-sum matrix games. Since in matrix games, there are no transition probability functions, matrix games are static.

3.2 Stochastic Games Framework

A stochastic game can be defined by a tuple $< n, S, A^1, \ldots, A^n, P, R^1, \ldots, R^n >$. The elements of the tuple are as follows.

- n denotes the number of agents/players/decision makers.
- S denotes the finite set of states of the environment.
- $A^1, \ldots, A^n$ denote the collection of finite set of actions available to the agents $1, \ldots, n$, where $m^1(s) = |A^1(s)|, \ldots, m^n(s) = |A^n(s)|$ are the cardinalities of the action spaces in state s.
- $P : S \times A^1 \times \ldots \times A^n \rightarrow \prod(S)$ is the transition function, where an element $p(s'|s, a^1, \ldots, a^n)$ of P is the probability of reaching state s' as a result of action a^1 through a^n chosen by players 1 through n in state s.
- $R^i : S \times A^1 \times \ldots \times A^n \rightarrow \Re$ is the reward function that gives the expected immediate reward gained by the agent i for each set of actions of the agents in each state. For example, an element $r^i(s, a^1, \ldots, a^n)$ of R^i is the expected reward of agent i in state s when agents 1 through n choose actions a^1 through a^n respectively.

In a stochastic game, the transition probabilities and the reward functions depend on the choices made by all the agents. Thus, from the perspective of an agent, the game environment is nonstationary during its evolutionary phase. However, for stochastic games, optimal strategies constitute stationary policies and hence it is sufficient to consider only the stationary strategies [8]. We define $\pi^i(s)$ as the mixed strategy at state s for agent i, which is the probability distribution over available action set of agent i. Thus $\pi^i(s) = \{\pi^i(s, a) : a \in A^i(s)\}$, where $\pi^i(s, a)$ denotes the

probability of agent i choosing action a in state s, and $\sum_{a \in A^i(s)} \pi^i(s,a) = 1$. Then $\pi = \{\pi^i(s) : s \in S, i = 1, \ldots, n\}$ denotes a policy.

Under policy π, the transition probability can be given as

$$P(s'|s,\pi) = \sum_{a^1=1}^{m^1(s)} \ldots \sum_{a^n=1}^{m^n(s)} p(s'|s,a^1,\ldots,a^n)\pi^1(s,a^1)\ldots\pi^n(s,a^n).$$

The immediate expected reward of player i induced by π in a state s is given as

$$r^i(s,\pi) = \sum_{a^1=1}^{m^1(s)} \ldots \sum_{a^n=1}^{m^n(s)} r^i(s,a^1,\ldots,a^n)\pi^1(s,a^1)\ldots\pi^n(s,a^n).$$

Then the overall discounted value of a policy π to agent i starting in state s can be given as

$$V^i_\beta(s,\pi) = \sum_{t=0}^{\infty} \beta^t E_s(r^i_t) = \sum_{t=0}^{\infty} \beta^t \sum_{s' \in S} P^t(s'|s,\pi) r^i(s',\pi),$$

where $P^t(\cdot)$ denotes the t^{th} power of P.

The overall average value of a policy π to agent i starting in state s can be given as

$$V^i_\alpha(s,\pi) = \lim_{T\to\infty} \frac{1}{T+1} \sum_{t=0}^{T} E_s(r^i_t) = \lim_{T\to\infty} \frac{1}{T+1} \sum_{t=0}^{T} \sum_{s' \in S} P^t(s'|s,\pi) r^i(s',\pi).$$

Clearly, in noncooperative games, every agent has separate reward functions and value functions. The agents are only interested in maximizing their individual values. However, a strategy by which all agents can achieve their maximum values may not exist. This gives rise to a conflict. Game theory is the study of such conflicts between two or more intelligent rational decision makers. The concept of *Nash equilibrium* from game theory is useful for conflict resolution in stochastic games.

3.2.1 Stochastic Game VS Matrix Game

Matrix games can be viewed as recursive stochastic games with single state. On the other hand, stochastic games can be viewed as extensions of matrix games from a single state to a multi-state environment.

This above idea is presented through a schematic diagram in Figure 3.1. The figure shows that when system time $T = t$, the system is at state s, and $[r^i(s, a^1, \ldots, a^n)]_{a^1=1,\ldots,a^n=1}^{a^1=m^1(s),\ldots,a^n=m^n(s)}$ is the immediate payoff matrix for player i. Thus, state s has associated with it an n-dimensional matrix game. If the players choose strategy π, then at time $T = t + 1$, the system evolves to state $s' \in \{s^1, \ldots, s^m\}$ according to the transition probability $P(s'|s, \pi)$, where m is the total number of states. For each of the new possible states, there is a corresponding matrix game. So a stochastic game involves multiple matrix games, which are connected by transition probabilities. But one can not just solve for the immediate payoffs of the matrix games separately to get the equilibrium strategies for the states, since in addition to the immediate payoffs the opportunities in future states must also be considered.

3.3 Discounted Reward Stochastic Games

Due to the economic meaning of the discount factor and also the mathematical convenience that it provides to bound the infinite sum, discounted stochastic games have been studied extensively. Most of what is presented in this section are multi-player extension of the treatment of two-player games by Filar and Vrieze [8].

3.3.1 Definition

The discounted reward stochastic games are defined as games in which future payoffs are discounted by a discount factor β. A β discounted stochastic game

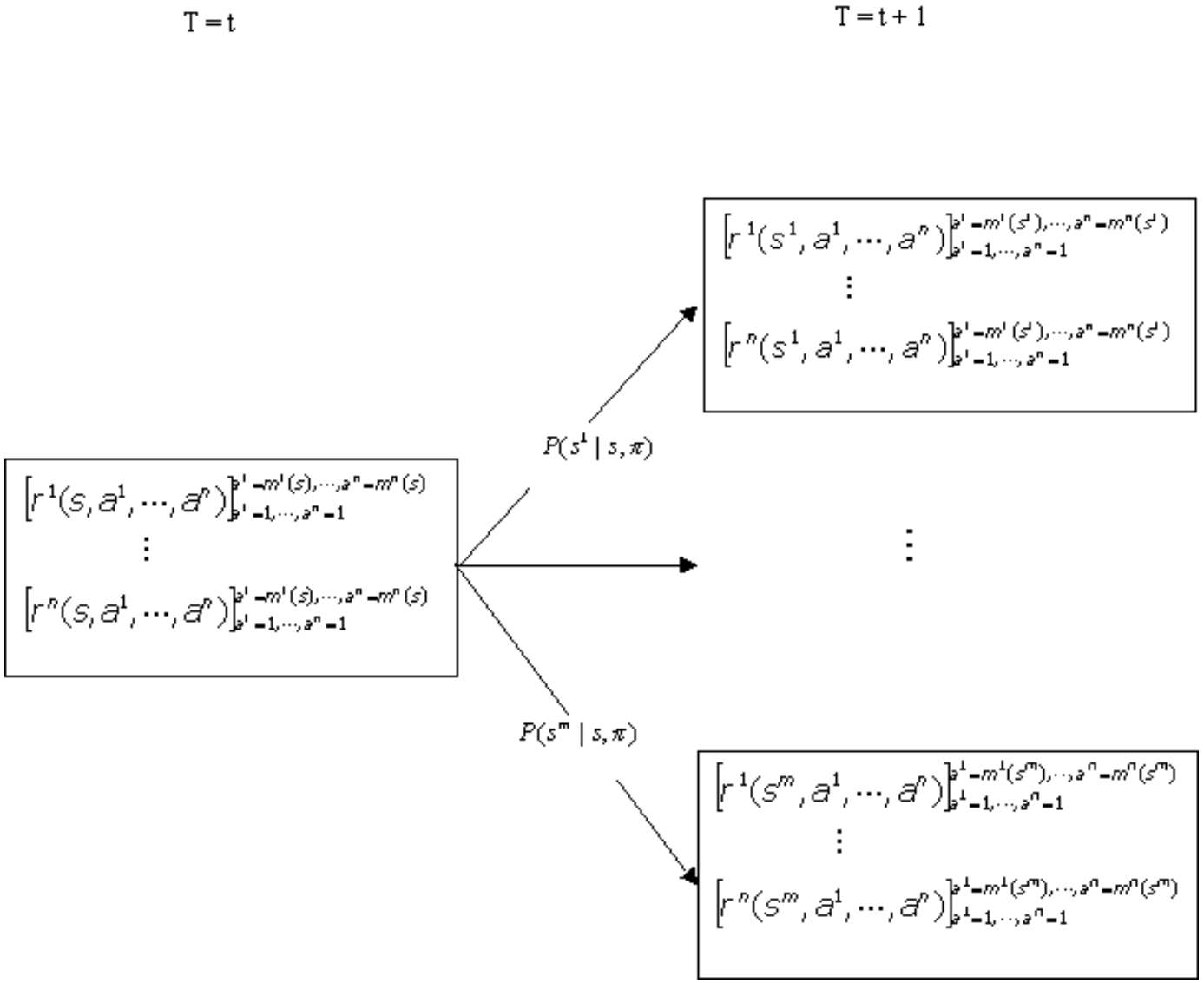

Figure 3.1. Stochastic game as multi-state matrix game

can be viewed as a game for which the stopping probability in each state is $1-\beta$. Consequently, a normalization factor $(1-\beta)$ can be introduced to the definition of the discounted reward as.

$$V_\beta^k(\pi^1,\ldots,\pi^n) = (1-\beta)\sum_{t=0}^{\infty}\beta^t P^t(\pi^1,\ldots,\pi^n)r^k(\pi^1,\ldots,\pi^n). \tag{3.1}$$

The discounted reward models with or without the factor $(1-\beta)$ are mathematically equivalent.

3.3.2 Nash Equilibrium Definition

Let $\pi_* = \{\pi_*^1,\ldots,\pi_*^n\}$ and $(\pi_*^{-k},\pi^k) = \{\pi_*^1,\ldots,\pi_*^{k-1},\pi^k,\pi_*^{k+1},\ldots,\pi_*^n\}$, where the latter indicates that only player k plays arbitrary strategy π^k while the other players play the strategies defined by π_*. The strategy π_* denotes the *Nash equilibrium point* for discounted stochastic game if $V_\beta^k(\pi_*) \geq V_\beta^k(\pi_*^{-k},\pi^k)$ for all $k = 1,2,\ldots,n$. The discounted reward given in (3.1) can be rewritten in component notation in terms of the expected immediate reward and the expected discounted value of the next state as follows

$$V_\beta^k(s,\pi) = (1-\beta)r^k(s,\pi) + \beta\sum_{s'\in S} p(s'|s,\pi)V_\beta^k(s',\pi), \tag{3.2}$$

from which the definition of *Nash equilibrium* can be expanded to

$$\begin{aligned}(1-\beta)r^k(s,\pi_*)+ \quad \beta\textstyle\sum_{s'\in S} p(s'|s,\pi_*)V_\beta^k(s',\pi_*) \geq (1-\beta)r^k(s,\pi_*^{-k},\pi^k) + \\ \beta\textstyle\sum_{s'\in S} p(s'|s,\pi_*^{-k},\pi^k)V_\beta^k(s',\pi_*^{-k},\pi^k)\end{aligned} \tag{3.3}$$

Solving for Nash equilibrium using the above inequality is difficult, even when the reward functions and transition probabilities are available. But since there are

methods to solve for equilibrium points of matrix games, some researchers have tried to study stochastic games by exploiting their connection with matrix games.

3.3.3 Alternative Nash Equilibrium Definition in Matrix Game Form

Filar and Vrieze [8] combined the theories of discounted Markov decision processes and matrix games to develop an auxiliary bi-matrix game for two-player discounted stochastic games. In what follows, the above technique is extended to n-player game and construct n-dimensional equivalent auxiliary matrices $Q^k(\cdot)$ for all players $k = 1, \ldots, n$. The elements of these matrices are payoffs for all possible pure action sets $(a^1, \ldots, a^n)$, which take into account both the immediate reward and the future opportunities. For $s \in S$, the matrix with size $|m^1(s)| \times |m^2(s)| \times \cdots \times |m^n(s)|$ for the k^{th} player can be given as

$$Q^k(s) = \left[(1-\beta) r^k(s, a^1, \ldots, a^n) + \beta \sum_{s' \in S} p(s'|s, a^1, \ldots, a^n) V_\beta^k(s', \pi_*) \right]_{a^1=1,\ldots,a^n=1}^{a^1=m^1(s),\ldots,a^n=m^n(s)}, \tag{3.4}$$

where $V_\beta^k(s', \pi_*)$ is the equilibrium value for the stochastic game.

If players use a strategy π, the value of the above matrix game can be obtained as:

$$\begin{aligned} Val[Q^k(s), \pi] &= \sum_{a^1=1}^{m^1(s)} \cdots \sum_{a^n=1}^{m^n(s)} \pi^1(s, a^1) \ldots \pi^n(s, a^n) Q^k(s, a^1, \ldots, a^n) \\ &= \sum_{a^1=1}^{m^1(s)} \cdots \sum_{a^n=1}^{m^n(s)} \pi^1(s, a^1) \ldots \pi^n(s, a^n) \{ (1-\beta) r^k(s, a^1, \ldots, a^n) + \\ &\qquad \beta \sum_{s' \in S} p(s'|s, a^1, \ldots, a^n) V_\beta^k(s', \pi_*) \} \\ &= (1-\beta) r^k(s, \pi) + \beta \sum_{s' \in S} p(s'|s, \pi) V_\beta^k(s', \pi_*). \end{aligned}$$

For the matrix game (3.4), the equilibrium point π_* is defined such that

$$Val[Q^k(s), \pi_*] \geq Val[Q^k(s), \pi_*^{-k}, \pi^k],$$

which can be expanded to

$$(1-\beta)r^k(s,\pi_*)+\beta \sum_{s' \in S} p(s'|s,\pi_*)V_\beta^k(s',\pi_*) \geq (1-\beta)r^k(s,\pi_*^{-k},\pi^k)+\beta \sum_{s' \in S} p(s'|s,\pi_*^{-k},\pi^k)V_\beta^k(s',\pi_*). \tag{3.5}$$

The difference between the definitions of Nash equilibria for discounted reward stochastic game (3.3) and auxiliary matrix game (3.5) is subtle. In the following theorem, we show that the two equilibria are equivalent.

Theorem 1 *The following assertions are equivalent:*
(i) π_* is an equilibrium point in the discounted reward stochastic game with equilibrium payoffs $(V_\beta^1(\pi_*), \ldots, V_\beta^n(\pi_*))$.
(ii) For each $s \in S$, the strategy π_* constitutes an equilibrium point in the static n-dimensional matrix game $(Q^1(s), \ldots, Q^n(s))$ with equilibrium payoffs $(V_\beta^1(s,\pi_*), \ldots, V_\beta^n(s,\pi_*))$. The entry of $Q^k(s)$ corresponding to actions $(a^1, \ldots, a^n)$ is given by $Q^k(s,a^1,\ldots,a^n) = (1-\beta)r^k(s,a^1,\ldots,a^n) + \beta \sum_{s' \in S} p(s'|s,a^1,\ldots,a^n)V_\beta^k(s',\pi_*)$, for $k = 1,\ldots,n$, where $(a^1,\ldots,a^n) \in \prod_{i=1}^n A^i(s)$.

Proof If (i) is true, it means that π_*^k is the best response of player k to the equilibrium responses of other players. Then player k is not able to improve his/her reward by deviating from the strategy π_*^k even for one step (a similar argument can be found in many cited works including [8][37]). Then we write that $V_\beta^k(s,\pi_*) \geq (1-\beta)r^k(s,(\pi_*^{-k},\pi^k)) + \beta \sum_{s' \in S} p(s'|s,(\pi_*^{-k},\pi^k))V_\beta^k(s',\pi_*)$, which implies inequality (3.5). So π_* is the equilibrium point of the matrix game in (ii).

If (ii) is true, then the equilibrium points satisfy the inequality (3.5). Then the left side of (3.5) is $V_\beta^k(s, \pi_*)$, which in matrix form gives $V_\beta^k(\pi_*) \geq (1-\beta) r^k(\pi_*^{-k}, \pi^k) + \beta P(\pi_*^{-k}, \pi^k) V_\beta^k(\pi_*)$. Then by iteration of this inequality infinite times, we get $V_\beta^k(\pi_*) \geq (1-\beta) r^k(\pi_*^{-k}, \pi^k) + (1-\beta)\beta P(\pi_*^{-k}, \pi^k) r^k(\pi_*^{-k}, \pi^k) + (1-\beta)\beta^2 P^2(\pi_*^{-k}, \pi^k) r^k(\pi_*^{-k}, \pi^k) + \ldots + (1-\beta) \lim_{t\to\infty} \beta^t P^t(\pi_*^{-k}, \pi^k) V_\beta^k(\pi_*)$. The limiting value of the last item is 0. Then based on the definition of discounted value in (3.1), the above inequality gives $V_\beta^k(\pi_*) \geq V_\beta^k(\pi_*^{-k}, \pi^k)$. Hence π_* is an equilibrium point of the stochastic game in (i). ■

Note that we can not interpret any of the elements of (3.4) as the expected discounted reward gained when all players play the corresponding pure action set in that state. This is due to the fact that the expected utility in stationary strategies does not hold for the class of stochastic games [37]. Once the matrices $Q^k(\cdot)$ are constructed, linear or quadratic programming methods can be used to solve for Nash equilibrium. However, to construct the static matrix games, we need to know the reward functions $r^k(\cdot)$, transition probability functions $P(\cdot)$, and the equilibrium value $V_\beta^k(\pi_*)$. We may know the reward and transition probability, but not the equilibrium value before the problem is solved. Filar and Arieze [8] presented a nonlinear complementarity programming approach for solving two player matrix games.

3.3.4 Learning Discounted Reward Stochastic Games

In a single-agent model, the agent treats anything outside itself as environment. And this environment can only be effected by this agent hence is stationary. But in a multi-agent model, environment is affected by all the agents' decisions. The

idea is represented in Figure 3.2. Hence single agent algorithm won't work in all multi-agent problems.

We note that, the entries in the matrix game (3.4) have similar structure to the Bellman's optimality equation for discounted MDP. Well known algorithms to solve Bellman's discounted optimality equation are value iteration and policy iteration. Shapley [6] extended the value iteration and redefined the value operator so as to solve stochastic games.

$$G(s,a,V) = \left[r(s,a) + \beta \sum_{s' \in S} p(s'|s,a) V(s') \right],$$

$$V(s') \leftarrow Val[G(s'), \pi^*].$$

This algorithm replaces the "max" operator in value iteration for MDPS with Nash equilibrium value operator.

There exsit the learning algorithms that attempt to learn the entries of the $Q^k(\cdot)$ matrices. The matrices are updated during each stage and are expected to converge to their optimal forms. Littman [13] presented Minmax Q-learning algorithm for discounted zero-sum games.

$$Q^1(s,a^1,a^2) = (1-\alpha)Q(s,a^1,a^2) + \alpha(r(s,s',a^1,a^2) + \beta V(s')),$$

$$V(s') \leftarrow \max_{\pi^1(s')} \min_{a^2} \sum_{a^1} \pi^1(s',a^1) Q^1(s',a^1,a^2).$$

Hu and Wellman [38] presented Nash Q-learning for discounted general-sum games (see also [15] [39]).

$$Q^1(s,a^1,\cdots,a^n) = (1-\alpha)Q(s,a^1,\cdots,a^n) + \alpha(r(s,s',a^1,\cdots,a^n) + \beta V(s')),$$

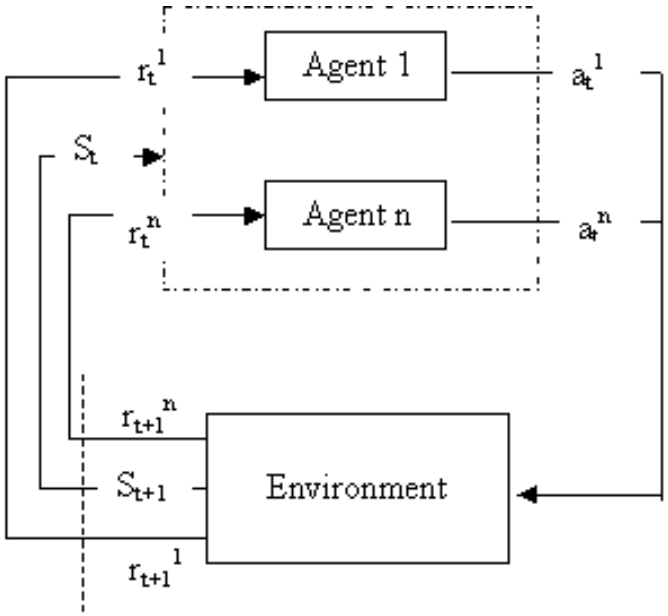

Figure 3.2. The multi-agent learning framework

$$V(s') \leftarrow Val[Q^k(s'), \pi^*].$$

Both minmax and Nash-Q learning algorithms are extensions of the model-free reinforcement Q-learning [29][30]. Bowling [16] presented a table summarizing the available stochastic game algorithms.

For MDPs, there is only one optimal value for each state. But for stochastic games, there may be several Nash equilibrium points and perhaps several equilibrium values. Theorem 1 says that each equilibrium value has a corresponding matrix game. So there can be several possible equivalent matrix games for each state of a stochastic game. But in a machine learning based approach, where each agent usually can only learn one matrix game, assuring or verifying that all the matrices learned by different players correspond to the same equilibrium value is difficult. In other words, it is possible that strategies learned by the players may not constitute a Nash equilibrium policy. For zero-sum games with unique Nash equilibrium value, such a problem

does not arise. Nash-Q algorithm [15] for general-sum games is developed only for problems having a unique Nash equilibrium value.

CHAPTER 4

AVERAGE REWARD STOCHASTIC GAMES

In this chapter we explore the average reward stochastic games by examining the relationship between the discounted reward V_β and the average reward V_α. For average reward stochastic games, we will construct auxiliary matrix games having equivalent Nash equilibria and values.

4.1 Preliminaries

For some special cases (e.g., irreducible games), the limiting average criterion can be treated as the limit of the β discounted criterion with β going to 1. One of the underlying mathematical theory for the above is a classical result referred to as the Hardy-Littlewood Theorem [8].

Theorem 2 (Hardy and Littlewood) *Let* $\{a_n\}_{n=0}^{\infty}$ *be a bounded sequence of real numbers.*
$f(\beta) := (1-\beta)\sum_{n=0}^{\infty}\beta^n a_n$ *and* $\sigma_N := \frac{\sum_{n=0}^{N} a_n}{N+1}$
If $\lim_{\beta\to 1^-} f(\beta) = a$ *(Abel-summable), then* $lim_{N\to\infty}\sigma_N = a$ *(Cesaro-summable).*

Theorem 2 is stated for a sequence of real numbers, whereas the study of stochastic games involve a sequence of transition probability matrices $P(\cdot)$. In the following, we study the behavior of this matrix sequence.

Proposition 3 *For any* $\beta \in (0,1)$, $(I-\beta P)$ *is nonsingular and* $(I-\beta P)^{-1} = \sum_{t=0}^{\infty}\beta^t P^t$.

One can verify that the row sums of $(I - \beta P)^{-1}$ equal $\frac{1}{1-\beta}$, which mathematically explains why the normalization factor $(1 - \beta)$ is added to the discounted reward definition.

Define the cesaro limit of P as $Q := \lim_{T\to\infty} \frac{1}{T+1} \sum_{t=0}^{T} P^t$.

Proposition 4 $Q = \lim_{\beta\to 1^-} (1 - \beta)(I - \beta P)^{-1}$.

The above proposition follows easily from the Hardy and Littlewood theorem that a Cesaro-summable sequence is also Abel-summable and their limit sums are the same.

In Markov control problems, the fundamental matrix $H_P := (I - P + Q)^{-1}(I - Q)$ plays an important role.

Proposition 5

$$H_P = \lim_{N\to\infty} \frac{1}{N} \sum_{k=0}^{N-1} \sum_{i=0}^{k} (P^i - Q). \tag{4.1}$$

4.2 Average Reward VS Discounted Reward

For irreducible stochastic games, there is only one ergodic class independent of the strategies, and the invariant probabilities Q are the same for each starting state. Hence, for average reward $V_\alpha = \lim_{T\to\infty} \frac{1}{T+1} \sum_{t=0}^{T} P^t r$, the reward vector r can be put outside of the $\lim_{T\to\infty}$ operator. Then based on Proposition 4, the average reward value can be written as

$$V_\alpha = \lim_{\beta\to 1^-} (1 - \beta)(I - \beta P)^{-1} r. \tag{4.2}$$

Similarly, we can rewrite the discounted reward as

$$V_\beta = (1 - \beta)(I - \beta P)^{-1} r. \tag{4.3}$$

By comparing the equations (4.2) and (4.3), we can see that if the P and r for discounted reward game and average reward game are identical, the average reward value is the same as the limiting discounted reward value with β going to 1. This implies that for any given strategy π independent of β, we have $V_\alpha(\pi) = \lim_{\beta \to 1^-} V_\beta(\pi)$. What is of concern here is that, whether $V_\alpha(\pi_1) = \lim_{\beta \to 1^-} V_\beta(\pi_\beta)$ holds when $\pi_\beta \to \pi_1$. Since the P matrix and the r vector are functions of the strategies, the above can be expanded to

$$\lim_{\beta \to 1^-} (1-\beta)[I - \beta P(\pi_1)]^{-1} r(\pi_1) = \lim_{\beta \to 1^-} (1-\beta)[I - \beta P(\pi_\beta)]^{-1} r(\pi_\beta).$$

We know that as β changes, the optimal strategy π_β for discounted reward changes resulting in the changes of $P(\pi_\beta)$ and $r(\pi_\beta)$. Hence the key issue here is the continuity of $P(\cdot)$ and $r(\cdot)$ functions on the strategy space. Continuity of $P(\cdot)$ and $r(\cdot)$ will ensure that $\lim_{\beta \to 1^-} P(\pi_\beta) = P(\pi_1)$ and $\lim_{\beta \to 1^-} r(\pi_\beta) = r(\pi_1)$ when $\pi_\beta \to \pi_1$. For an irreducible Markov chain with finite action space and bounded reward, it is well known that $P(\cdot)$ and $r(\cdot)$ are continuous functions on π. Hence, for irreducible stochastic games, if $\pi_1 = \lim_{\beta \to 1^-} \pi_\beta$, then $V_\alpha(\pi_1) = \lim_{\beta \to 1^-} V_\beta(\pi_\beta)$.

4.3 Average Reward Irreducible Stochastic Games

The following theorem for average reward irreducible games follows from the above preliminaries and results. The theorem is an extension of the result obtained by Filar and Arieze [8] for two player games to multiplayer games.

Theorem 6 *Let $\pi_{\beta *}$ be the β-discounted equilibrium point and let $\pi_* := \lim_{\beta \to 1^-} \pi_{\beta *}$. If the stochastic game is irreducible, then $\lim_{\beta \to 1^-} V_\beta^k(\pi_{\beta *}) = V_\alpha^k(\pi_*)$,*
and π_* is an average reward equilibrium point.

The only thing that requires proof in Theorem 6 is that π_* is the equilibrium point for average reward games. Because $\pi_{\beta*}$ is the β-discounted equilibrium point, $V_\beta^k(\pi_{\beta*}) \geq V_\beta^k(\pi_{\beta*}^{-k}, \pi^k)$. Taking limit of β to 1^-, we get $V_\alpha^k(\pi_*) \geq V_\alpha^k(\pi_*^{-k}, \pi^k)$. Hence π_* is an average reward equilibrium point. Since every nonzero-sum discounted stochastic game possesses at least one equilibrium point in stationary strategies, Theorem 6 implies that equilibrium points exist in every average reward irreducible stochastic game. But lack of knowledge of the structure of the equilibrium points makes their computation difficult. Theorem 1 indicated the structure of the equilibrium points in discounted reward stochastic games. We exploit this to gain insight into the structure of the equilibrium points for average reward stochastic games. This is achieved through a Laurent series expansion approach presented below.

4.3.1 Laurent Series Expansion

In Section 2, we referred to the use of Laurent series expansion in obtaining average reward optimality function in terms of gain and bias from the discounted reward optimality function for MDPs. For stochastic games, we use the same approach instead of the Puiseux series used in [8], since the elements of Puiseux series are not easily amenable to physical interpretation. The propositions and theorem in this section are adopted from Puterman's book [2], some of which are extended for games.

Define *bias* (h) of irreducible stochastic games for player k and starting state $s \in S$ as

$$h^k(s) = E_s\{\sum_{t=0}^{\infty}[r^k(t) - V_\alpha^k(t)]\}.$$

The term V_α^k is called the gain for player k, which represents the average reward per period for a system in steady state, also referred to as the *stationary reward*.

Note that the fundamental matrix H_P in (4.1) is the Cesaro limit of $\sum_{i=0}^{k}(P^i - Q)$. Based on this proposition, we can write $h^k = H_P r^k$. Let $\beta = (1+\rho)^{-1}$. We can interpret ρ as the interest rate. Then we can write $V_\beta^k = (1-\beta)(I-\beta P)^{-1} r^k = (1-\beta)(1+\rho)(\rho I + [P-I])^{-1} r^k$. A matrix decomposition method can be used to derive Laurent series expansion for $(\rho I + [P-I])^{-1}$.

Proposition 7 *For* $0 < \rho < \sigma(I-P)$, $(\rho I + [P-I])^{-1} = \rho^{-1}Q + \sum_{n=0}^{\infty}(-\rho)^n H_P^{n+1}$.

Based on the above proposition, the following Laurent series expansion of V_β^k is obtained.

Theorem 8 $V_\beta^k = (1-\beta)(1+\rho)[\rho^{-1} y_{-1}^k + \sum_{n=0}^{\infty} \rho^n y_n^k]$
where $y_{-1}^k = Qr^k = V_\alpha^k, y_0^k = H_P r^k = h^k$, *and* $y_n^k = (-1)^n H_P^{n+1} r^k$, *for* $n = 1, 2, \ldots$.

For average reward criterion, it will suffice to use the following truncated Laurent series expansion of V_β^k.

Theorem 9 *If reward* r^k *is bounded for all player* k, *then*

$$V_\beta^k = V_\alpha^k + (1-\beta)h^k + (1-\beta)f^k(\beta) \tag{4.4}$$

where $f^k(\beta)$ *denotes a vector which converges to zero as* $\beta \uparrow 1$.

Expressing V_β^k in Theorem 8 in terms of β and adding and substracting h^k, we obtain $V_\beta^k = V_\alpha^k + (1-\beta)h^k + (1-\beta)\left[\frac{(1-\beta)}{\beta}h^k + \frac{1}{\beta}\sum_{n=0}^{\infty}(-1)^n(\frac{1-\beta}{\beta})^n y_n^k\right]$. The last term within square brackets converges to 0 as $\beta \to 1$ and is written as $f^k(\beta)$ in (9).

Theorem 9 establishes a relationship of the average reward value with the discounted reward value and the bias. In what follows, with the above relationship, we develop auxiliary matrix games for average reward irreducible stochastic games.

4.3.2 Nash Equilibrium for Average Reward Irreducible Stochastic Games

The *Nash equilibrium point* for average reward stochastic games is defined as $V_\alpha^k(\pi_*) \geq V_\alpha^k(\pi_*^{-k}, \pi^k)$ for all $k = 1, 2, \ldots, n$. It was discussed earlier that nonzero-sum discounted stochastic games possess at least one Nash equilibrium point $\pi_{\beta*}$. According to Theorem 1, the equilibrium points $\pi_{\beta*}$ are equivalent to the ones defined in inequality (3.5). Since the left side of inequality (3.5) is $V_\beta^k(s, \pi_{\beta*})$, we can transform the inequality by changing sides as

$$0 \geq r^k(\pi_{\beta*}^{-k}, \pi^k) + \frac{1}{1-\beta}(\beta P(\pi_{\beta*}^{-k}, \pi^k) - I)V_\beta^k(\pi_{\beta*}).$$

By substituting $V_\beta^k(\pi_{\beta*})$ from (4.4) into the above inequality, we obtain

$$0 \geq r^k(\pi_{\beta*}^{-k}, \pi^k) + \frac{1}{1-\beta}(\beta P(\pi_{\beta*}^{-k}, \pi^k) - I)\left[V_\alpha^k(\pi_{\beta*}) + (1-\beta)h^k(\pi_{\beta*}) + (1-\beta)f^k(\beta)\right]. \tag{4.5}$$

Applying the limit $\beta \uparrow 1^-$ on (4.5), we obtain the following result.

Proposition 10 *The equilibrium points π_* for average reward irreducible stochastic games satisfy the following inequality:*

$$h^k(\pi_*) \geq r^k(\pi_*^{-k}, \pi^k) - V_\alpha^k(\pi_*) + P(\pi_*^{-k}, \pi^k)h^k(\pi_*). \tag{4.6}$$

Proof Let $\pi_* = \lim_{\beta \to 1^-} \pi_{\beta*}$. Applying limit to (4.5), we get

$$\begin{aligned} 0 \quad &\geq \quad \lim_{\beta \to 1^-} \{r^k(\pi_{\beta*}^{-k}, \pi^k) + \frac{1}{1-\beta}(\beta P(\pi_{\beta*}^{-k}, \pi^k) - I) * \\ &\quad \left[V_\alpha^k(\pi_{\beta*}) + (1-\beta)h^k(\pi_{\beta*}) + (1-\beta)f^k(\beta))\right]\} \\ 0 \quad &\geq \quad r^k(\pi_*^{-k}, \pi^k) + \lim_{\beta \to 1^-} \frac{1}{1-\beta}\left(\beta P(\pi_{\beta*}^{-k}, \pi^k) - I\right)V_\alpha^k(\pi_{\beta*}) + \left(P(\pi_*^{-k}, \pi^k) - I\right)h^k(\pi_*) \end{aligned}$$

Since $V_\alpha^k(\pi_*)$ is constant for each state in irreducible average reward games, the limit of the second item on the r.h.s becomes $-V_\alpha^k(\pi_*)$. From the above, we get the desired inequality (4.6). According to Theorem 6, π_* in (4.6) is the Nash equilibrium point for average reward stochastic games. ■

Similarly, by substituting $V_\beta^k(\pi_{\beta*})$ from (4.4) into the discounted value equation (3.2), the value equation for $h^k(\pi_*)$ is obtained as

$$h^k(\pi_*) = r^k(\pi_*) - V_\alpha^k(\pi_*) + P(\pi_*)h^k(\pi_*).$$

We call $V_\alpha^k(\pi_*)$ the *gain equilibrium value*, and $h^k(\pi_*)$ the *bias equilibrium value*. Now we state the main theorem for the average reward irreducible stochastic games.

Theorem 11 *The following assertions are equivalent:*
(i) π_* is an equilibrium point in the average reward irreducible stochastic game with bias equilibrium value $h^k(\pi_*)$ and gain equilibrium value $V_\alpha^k(\pi_*)$ for $k = 1, 2, \cdots, n$.
(ii) For each $s \in S$, the strategy set $\pi_*(s)$ constitutes an equilibrium point in the static n-dimensional matrix game $(R^1(s), \ldots, R^n(s))$ with bias equilibrium value $h^k(s, \pi_*)$ and gain equilibrium value $V_\alpha^k(\pi_*)$ for $k = 1, 2, \cdots, n$. For $(a^1, \ldots, a^n) \in \prod_{i=1}^n A^i(s)$, the corresponding entry of $R^k(s)$ is given by

$$R^k(s, a^1, \ldots, a^n) = r^k(s, a^1, \ldots, a^n) - V_\alpha^k(\pi_*) + \sum_{s' \in S} p(s'|s, a^1, \ldots, a^n)h^k(s', \pi_*). \quad (4.7)$$

Proof According to Proposition 10, the equilibrium points π_* in average reward irreducible stochastic games satisfy the inequality (4.6). For n-dimensional matrix

game $R^k(s)$, the value for a strategy π is defined as:

$$\begin{aligned} Val[R^k(s),\pi] &= \sum_{a^1=1}^{m^1(s)} \cdots \sum_{a^n=1}^{m^n(s)} \pi^1(s,a^1)\ldots\pi^n(s,a^n)R^k(s,a^1,\ldots,a^n) \\ &= r^k(s,\pi) - V_\alpha^k(\pi_*) + \sum_{s'\in S} p(s'|s,\pi)h^k(s',\pi_*). \end{aligned}$$

The equilibrium points π_* satisfy $Val[R^k(s),\pi_*] \geq Val[R^k(s),\pi_*^{-k},\pi^k]$. Then we can get,

$$\begin{aligned} r^k(s,\pi_*) - V_\alpha^k(\pi_*) &+ \sum_{s'\in S} p(s'|s,\pi_*)h^k(s',\pi_*) \geq r^k(s,\pi_*^{-k},\pi^k) - V_\alpha^k(\pi_*) \\ &+ \sum_{s'\in S} p(s'|s,\pi_*^{-k},\pi^k)h^k(s',\pi_*), \end{aligned}$$

which is the same as inequality(4.6). So the two assertions are equivalent. ■

For matrix games (4.7), the gain equilibrium values $V_\alpha^k(\pi_*)$ is constant (scalar). Adding or subtracting a constant to every entry in any player's matrix will not alter its Nash equilibrium policies. Altman et al. [9] also gave a structure of equilibrium points for average reward stochastic games with countable states. It was shown in their paper that an equilibrium point π_* satisfies the following optimality equations:

$$V_\alpha^k(s) = max_{a\in A^k(s)} \sum_{s'} p(s'|s,\pi_*^{-k},a)V_\alpha^k(s'), \tag{4.8}$$

$$h^k = \max_{a\in A^{k,*}(s)} \left[r^k(s,\pi_*^{-k},a) - V_\alpha^k(s) + \sum_{s'} p(s'|s,\pi_*^{-k},a)h^k(s') \right],$$

where $A^{k,*}(s) = \{a \in A^k(s) | V_\alpha^k(s) = \sum_{s'\in S} p(s'|s,\pi_*^{-k},a)V_\alpha^k(s')\}$.

The above structure is not limited to irreducible stochastic games, and hence requires an additional constraint (4.8) to account for the dependence of V_α^k on the

states. The structure of our result in Theorem 11 differs from the above owing to its matrix game form. This was motivated by our intention to benefit from the available computational methods to solve matrix games.

CHAPTER 5

LEARNING AVERAGE REWARD STOCHASTIC GAMES

Reinforcement learning (RL) has been successful at finding optimal control policies for a single agent operating in a stationary environment [29][30]. In recent years RL has been applied to discounted reward stochastic games [11][13][15]. In this chapter, a reinforcement learning algorithm for average reward stochastic games is presented. Average reward RL algorithms are commonly referred to as the R-learning algorithms.

5.1 A Nash R-learning Algorithm

We consider a multiplayer irreducible stochastic game on a finite state space and a finite action space. We are interested in finding stationary Nash equilibrium policies.

5.1.1 Optimality Equation

Recall that Nash equilibrium policies exist under the conditions stated in Theorem 6. The structure of Nash equilibria is given in (4.7), which is similar to Bellman's optimality equation for average reward MDP. Since it can be shown (from

the argument used in the proof of Theorem 11) that $Val[R^k(s), \pi_*] = h^k(s, \pi_*)$, we can rewrite (4.7) as follows,

$$R^k(s, a^1, \ldots, a^n) = r^k(s, a^1, \ldots, a^n) - V_\alpha^k(\pi_*) + \sum_{s' \in S} p(s'|s, a^1, \ldots, a^n) Val[R^k(s'), \pi_*], \tag{5.1}$$

where π_* is the Nash equilibrium point for matrix game $R(\cdot)$, and

$$Val[R^k(s), \pi_*] = r^k(s, \pi_*) - V_\alpha^k(\pi_*) + \sum_{s' \in S} p(s'|s, \pi_*) Val[R^k(s'), \pi_*]. \tag{5.2}$$

The fixed point of equation (5.1) is the auxiliary matrices for the stochastic game, whose value is the Nash equilibrium bias value $h^k(\pi_*)$.

5.1.2 Updating Rules

In most real applications, the transition probabilities are extremely difficult to obtain. Hence we use adaptive learning mechanisms such as simulation based reinforcement learning to intelligently build $R^k(\cdot)$ matrices. There exist three main classes of reinforcement learning (RL) mechanisms: value iteration based methods such as Q-learning and R-learning algorithms, policy iteration based methods, and combination of value and policy iteration based methods referred to as the Actor-Critic algorithm [11]. In this dissertation, we use a value iteration based approach. In this approach that uses value function based RL algorithms [40], an estimate of the optimal value function is built gradually from the decision maker's experiences, and these estimates are often used for control of the learning process. For stochastic games, we extend a value function based two time scale R-learning algorithm, discussed in Section 2, to a multi-agent scenario. It is considered that each player learns not only her own R values but also other players' R values based on the observa-

tion of all other players' immediate rewards and chosen actions. Hence the auxiliary matrices $R(\cdot)$ are developed simultaneously by each player using the learning scheme presented below. Let at the t^{th} stage, the system state is s and the action combination $(a^1, \ldots, a^n)$ is chosen by the players. Also let the system state be s' at the $(t+1)^{th}$ stage. The $R(\cdot)$ matrix and the average reward value for the k^{th} player are updated as follows

$$R^k_{t+1}(s, a^1, \ldots, a^n) = R^k_t(s, a^1, \ldots, a^n) +$$
$$\alpha_t \left[r^k(s, s', a^1, \ldots, a^n) - G^k_t + Val[R^k_t(s'), \pi_{t*}] - R^k_t(s, a^1, \ldots, a^n) \right], \qquad (5.3)$$

and

$$G^k_{t+1} = G^k_t + \beta_t \left(\frac{(r^k(s, s', a^1, \ldots, a^n) + tG^k_t)}{t+1} - G^k_t \right). \qquad (5.4)$$

In the above updating scheme, π_{t*} represents a Nash equilibrium policy of the matrix game $R^t(s')$ at the t^{th} stage. The element $Val[R^k_t(s'), \pi_{t*}]$ reprerents the Nash equilibrium value for player k in state s' which is obtained as

$$Val[R^k_t(s'), \pi_{t*}] = \sum_{a^1, \ldots, a^n} \pi^1_{t*}(s', a^1) \ldots \pi^n_{t*}(s', a^n) R^k(s', a^1, \ldots, a^n).$$

Thus (5.3) and (5.4) present the two time scale iteration scheme, in which the parameters α_t and β_t represent the step sizes at time t.

5.1.3 Assumptions

The simulation process on which the learning scheme is applied is assumed to satisfy the following assumption.

Assumption 1 Every state and every action of the players are visited infinitely often.

In the above two time scale learning scheme, two diminishing step sizes α_t and β_t are needed to update the $R_t(\cdot)$ matrices and the average reward G_t. We make the following assumption for α_t and β_t.

Assumption 2 $\{\alpha_t\} \subset (0, \infty)$, with $\sum_t \alpha_t = \infty$, $\sum_t \alpha_t^2 \leq \infty$.
$\{\beta_t\} \subset (0, \infty)$, with $\sum_t \beta_t = \infty$, $\sum_t \beta_t^2 \leq \infty$. Also, $\beta_t = O(\alpha_t)$, which denotes $\lim_{t\to\infty} \frac{\beta_t}{\alpha_t} = 0$.

The assumption $\beta_t = O(\alpha_t)$ made above implies that the updating of G_t in iteration (5.4) proceeds at a "slower rate" than R_t in the iteration (5.3). Hence, in the analysis of R_t, G_t can be viewed as quasistatic.

It can be seen from (5.3) that updating of $R_t(\cdot)$ matrices requires the Nash equilibrium value $Val[R_t^k(s'), \pi_{t*}]$ which in turn requires the Nash equilibrium policy π_{t*} for stage matrix game $R_t(s')$. The following assumption is needed for the Nash equilibrium policy π_{t*}.

Assumption 3 A Nash equilibrium policy π_{t*} for any n-dimensional stage matrix game $[R_t^1(\cdot), \ldots, R_t^n(\cdot)]$ satisfies one and only one of the following properties.

- The Nash equilibrium is global optimal for which $Val[R_t^k(s), \pi_{t*}] \geq Val[R_t^k(s), \pi]$, $k = 1, \ldots, n, \forall s \in S, \forall \pi$.

- The Nash equilibrium is a saddle point or adversarial equilibrium for which $Val[R_t^k(s), \pi_{t*}] \leq Val[R_t^k(s), (\pi^{-k}, \pi_{t*}^k)]$, $k = 1, \ldots, n$, $\forall s \in S$, $\forall \pi$. This property implies that an agent receives a higher payoff when other agents deviate from the Nash equilibrium strategy.

The assumption that the Nash equilibrium for each stage matrix game can only be either a global optimal point or a saddle point guarantees that there is only one unique Nash equilibrium value [38][41]. Uniqueness of the Nash equilibrium value implies a unique auxiliary matrix game, which again suggests that the auxiliary matrices

learned by the players converge to the same matrices. The above assertion will be further addressed in our convergence analysis.

5.2 Stochastic Approximation and ODE Method

Reinforcement learning algorithms such as Q-learning and R-learning are iterative algorithms and also inherently of the stochastic approximation type. We try to find the optimal value of MDP problems, but the value function is not known due to unknown transition probabilities and reward functions. But through simulation or experiment, we are able to take "noisy" measurements at any desired value. Stochastic iterative algorithms can operate in the presence of noise which differ from deterministic iterative algorithms.

5.2.1 Stochastic Approximation Standard Form

We are usually interested in solving a system of equations of the form

$$H(r) = r. \tag{5.5}$$

Bellman optimality equations and the optimality equation in section 5.1.1 are of this form. The function of H is not known precisely, but we can have access to a random variable s of the form $s = H(r) + w$, where w is a random noise term. The original work in recursive stochastic algorithms was by Robbins and Monro, who developed and analyzed a recusive procedure for finding the fixed point of (5.5) [42][43]. The *Robbins-Monro* stochastic approximation algorithm is of the form

$$r_{n+1} = (1 - \alpha_n)r_n + \alpha_n(H(r_n) + w_n), \tag{5.6}$$

The stepsize is denoted as α_n, which should satisfy the following two assumptions with probability 1

- $\sum_{n=0}^{\infty} \alpha_n^2 < \infty$,
- $\sum_{n=0}^{\infty} \alpha_n = \infty$.

The first assumption implies that the stepsize decreases to zero. So the effect of the noise becomes vanishingly small. On the other hand, the stepsize cannot be allowed to decrease too quickly. Because if the algorithm starts at the point, which is far away from the desired solution $r*$, the algorithm will never succeed in getting to $r*$. This motivates the second assumption.

A simple example is considered to illusrate this idea. We want to calculate the mean of an IID random variable. Let x_t be the IID random variable with unknown mean μ. The iteration should be

$$r_{t+1} = (1 - \alpha_t)r_t + \alpha_t x_t$$

if we set the stepsize $\alpha_t = \frac{1}{t}$ we know from the strong law of large numbers that:

$$r_t \to \mu \quad a.s.$$

5.2.2 Ordinary Diffenrential Equation (ODE) Framework

The existing approaches to convergence analysis include smooth potential function such as the stochastic gradient algorithm [44], contraction map [32][40], and ODE methods [42][43][45][46][33][47][36]. Our focus here is to analyze the asymptotic properties of the stochastic approximation process involves showing that the asymptotic paths are those of the ODE determined by the "mean" dynamics of the

algorithm. The solutions of such ODEs will appear as "limits" of the paths of stochastic approximation algorithms. The ODE for (5.6) is

$$\dot{r}_t = H(r) - r. \tag{5.7}$$

The *critical point* (alteratively called equilibrium point, stable point or fixed point) is defined as the vector $r*$ such that $H(r*) - r* = 0$ in (5.7).

Let $r*$ be the fixed point of (5.7). Let $B(r*, R)$ be the ball (open sphere) of radius R centered around $r*$. Then $r*$ is said to be an *asymptotically stable critical point* if and only if for all values of $\epsilon > 0$, there exists a scalar quantity $R_\epsilon > 0$ such that if $r_0 \in B(r*, R_\epsilon)$, then $r_t \in B(r*, R_\epsilon)$ for all t [48], and

$$\lim_{t\to\infty} r_t = r*.$$

5.2.3 Exsitence of a Solution to an ODE

The asymptotic properties of ODE will tell us what we wish to know about the tail behavior of r_n. The exsitence of a solution to an ODE was introduced in [42]. The stochastic approximation process r_n in (5.6) will be piecewise linearly interpolated into a continuous parameter process with interpolation intervals α_n.

Define $t_n = \sum_{i=0}^{n-1} \alpha_i$. And define the interpolations $r^0(\cdot)$ and $W^0(\cdot)$ of r_n and w_n, respectively, by

$$\begin{aligned} r^0(t_n) &= r_n, \\ r^0(t) &= \frac{t_{n+1} - t}{\alpha_n} r_n + \frac{t - t_n}{\alpha_n} r_{n+1}, t \in (t_n, t_{n+1}), \\ W^0(t_n) &= \sum_{i=0}^{n-1} \alpha_i w_i, \end{aligned}$$

$$W^0(t) = \frac{t_{n+1}-t}{\alpha_n}W^0(t_n) + \frac{t-t_n}{\alpha_n}W^0(t_n), t \in (t_n, t_{n+1}),$$

Let $\bar{r}^0(t) = r_n$ in $[t_n, t_{n+1})$, a piecewise constant right continous interpolation. Now we may write

$$r^0(t) = r^0(0) + \int_0^t H(\bar{r}^0(s))ds + W^0(t).$$

Then we define a sequence of left shifts which bring the "asymptotic part" of r_n to a neiborhood of the time origin.

$$r^n(t) = \begin{cases} r^0(t+t_n), & t \geq -t_n \\ r_0, & t \leq -t_n \end{cases}$$

$$W^n(t) = \begin{cases} W^0(t+t_n) - W^0(t_n), & t \geq -t_n \\ -W^0(t_n), & t \leq -t_n \end{cases}$$

Now we get the following

$$\begin{aligned} r^n(t) &= r^n(0) + \int_0^t H(r^n(s))ds + W^n(t) + e^n(t). && \text{t}\geq -t_n \\ &= r_0. && \text{t}\leq -t_n \end{aligned}$$

where $e^n(t)$ is a function which tends to zero as $n \to \infty$. If $W^n(\cdot) \to 0$ uniformly on finite time intervals, the sequence $r^n(\cdot)$ is bounded and equicontinuous on $(-\infty, \infty)$. Extract a convergent subsequence, also indexed by n, and denote the limit by $r(\cdot)$. Then $r(\cdot)$ satisfies

$$\dot{r} = H(r), \quad r(\cdot) \textit{ bounded on } (-\infty, \infty).$$

If this ODE has an asymptotically stable point $r*$, $r(t)$ is identically equal to $r*$. This implies that $r_n \to r*$.

5.2.4 Stochastic Approximation with Two Time Scales

There are some situations to consider a two time scale stochastic approximation. In R-learning, each state-action combined R value is updated on a time scale. But the updating of R values requires a further averaging of the cost, which is updated on a different time scale. A typical two time scale stochastic approximation algorithm given by, respectively, d and l dimensional coupled iterations has the following form

$$X(n+1) = X(n) + a(n) \times (F(X(n), Y(n)) + M(n+1)), \tag{5.8}$$

$$Y(n+1) = Y(n) + b(n) \times (G(X(n), Y(n)) + N(n+1)). \tag{5.9}$$

Borkar [49] discussed the convergence conditions by using perturbed ODE method. Here we list out the conditions:

- $F : \Re^{d+l} \to \Re^d$, $G : \Re^{d+l} \to \Re^l$ are Lipschitz,
- $$\sum_n a(n) = \sum_n b(n) = \infty$$
  $$\sum_n a(n)^2 = \sum_n b(n)^2 < \infty, a(n) = o(b(n)).$$
- For $\mathcal{F}_n = \sigma(X(m), Y(m), M(m), N(m), m \leq n)$, $n \geq 0$, $(M(n), \mathcal{F}_n)$ and $(N(n), \mathcal{F}_n)$ are sequences of random variables satisfying

  $$\sum_n a(n)M(n), \quad \sum_n b(n)N(n) < \infty \quad a.s.$$

- For each $x \in \Re^d$, The ODE

  $$\dot{y}(t) = G(x, y(t))$$

has a unique global asymptotically stable equlibrium $\lambda(x)$ such that $\lambda : \Re^d \rightarrow \Re^l$ is Lipschitz.

- The ODE

$$\dot{x}(t) = F(x(t), \lambda(x(t)))$$

has a unique global asymptotically stable equlibrium $x*$.

The condition $a(n) = o(b(n))$ implies that the first iteration (5.8) proceeds at a "slower rate" than the second (5.9), so Intuitively the component $y(\cdot)$ will see the slow component as quasi-static while the slow component sees the fast one as "equilibrated".

The final result from Borkar is shown below

Theorem 12 *The iterations (5.8) and (5.9) converge to* $(x*, \lambda(x*))$ *a.s. on the set* $Q = \sup_n X(n) < \infty, \sup_n Y(n) < \infty$.

5.3 Nash R-learning Algorithm Convergence Analysis

The convergence of R_t and G_t given by (5.3) and (5.4) is the consequence of stochastic approximations. The proposed Nash R-learning algorithm like most learning algorithms involves fixed-point computation. Our analysis is inspired by a similar analysis by Abounadi [33] and Gosavi [36] based on Bokar's approach to stochastic approximation with two time scales [49].

5.3.1 Two Time Scales Stochastic Approximation Form

The standard form of stochastic approximation with two time scales as in our algorithm has the following structure. Let $H(R_t^k, G_t^k)$ denote the expected value of R_{t+1}^k given the history up to time t. But, we can only obtain observed values

that include noise. Let the noise be denoted by M_t^k. Similarly, $F(G_t^k)$ denotes the expected average reward G_{t+1}^k given the history, and N_t^k denotes the corresponding noise. Then we can have that

$$R_{t+1}^k = R_t^k + \alpha_t(H(R_t^k, G_t^k) - R_t^k + M_t^k), \tag{5.10}$$

$$G_{t+1}^k = G_t^k + \beta_t(F(G_t^k) - G_t^k + N_t^k). \tag{5.11}$$

Using $E(\cdot)$ to denote expected value we can write the following.

$$\begin{aligned} H(R_t^k, G_t^k) &= E\left(r(s, s', a^1, \ldots, a^n) - G_t^k + Val[R_t^k(s'), \pi_{t*}]\right). \\ M_t^k &= r(s, s', a^1, \ldots, a^n) - E(r(s, s', a^1, \ldots, a^n)) + G_t^k - E(G_{t))+}^k \\ & \quad Val[R_t^k(s'), \pi_{t*}] - E\left(Val[R_t^k(s'), \pi_{t*}]\right). \\ F(G_t^k) &= E\left(\frac{r(s, s', a^1, \ldots, a^n) + tG_t^k}{t+1}\right). \\ N_t^k &= \frac{r(s, s', a^1, \ldots, a^n) - E(r(s, s', a^1, \ldots, a^n)) + tG_t^k - E(tG_t^k)}{t+1}. \end{aligned} \tag{5.12}$$

$F(G_t^k)$ is a linear function, so it is Lipschitz continuous. The Lipschitz continuity for $H(\cdot)$ will be discussed later.

5.3.2 Noises Analysis

Let $\mathcal{F}(t)$ denote the history of the algorithm up to and including the point at which the step sizes α_t and β_t for the t^{th} iteration are selected, but just before the noise terms M_t^k and N_t^k are generated. That is,

$$\mathcal{F}(t) = \sigma\left(R_0^k, G_0^k, \alpha_0, \beta_0, M_0^k, N_0^k, \ldots, M_{t-1}^k, N_{t-1}^k, R_t^k, G_t^k, \alpha_t, \beta_t\right).$$

We note that for all t and k,

$$E[M_t^k|\mathcal{F}(t)] = 0, \tag{5.13}$$

$$E[N_t^k|\mathcal{F}(t)] = 0. \tag{5.14}$$

Based on the above results, $\sum_{i=0}^{t} \alpha_i M_i^k$ and $\sum_{i=0}^{t} \beta_i N_i^k$ are the $\mathcal{F}(t)$-martingales. So the noise terms M_t^k and N_t^k are "martingale difference" sequences. Now $E[(M_t^k)^2|\mathcal{F}(t)]$ is the conditional variance of the noise term M_t^k. Using $Var(\cdot|\mathcal{F}(t))$ to denote the conditional variance, we can write from (5.12) that:

$$E[(M_t^k)^2|\mathcal{F}(t)] = Var(r(s, s', a^1, \ldots, a^n)|\mathcal{F}(t)) + Var(Val[R_t^k(s'), \pi_{t*}]|\mathcal{F}(t)). \tag{5.15}$$

Since the reward $r(s, s', a^1, \ldots, a^n)$ is bounded, the conditional variance is bounded. Let C denote the bound of $Var(r(s, s', a^1, \ldots, a^n)|\mathcal{F}(t))$. The conditional variance of $Val[R_t^k(s'), \pi_{t*}]$ given $\mathcal{F}(t)$ is bounded above by the largest possible value that this random variable could take. Then we can write (5.15) as:

$$E[(M_t^k)^2|\mathcal{F}(t)] \leq C + \|(R_t^k)^2\|. \tag{5.16}$$

Similarly, we can get:

$$E[(N_t^k)^2|\mathcal{F}(t)] \leq C.$$

From the above, we can see that $sup_t E[(N_t^k)^2|\mathcal{F}(t)] < \infty$. If R_t is a bounded sequence, then $sup_t E[(M_t^k)^2|\mathcal{F}(t)] < \infty$. These imply that $\sum_{i=0}^{t} \alpha_i M_i^k$ and $\sum_{i=0}^{t} \beta_i N_i^k$ converge with probability one. Then $\sum_{i=0}^{t+1} \alpha_i M_i^k - \sum_{i=0}^{t} \alpha_i M_i^k \to 0$ and $\sum_{i=0}^{t+1} \beta_i N_i^k - \sum_{i=0}^{t} \beta_i N_i^k \to 0$ a.s. as $t \to \infty$. Thus the errors due to the noise in the two approximations (5.10) and (5.11) become asymptotically negligible.

In that case, the time asymptotic part of recursion R_t^k (5.10) tracks the ODE

$$\dot{R}_t^k = H(R_t^k, G) - R_t^k, \tag{5.17}$$

where G is treated as a fixed parameter, since $\beta_t = o(\alpha_t)$ [49].

Using the same argument used for R_t^k, we can say that the time asymptotic parts of G_t^k (5.11) tracks the O.D.E

$$\dot{G}_t^k = F(G_t^k) - G_t^k. \tag{5.18}$$

The set of equilibrium points R^{k*} and G^{k*} of these two ODEs respectively are precisely the corresponding fixed points of $H(\cdot)$ and $F(\cdot)$, which are $H(R^{k*}, G) = R^{k*}$ and $F(G^{k*}) = G^{k*}$. The ODE (5.18) is independent of ODE (5.17). Hence we consider ODE (5.18) first. We present the following lemmas that are based on the results of [50], and [33].

Lemma 13 *The ODE (5.18) has a unique global asymptotically stable equilibrium point G^*.*

Since $F(G^k)$ is a non-expansive mapping, and the fixed point set is not empty, the solution of the differential equation converges to an asymptotically stable equilibrium point.

We have shown earlier that the Nash equilibrium point exists for an average reward irreducible stochastic game (Theorems 6 and 11), and G functions as a scalar in the matrix game, and it does not affect the calculation of Nash equilibrium policies. Though the convergence of G is required, it is not essential that the converged value is the optimal average cost. So for any fixed G value, fixed point R^{k*} exists. And

assumption 3 assures the uniqueness of the fixed point. In next section, we show $H(\cdot)$ is non-expansive with respect to the sup-norm. Based on the result from [50], we get

Lemma 14 *The O.D.E. (5.17) has a global asymptotically stable equilibrium $R^*(G)$.*

5.3.3 Boundedness and Convergence

From (5.4), boundedness of $r^k(\cdot)$ implies that G_t^k is bounded. We now address the boundedness of R_t^k. Our approach is based on the following theorem by [45].

Theorem 15 *Given the properties (5.13) and (5.16) of M_t^k, if $H(R_t)$ is Lipschitz. For any $\mu > 0$, the scaled function is defined by: $H_\mu(R) = h(\mu R)/\mu, R \in \Re^d$. And there exists a function $H_\infty : \Re^d \to \Re^d$ such that*

$$\lim_{\mu \to \infty} H_\mu(R) = H_\infty(R),$$

Further more, the origin in $\Re^d$ is an asymptotically stable equilibrium for the o.d.e. $\dot{R}_t^k = H_\infty(R_t^k) - R_t^k$. Then the sequence R_t^k is bounded a.s.

For this, consider H^0 defined by

$$H^0(R^k(s, a^1, \ldots, a^n) = E\left(Val[R_t^k(s'), \pi_{t*}]\right).$$

Because $Val[\mu * R] = \mu * Val[R]$, for $\mu > 0$, then

$$\lim_{r \to \infty} \frac{H(rR^k, G)}{r} = H^0(R^k).$$

And the O.D.E. $\dot{R}_t^k = H^0(R_t^k) - R_t^k$, has the origin as the globally asymptotically stable equilibrium, which is a special case of Lemma 14 with $r(\cdot) = 0$. We also need to study the Lipschitz condition of $H(\cdot)$.

For any two n-dimensional matrices R_t^k and $R_t^{'k}$,

$$H(R_t^k(s,a^1,\ldots,a^n)) - H(R_t^{'k}(s,a^1,\ldots,a^n)) =$$
$$\sum_{s'} p(s'|s,a^1,\ldots,a^n)\left(Val[R_t^k(s'),\pi_{t*}] - Val[R_t^{'k}(s'),\pi_{t*}^{'}]\right), \qquad (5.19)$$

where π_{t*} and $\pi_{t*}^{'}$ are Nash equilibrium points for $R_t^k(s)$ and $R_t^{k'}(s)$. Based on Assumption 3, there are two possible cases as follows.

Case 1: π_{t*} and $\pi_{t*}^{'}$ are adversarial equilibria. If $H(R_t^k(s,a^1,\ldots,a^n)) > H(R_t^{'k}(s,a^1,\ldots,a^n))$, then we have from (5.19) that

$$\begin{aligned}
& H(R_t^k(s,a^1,\ldots,a^n)) - H(R_t^{'k}(s,a^1,\ldots,a^n)) \\
\leq\ & \sum_{s'} p(s'|s,a^1,\ldots,a^n)\left(Val[R_t^k(s'),\pi_{t*}] - Val[R_t^{'k}(s'),(\pi_{t*}^{'-k},\pi_{t*}^k)]\right) \qquad (5.20)\\
\leq\ & \sum_{s'} p(s'|s,a^1,\ldots,a^n)\left(Val[R_t^k(s'),(\pi_{t*}^{'-k},\pi_{t*}^k)] - Val[R_t^{'k}(s'),(\pi_{t*}^{'-k},\pi_{t*}^k)]\right) \qquad (5.21)\\
\leq\ & \sum_{s'} p(s'|s,a^1,\ldots,a^n) \sum_{a^1,\ldots,a^n} \pi_{t*}^{'1}(s',a^1)\ldots\pi_{t*}^k(s',a^k)\ldots\pi_{t*}^{'n}(s',a^n) \\
& \max_{(a^1,\ldots,a^n)\in A(s')} \mid R_t^k(s',a^1,\ldots,a^n) - R_t^{'k}(s',a^1,\ldots,a^n) \mid \\
=\ & \sum_{s'} p(s'|s,a^1,\ldots,a^n) \max_{(a^1,\ldots,a^n)\in A(s')} \mid R_t^k(s',a^1,\ldots,a^n) - R_t^{'k}(s',a^1,\ldots,a^n) \mid
\end{aligned}$$

The first inequality (5.20) is based on the property of Nash equilibrium policy $\pi_{t*}^{'}$. The inequality (5.21) is based on the property of adversarial equilibrium π_{t*}.

If $H(R_t^k(s,a^1,\ldots,a^n)) < H(R_t^{'k}(s,a^1,\ldots,a^n))$, the

$$\begin{aligned}
& H(R_t^{'k}(s,a^1,\ldots,a^n)) - H(R_t^k(s,a^1,\ldots,a^n)) \\
\leq\ & \sum_{s'} p(s'|s,a^1,\ldots,a^n)\left(Val[R_t^{'k}(s'),(\pi_{t*}^{-k},\pi_{t*}^{'k})] - Val[R_t^k(s'),(\pi_{t*}^{-k},\pi_{t*}^{'k})]\right)
\end{aligned}$$

and the rest of the proof is similar to the above. Thus

$$| H(R_t^k(s,a^1,\ldots,a^n)) - H(R_t^{'k}(s,a^1,\ldots,a^n)) | \leq \sum_{s'} p(s'|s,a^1,\ldots,a^n) \| R_t^k(s') - R_t^{'k}(s') \|$$

.

Case 2: π_{t*} and $\pi_{t*}^{'}$ are global optimal points, if $H(R_t^k(s,a^1,\ldots,a^n)) > H(R_t^{'k}(s,a^1,\ldots,a^n))$, then we can write that

$$\begin{aligned}
& H(R_t^k(s,a^1,\ldots,a^n)) - H(R_t^{'k}(s,a^1,\ldots,a^n)) \\
\leq & \sum_{s'} p(s'|s,a^1,\ldots,a^n) \left(Val[R_t^k(s'),\pi_{t*}] - Val[R_t^{'k}(s'),\pi_{t*}] \right) \qquad (5.22) \\
= & \sum_{s'} p(s'|s,a^1,\ldots,a^n) \sum_{a^1,\ldots,a^n} \pi_{t*}^1(s',a^1)\ldots\pi_{t*}^n(s',a^n) \\
& \left(R_t^k(s',a^1,\ldots,a^n) - R_t^{'k}(s',a^1,\ldots,a^n) \right) \\
\leq & \sum_{s'} p(s'|s,a^1,\ldots,a^n) \max_{(a^1,\ldots,a^n)\in A(s')} | R_t^k(s',a^1,\ldots,a^n) - R_t^{'k}(s',a^1,\ldots,a^n) | .
\end{aligned}$$

The first inequality (5.22) is based on the property of global optimal policy $\pi_{t*}^{'}$.

If $H(R_t^k(s,a^1,\ldots,a^n)) < H(R_t^{'k}(s,a^1,\ldots,a^n))$, the

$$\begin{aligned}
& H(R_t^{'k}(s,a^1,\ldots,a^n)) - H(R_t^k(s,a^1,\ldots,a^n)) \\
\leq & \sum_{s'} p(s'|s,a^1,\ldots,a^n) \left(Val[R_t^{'k}(s'),\pi_{t*}^{'}] - Val[R_t^k(s'),\pi_{t*}^{'}] \right)
\end{aligned}$$

and the rest of the proof is similar to the above.

Thus in both cases, we get

$$| H(R_t^k(s,a^1,\ldots,a^n)) - H(R_t^{'k}(s,a^1,\ldots,a^n)) | \leq \sum_{s'} p(s'|s,a^1,\ldots,a^n) \| R_t^k(s') - R_t^{'k}(s') \|. \qquad (5.23)$$

We continue to take the maximum of $\| R_t^k(s') - R_t^{'k}(s') \|$ over s', we can conclude that

$$
\begin{aligned}
\mid H(R_t^k(s,a^1,\ldots,a^n)) - H(R_t'^k(s,a^1,\ldots,a^n)) \mid &\leq \sum_{s'} p(s'|s,a^1,\ldots,a^n)\|R_t^k - R_t'^k\| \\
&= \|R_t^k - R_t'^k\| \\
\|H(R_t^k) - H(R_t'^k)\| &\leq \|R_t^k - R_t'^k\| \qquad (5.24)
\end{aligned}
$$

Hence "nonexpansivity" property of $H(R)$ is verified. We proved the boundedness for R_t^k matrices and also the Lipschitz continuity of $H(\cdot)$. With the boundedness of R_t^k and G_t^k, and the fact that $H(\cdot)$ and $F(\cdot)$ are Lipschitz, it can be said that the time asymptotic parts of the two time scale stochastic approximations track ODE (5.17) and (5.18).

We shall also need

Lemma 16 *$R^{k*}(G^k)$ is Lipschitz continuous.*

Proof Since $R^{k*}(s,G^k) = r^k(s) - G^k + \sum_{s'\in S} p(s'|s,\pi_*)Val[R^{k*}(s',G^k),\pi_*]$, we can get

$$
\begin{aligned}
\mid R^{k*}(s,G^k) - R^{k*}(s,G'^k) \mid \leq & \mid G^k - G'^k \mid + \sum_{s'\in S} p(s'|s,\pi_*) \mid Val[R^{k*}(s',G^k),\pi_*] \\
& -Val[R^{k*}(s',G'^k),\pi_*] \mid
\end{aligned}
$$

As we mentioned before, G functions as a scaler in the matrix games. The nash equilibrium policy remains the same for different G, and $Val[R^{k*}+a,\pi_*] = Val[R^{k*},\pi_*]+a$. Hence

$$\|R^{k*}(G^k) - R^{k*}(G'^k)\| \leq 2\|G^k - G'^k\|.$$

■

Based on the above analysis and the Theorem 1.1 from [49], we make the following conclusion.

Theorem 17 *For the Nash R-learning algorithm (5.3)(5.4),* $(R_t, G_t) \to (R^*, G)$ *a.s.*

5.3.4 The Asynchronous Case

The result above can be extended to the model of asynchronous stochastic approximation. Let $\Phi = \Phi^1, \Phi^2, \cdots$ is the sequence of state-action pairs tried in the learning process, and Φ^k is the state-action pair tried in the kth epoch. The asynchronous version is

$$\begin{aligned} R^k_{t+1}(s, a^1, \ldots, a^n) &= R^k_t(s, a^1, \ldots, a^n) + \alpha_t(\nu_t(s, a^1, \ldots, a^n))\{(r^k(s, s', a^1, \ldots, a^n) - \\ &\quad G^k_t + Val[R^k_t(s'), \pi_{t*}] - R^k_t(s, a^1, \ldots, a^n)\} I\{(s, a^1, \ldots, a^n) \in \Phi^k\}, \\ G^k_{t+1} &= G^k_t + \beta_t \left(\frac{(r^k(s, s', a^1, \ldots, a^n) + tG^k_t)}{t+1} - G^k_t \right). \end{aligned}$$

Where $I(\cdot)$ denotes an identity function which takes the value of 1 when the condition within the round brackets is satisfied and 0 when it is not. $\nu_t(s, a^1, \ldots, a^n) = \sum_{m=0}^{t} I\{(s, a^1, \ldots, a^n) \in \Phi^m\}$, the number of times that the state-action pair was tried till that decision epoch. Hence at every epoch, only the visited state-action value undergos a change, others remain unchanged.

We can reformulate our Assumption 1 that there exists $A > 0$ such that for all $s \in S$ and $(a^1, \ldots, a^n) \in A$,

$$\liminf_{t \to \infty} \frac{\nu_t(s, a^1, \ldots, a^n)}{t} \geq A \ a.s.$$

This ensures that all components are updated comparably often.

The analysis about asynchronous stochastic approximations in [51] requires the following additional conditions on $\alpha(t)$.

Assumption 4 Let $[\cdots]$ stands for "the interger part of $\cdots$", then for $x \in (0,1)$,

$$sup_k \frac{\alpha([xk])}{\alpha(k)} < \infty$$

and

$$\frac{\sum_{m=0}^{[yk]} \alpha(m)}{\sum_{m=0}^{k} \alpha(m)} \to 1 \; uniformly \; in \; y \in [x,1].$$

Examples satisfying this assumption are $\alpha(n) = \frac{1}{n}, \frac{1}{n \log n}, \frac{\log n}{n}$ etc., for $n \geq 2$.

With the additional assumption, we can conclude based on the results in [51] that Theorem 17 holds for the asynchronous case.

5.4 Numerical Experiment

The grid games have been popular testbeds for evaluating and benchmarking of multiagent learning algorithms since these games possess all the key elements of dynamic games. [13] implemented his MinMax Q-learning algorithm on a two-person zero-sum soccer grid game. [15] implemented their Nash Q-learning algorithm on two-person general-sum grid-world games. We adopt one of the grid games used by Hu and Wellman to test and benchmark our Nash R-learning algorithm.

5.4.1 A Grid-World Game

As shown in Figure 5.1, Player A starts from the lower left cell and tries to reach the upper right cell, her goal state. Player B starts from the lower right cell and tries to reach the upper left cell. The players can only move *up, down, left, or right* to the adjacent cells. After both players select their actions, the two moves are

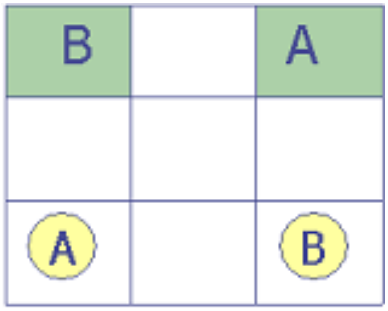

Figure 5.1. A Grid-World Game.

executed at the same time. If they collide with each other, the players bounce back to their previous cells and get punished with reward of -1. The game terminates as soon as any player reaches her goal state, upon which the player gets a reward of 100.

The objective of a player is to reach her goal state through the shortest path, which satisfies both the average reward and the discounted reward criteria. Hence, as in the work of Hu and Wellman, two noninterfering shortest paths of the players, that are constituted of the best responses, represent a Nash equilibrium. In Figure 5.2, we identify some of the Nash equilibrium paths for this grid game. The Nash equilibrium paths take both players four steps to reach their goals. We can see that there are multiple Nash equilibrium policies, but there is only one unique Nash equilibrium average cost, which is $100/4 = 25$. In this grid game, the players' joint positions

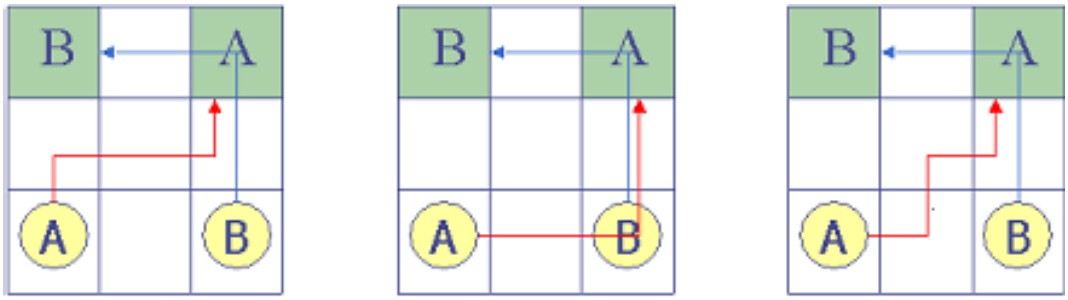

Figure 5.2. Some Nash equilibrium paths for the grid-world game.

define the system state. Since there are nine cells and the players can not be in

the same cell, the total number of states is seventy two including fifteen terminating states (in which at least one player is in her goal state). It is considered that the players do not know their goal states and the payoff functions. They choose their actions independently and simultaneously. They can observe the opponent's previous actions, immediate rewards and current state.

5.4.2 Testing and Benchmarking Approach

A major assumption in all learning algorithms is that all system states and action combinations are visited infinitely often. This is addressed by adopting a suitable exploration-exploitation strategy. Since runing time is not an issue for the grid game having limited state space, we adopt a *non-exploitive exploration* strategy. According to this strategy, the players choose their available actions with equal probability. Let m be the number of steps in the game and $n(s, a^1, a^2)$ denote the frequency of the tuple (s, a^1, a^2) being visited. Then the learning rates are obtained as $\alpha(s, a^1, a^2) = \frac{1}{n(s,a^1,a^2)}$ and $\beta = \frac{1}{m}$. Since $\alpha \geq \beta$, the updating of G^k is performed at a "slower rate" than $R^k(\cdot)$. We define a training (learning) episode as a process that starts when the players are assigned random positions (except goal positions) and ends when either player reaches her goal position. We choose 5000 training episodes such that the learning rate α would reach approximate 0.01 or lower at the end of a training. During the experiments, we found that one run with 5000 episodes usually takes 40,000 steps ($m = 40,000$) (same as in Hu and Wellman's paper). The total number of state-action tuples in this grid game is 424, and each tuple on average is visited 95 times.

We implemented four different learning schemes on the above grid game, for the purpose of testing and benchmarking. In the first scheme, each player ignores the existence of the other player, and uses the independent single agent R learning algorithm

for MDPs. We refer to this scheme as MDP-RL. The players perform their actions, obtain a reward and update their $R^k(s, a^k)$ values without regard to the actions performed by the other player and update the values by $max_{a^k} R^k(s', a^k)$ operator. In the second scheme, the players observe the opponents' action (not the reward) and update their $R^k(s, a^1, a^2)$ values using the $max_{(a^1,a^2)} R^k(s', a^1, a^2)$ operator. We refer to this scheme as MMDP-RL. The MDP based schemes are used commonly for games due to either a lack of computationally feasible methods to solve games or the lack of complete information about the other players; these served as our motivation to consider the first two schemes. The remaining two schemes are derived from the Nash-R learning algorithms. In the Nash-R learning algorithms for the grid game, as shown in Figure 5.3, each player maintains her own matrices as well as the other player's matrices. In one of the schemes, each player randomly uses any one of the Nash equilibrium policies of the stage game in updating her $R_t^k(s, a^1, a^2)$ matrices. Note that a stage matrix game may have multiple equilibria. If the players at any stage of learning use different policies to update their matrices, the final form of the matrices learned by the players may not be the same. As a result, the policies learned by each player may not constitute a Nash equilibrium. In the analysis of convergence of the Nash-R algorithm it was assumed for the stage games to have a unique Nash equilibrium value with either a global optimal policy or a saddle point policy. The purpose of studying the third scheme (where player can use different Nash equilibrium value for updating purposes) is to assess the impact of deviation from the above assumption. In the last of the four schemes, the players always use the same Nash equilibrium value for updating their matrices.

For each of the four schemes, the players were allowed to learn their R-values (for MDP-RL and MMDP-RL schemes) and R-matrices (for the two Nash-R based schemes) over 5000 training episodes. After each training session, the players were

Nash-R Algorithm for a Grid-World Game:

1. Let time step $m = 1$. Initialize matrices $R^k(s, a^1, a^2) = 0$ and average cost $G^k = 0$ for all $s \in S$, $a^1 \in A^1(s)$, $a^2 \in A^2(s)$ and $k = 1, 2$. Set the frequency for the tuple (s, a^1, a^2) being visited $n(s, a^1, a^2) = 0$. Set the number of learned episodes $episode = 0$. Start system simulation.

2. While $episode < 5000$ do

   (a) Randomly generate the initial positions for the players.

   (b) While neither player is in the goal position do

      i. Choose for each player a random action.

      ii. Simulate the chosen actions. Let the system state at the next decision epoch be s', and $r^k(s, s', a^1, a^2)$ be the immediate reward for player k earned as a result of actions (a^1, a^2) chosen in state s.

      - If player k enters her goal position, set immediate reward $r^k(s, s', a^1, a^2) = 100$, for $k = 1, 2$. $episode \leftarrow episode + 1$, go to step 2.
      - If the players collide with each other in a cell other than the goal states, set $r^k(s, s', a^1, a^2) = -1, k = 1, 2$, and $s' = s$.
      - In all other cases $r^k(s, s', a^1, a^2) = 0, k = 1, 2$.

      iii. Calculate the Nash equilibrium policies for the stage game $R^k(s')$. Choose for each player an equilibrium policy π^k_{m*} based on a given scheme and calculate the Nash equilibrium value $Val^k(\pi^k_{m*})$.

      iv. $n(s, a^1, a^2) \leftarrow n(s, a^1, a^2) + 1$. Update the steps sizes as: $\alpha_m = 1/n(s, a^1, a^2)$, and $\beta_m = 1/m$.

      v. Update stage matrices $R^k(s, a^1, a^2)$ and the G^k values for each player as follows, for $k = 1, 2$.

$$\begin{aligned} R^k(s, a^1, a^2) &= (1 - \alpha_m) R^k(s, a^1, a^2) + \alpha_m \left\{ r^k(s, s', a^1, a^2) \right. \\ &\quad \left. - G^k + Val^k(\pi^k_{m*}) \right\}. \\ G^k &= (1 - \beta_m) G^k + \beta_m \left[\tfrac{r^k(s, s', a^1, a^2) + t G^k}{t+1} \right]. \end{aligned}$$

      vi. Set $s \leftarrow s'$, and $m \leftarrow m + 1$, go to step $2(b)i$.

Figure 5.3. A Nash-R reinforcement learning algorithm for computing Nash equilibrium policies for a grid-world game.

allowed to test the strategies calculated based on the learned values/matrices for one episode. Starting positions for both the players were always set to the lower corners as shown in Figure 5.1. During the one episode testing phase, for mixed strategies, random numbers were generated to select actions. We labeled a testing phase as "success" if both players reached their respective goal states in four steps (which is optimal). For every scheme, two hundred training-testing runs were conducted and the respective "success" rates were obtained. We note that our method of establishing success rate is different from that of [38]. It appears that they compared the Nash equilibria obtained from the learned Q-matrices with the theoretical results, and a match was defined as a "success".

5.4.3 Results and Discussion

The success rates were calculated for each of the four benchmarking schemes. Note that the success rates are binomial proportions obtained from 200 Bernoulli trials consisting of 5000 training episodes followed by one testing episode. We used a normal approximation to obtain 95% confidence intervals on the success rates. The results are shown in Table 5.1. Several observations can be made from the results.

- The single player learning scheme (MDP-RL) which ignores the existence of the other player attained optimal strategy only 16% of time. This is quite expected, since it is well known that in a game, considering other players as part of the stationary environment yields poor result.

- In the MMDP-RL scheme, the players observe other's actions and use that information in their single player learning scheme. With more information than

Table 5.1. Testing and benchmarking results for four different learning algorithms

Benchmarking schemes	Success rate $\pm 95\% C.I.$
MDP-RL (learning without observing another player's action)	$16\% \pm 5.1\%$
MMDP-RL (learning with observing another player's action)	$30.0\% \pm 6.3\%$
Nash R learning (updated with possible different Nash values)	$33.5\% \pm 6.5\%$
Nash R learning (updated with the same Nash values)	$98\% \pm 1.9\%$

MDP-RL, this algorithm performed much better than MDP-RL and obtained a success rate of around 30%.

- For the Nash-R learning scheme that uses possibly different stage Nash-equilibria for updating , the success rate was significantly lower than the other Nash-R scheme that uses the same Nash equilibrium . Recall that in the convergence analysis of Nash-R algorithm, a necessary assumption (Assumption 3) was to have a unique Nash equilibrium value. We conclude from the poor performance of the third scheme that the assumption has been violated.

- An arbitrary Nash-R scheme that violates Assumption 3 may not perform significantly better than the MMDP-RL scheme.

- Even though it is clear that the grid game does not satisfy Assumption 3, ensuring the use of same Nash equilibrium value by both players produces near perfect result.

CHAPTER 6

APPLICATION STUDY: DEREGULATED POWER MARKET

In the late 1990s, several U.S. states or control areas such as California (CA), Pennsylvania-New Jersey-Maryland (PJM) interchange, New York (NY), and New England (NE) established market for electricity. Policy makers believed restructuring would impose market discipline and thus lead to lower cost of production from existing generation units and could encourage more efficient investments. Major service sectors like, banking, airlines and telecommunications benefited immensely from the competition resulting from deregulation. Unfortunately, soaring electricity prices throughout the western U.S. have called into question the entire restructuring movement. Restructured eastern U.S. markets appear to have had a relatively successful experience compared to California. It has been argued that the design of the eastern markets have lead to less price volatility and limited the degree to which firms exercise market power.

6.1 Electricity Market Overview

The electricity market is fundamentally a commodity market. But it differs from other commodity market in two major respects:

- the inability to store electrical energy, and
- the sharing of a common, essentially uncontrollable, transmission network.

As a result of these differences, the electricity market didn't evolve in the same way as other commodity markets such as airline, banking, and telecommunication. These commodities established markets spontaneously and developed trading rules organically without any government oversight beyond the normal provisions of law. Electricity market was run either by government or under significant regulation. Improvements in the transmission technology have made the trading and competition possible in the electricity markets. The two key common aspects of the transition towards competitive electricity markets in the U.S. and around the world are competitive generation sector and open access to the transmission system [52] [53][1][54].

6.1.1 Wholesale Market Participants

An electricity market has the following sectors: 1) generation, which produces power using thermal, hydro, nuclear, or other technologies, 2) transmission, which carries power from the generation points to the load center through high voltage lines, 3) distribution, which distributes power (in low voltage) to the consumers from the load centers, and 4) customer service, which performs activities such as metering the power and billing the customers [52][1]. Figure 6.1 shows a schematic diagram of the market structure [1]. Wholesale market participants include the generators and the distributors. In this work, distributors are considered as end customers in the wholesale market. The design of the wholesale market should accommodate access for both suppliers (generators) and customers' competition while recognizing the special requirements of reliability in the transmission grid. A short-term electricity market coordinated by a system operator provides a foundation for a competitive electricity market. The coordination function served by independent system operator (ISO) is vital for a power market. PJM defines ISO as [55]

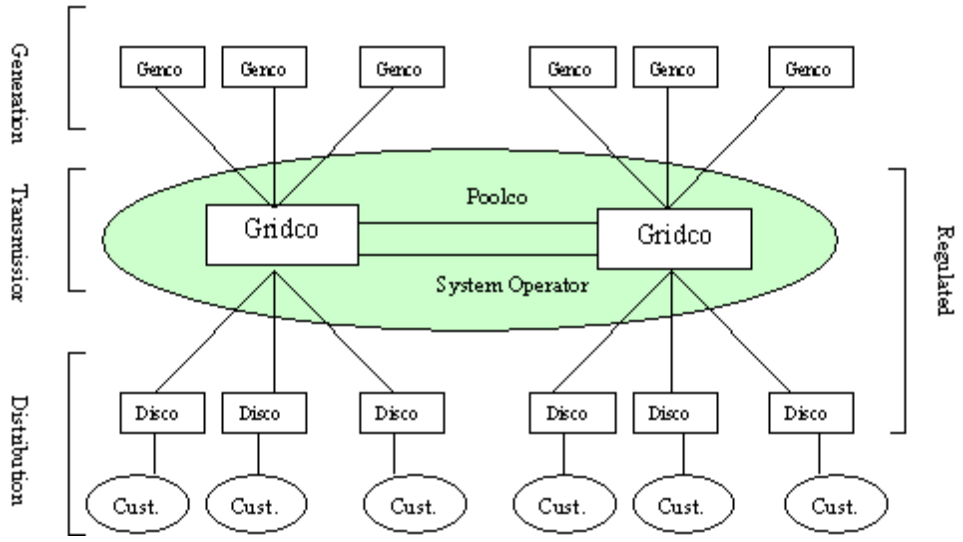

Figure 6.1. Competitive Wholesale Electricity Market Structure [1]

> "It coordinates the continuous buying, selling and delivery of wholesale electricity through the energy market. In its role as market operator, SO balances the needs of suppliers, wholesale customers and other market participants and monitors market activities to ensure open, fair and equitable access. "

PJM Energy Market operates much like a stock exchange, in which the market participants establish a price for electricity by matching supply and demand.

6.1.2 Market Designs

Two major types of market design have emerged in the restructuring process, which either have been implemented or are currently being proposed for various markets in the U.S [54] [56]. The first one is *pool* type market that uses a centralized dispatch. Some of the existing POOLCO markets are PJM, NYISO and ISO-NE. In a pool market, A ISO runs a real time market with centralized dispatch. The supply bids are complex multi-part supply bids with detailed information about

the generators. ISO selects the prices and quantities for all the generators by solving a nonlinear programming model, which also addresses both the unit commitment (UC) and the optimal power flow. The dispatch decisions are made independent of and without any recognition of any bilateral forward contracts. Bilateral trades are purely financial in nature and do not get scheduling priority. In this way, electricity is not like other commodities. We will discuss this in detail in later sections.

The other market design known as *exchange* markets, is a more decentralized approach. CA ISO adopts a combination of public exchange and private exchanges and dealers (the other scheduling coordinates). As is typical of exchanges, simple bids are used in a power exchange. Generators only bid energy quantity and price and can not take account of their startup and no-load costs. The UC problem is solved by the independent suppliers themselves. The ISO determines the market clearing prices by solving an auction problem.

6.1.3 Forward Market and Real Time Market

Generators and customers trade electricity in both forward market and real time market. Bilateral contracts and day-ahead transactions run by ISO occur in the forward market. Actual power trading occurs in the real time market. The notion of bilateral contracts like other commodity markets is defined where the customers and the generators make agreements to trade at prices that are independent of recognition of SO. In a pool based market, SO runs a two-settlement system with day-ahead bids and real time scheduling. In the day ahead (DA) market, the SO accepts bids and determines an appropriate market clearing equilibrium and associated payment settlement. This schedule then defines a set of commitments for delivering and taking power in the short-run dispatch. In the real time market, the actual dispatch usually

differs from the scheduled commitments and appropriate balancing settlements are arranged.

The forward markets are financial markets in the sense that the delivery of power is optional and the seller's only real obligation is financial. In many forward markets, including many DA markets, traders selling electricity need not be a generator. Any power that is sold in the forward market but not delivered in real time is deemed to be purchased in real time at the spot price of energy.

6.1.4 Transmission and Congestion

Deregulation has increased competition among generators to use transmission networks effectively. In reality, generators and customers are connected through a free-flowing grid of transmission and distribution lines. If a line joining a low cost source to a high priced node has a limit, then it may not be possible to supple sufficient power from the low cost to the high cost location. In which case, Higher cost generator closer to the load may need to be used to meet demand. This results in the locational marginal prices (LMP). LMP is not a new concept to power system operators. For many years, system operators have managed congestion using least-cost security constrained dispatch uses similar programs that calculate LMP. Under locational pricing, the cost of transmission congestion is obtained as the difference in energy prices between two locations. Thus the cost to operate more expensive generation are translated into transmission congestion costs in LMP calculation.

Locational pricing is usually accompanied by a system of allocating transmission capability through financial contracts. Market participants can hedge volatile congestion costs by buying these contracts, thus increasing transmission price certainty. Such contracts are known as either Fixed Transmission Rights (FTR) or Transmission Congestion Contracts (TCC). During post-dispatch financial settlements, the

ISO collects congestion fees in the form of energy and/or transmission charges from system users, and pays congestion rents to TCC owners.

6.1.5 Pricing and Settlements

Efficient pricing is a central feature of a competitive electricity market which will signal and provide good incentives to appropriate levels of consumption and supply or the appropriate levels and locations of new generation and transmission investment [57]. The LMPs mentioned in the previous section are simply the market-clearing prices based on all the bids and the details of the requirements of network operations. Using the bids as the representation of these benefits and costs, the corresponding economic dispatch produces the same outcome as a competitive equilibrium [57]. The LMPs determined by the economic dispatch consider generation marginal cost, transmission congestion cost and cost of marginal losses.

Nodal pricing and zonal pricing have evolved to be the two distinct approaches . In the nodal pricing scheme, during the transmission congestion period, the market clearing price (MCP), varies at different nodes. The difference in the LMP between any two nodes is the congestion charges between the two nodes. In the zonal method, the entire transmission line is divided into different congestion zones, and the price vary across the zones [58]. Though the zonal method is easier to implement in the sense that the number of places (zones) at which the prices vary is less, it brings another critical question of how to decide the congestion zones. The trend is now to move towards the nodal pricing system, which has been proven to be effective.

Most electricity markets have a two settlement system such as financial settlements and physical settlements. A simplified two-settlement system is described in Figure 6.2. Not like other commodity, bilateral contracts, day-ahead bidding and transmission rights in electricity markets are financial in nature and not connected

with the physical operations, which manage the flow of money not the flow of power. In the real time, power is dispatched independent of and without any recognition of forward contract, bidding and transmission rights.

When transmission constraints are binding, congestion costs will change prices at different locations. The forward prices are also just estimates, sometimes very rough estimates. In a competitive market the real time prices are true marginal costs. But the real time (spot) market price can be volatile, which presents certain risks for both generators and customers. The forward market trades mitigate or share the risk. Contracts for differences (CFDs) insulate bilateral trades from all risks of spot price fluctuations while allowing the inevitable inefficiencies of forward trading to be corrected by accurate real time price signals. Both CFDs and two-settlements preserve real time incentives [52]. If a generator sells Q_1 in the forward market with price P_1 and delivers Q_0 to the real time market with real time price P_0, it will be paid $Q_1 \times P_1 + (Q_0 - Q_1) \times P_0$.

In the presence of transmission congestion and losses, the forward market is not sufficient to provide the necessary price hedge. It is possible to arrange an FTR that provides compensation for differences in LMPs. The generator obtains an FTR for Q MW between node G and L. If the real time LMPs in two nodes are P_G and P_L, the settlement system should pay the generator $(P_L - P_G) \times Q$.

6.2 Market Equilibrium Models Review

The objective of restructuring the electricity market is to create a competitive environment for trade of electricity that ultimately benefits the society by lowering prices and attracting new investments. The realization of these potential benefits depends on efficient market design and operation that limit the opportunities for participants to achieve market power. In electricity markets, the physical

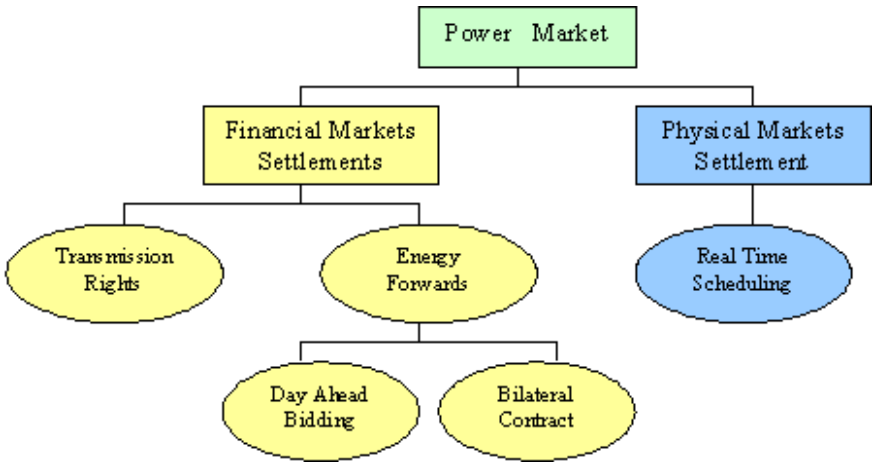

Figure 6.2. A Simplified two-Settlement Electricity Market

properties of electricity, economic aspects of electricity supply and demand, and market design/institutional features may create opportunities for the exercise of market power by participants. Several approaches have been used to evaluate power market design, such as empirical analysis, laboratory experiments, modeling and simulation studies. The modeling approach is more easily generalized and analyzed for sensitivity [23]. In a competitive market, the strategic interactions of various decision makers need to be incorporated into the model. In examining this issue, a complementary question has given rise to a theory about an individual firm's behavior toward its rivals and their possible reactions, and how should such behavior be modeled from an analytical perspective. To answer this question, the theory of noncooperative game has been applied extensively [59]. *Gaming* refers to taking unanticipated advantage of market rules [20]. While any economic mechanism can be modeled as a game, and all evoke particular strategies from the players, certain mechanisms are considered to be particularly susceptible to gaming. These are mechanisms that evoke strategies that are unanticipated and that subvert the intentions of the mechanism's designer. The concept of Nash equilibrium is widely used in economic theory to model the behavior of firms in competitive markets. According to the word presented in [60],

two major reasons why it is in the best interests of profit-maximizing firms to adopt a Nash equilibrium strategy in the power market are as follows: 1) By the definition of Nash equilibrium, it is rewarding for a firm to bid according to the Nash equilibrium strategy when competing firms also bid according to the Nash equilibrium strategy. 2) The Nash equilibrium is stable, the firm that decides to deviate from this strategy has a strong incentive to return to it.

6.2.1 Introduction

The three primary equilibrium models formed in the literature that have been applied to examine market power are Cournot, Bertrand, and Supply Function Equilibrium (SFE) models of imperfect competition. The common assumption of these models is that each individual competing firm seeks to maximize its profit for given:

- demand conditions,
- cost structure,
- other relevant market conditions, and
- assumption about how rivals will respond to its decisions.

The key difference among Cournot, Bertrand and SFE models lies in how each generating firm anticipates the rivals' reaction to its decision concerning either of quantity, price and supply function. In the Bertrand model, for example, players believe that their rivals will not change their prices. In the Cournot model, each individual player assumes that its output affects the price but that its output decision does not affect the output of the rivals. In the SFE model, the firms are assumed to bid entire supply

functions, and the resulting price equilibria generally are between the Bertrand and Cournot outcomes.

In the following the discussion, let q_i, p_i denote the quantity and price of firm i, and q_{-i}, p_{-i} denote the quantity and price of the rivals of the ith firm. The profit function of firm i is $U_i = q_i p_i - C_i(q_i)$, where $C_i(q_i)$ denotes the cost function corresponding to quantity q_i. Let $D(p)$ denote the demand as a function of p, the price.

6.2.2 Bertrand-Nash Model

The game of price competition was first studied by Bertrand, and is now known as Betrand game. In this game, two firms compete against each other using prices as strategy choices. Each firm is assumed to correctly anticipate its rival's price choice, and then chooses a price to maximizes its own profit for a market demand is $D(p)$. The firms always supply to meet demand. The consumers are assumed to buy the cheaper one, or to purchase half from each if the prices are the same, which is represented as:

$$D_i(p_i, p_j) = \begin{cases} D(p_i) & p_i < p_j \\ 0.5D(p_i) & p_i = p_j \\ 0 & p_i > p_j \end{cases}$$

So the ith firm's profit function is $U_i(p_i, p_j) = D_i(p_i, p_j)p_i - C_i(p_i)$. The Bertrand-Nash equilibrium (p_i^*, p_j^*) is defined as:

$$U_i(p_i^*, p_j^*) \geq U_i(p_i, p_j^*).$$

It is equivalent to solve the following function for each firm

$$\max_{p_i} U_i(p_i, p_j) = D_i(p_i, p_j)p_i - C_i(p_i).$$

To calculate the Nash equilibrium, the above profit function can be differentiated for each firm and set the first order condition to zero (Kuhn-Karesh-Tucker/KKT conditions), and this is where the Bertrand assumption enters. Because we assume that the price of the rival is constant in its profit function, $\frac{dp_j}{dp_i} = 0$. Then the first-order condition is :

$$\frac{dU_i(p_i, p_j)}{dp_i} = \frac{dD_i(p_i, p_j)}{dp_i} \times p_i + D_i(p_i, p_j) - \frac{dC_i}{dp_i} = 0$$

The equilibrium solution satisfies the first-order condition for each firm while matching demand and supply. If both firms have identical unit production costs, the Nash equilibrium is that both firms set price at marginal cost hence do not make positive, economic profit. These results constitute the Bertrand paradox because they imply that a competitive outcome occur even in an industry with only two competitors [59].

The first application of Bertrand game in electricity market was proposed by Hobbs [19]. An important rationale for the application is that, since electricity can not be stored, it may be subject to significant short-run price competition leading to Bertrand assumption. The motivation is that as long as price exceeds marginal cost and producers have sufficient capacity to meet demand, they will undercut each other's prices in an effort to gain market share. Under the existence of capacity constraints and transmission cost, which introduces spatial heterogeneity to production cost, generators sell at the marginal generation and transportation cost of their closest competitors which is above their own marginal cost. Hence the generators use geography to exert market power.

If the competing generators' cost functions are relatively flat and excess capacity exists, Bertrand competition may realistically represent firm behavior. But in electricity market, the generators sometimes face peak demand periods resulting in

significant capacity constraints exist. In that case, the Bertrand model may be questionable. There is another feature in electricity market, the transmission limits should be recognized, they affect the market clearing prices. Hence Bertrand assumption is not realistic in models with transmission constraints [61].

6.2.3 Cournot-Nash Model

Cournot published his theory of oligopolistic competition in the 1830's. In the classic model of Cournot duopoly, firms compete against each other using quantities as strategy choices. Each firm anticipates its rival's quantity choice, such that the quantity it chooses in equilibrium maximizes its profit (given the quantity choice by its rival). Price is given by the inversed demand function, $p = P(q_i + q_{-i})$. So firm i's profit function is $U_i(q_i, q_{-i}) = P(q_i + q_{-i})q_i - C_i(q_i)$. The Cournot Nash equilibrium (q_i^*, q_{-i}^*) is defined as:

$$U_i(q_i^*, q_{-i}^*) \geq U_i(q_i, q_{-i}^*).$$

To calculate the Nash equilibrium, the profit function for each firm is differentiated and is set to zero. Also used here is the Cournot assumption that the quantity of its rival is constant is its profit function, $\frac{dq_{-i}}{dq_i} = 0$. The first-order condition is :

$$\frac{dU_i(q_i, q_{-i})}{dq_i} = P(q_i + q_{-i}) + \frac{dP(q_i + q_{-i})}{dq_i} \times q_i - \frac{dC_i(q_i)}{dq_i} = 0.$$

Stoft [20] gave a simple Cournot example as follows. Let generators 1 and 2 have constant marginal costs of $20/MWh$ and $40/MWh$. Assume they have no capacity and the demand is $Q_D = 2(100-P)$. Then the price is given by $P = 100-(Q_1+Q_2)/2$.

The profit function is $U_i = (100 - (Q_1 + Q_2)/2)Q_i - c_i Q_i, i = 1, 2$. Setting the first order of the profit function to 0, the Nash equilibrium is $Q_1^* = 66\frac{2}{3}, Q_2^* = 26\frac{2}{3}$.

Cournot competition may realistically represent electricity market behavior where competing generators' marginal costs are relatively steep and capacity constraints exist. Smeers [21] thought it was natural to use Cournot behaviour to analyze natural gas competition, since it is mainly traded in Europe through long term contracts and competition can thus be expected to take place through quantities. Since long term contract trading consists of large proportion of energy transaction, competition is likely to be in quantities. Hobbs built two Counot models to simulate bilateral markets including a congestion pricing scheme for transmission and Kirchhoff's laws via a DC approximation [61].

In the Cournot model, price is exclusively determined by the intersection of the aggregate quantity offered and demand curve. But in electricity market, it is difficult to specify the market demand curve. The demand in the short-run market is not so elastic, hence price predictions from Cournot models are not particularly reliable. Since Bertrand assumption is not realistic in models with transmission constraints, the Cournot behavior is assumed in most of models with transmission models. Stoft [62] used Cournot model to study market power when generators faced a demand curve that is limited by transmission constraints, which also included TCC into the profit function. Kamat and Oren [56] analyzed welfare properties of two-settlement systems for electricity in the presence of network uncertainty and market power. They model a spot market where generators use a Cournot conjectural variation in period 2. Cunningham et al. [63] model Cournot equilibrium of three market players in a transmission constrained system and consider nonconstant marginal cost. Metzler [64] model Nash Counot equilibria in power markets on a linearized DC network.

6.2.4 Supply Function Equilibrium Model

The third model for the analysis of imperfect competition is the supply function equilibrium model (SFE), in which firms compete with each other through the simultaneous choice of supply functions. The SFE model is more intuitively appealing than the Bertrand and Cournot models because it allows for a strategy space in which competing firms choose entire supply functions. The strategies of the Bertrand and Cournot models are limited because firms choose either prices or quantities. Green and Newbery [65] used SFE to describe the electricity spot market in England and Wales. Each firm submits its supply function to the grid dispatcher simultaneously, and the dispatcher then determines the spot price and each firm's supply by solving for the price-quantity pair that equates supply to demand. The total output supplied at the market-clearing price p must equal the demand with that price at that time $D(p) = \sum q_i(p)$. Each firm's profit function $U_i = pq_i - C_i(q_i)$, can be expanded as an function of p as follows:

$$U_i(p) = p\left(D(p) - \sum_{j \neq i} q_j(p)\right) - C_i(D(p) - \sum_{j \neq i} q_j(p)).$$

The first-order condition can be written as:

$$\frac{dU_i}{dp} = D(p) - \sum_{j \neq i} q_j(p) + (p - C_i'(q_i(p))) \times \left(\frac{dD(p)}{dp} - \sum_{j \neq i} \frac{dq_j}{dp}\right).$$

To maximize the profit function, the derivative is set to zero and get another differential equation for the function:

$$q_i(p) = (p - C_i'(q_i(p))) \left(-\frac{dD}{dp} + \sum_{j \neq i} \frac{dq_j}{dp} \right). \tag{6.1}$$

The basic equation (6.1) governing the SFE solution is provided by Green [66] and does not depend upon particular functional forms. When we substitute supply functions, cost functions and demand function to (6.1), any solution to the above coupled differential equations when each firm submits a non-decreasing supply function is a SFE.

Green concentrated on the unique linear solution in [66]. Each firm has quadratic costs to give linear marginal costs to outputs $C_i'(q_i) = c_i q_i$. Each firm's supply function takes the form $q_i(p) = \beta_i p$. So each firm's objective function is to: $\max_{\beta_i} U_i$. It is equivalent to solve equations (6.1) noting that $\frac{dq_j}{dp} = \beta_j$. Baldick, Grant and Kahn [67] extended the linear version of SFE. They generalized this to the case of asymmetric plants with affine marginal costs and propose an ad hoc approach to deal with capacity constraints.

SFE is more realistic regarding the competing behavior of generators in electricity markets. In a centralized market-clearing mechanism, such as POOLCO, the ISO requires all generator to offer a supply function bid. SFE can estimate equilibrium mark-ups. Equilibrium mark-ups are less transparent in the Cournot framework than in the SFE framework. Cournot (vertical supply curves) and Bertrand (horizontal supply curves) equilibriums are extreme solutions in SFE. The SFEs generally lie between Betrand and Cournot equilibria. Baldcik [22] demonstrated with examples that some SFE results presented in the literature are in fact artifacts of assumptions about the choices of particular bid parameters ($C_i'(q_i) = c_i + R_i q_i$). Unfettered choices

of convex quadratic bid cost functions can be expected to lead results that are closer to Cournot outcomes. Hobbs [23] only allow firms to manipulate the intercept of the bid functions, and not its slope in his Mathematical Program with Equilibrium Constraints approach.

A major obstacle to the realistic application of SFE to models of electricity networks is computational difficulty. Either very simple systems are considered (1 or 3 nodes) or restrictive assumptions had to made about the form of the supply functions. With the representation of large transmission networks and many generators subject to capacity constraints, a generator's optimization problem is often non-convex and may yield multiple, local optima. Day et al. [68] use conjectured supply function (CSF) approach to modeling oligopolistic competition on power network, which is computational feasible for large system. Pang et al. [69] use CSF approach to analyze the existence and uniqueness properties of solutions for a linearized DC load flow model. But the CSF model needs behavioral parameters which is not directly observable. Another obstacle is the existence of multiple SFEs when generators have full discretion in choosing the bid supply function parameters and are not constrained to bid the same supply function over multiple pricing-periods. This makes no prediction.

6.2.5 Summary

Three primary equilibrium models, Cournot, Bertrand and Supply Function Equilibrium (SFE), all have their advantages and disadvantages as discussed above. Each model should be applied to specific problem settings. SFEs are more realistic regarding the competing behavior of firms in electricity markets. Since in a pool market, the ISO requires all generator to offer a supply function bid. Most SFE calculation like other models is based on the Kuhn-Karesh-Tucker (first order) condition, which requires that the optimization problem is convex. But in most real

scenarios, the assumption will be violated, hence it is hard to solve mathematical programs. Hobbs used an advanced interior point algorithm to solve a bilevel nonconvex problem [23], but there is no convergence proof.

The stochastic model of bidding problems have been modeled as MDPs [70][71], but for a multi decision maker's problem, as presented in the previous chapter, learning without recognizing the existence of other players produces poor result.

The SFEs concept is adopted in this research. It is also considered that electricity auctions are repeated on a daily basis, and this may give generators an opportunity to participate in more complex strategic moves. Hence instead of being modeled as one shot games or MDPs, the generators' bidding problems have been modeled as stochastic games in the next chapter.

CHAPTER 7

POWER MARKET MODEL FORMULATION AND IMPLEMENTATION

The coexistence of competition and regulation is by nature imperfect, and may result in behaviors that are far from the ideal paradigm of perfect competition. Our research objective is to obtain the equilibrium strategies in order to identify the behaviors of competing generators. In this chapter, a formal model is presented, which embeds in it various market and institutional aspects. Institutional aspects may include decisions of public authorities that influence the organization of the market, such as pool or power exchange market design, and transmission pricing polices. Market aspects include separate transmission and energy market, two settlement system, and other market rules governing capacity and cost.

7.1 Problem Statement and Assumptions

The problem considered here is to develop a model that integrates several physical and economic features as well as assumptions about the pool type electricity market as described below.

- Bilateral contract

  The generators are assumed to compete in the bilateral, day ahead, and real time market. It is assumed that the players are risk neutral, and hence the bilateral contract price is considered to be the demand-weighted average of the

prices in the past spot markets. Generators are assumed to game with the bilateral contract quantities only.

- Day ahead market

  Based on the forecasted demand, each generator submits a bid for 24 hours to an ISO. Each bid is in the form of a piece-wise linear functions of cost versus quantity. The ISO then decides on the supply quantities for the generators and locational marginal prices based on the optimal power flow (OPF) model.

- Transmission pricing

  Point-to-point (PTP) financial transmission rights are considered. It is assumed that each generator is a PTP option holder, which is entitled to payments when the differences in the locational prices are positive, but will not be obligated to make payments when the locational differences are negative. The FTRs are assigned based on simultaneous feasibility test to assure the revenue sufficiency.

- Real time scheduling

  The demand experiences some variation in real time. It is assumed that the generators can not adjust their bids. The ISO recalculates the OPF model to schedule real time power dispatch.

- Demand elasticity

  Demand bidding is not considered in this model. Demand is a function of price $D(p)$ and considered elastic to certain degree with a downward sloping curve. In this model, demand is considered to responds elastically to the previous day spot price p^{T-1}. The delay is due to the fact that the electricity customer has no real time billing. In the day ahead market, we forecast demand based on $D(p^{T-1}) = q_{max} - \alpha \times p^{T-1}$, where p^{T-1} is the spot price from previous day. The

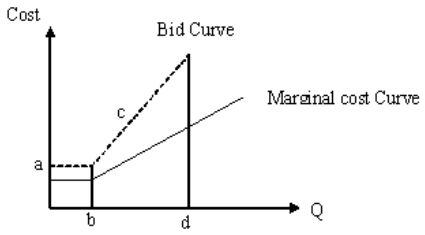

Figure 7.1. Marginal Cost (Supply) Function and Bid Curve

real time demand generally deviates from the forecasted demand, It is assumed that the real time demand follows a normal distribution with mean $D(p^{T-1})$ and a given variance based on factors such as market, season and day of the week. All real time market demands must be satisfied.

7.2 Problem Formulation

This model is represented as a two-level optima ion problem for each player. At the upper level, generators choose the parameters of bidding curves for the day ahead and real time markets, and quantities for the bilateral contract. At the lower level, an OPF model is used to clear the market. The upper level is modeled as a competitive Markov decision process (CMDP)/stochastic game (SG), while the OPF at the lower level is a nonlinear program.

7.2.1 A CMDP/SG Model for Generator Game Behavior

Generators compete on a daily basis in the day ahead and real time energy markets by submitting n-part piecewise linear bid curves for each of the 24 hours. The ISO performs OPF calculation for day ahead market and real time market. As shown in the Figure 7.1, a two part piece-wise supply bid curve can be specified by the parameters $\{a, b, c, d\}$, where a, b, c, d can be interpreted as minimum cost, minimum

capacity, price and maximum capacity respectively. It is assumed that generators bid at or above their marginal cost curves. Each generator's objective is to maximize the average profit over infinite horizon, which can be modeled as a CMDP. Following notation is used in the formulation of this problem.

- N: the number of generators.
- M: the number of load points.
- T: denotes the day. $T = 1, 2, \ldots$
- i: index of generation nodes (busses) in the network.
- j: index of load nodes (busses) in the network.
- $\mathcal{G}$: set of all generators' nodes.
- $\mathcal{L}$: set of all load nodes.
- L_i: set of all possible load nodes which can make contracts with generator i. $L_i \in \mathcal{L}$
- ℓ_j: load demand for load node $j \in \mathcal{L}$.
- p_i^d: day ahead price of power at node i.
- q_i^d: day ahead quantity of generation power at node i, for $i \in \mathcal{G}$.
- p_i: spot price of power at node i.
- q_i: spot quantity of generation power at node i, for $i \in \mathcal{G}$.
- $p_{i,j}^b$: bilateral contract price between generator i and load j.
- $q_{i,j}^b$: bilateral contract quantity between generator i and load j.

- $T_{i,j}$: TCC right owned by generator i for arc ij.

Let the system state at day T is given by $S = \{\ell, p\}$, where $\ell = \{\ell_j, j \in \mathcal{L}\}$, and ℓ_j denotes the forecasted demand for the load j. This demand is calculated based on the demand function of previous spot price. Aslo, let $p = \{p_j, j \in \mathcal{L}\}$, where p_j denotes the forcasted load nodal price and is considered to be the same as the actual spot prices for the loads in the previous day.

Action space $A_i = \{q^b_{i,j}, B_i, j \in L_i, \}$ for each generator consists of two decisions. where $q^b_{i,j}$ is to decide the quantity for each possible contract, B is the bidding curve. Let $B^t_i = \{a_i, b_i, c_i, d_i\}$, which consists of four bidding parameters. However, the bilateral quantities and the bidding parameters are discretized, the latters vary based on the base cost curve.

After each generator takes action and ISO calculates p^d_i and q^d_i based on the forecasted demand. The ISO also calculates p_i and q_i based on the real time demand for each node. The profit for each generator is obtained as $R^i = p^d_i q^d_i + \sum_{j \in L_i} p^b_{i,j} q^d_{i,j} + \sum_j T_{i,j}(q^d_j - q^d_i) + p_i(q_i - q^d_i - \sum_j q^b_{i,j)})$.

7.2.2 OPF Model–Non Linear Programming

The optimal prices in a transmission network are the nodal prices resulting from an OPF performed by a centralized dispatcher (ISO). The OPF model is implemented in parts of the United States such as PJM. It is solved in order to minimize the total bid cost, subject to the demand and supply constraints, voltage constraints, thermal limit constraints, and the constraints of power supply. The constraint set also considers the Kirchhoff's power flow equation.

Let $f_{1,i}$ and $f_{2,i}$ denote the costs of active and reactive power generation, respectively, for generator i at a given dispatch point. Both $f_{1,i}$ and $f_{2,i}$ are assumed to be a polynormial or piecewise-linear functions. Also let $Q_{1,i}$ and $Q_{2,i}$ denote the quantities

of the active and the reactive power generation respectively by the generator i. Then the OPF formulation is given as

$$min \sum_{i \in N} f_{1,i} Q_{1,i} + f_{2,i} Q_{2,i}$$

subject to

$$\sum_{i \in N} Q_{1,i} - Q_{1,\ell} - Q_1(V, \theta) = 0 \tag{7.1}$$

$$\sum_{i \in N} Q_{2,i} - Q_{2,\ell} - Q_2(V, \theta) = 0 \tag{7.2}$$

$$S_{y,z} \leq S_{y,z}^{max} \tag{7.3}$$

$$Q_{1,i}^{min} \leq Q_{1,i} \leq Q_{1,i}^{max} \tag{7.4}$$

$$Q_{2,i}^{min} \leq Q_{2,i} \leq Q_{2,i}^{max} \tag{7.5}$$

Contraints (7.1 and 7.2) are active and reactive power balance equations, where $Q_{1,\ell}$ and $Q_{2,\ell}$ denote active and reactive load demand respectively. The two constraints ensure all the demand and transmission losses are met by the generations. Constraint (7.3) ensures that all the maximum flow limit constraints in both directions are not violated, where $S_{y,z}$ denotes the power flow from bus y to bus z. The constraints (7.4 and 7.5) are used to maintain the active and reactive power generation limits.

Matpower, a matlab power system simulation package can be used for OPF simulation, which was developed by the Power System Engineering Research Center (PSERC) at Cornell University [72][73].

7.3 Experiments and Analysis

The model and the solution framework, as presented in the previous chapter, are tested using a power network example, for which the day ahead and real time

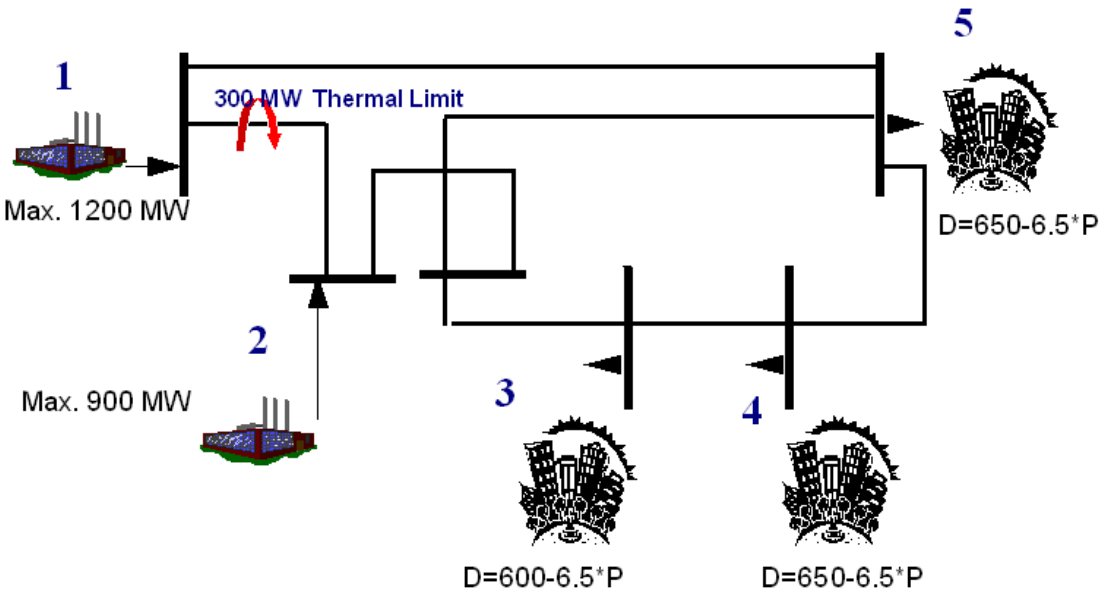

Figure 7.2. A 2-Supplier and 3-Retailer Power Network with Congestion

markets are considered. The objectives of this study are threefold: 1) to implement the modeling and solution framework, 2) to investigate the pricing behavior of generators with or without the existence of transmission constraints, and 3) to investigate the effect of TCC allocation on generators' pricing behavior.

7.3.1 A Sample Power Network

The example network consists of two generators and three load nodes. A congested version of the network is shown in Figure 7.2 where congestion is caused by a transmission limit in the line connecting Bus 1 and Bus 2. The details (capacitance, inductance, etc.) of the six-Bus network are identical to the tutorial problem that is available in the PJM 101 Training Materials at the PJM website. The figure shows the maximum capacities of the suppliers, demand function for each load, and the transmission limit. It is also considered that the generator at node 1 can acquire transmission congestion contract (TCC) on the line connecting Bus 1 and Bus 2.

7.3.2 Simulation and Decision Parameters

The numerical study includes learning and testing phases. For each episode of learning and testing, the day ahead market and real time market are simulated for a given TCC allocation. ISO forecasts day ahead demand based on the previous price. Generators submit bids and bilateral contract quantities. Matpower simulation package [72][73] is used to calculate prices and quantities for day ahead market. The real time market is simulated where a normal distribution random number generator is used to generate real demand for each load. Then Matpower is used again for real time schedule. The marginal cost curves for generator 1, 2 are $(60, 10, 18, 1200)$ and $(70, 10, 23, 900)$ respectively. Generators are assumed to bid price parameter c at or above the marginal-cost price or bid capacity parameter d at or below the maximum capacities. ($\{36, 25, 30, 18\}$ for generator 1 price bid set, $\{23, 27, 34, 40\}$ for generator 2 price bid set, $\{900, 1000, 1100, 1200\}$ for generator 1 capacity bid set, $\{700, 750, 800, 900\}$ for generator 2 capacity bid set). Generator 1 is assumed to enter into bilateral contract with load 3 at quantity level $\{0, 100, 220, 300\}$. The bilateral contract quantities that generator 2 can sign up with load 4 are $\{0, 80, 200, 280\}$.

The step by step details of the reinforcement learning algorithm is similar to the algorithm (Figure 5.3) for grid game. The learning rates are decayed to 0 as the learning progresses through the 100,000 learning episodes. An exploration-exploitation strategy is used to deal with the large state-action space is big. For the first 5000 episodes, generators pick actions randomly, then the exploration probability (p=0.2) is applied to allow 80% exploitation and 20% of exploration. After 50,000 episodes, a greedy strategy is applied for generators to choose actions based on current Nash equilibrium policies.

After learning is completed, 5000 testing episodes are run based on the learned R matrices. For each testing run, the Nash equilibrium policy is implemented and the average profit per day is obtained.

7.3.3 Numerical Results

Two separate cases of the network are considered here, which are constrained case and unconstrained case. In unconstrained case, the transmission limit (300 MW in constrained case) on the line connecting Bus 1 and Bus 2 in Figure 7.2 sets to infinity, hence no congestion will occur. For each case of the two cases, the perfect competition scenario is implemented for the purpose of comparison. In perfect competition scenario, all generators bid at their marginal cost curves. For constrained case, three different scenarios were considered with generator 1 having TCC for 0MW,100MW or 300MW respectively.

The sample learning curves of average profits (G) for two generators are shown in Figure 7.3, which indicate the convergence of their learning processes. The day ahead demand and real time demand are presented in Figure 7.4, which shows that the real time market bears more variability than day ahead market. For multi-settlement system, forward market provides some hedge against the volatile real time market.

Tables 7.1 through 7.6 show the results that were obtained for various scenarios of the sample network. The tables contain profits, real time prices, quantities and bilateral quantities for the generators, and also the prices and demands at the load nodes. The results are obtained as average of 5000 testing episodes which were run after the completion of the learning phase. It can be noted from these Tables that the locational marginal prices (LMPs) are the same for all nodes in unconstrained cases. The LMPs vary among the nodes in constrained cases due to the congestion effect of transmission constraint. In the unconstrained case with perfect competition (Table

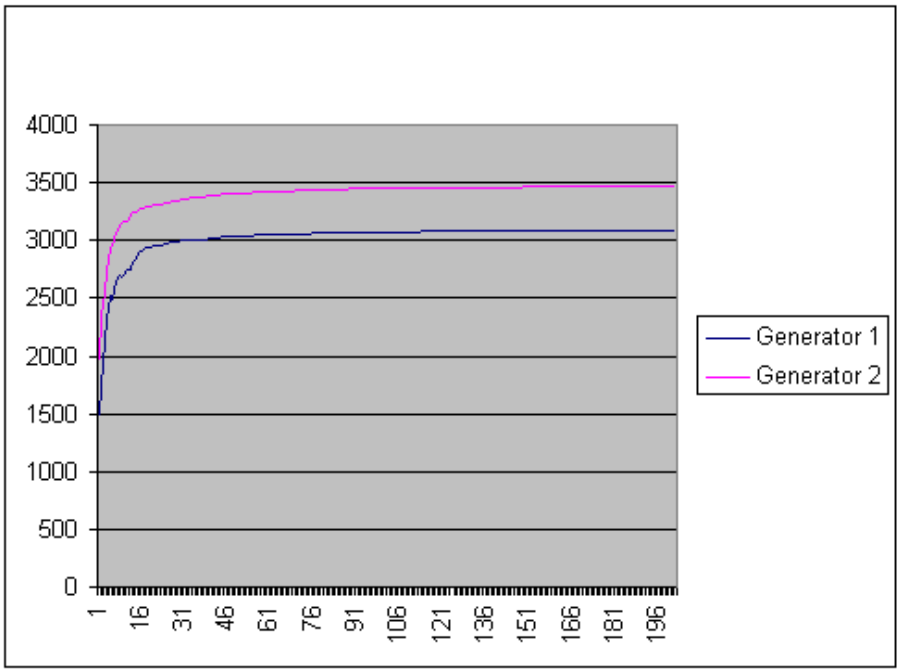

Figure 7.3. Learing Curves of Average Profits under Nash-R Algorithm

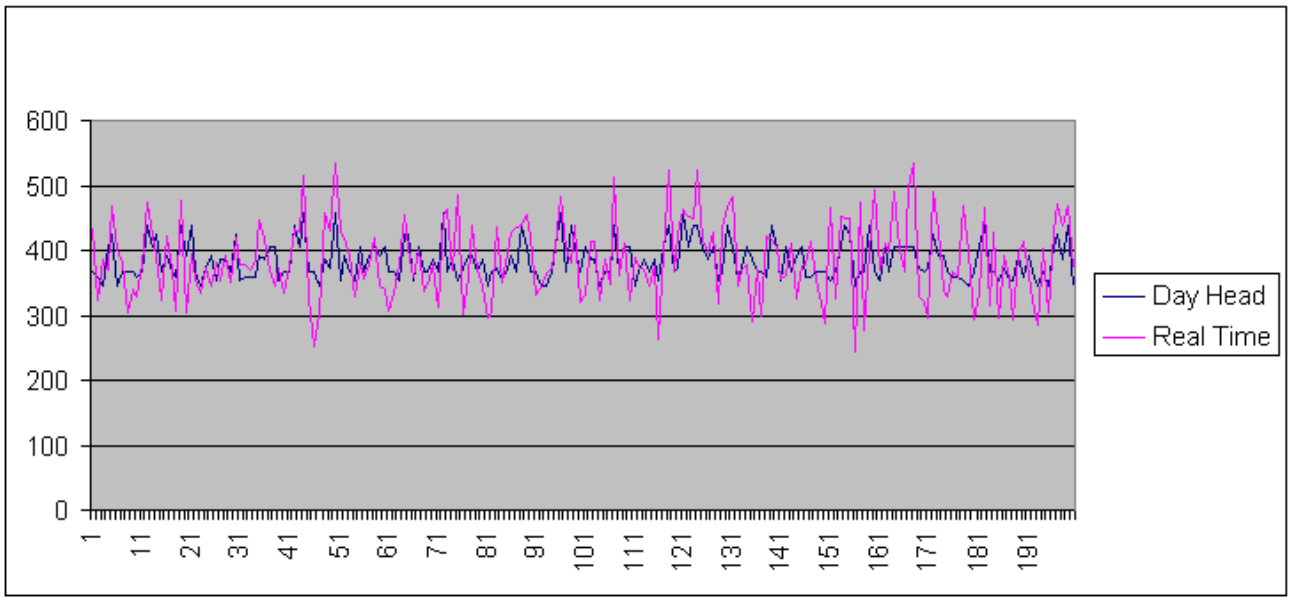

Figure 7.4. Day Ahead and Real Time Demand

Table 7.1. Unconstrained Case with Perfect Competition

	Generator 1	Generator 2	Load 3	Load 4	Load 5
Average Profit	6980	1490			
Average Real Time Price	22.9	22.9	22.9	22.9	22.9
Average Real Time Quantity	1199.4	255.8	451.0	501.6	501.5

Table 7.2. Unconstrained Case with Imperfect Competition

	Generator 1	Generator 2	Load 3	Load 4	Load 5
Average Profit	11893.2	6081.5			
Average Real Time Price	31.7	31.7	31.7	31.7	31.7
Average Real Time Quantity	800.2	483.3	393.9	444.7	444.2
Average Bilateral Quantity	153.8	128.6			

7.1), each generator was paid at \$22.9 which was almost generator 2's marginal-cost price \$23, and generator 1 with lower marginal cost gets to supply a nearly maximum quantity (1200 MW). While in the constrained case (Table 7.3), the generators 1 and 2 get paid at their own marginal-cost prices \$ 18 and \$ 23 respectively. Also, generator 1 is no longer able to obtain the maximum quantity. Thus generator 1 loses its low marginal cost advantage resulting in a drop of profit from \$ 6980 to \$ 980. We can also interpret this difference of \$ 5 in the nodal prices between generator 1 and 2 as the congestion cost to generator 1.

Table 7.3. Constrained Case with Perfect Competition

	Generator 1	Generator 2	Load 3	Load 4	Load 5
Average Profit	980	1490			
Average Real Time Price	18	23	22	22	21
Average Real Time Quantity	830.8	648.4	456.9	507.5	513.9

Table 7.4. Constrained Case with Imperfect Competition and TCC=0

	Generator 1	Generator 2	Load 3	Load 4	Load 5
Average Profit	8269.6	8002.6			
Average Real Time Price	29.1	34.2	33.1	33.0	31.7
Average Real Time Quantity	639.0	626.3	384.4	436.0	444.2
Average Bilateral Quantity	155.6	125.5			

In the case of imperfect competition which represents the real life working condition, generators could bid above marginal-cost price or withhold some capacity to attain more profit. It is considered that market rule prohibits the generators to withhold capacity below a certain level except for outages. The results show that in both unconstrained and constrained cases, the prices with imperfect competition are much higher than the prices with perfect competition. Table 7.2 shows that generator 1 does not supply at its maximum capacity as in Table 7.1, but it gets higher price and profit. All of these results indicate the possible existence of market power.

The load prices ($ 33.1, $ 33.0,$ 31.7) in Table 7.4 are higher than ($ 31.71, $ 31.71, $ 31.7) in Table 7.2. It is noted that the prices of load node 5 did not differ much with or without congestion, while the price of load 3 increased significantly in constrained case. It was also observed that generator 2 dictates much higher prices and profits in constrained cases than in the unconstrained cases. This indicates that generator 2 is able to exercise higher level of market power under congestion.

Results show that, under different scenarios, the quantities of bilateral contracts do not vary much. This seems due to the assumption of risk-neutral strategy towards bilateral contracts. Since we use demand weighted average price as bilateral contract prices, the quantity did not make much difference for different cases.

Table 7.5. Constrained Case with Imperfect Competition and TCC=100

	Generator 1	Generator 2	Load 3	Load 4	Load 5
Average Profit	8650.8	7695.2			
Average Real Time Price	28.2	34.0	32.76	32.7	31.22
Average Real Time Quantity	667.1	605.3	386.5	437.9	447.5
Average Bilateral Quantity	156.4	130.2			

Table 7.6. Constrained Case with Imperfect Competition and TCC=300

	Generator 1	Generator 2	Load 3	Load 4	Load 5
Average Profit	9737.2	7803.2			
Average Real Time Price	28.0	34.1	32.78	32.7	31.17
Average Real Time Quantity	664.7	607.7	386.2	437.7	447.8
Average Bilateral Quantity	159.0	123.3			

Finally, the effect of TCC was investigated. We ignore TCC purchasing cost here since it is a constant in our learning processes. The price of generator 1 drops from \$29.1 (Owning 0MW TCC in Table 7.4) to \$ 28.2 (Owning 100MW TCC in Table 7.5), but the output quantity and profit of generator 1 increased in the latter case. When generator 1 owns TCC 300MW (Table 7.6), the price keeps dropping to \$28.0. It is observed that by owning TCC, generator 1 lowered the bid to lower the price and gain more output, which results in bigger nodal price difference hence more congestion rent.

CHAPTER 8
CONCLUSION AND FUTURE WORK

8.1 Summary of Results

The theoretical framework for modeling and solving stochastic games has been developed and applied to model and examine the bidding behavior of generators in power market.

It is shown that there exist equivalent matrix games to average reward irreducible stochastic games. A simulation based two time scale stochastic approximation scheme using reinforcement learning has been developed. An ODE approach for this two time scale stochastic approximation scheme is presented for convergence analysis. The scheme is tested and then benchmarked with MDP based learning schemes using a grid game.

The convergence analysis requires a strict assumption for the stage matrix games (i.e., uniqueness of the Nash equilibrium value for the stage games). For many problems, it may be difficult to meet the above assumption. But the experimental results suggest that, even when there are multiple Nash equilibrium values for stage games, optimal policies can be attained most of the times by ensuring that the players use identical stage Nash equilibrium value in updating their R-matrices. However we note that our learning scheme requires the computation of Nash equilibria for matrix games during each iteration. This presents a computational challenge.

The stochastic games in this research are noncooperative general-sum games. But the studies can be extended to zero-sum stochastic games by just replacing the value operator as max-min operator, and Assumption 3 is not required for zero-sum case which guarantees the uniqueness of Nash equilibrium value. If the games are purely collaborative, then players can use learning strategies for MDP environment (such as single agent reinforcement learning scheme), since the best interest of one player is the best interest of all other players.

In the existing power market literature, the three primary equilibrium models applied to examine market power are the Cournot, Bertrand and Supply Function Equilibrium (SFE). SFEs are more realistic regarding the competing behavior of firms in electricity markets. The SFEs concept is adopted in our approach, instead of being modeled as one shot games or MDPs, the generators' bidding problems have been modeled as stochastic games. The model is formulated based on the pool market design. It is a bilevel optimization problem. At the upper level, generators choose the parameters of the bidding curves independently and strategically. At the lower level, the market clearing mechanism of the ISO is been formulated as an Optimal power flow problem. The results of our experiments show that the model approach is able to identify market power in a network when it exists. Generators can take advantage of transmission constraints to exert more market power. Results also show that the TCC acquisition can have significant impact on the generators bidding strategies and the resulting profits.

8.2 Future Work

The existence of Nash equilibrium for n players stochastic games is established, but the implementation requires us to have a computational algorithm to solve Nash equilibrium for n-player matrix games. Right now Lemke-Howsen method

[38][74] is used for 2-player games. If an efficient algorithm for n-player matrix games can be found, the experiments can be extended from 2-player to n-player stochastic games. Gambit is a library of game theory software and tools for the construction and analysis of finite extensive and normal form games [74], which shows that *Simp-DivSolve* algorithm is guaranteed to find at least one mixed strategy equilibrium to any n-player game algorithm and works well. This algorithm needs to be examined and performed in further research.

In this research, stochastic games have been related with matrix games. There is another type of game, extensive-form-game, which can be explored to study stochastic games. Nash equilibrium concept is adopted here, other equilibrium concepts can also be of interest to be investigated.

There is a rich literature on repeated game. Stochastic games can also be transformed to study repeat games with single state, which reinforcement learning algorithm helps solve large scale problems. Then a repeat game model of power market can be studied.

Power market is a complex market which includes components that are themselves markets. It seems to be fruitful area for future research. To name a few, demand side bidding can be added into our model, FTR auction is another direction to be explored.

REFERENCES

[1] William W. Hogan. Competitive electricity market design: A wholesale primer. Working paper, Center for Business and Government, John F. Kennedy School of Government, Harvard University, 1998.

[2] M.L. Puterman. *Markov Decision Processes.* John Wiley and Sons, New York Chichester Brisbane Toronto Singapore, 1994.

[3] T.K. Das, A. Gosavi, S. Mahadevan, and N. Marchalleck. Solving semi-Markov decision problems using average reward reinforcement learning. *Management Science*, 45(4):560–574, 1999.

[4] E. Rasmusen. *Games Information, 3rd Edition.* Blackwell publisher, 2001.

[5] R.B. Myerson. *Game Theory– Analysis of Conflict, 3rd Printing.* Harvard University Press, 1997.

[6] L.S. Shapley. Stochastic game, 1953. In H.W Kuhn, editor, *Classics in Game Theory.* Princeton University Press, 1997.

[7] John Nash. Non-cooperative games. *The Annals of Mathematics*, 54(2):286–295, 1951.

[8] J. Filar and K Vrieze. *Competitive Markov Decision Process.* Springer, Verlag New York Berlin Heidelberg, 1997.

[9] E. Altman and F.M. Spieksma. Contraction conditions for average and α-discount optimality in countable state Markov games with unbounded rewards. *Journal of Cognitive Systems Research*, 2:55–66, 1997.

[10] Eitan Altman and Odile Pourtallier. Approximating Nash equilibria in nonzero-sum games. *International Game Theory Review*, 2:155–172, 2000.

[11] V. S. Borkar. Reinforcement learning in Markovian evolutionary games. *Advances in Complex Systems*, 5(1):55–72, 2002.

[12] M.L. Littman. Value-function reinforcement learning in Markov games. *Journal of Cognitive Systems Research*, 2:55–66, 2001.

[13] M.L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *In Proceedings of the Eleventh International Conference on Machine Learning*, pages 151–163, San Francisco, CA, 1994.

[14] R.I. Brafman and M. Tennenholtz. A near optimal polynomial time algorithm for learning in certain classes of stochastic games. *Artificial Intelligence*, 121, 2000.

[15] J. Hu and M. P. Wellman. Multiagent reinforcement learning: Theoretical framework and an algorithm. In *In Proceedings of the Fifteenth International Conference on Machine Learning*, 1998.

[16] M. Bowling and M.M. Veloso. An analysis of stochastic game theory for multiagent reinforcement learning. Technical Report CMU-CS-00-165, Computer Science Department, Carnegie Mellon University, 2000.

[17] K. K. Ravulapati, J. Rao, and T. K. Das. A reinforcement learning approach to stochastic business games. *In review with IIE Transactions on Scheduling and Logistics*, 2002.

[18] J. Rao, K. K. Ravulapati, and T. K. Das. A simulation based approach to study stochastic inventory planning games. *In review with International Journal of Systems Science*, 2002.

[19] Benjamin F. Hobbs. Network models of spatial oligopoly with an application to deregulation of electricity generation. *Operations Research*, 34(3):395–409, 1986.

[20] Steven Stoft. Using game theory to study market power in simple networks. Technical report, Federal Energy Regulatory Commission, http://www.stoft.com, 1998.

[21] Yves Smeers. Computable equilibrium models and the restructuring of the European electricity and gas markets. *Energy Journal*, 18(4):1–31, 1997.

[22] Ross Baldick. Electricity market equilibrium models: The effect of parametrization. *IEEE Transactions on Power Systems*, 17(4):1170–1176, 2002.

[23] Benjamin F. Hobbs, Carolyn B. Metzler, and Jong shi Pang. Strategic gaming analysis for electric power systems: An mpec approach. *IEEE Transactions on Power Systems*, 15(2):638–645, 2000.

[24] Frederick S. Hillier and Gerald J. Lieberman. *Introduction to Operations Research*. McGraw-Hill, 1995.

[25] R. Bellman. The theory of dynamic programming. *Bull. Amer. Math. Soc*, 60:503–516, 1954.

[26] Dimitri P. Bertsekas. Distributed dynamic programming. *IEEE Transactions on Automatic Control*, (3), 1982.

[27] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control.* Athena Scientific, Belmont, MA, 1995.

[28] Z. Ghahramani. Learning dynamic Bayesian networks. In C.L. Giles and M. Gori, editors, *Adaptive Processing of Sequences and Data Structures*, pages 168–197. Springer-Verlag, Berlin, 1998.

[29] R.S. Sutton and A.G. Barto. *Reinforcement Learning.* The MIT press, 1998.

[30] L.P. Kaelbling, M.L. Littman, and A.W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[31] Christopher J.C.H. Watkins and Peter Dayan. Q-learning. *Machine Learning*, 8(3):279–292, 1992.

[32] John N. Tsitsiklis. Asynchronous stochastic approximation and q-learning. *Machine Learning*, 6:185–202, 1994.

[33] J. Abounadi, D.P. Bertsekas, and V. S. Borkar. Learning algorithms for Markov decision processes with average cost. *SIAM Journal on Control and Optimization*, 40(3):681–698, 2001.

[34] J.N. Tsitsiklis and B.V. Roy. Average cost temporal-difference learning. Technical Report LIDS-P-2390, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139, 1997.

[35] A. Gosavi. *An Algorithm for Solving Semi-Markov Decision Problems Using Reinforcement Learning: Convergence Analysis and Numerical results.* PhD thesis, University of South Florida, Department of Industrial and Management System Engineering, 1999.

[36] A. Gosavi. A convergent reinforcement learning algorithm for solving semi-Markov decision problems under average cost for an infinite time horizon. *In press, European Journal of Operational Research*, 2003.

[37] P. Jean-Jacques Herings and Ronald J.A.P Peeters. Stationary equilibria in stochastic games: Structure, selection, and computation. Research memoranda 004, Maastricht : METEOR, Maastricht Research School of Economics of Technology and Organization, 2000.

[38] J. Hu and M. P. Wellman. Nash q-learning for general-sum stochastic games. *to appear, Journal of Machine Learning Research*, 2003.

[39] M. Bowling. Convergence problems of general-sum multiagent reinforcement learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, 2000.

[40] C. Szepesvari and M.L. Littman. A unified analysis of value-function-based reinforcement-learning algorithms. *Neural Comput.*, 11:2017–2059, 1999.

[41] M.L. Littman. Friend or foe Q-learning in general-sum Markov games. In *In Proceedings of the Eighteeth International Conference on Machine Learning*, 2001.

[42] H.J. Kushner and D.S. Clark. *Stochastic Approximation Methods for Constrained and Unconstrained Systems.* Springer-Verlag, New York, 1978.

[43] H.J. Kushner and G.G. Yin. *Stochastic Approximation Algorithms and Applications.* Springer-Verlag, New York, 1997.

[44] Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming.* Athena Scientific, 1996.

[45] V. S. Borkar and S.P. Meyn. The o.d.e method for convergence of stochastic approximation and reinforcement learning. Working paper, Indian Institute of Science, Bangalore, India, 1999.

[46] J. Abounadi, D.P. Bertsekas, and V. S. Borkar. Ode analysis for q-learning algorithms. Lids report, MIT, Cambridge, MA, 1996.

[47] J. Abounadi, D. Bertsekas, and V. S. Borkar. Stochastic approximation for nonexpansive maps: Application to q-learning algorithms. *SIAM Journal on Control and Optimization*, 41(1):1–22, 2002.

[48] Walter Wolfgang. *Ordinary Differential Equations.* Springer-Verlag, New York, 1998.

[49] V. S. Borkar. Stochastic approximation with two-time scales. *System and Control Letters*, 29, 1997.

[50] V. S. Borkar and K. Soumyanath. An analog scheme for fixed point computation, part i: theory. *IEEE Transactions Circuits and Systems I. Fundamental Theory and Application*, 44, 1997.

[51] V. S. Borkar. Asynchronous stochastic approximations. *SIAM J. Control Optim.*, 36(3):840–851, May 1998.

[52] Steven Stoft. *Power System Economics.* IEEE Press, John Wiley and Sons, New York, 2000.

[53] Sally Hunt. *Making Competition Work In Electricity.* John Wiley and Sons, New York, 2002.

[54] Hung po Chao and Robert Wilson. Design of wholesale electricity markets. Draft 990101, Electric Power Research Institute, 1999.

[55] Pennsylvania-New Jersey-Maryland (PJM) Interchange. Technical report, http://www.pjm.com.

[56] Rajnish Kamat and Shmuel S. Oren. Two-settlement systems for electricity markets: Zonal aggregation under network uncertainty and market power. Working paper, pwp-091, University of Carlifornia Energy Institute, 2002.

[57] William W. Hogan. Electricity market restructuring: Reforms of reforms. In *20th Annual conference, center for research in regulated industries*, May 2001.

[58] William W. Hogan. Nodes and zones in elelctricity markets: Seeking simplified congestion pricing. In Hung-po Chao and Hillard G. Huntington, editors, *Designing Competitive Electricity Markets.* Kluwer Academic Publishers, 1998.

[59] Jean Tirole. *The Theory of Industrial Organization.* The MIT Press, 1988.

[60] Aleksandr Rudkevich, Max Duckworth, and Richard Rosen. Modeling electricity pricing in a deregulated generation industry: The potential for oligopoly pricing in a poolco. *Energy Journal*, 19(3):19–48, 1998.

[61] Benjamin F. Hobbs. Linear complementarity models of Nash-Cournot competition in bilateral and POOLCO power markets. *IEEE Transactions On Power Systems*, 16(2):194–202, May 2001.

[62] Steven Stoft. Financial transmission rights meet Cournot: How TCC's curb market power. *Energy Journal*, 20(1):1–23, 1999.

[63] Lance B. Cunningham, Ross Baldick, and Martin L. Baughman. An empirical study of applied game theory: Transmission constrained Cournot behavior. *IEEE Transactions On Power Systems*, 17(1):166–172, 2002.

[64] Carolyn Metzler, Benjamin F. Hobbs, and Jong-Shi Pang. Nash-cournot equilibria in power markets on a linearized dc network with arbitrage: formulations and properties. *Networks and Spatial Economics (forthcoming)*, 2003.

[65] Richard J. Green and David M. Newbery. Competition in the british electricity spot market. *The Journal of Political Economy*, 100(5):929–953, 1992.

[66] Richard J. Green. Increasing competition in the british electricity spot market. *The Journal of Industrial Economics*, 44(2):205–216, 1996.

[67] Ross Baldick, Ryan Grant, and Edward Kahn. Linear supply function equilibrium: Generalizations, application, and limitations. Working paper pwp-078, 2000.

[68] Christopher J. Day, Benjamin F. Hobbs, and Jong-Shi Pang. Oligopolistic competition in power networks: A conjectured supply function approach. Working paper pwp-090, 2002.

[69] Jong-Shi Pang, Benjamin F. Hobbs, and Christopher J. Day. Properties of oligopolistic market equilibria in linearized dc power networks with arbitrage and supply function conjectures. In *Proceedings of the IFIP TC7 20th Conference on System Modeling and Optimization, July 23-27, Trier, Germany*, 2003.

[70] Haili Song, Chen-ching Liu, Jacques Lawarree, and Robert W. Dahlgren. Optimal electricity supply bidding by Markov decision process. *IEEE Transactions on Power Systems*, 15(2):618–624, May 2000.

[71] G.R. Gajjar, S.A. Khaparde, P. Nagaraju, and S.A. Soman. Application of actor-critic learning algorithm for optimal bidding problem of a Genco. *IEEE Transactions on Power Systems*, 18(1):11–18, 2003.

[72] Ray D. Zimmerman and Deqiang Gan. Matpower–a Matlab power system simulation package. User's manual, Power System Engineering Research Center (PSERC), School of Electrical Engineering, Cornell University, http://blackbird.pserc.cornell.edu/matpower/, 1997.

[73] Tarjei Kristiansen. Utilizing matpower in optimal power flow. *Modeling, Identification and Control (MIC), Special Issue on Numverical Software*, 2003.

[74] Richard Mckelvey, Andrew Mclennan, and Theodore Turocy. Gambit: Software tools for game theory. Technical report.

ABOUT THE AUTHOR

Jun Li received a Bachelor's Degree in Control Engineering from Central South University of Technology (Changsha, China) in 1996 and a M.S. in System Engineering from Huazhong University of Science and Technology (Wuhan, China) in 1999. Then she came to Univeristy of South Florida and entered the Ph.D program in Department of Industrial Management and System Engineering.

Jun Li served as Teaching Assistant and Research Assitant in her Ph.D years. She also taught Engineering Statistics for three semesters. Her research focuses on Operations Research, and her fields of interest are Optimization,Stochastic Processes, Machine Learning, Design of Statistic Models, Modeling and Simulation, Computable Models of Market Equilibria.