Modeling of suppliers' learning behaviors in an electricity market environment

by

Nanpeng Yu

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Electrical Engineering

Program of Study Committee:
Chen-Ching Liu, Major Professor
Leigh Tesfatsion
Venkataramana Ajjarapu

Iowa State University

Ames, Iowa

2007

UMI Number: 1447491

UMI®

UMI Microform 1447491

TABLE OF CONTENTS

iv

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

In order to study the strategic bidding behavior of electricity suppliers and test the electricity market design, the Day-Ahead electricity market is modeled as a multi-agent system with interacting agents including supplier agents, load serving entities, and a market operator. The profit maximizing objective of a supplier naturally requires the player to learn from its bidding experience and behave in an anticipatory way. With volatile Locational Marginal Prices (LMPs), ever-changing transmission grid conditions, and incomplete information about other market participants, decision making for a supplier is a complex task. A learning algorithm that does not require an analytical model of the complicated market but allows suppliers to learn from the past experience and act in an anticipatory way is a suitable approach to this problem. Q-Learning, an anticipatory reinforcement learning technique, has all these desired properties. Therefore, it is used in this research to model the learning behaviors of electricity suppliers in a Day-Ahead electricity market. Simulation of the market clearing results under the scenarios in which agents have learning capabilities is compared with the scenario where agents report true marginal costs. It is shown that, with Q-Learning and strategic gaming, electricity suppliers are making more profits compared to the scenario without learning. As a result, the LMP at each bus is substantially higher.

CHAPTER 1. INTRODUCTION

1.1 Research Motivation and Background

Strategic bidding is an important issue that has been raised in many electricity markets, such as England and Wales wholesale electricity market in the 90s, and California electricity market during the energy crisis. Electricity prices change as a result of transmission network congestion, which may be caused by strategic bidding or heavy load. For PJM, the total congestion costs were $750 million in 2004 and $2.09 billion in 2005. Learning may allow larger electricity suppliers to use their market power and bid strategically. The most recognizable case is in England and Wales electricity market where the two largest firms bid strategically in a non-competitive manner to keep the price significantly higher than the competitive level. In California [1], electricity expenditure in the wholesale market increased from $2.04 billion in summer 1999 to $8.98 billion in summer 2000. It is estimated that 59% of this increase was due to increased market power. Learning to bid in the wholesale market is also crucial for smaller electricity suppliers who have a desire to recover the cost of their investment in generation by avoiding over or under-bidding. Therefore, research on the learning behavior of electricity suppliers will provide insights into gaming in the market. This may allow market designers to develop appropriate market rules to discourage strategic bidding and enhance the market efficiency.

Researchers have used various learning methods to model electricity suppliers' behavior. The learning configuration for suppliers in [2] is a version of stochastic reactive reinforcement learning developed by Alvin Roth and Ido Erev. In this configuration, agents have finite fixed action domains, are backward looking, and rely entirely on response learning. Average reward γ-greedy reinforcement learning was used in [3] to model the learning and bidding processes of suppliers. With this scheme, each supplier uses greedy

selection as its action choice rule with probability (1- γ), and random action selection with probability γ. Thus, γ determines the trade-off between exploitation of available information and exploration of untested actions. The trading agents modeled in [4] use GP-Automata to compute their bidding strategies for the current market conditions.

The objective of this thesis is to model the learning behaviors of suppliers in the electricity market. This thesis is focused on how to model electricity suppliers' learning behavior by Q-Learning. The electricity market will be modeled as a Multi-agent system with three types of interacting agents: supplier agents, load serving entities, and the market operator. The effect of the suppliers' learning behavior on the market clearing results is examined. In addition, load serving entities with a simple demand-side response behavior are considered in this multi-agent electricity market environment.

1.2 Thesis Organization

This thesis consists of five chapters. Chapter 2 provides a review on multi-agent systems, their applications to power systems and presents a multi-agent technique to model the Day-Ahead electricity market. Chapter 3 presents a literature review on multi-agent learning algorithms and applied Q-Learning to the modeling of suppliers' learning behavior. Chapter 4 presents a case study of the proposed methods on a 5-bus transmission grid. Chapter 5 includes the conclusion of this research, and the suggestions for future work.

1.3 Contents of This Thesis

The objective of this thesis is to examine the learning behavior of electricity suppliers' and its impact on the Day-Ahead electricity market. In addition, this thesis is concerned with the development of a Day-Ahead electricity market model using the multi-agent methodology.

Chapter 2 provides an introduction to multi-agent systems and a review of their applications to power systems. This chapter discusses the Foundation for Intelligent Physical

Agents, a popular standard that is used in most industrial and commercial multi-agent system applications. A multi-agent system model of the Day-Ahead electricity market is presented. The market operator, load serving entities, and supplier agents' models are incorporated.

Chapter 3 provides a literature review of the multi-agent learning algorithms. Multi-agent learning algorithms are classified into three categories: Model-based approaches, Model-free approaches, and regret minimization approaches. In this research, Q-Learning, an anticipatory reinforcement learning technique, is selected for the study of the electricity suppliers' learning behavior.

Chapter 4 presents a case study of the proposed modeling methods on a 5-bus transmission grid. Simulation results of both the no-learning scenario and two learning scenarios are presented and compared. Simulation results indicate that Q-Learning helps electricity suppliers learn how to bid strategically under the condition of a simple demand-side response model. With Q-Learning capabilities, electricity suppliers find their way to make more profits in the long term by sacrificing the immediate profits.

Chapter 5 provides the key conclusions of this research and suggestions for the future work.

CHAPTER 2. MULTI-AGENT SYSTEMS

2.1 Introduction

2.1.1 What is an Agent?

There is not a single definition of an agent that is universally accepted. However, the following definition from the Wooldridge and Jennings [5] is commonly adopted in the field. An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives.

Note that the agent discussed here is actually a software entity. The two basic properties that an agent must have are autonomous, and situated. Autonomous means that a software agent must be able to operate without the direct intervention of people or other agents and has control over its own action and internal state. Situated means that a software agent is situated in some type of environment. These environments may be dynamic, unpredictable and unreliable.

According to Jennings and Wooldridge, to make an agent "intelligent", the software agent should be able to take flexible autonomous actions in order to meet its design objectives [6]. Flexible means an agent is reactive, proactive and social. By reactive, it is meant that the agent perceives its environment and responds in a timely fashion to changes that occur in the environment. By proactive, it is meant that the agent does not simply act in response to its environment but is able to achieve a goal by taking the initiative. By social, it is meant that in order to achieve its goals, the agent interact with people or other agents.

2.1.2 What is a Multi-Agent System?

A Multi-Agent System (MAS) is an organization of heterogeneous and self-motivated agents that interact with one another. The agents in MAS could have conflicting interests or they could coordinate with one another to accomplish the same mission.

2.1.3 When and Why are Agents Useful?

Reactive systems that maintain an ongoing interaction in some environment are inherently more difficult to design and implement [6]. One can classify these systems into three categories: open systems, complex systems, and ubiquitous computing systems. Some of the characteristics that these systems have are dynamic, highly complex, and unpredictable. With a better encapsulation, and modularity, the agent paradigm can develop a number of modular components that are specialized at solving a particular aspect of the complex, unpredictable system. In addition, with reactiveness and proactivenss, an agent can be relied upon to persist in achieving its goals, trying alternatives that are appropriate to the changing environment without continuous supervision and checking [7]. The Agent technology also helps to improve the efficiency of software development, especially when the data, control, expertise, or resources are physically or logically distributed.

2.1.4 Agent Oriented Programming versus Object Oriented Programming

Agent Oriented Programming has a higher level of encapsulation than Object Oriented Programming. An object encapsulates some state, and has some control over its own state in that it can only be accessed or modified via the methods that the object provides [6]. An agent encapsulates not only state, but also its own behavior. In contrast, an object does not encapsulate behavior: in other words, it has no control over the execution of its own methods. Note that the autonomous property of an agent allows it to have control over its

own actions. Due to this distinction, one should not think of agents as invoking methods (actions) on agents. Rather, agents are requesting actions to be performed [6]. The agent itself could also decide whether to act upon the request.

2.1.5 Applications of Multi-Agent Systems in Power Engineering

The MAS technologies are being applied to an increasingly wide range of applications in power systems. These applications falls into four major categories: Modeling and Simulation, Monitoring and Diagnostics, System Restoration and Reconfiguration, and System Controls.

2.1.5.1 Modeling and Simulation

The MAS technologies have been used for modeling and simulation of different aspects of power engineering. A major application is the simulation of restructured electricity market. With the embedded learning capabilities, agents that are autonomous, proactive and reactive are well suited for modeling of various market participants in the electricity market. It has been shown that a well-designed software agent can emulate the offer behavior of human agents [8]. Thomas et al. proposed to use software agents to test electricity markets [9]. Five standardized agents – four different types of speculators and a marginal cost offer agent are designed to compete with human subjects in a central auction market. A multi-agent trading platform for electricity contract market is constructed [10]. Customers' response under time-of-use electricity pricing is studied in a Multi-Agent system [11]. An agent-based model is designed in [12] to support decentralized generation expansion in electricity market.

2.1.5.2 Monitoring and Diagnostics

Major challenges in the power system diagnostic and monitoring applications include how to handle large volumes of raw data from different sources, how to convert those raw

data into meaningful information, and how to provide power engineers with correct information to support the decision making. These challenges could be overcome with the help of MAS technology. In [13], the authors designed and constructed the Protection Engineering Diagnostic Agents system (PEDA) for automated disturbance diagnosis. The PEDA system was implemented as an on-line post-fault analysis system for the Scottish Power Systems which significantly reduces the data retrieval, collection and interpretation burden on protection engineers. Condition Monitoring Multi-Agent System (COMMAS) for transformer condition monitoring was developed in [14]: the system is intended to provide decision support for operational engineers. A MAS was designed for fault detection, diagnostics, and prognostics of navy All-Electric Ships (AES) [15]. This fault diagnosis and prognosis tool will improve the reliability, availability, and survivability of AES, and support the drastic manning reduction requirements for future navy ships.

2.1.5.3 System Restoration and Reconfiguration

Nagata et al. suggests a multi-agent approach to restore a power system to a target network that has as many buses as possible [16]. In the proposed MAS, local bus agents formulate a restoration plan through negotiation, and then check the restoration plan with a global facilitator. In [17], a multi-agent-based approach for navy ship system electric power restoration is provided to restore the capacity as much as possible to serve the loads. Through negotiation among three different types of agents, the system can perform the restoration work using local information without a central controller.

2.1.5.4 System Controls

The distributed properties of MAS, and potential of local decision making make it better suited for certain control scenarios in a power system relative to conventional centralized control. There are some common features of those control scenarios. The system is highly complex so that optimum control is difficult to accomplish even with centralized

control and the control decision making time is limited. For example, in microgrid control, the operation of micro-sources, storage devices, and controllable loads is highly complex. In [18], MAS approach was used to control the microgrid. In the proposed method, Microgrid Central Controller coordinates the local controllers and decides whether to connect to the main grid, whereas Local Controllers control the distributed energy resources, production and storage units, and some of the local loads. Jung et al. proposed an application of multi-agent system technologies for the development of strategic power infrastructure defense (SPID) system that is designed to prevent catastrophic failures and cascading sequences of events, an application of which is on adaptive load shedding [19].

2.2 The Foundation for Intelligent Physical Agents (FIPA)

FIPA was established in 1996 as an international non-profit association to develop a collection of standards relating to software agent technology [20]. FIPA was formally reincorporated in mid-2005 as a standards committee of the IEEE Computer Society, lending credibility to the use of FIPA as standards for industrial and commercial multi-agent system applications. FIPA standards govern the basics of an agent architecture, including agent lifecycle management, inter-agent message transport, message structure, inter-agent interaction protocols, and security. Users are left with the flexibility to design an agent that accomplishes its goals. The most important ideas of FIPA are agent communication, agent management, and agent architecture.

2.2.1 Agent Communication

The FIPA-Agent Communication Language (ACL) states the message representing actions or communicative acts that are called speech acts or performatives [20]. There are 22 performatives in communicative act library, which has 4 basis performatives: request, inform, confirm, and disconfirm. FIPA also standardizes a set of interaction protocols such as requests, query to coordinate multi-message actions. Different content languages can be

employed to express the content of FIPA-ACL. The most popular language FIPA semantic language (SL) is standardized and specified in [21].

2.2.2 Agent Management

The second fundamental aspect of FIPA is addressed by agent management that establishes the logical reference model for creation, registration, location, communication, migration and operation of the agents. It specifies how a FIPA compliant agent can exist, operate and be managed. A FIPA compliant Agent Platform (AP) provides the physical infrastructure that consists of the machines, operating system, FIPA agent management components, the agents themselves, and any additional support software [20]. An AP has two utility agents: the Agent Management System (AMS) and the Directory Facilitator (DF). The AMS is mandatory, as it allocates agent identifiers (AIDs) to each agent that registered with it, keeps track of the status of an agent, and terminates the life an agent when it deregistered. The DF is optional; it provides yellow page services that allow every agent to advertise its services on a non-discriminatory basis. An AP also provides a Message Transport Service (MTS) to transport FIPA-ACL messages between agents on the same platform or within different platforms.

2.2.3 Agent Architecture

The FIPA Abstract Agent Architecture provides a common, unchanging point of reference for FIPA-compliant implementations that capture the most critical and salient features of an agent system [22]. Most important mandatory items specified in the architecture are the ACL message structure, message transport, agent directory services, and service directory services. As described in Section 2.2.2, the communication between two agents relies on a message transport service that transports FIPA-ACL messages. As mentioned in Section 2.2.1 the structure of a message is a set of key values written in FIPA-ACL. The content of the message is expressed in a content language, such as FIPA-SL or

FIPA KIF [22]. Essentially, the two directory services allow agents to register themselves or the services that they provide, and to search for specific agents for services.

2.3 Multi-Agent Approach to Day-Ahead Electricity Market Modeling

The Day-Ahead electricity market is modeled as a multi-agent system with three types of agents interacting with one another. These agents are supplier agents, Load Serving Entities (LSEs), and a Market Operator (MO). On the morning of day D, the MO sends messages to all supplier agents and LSEs to ask them to participate in the Day-Ahead market. Upon receiving the messages, the supplier agents and LSEs reply with their supply offers and demand bids. During the afternoon, the MO runs a market-clearing algorithm (similar to an optimal power flow), to match supply to demand and determine dispatch schedules and LMPs. At the end of the process, the MO sends messages to supplier agents and LSEs to inform them the dispatch schedules and LMPs for day D+1. The interaction among the MO, LSEs and supplier agents is shown in Fig. 1. The multi-agent system described above is developed with the Java Agent DEvelopment Framework (JADE) platform.
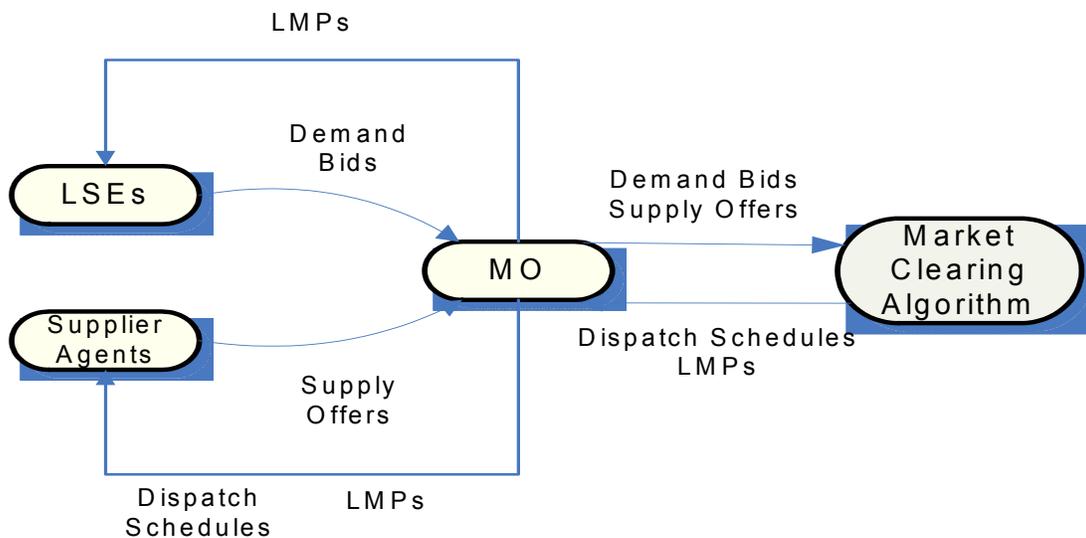

Figure 1. Multi-Agent Day-Ahead Market Environment

2.3.1 Load Serving Entity Model

LSEs purchase bulk power from the Day-Ahead market to serve load. Without loss of generality, it is assumed that LSEs do not have generation units and one LSE only serves load at one location in the power system. Suppose that the number of LSEs in the Day-Ahead market is J. On day D, LSE j submits a load profile for day D+1. This load profile specifies 24 hours of MW power demand $P_{Lj}(H)$, H = 0, 1 … 23.

It is assumed that demand-side response is available to LSEs. The demand-side response works as follows. If the day D peak hour LMP, $LMP_{Lj}(H_{peak})$, at the bus where LSE j is serving load, is higher than a critical value, then LSE j reduces its peak hour demand for day D+1 by 2%. If this LMP does not exceed the critical value, LSE j will not curtail its peak hour demand. Therefore, each LSE has two states. If the LMP at its node is below the critical value, it is in state 0, i.e., $S_{Lj} = 0$, and it will submit a normal load profile for day D+1. If the LMP at its node is above the critical value, it is in state 1, i.e., $S_{Lj} = 1$, and it will submit a curtailed load profile for day D+1.

2.3.2 Supplier Agent Model

Supplier agents sell bulk power to the Day-Ahead market. For simplicity, it is assumed that each supplier agent has only one generation unit. However, this model can be extended to permit suppliers with multiple generation units. Suppose the number of supplier agents in the Day-Ahead market is I, and the MW power output of generator i in some hour H is p_{Gi}. Generator i has lower and upper limits denoted by $p\min_i$ and $p\max_i$ for its hourly MW power output. For generator i, the hourly total production cost $C_i(p_{Gi})$ for production level p_{Gi} is represented by a quadratic form:

$$C_i(p_{Gi}) = a_i \cdot p_{Gi} + b_i \cdot p_{Gi}^2 + F_i \tag{2.1}$$

where a_i, b_i and F_i (pro-rated fixed cost) are given constants. By taking derivatives on both sides of (2.1), the marginal cost function for Generator i is obtained, i.e.,

$$MC_i(p_{Gi}) = a_i + 2 \cdot b_i \cdot p_{Gi} \tag{2.2}$$

On each day D, the supplier agents submit to the Day-Ahead market a supply offer for day D+1 that includes two components. The first component is its reported marginal cost function given by:

$$MC_i^B(p_{Gi}) = a_i^B + 2 \cdot b_i^B \cdot p_{Gi} \tag{2.3}$$

The second component is its hourly MW power output upper limit, denoted by $p\max_i^B$. Suppose, on day D, supplier agents submit their supply offers for day D+1 to the MO, and the market clearing program calculates LMPs and dispatch schedules. Let $LMP_{Gi}(H)$ denote the LMP for hour H at the bus where suppler i's generation unit is located, and let $p_{Gi}^*(H)$ denote the MW power output for hour H in the dispatch schedule posted by the MO. Supplier agent i's profit on day D is obtained by summing 24 hours of profits on that day:

$$\pi_{iD} = \sum_{H=0}^{23} [p_{Gi}^*(H) \cdot LMP_{Gi}(H) - C_i(p_{Gi}^*(H))] \tag{2.4}$$

The Accumulated profit of generator i on day D is given by:

$$AP_i(D) = AP_i(D-1) + \pi_{iD} \tag{2.5}$$

2.3.3 Market Operator Model

The MO for this Day-Ahead market is responsible for clearing the market based on the information submitted by LSEs and supplier agents. The MO uses a market clearing algorithm to determine the LMP at each bus and MW power output for each generation unit at each hour. Since only MW power is considered in this model, a DCOPF problem can be formulated as follows:

$$\min \sum_{i=1}^{I} (a_i^B \cdot p_{Gi} + b_i^B \cdot p_{Gi}^2) \tag{2.6}$$

subject to

$$P_k - P_{gk} + P_{dk} = 0, \qquad k = 1,...N_b \tag{2.7}$$

$$|H\delta| \leq F_{max} \tag{2.8}$$

$$p\min_i^B \leq p_{Gi} \leq p\max_i^B \tag{2.9}$$

where N_b denotes the total number of buses in the system, P_k represents the net power injection at bus k, P_{gk} denotes the total MW power generation at bus k, P_{gk} is the total MW demand at bus k, H denotes the line flow matrix, δ denotes the vector of voltage angle differences, and F_{max} is the vector of maximum line flows.

The objective of the DCOPF is to minimize the total variable generation cost based on supplier offers and LSE bids. The constraints are MW power balance constraints for each bus $k = 1,...N_b$, MW thermal limit constraints for each line, and MW production limits for each generator $i = 1,...I$. The DCOPF program of MATPOWER [23] applicable to large-scale power systems is used in this research.

2.4 Summary

This chapter provides an introduction to the multi-agent system technology by answering several basic questions, i.e., what is an agent, what is a multi-agent system, when and why are agents useful. In section 2.1.4, the agent oriented programming is compared with object oriented programming. Section 2.1.5 is an overview of the applications of multi-agent system in four areas of power engineering field: Modeling and Simulation, Monitoring and Diagnostics, System Restoration and Reconfiguration, and System Controls. Some core concepts of the FIPA specifications are discussed in section 2.2.

With the multi-agent system technology, the Day-Ahead electricity market is modeled as a multi-agent system with three types of agents: supplier agents, LSEs, and the Market Operator. Since JADE is an implementation of FIPA specification, it was used to develop the proposed multi-agent system. The models for supplier agents, LSEs, and MO are presented in detail in section 2.3.1-2.3.3.

CHAPTER 3. MULTI-AGENT LEARNING ALGORITHMS

3.1 Introduction

A basic question that was often asked by researchers in the field of Artificial Intelligence (AI) is how to design a learning algorithm that allows a machine to learn about the environment in which it resides and to maximize its chances of success.

Insightful observations and tools from statistics, computer science, psychology, cognitive science, and logic are utilized to develop learning algorithms that are implemented on machines in different contexts. Some of the key algorithms developed for single-agent learning are Artificial Neural Networks (ANN), Bayesian Learning (BL), Computational Learning Theory (CLT), Genetic Algorithms (GA), Analytical Learning (AL), and Reinforcement Learning (RL). The applications of these algorithms range from chess-play computer program Deep Blue that beats the world champion Garry Kasparov, to data mining programs that learn to approve bank loans to lower the bad loan rate, to autonomous cars that learn to drive safely from door to door.

In recent years, multi-agent learning takes the place of single-agent learning and becomes an important issue of learning that attracts the attention of many researchers in both computer science and game theory.

3.2 Literature Review

Three major classes of learning techniques were developed—the first one is representative of work in game theory, the second one is typical in AI, and the last one seems to have drawn equal attention from both communities [24]. The three approaches are model-based, model-free, and regret minimization approaches.

3.2.1 Model-based Approaches

In model-based learning algorithms, the presence of other decision making agents in the learning environment is taken into account. It usually begins with some models of the opponents' strategy, and then starts an iterative three-step learning process. First, it computes and plays the best action based on the model of opponents' strategies. Then, it observes the opponent's actions and updates the models of the opponents' strategies. Afterwards, it goes back to the first step.

The early model-based learning algorithm well known in game theory is called fictitious play. The model rests on traditional statistician's philosophy of basing future decisions on the relevant past history [25]. The opponent is assumed to pick an action at each turn according to a stationary probability distribution function (PDF). The algorithm keeps track of opponent's play, and chooses an action that is optimum against the estimates of the opponents' PDF based on the relative frequencies.

Fictitious play only allows the agent to exploit all the information that it has so far, and play the "optimum" action. The variants of fictitious play such as smooth fictitious play [26] and exponential fictitious play [27] allow the agent to explore other actions that is not "optimum".

3.2.2 Model-free Approaches

In model-free approaches, Q-Learning [28] allows agents to learn how to act in a controlled Markovian domain with unknown transition functions. A controlled Markovian domain implies that the environment is Markovian in the sense that state transition probabilities from state x to state y only depends on x, y and the action a taken by the agent, and not on other historical information. It works by successively updating estimates for the Q-values of state-action pairs. The Q-value $Q(x,a)$ is the expected discounted reward for taking action a at state x and following an optimal decision rule thereafter. Once these

estimates have converged to the correct Q-values, the optimal action in any state is the one with the highest Q-value.

By the procedure of Q-Learning, in the nth step the agent observes the current system state x_n, selects an action a_n, receives an immediate payoff r_n, and observes the next system state y_n. The agent then updates its Q-value estimates using a learning parameter α_n and a discount factor γ [28] as follows:

If $x = x_n$ and $a = a_n$,

$$Q_n(x,a) = (1-\alpha_n)Q_{n-1}(x,a) + \alpha_n[r_n + \gamma V_{n-1}(y_n)] \tag{3.1}$$

Otherwise,

$$Q_n(x,a) = Q_{n-1}(x,a) \tag{3.2}$$

where $V_{n-1}(y) \equiv \max_b\{Q_{n-1}(y,b)\}$ $\tag{3.3}$

It is proven by Watkins in [29] that if (1) the state and action-values are discrete, (2) all actions are sampled repeatedly in all states, (3) the reward is bounded, (4) the environment is Markovian and (5) the learning rate decays appropriately, then the Q-value estimates converge to the correct Q-values with probability 1.

The Q-Learning algorithm can be extended to the multi-agent environment by redefine the Q-values as a function of all the agents' actions:

$$Q_n(x,\vec{a}) = (1-\alpha_n)Q_{n-1}(x,\vec{a}) + \alpha_n[r_n + \gamma V_{n-1}(y_n)] \tag{3.4}$$

However, in the contexts where the actions taken by other agents are unknown such as the electricity market, it is impossible to apply this variation of Q-Learning algorithm. Therefore, in the above stated contexts, the only option left is to extend the Q-Learning to the multi-agent environment by having each agent simply ignore the other agents and pretend the environment is Markovian. The theoretical proof of convergence to the correct Q-values no longer holds when an opponent adapts its strategy based on the past experience. It is reasonable to expect that such a strong convergence result no long holds, in a non-Markovian environment where each agent is learning others' strategy.

3.2.3 Regret Minimization Approaches

In the regret minimization model, agents adjust their strategies probabilistically. This adjustment is guided by "regret measures" based on observations of the past period [30]. The assumption made in this model is that each agent knows the past history of all other agents, as well as its own payoff matrix. An instance of the no-regret learning algorithm is presented below. The regret of agent i for playing the sequence of actions s_i instead of playing action a_j, given that the opponents played the sequence s_{-i} is defined as $r_i^t(a_j, s_i)$ [24].

$$r_i^t(a_j, s_i | s_{-i}) = \sum_{k=1}^{t} R(a_j, s_{-i}^k) - R(s_i^k, s_{-i}^k) \tag{3.5}$$

At each round, an agent may either continue choosing the same strategy as in the previous round, or switch to other strategies that have positive regret with a probability proportional to $r_i^t(a_j, s_i)$.

3.3 Modeling of Suppliers' Learning Behavior by Q-Learning

A Generation Company (GENCO) usually has several generation plants located at different buses of the system. For simplicity, Q-Learning is used to model electricity suppliers that are assumed to have only one generation unit. Nevertheless, by a similar approach, Q-Learning could be implemented for supplier agents with multiple generation units at different locations.

A novel approach to the implementation of Q-Learning for a supplier agent is presented here. The supplier agent views the Day-Ahead market as a complex system with different states. The system state on day D, X^D, is defined as a vector for the states of all LSEs. Hence the state vector on day D can be expressed as $X^D = \{S_{L1}, S_{L2}, ..., S_{LJ}\}$, where J is the number of LSEs. The cardinality of the state space is 2^J since each LSE has two states, i.e., reduced peak load or not based on demand-side response. Electricity suppliers might have market power. Thus, it is assumed that supplier agents are capable of forecasting the LSEs' states. In other words, the state vector is predictable by the supplier agents.

The action domain of supplier agent i, AD_i, is defined as a vector of bidding information. This vector consists of the marginal cost function parameters a_i^B, $2 \cdot b_i^B$ and the hourly MW output upper limit $p\max_i^B$. The cardinality of the action domain, $M^a \times M^b \times M^{\max}$, is given by the product of the number of possible a_i^B, $2 \cdot b_i^B$ and $p\max_i^B$ values.

Consider the beginning of each day D. A supplier agent first makes a prediction of the system state, which is represented by x. It next chooses an action according to a Gibbs/Boltzmann probability distribution, i.e.,

$$p_D(x,a) = e^{Q(x,a)/T_D} / \sum_{b \in AD_i} e^{Q(x,b)/T_D} \tag{3.6}$$

where T_D, which depends on D, is a temperature parameter that models a decay over time.

Having chosen an action a, the supplier agent will submit its supply offer to the MO. Once the market is cleared, the supplier agent will receive its reward, which is the profit for day D+1. Then the agent will use this reward to update its Q-value estimates according to equations (3.1) to (3.3). The Q-value estimates of an agent are said to have converged if under all states x the agent chooses some action with probability 0.99 or higher. If the Q-value estimates of all the agents have converged, the simulation terminates.

The parameters that are used to implement the Q-Learning algorithm are set in the following way:

Discount factor $\gamma = 0.7$

Learning parameter α for a state-action pair (x,a) is set to be $\alpha = 1/T_{(x,a)}{}^\omega$, where $T_{(x,a)}$ is the number of times that action a has been taken in state x.

$\omega = 0.77$

The temperature parameter T_D is given by: $1/T_D = 1.7 \times 10^{-9} \times (D)^6$, where D is the number of days that have currently been simulated.

The cardinality of the action domain is $M^a \times M^b \times M^{\max} = 4 \times 4 \times 4$, in which a_i^B and b_i^B range from 1 to 3 times their true values, and $p \max_i^B$ ranges from 97% to 100% of the true upper limit.

3.4 Summary

In this chapter, the multi-agent learning techniques are organized into three categories: model-based approaches, model-free approaches and regret minimization approaches. Fictitious play, Q-Learning, and no-regret learning are described as representative of each of the approaches. Both model-based and regret minimization approaches assume that each agent knows all other agents' historical actions. However, this assumption is not valid in the electricity market context. Therefore, Q-Learning in the model-free approaches is selected to model the learning behavior of electricity supplier agents. A novel approach to the implementation of Q-Learning for an electricity supplier agent considering a simple demand-side response model of the Load Serving Entities is presented.

CHAPTER 4 CASE STUDIES AND RESULTS

4.1 Test System

The 5-bus transmission grid used here for simulation is taken from ISO-NE/PJM training manuals, where it is used to illustrate the determination of Day-Ahead market LMP solutions. A one-line diagram of the grid is shown in Fig. 2. Daily LSE load profiles are adopted from the dynamic 5-bus example in [2]; see Fig. 3. Line capacities, reactance levels, and generator cost data are also adopted from [2].
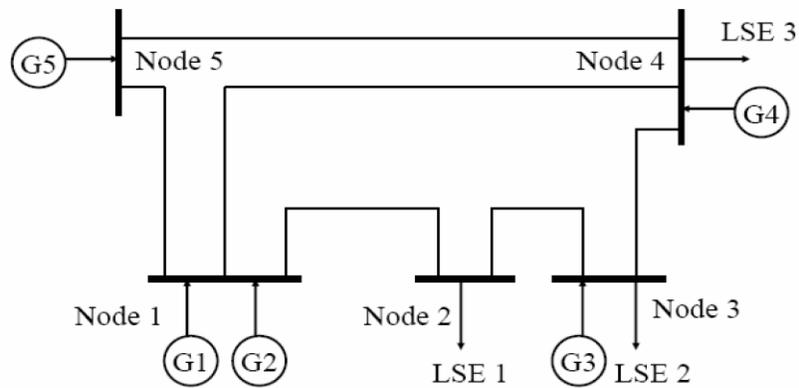

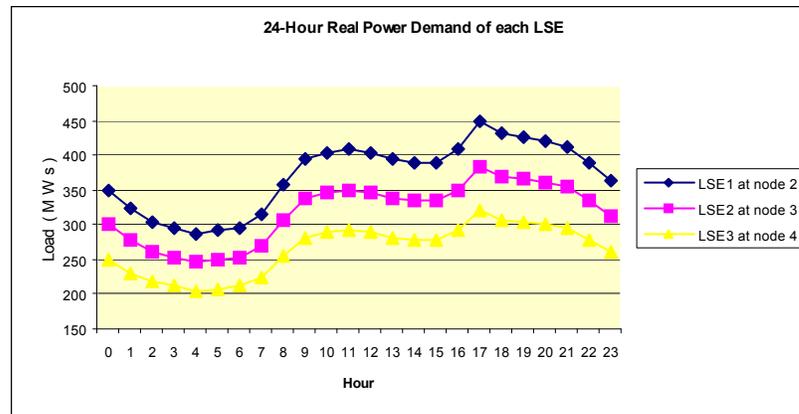**Figure 2. 5-Bus Transmission Grid**

Figure 3. 5-Bus Transmission Grid Daily Load Profiles

Detailed solution values for the scenario in which suppliers submit their true production data to the MO ("the no-learning scenario") are given in [2].

This study simulates two Q-Learning scenarios for this 5-bus test case. In the first scenario the LSEs have relatively low critical values for curtailing demand, whereas in the second scenario they have relatively high critical values. Simulation results for these learning scenarios are compared with the no-learning scenario.

4.2 Numerical Results

4.2.1 Review of Results from the No-Learning Scenario

In the no-learning scenario analyzed in [2], each generator submits a supply offer that includes its true marginal cost function and its true generation upper limit. The MW production level of each generator and the LMP at each bus that are cleared by the MO based on true cost data from generators are depicted in Fig. 4 and Fig. 5.
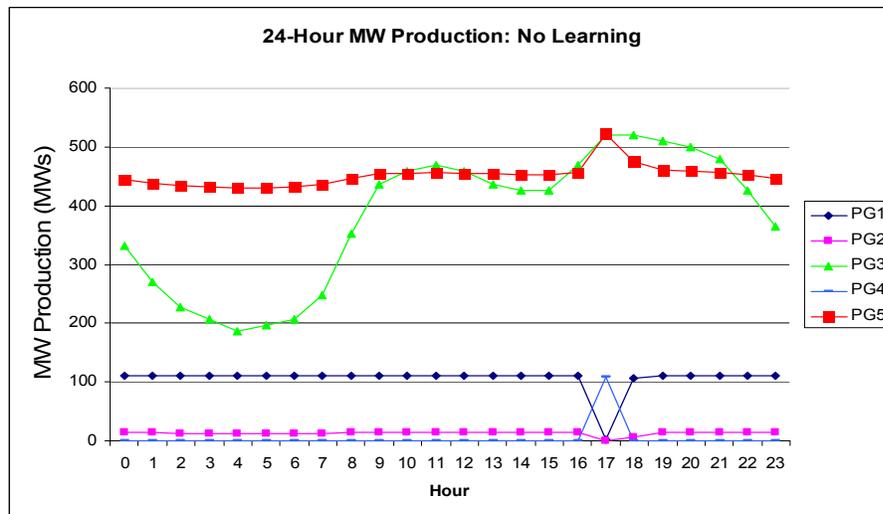

Figure 4. 5-Bus Simulation Results of 24-Hour MW Production (No-Learning Scenario)
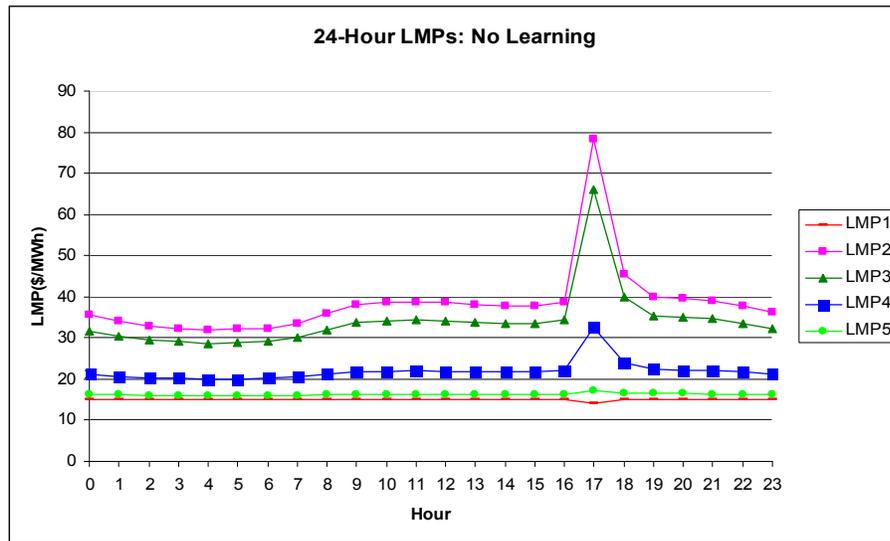
24-Hour LMPs: No Learning

Figure 5. 5-Bus Simulation Results of 24-Hour LMPs (No-Learning Scenario)

Generators 3 and 5 are the two largest units in the system with a combined capacity 1120MWs. The combined capacity of the three other small units is 410MW. The large units together with the high peak hour demand (1153.59MW) gives generators 3 and 5 potential market power. Note that the congestion between bus 1 and bus 2 exists for all 24 hours. This causes LMP separation between bus 1 and bus 2. During hour 17, the power flow on the line between buses 1 and 2 hits its upper thermal limit, and Generator 3 is dispatched at its upper production limit. Therefore, generator 4 that has the highest variable generation cost has to be dispatched to meet the demand. This results in a huge price spike at buses 2 and 3 at hour 17 that is about double of their LMP values at hour 16.

4.2.1 Results from the Two Learning Scenarios

Assume that the generators do not have to report their true marginal costs to the MO. Instead, the profit-seeking generators use Q-Learning to learn how to bid strategically to make more profits.

Since the system can be in several states, it does not have to stay in one single state in the long term. Rather, it may visit some states periodically or it may not even converge to a

periodic pattern. Therefore, one has to define convergence in a different way. The Day-Ahead market is said to be convergent if, at any state, each generator chooses one action in that state with probability 0.99 or higher.

Due to the probabilistic nature of the learning algorithm, the simulation does not converge to the same values for each run. In order to average out the random effects across different runs, 10 simulation runs are performed for each scenario and the mean values from the runs are reported.

In scenario one, LSEs have little tolerance for high LMPs. Their critical values for curtailing demand are only slightly higher than the LMPs that they will pay in the no-learning scenario. These critical values are given in Table 1.

Table 2 shows the number of days before convergence and final system states for 10 simulation runs. As can be seen, most of the time the system stays in state 8, in which every LSE is curtailing demand every day. It implies that generators are using very aggressive bidding strategies, and making full use of their market power. In this case, generators actually are making more profits by moving the system to state 8 because, even in the situation of less demand in peak hour the generators are still able to raise the price higher than the critical values of the LSEs.

Table 1. Critical Values for Learning Scenario 1

	LSE1	LSE2	LSE3
Critical Value ($/MWH)	115.5	98.0	47.5

Table 2. Number of Days before Convergence and Final States in Scenario 1

Simulation Run	1	2	3	4	5	6	7	8	9	10
# of Days before Convergence	83	189	81	77	77	189	78	81	86	230
Final state(s)	8	2↔8	8	8	8	2↔8	8	8	8	4↔8

2↔8 means the system visits state 2 and 8 periodically

In all 10 runs, all five generators converge by day 230 as shown in Table 2. The average number of days before convergence is 117.1. Note that in some cases the system moves back and forth between two states in a pattern of convergence.

In scenario two, the LSEs have high tolerance for high electricity prices. Their critical values for curtailing demand are higher than the critical values in scenario one. The critical values in this case are presented in Table 3. Table 4 shows the number of days before convergence and the final system states for the 10 simulation runs.

Table 3. Critical Values for Learning Scenario 2

	LSE1	LSE2	LSE3
Critical Value ($/MWH)	135.5	115.5	55.5

Table 4. Number of Days before Convergence and Final States in Scenario 2

Simulation Run	1	2	3	4	5	6	7	8	9	10
# of Days before Convergence	222	325	259	177	236	222	252	227	226	241
Final state	1↔8	1	1↔8	2↔8	1↔8	1↔8	1↔8	1	1↔8	1↔8

In all 10 runs, all five generators converged by day 325. The average number of days before convergence is 238.7. Note that, most of the time, the system ends up visiting state 1 and state 8 in turn. The day of convergence comes later if the system keeps visiting more than one state. From the results it is seen that, in fact, Q-Learning allows the generators to take advantage of the LSEs, whose demand-side response only has one day memory. First, by submitting low supply offers, the generators make sure that the LSEs do not curtail their demand tomorrow. Afterward they submit high supply offers and profit significantly from the LSEs that decrease their peak hour demand tomorrow. Then the generators submit a low supply offer again and so on. The simulation results show that Q-Learning helps generators make more profits by sacrificing today's benefit for more profits in the future. This scenario is a good illustration of anticipatory reinforcement learning.

Differences between the learning scenarios and the no learning scenario are discussed below. Furthermore, it is desirable to know to what extent Q-Learning is capable of helping generators exercise market power. Fig. 6 and 7 depict the mean values of MW production in learning scenarios 1 and 2, along with the corresponding simulation results obtained in the no-learning scenario. In the no-learning scenario, generator 4 is only dispatched at the peak hour. In both learning scenarios, in some simulation runs generator 4 is not dispatched. This is true when each generator is submitting an aggressive supply offer so that generator 4 is still the most expensive. However, in some simulation runs generator 4 chooses to submit less aggressive supply offers so that it becomes a relatively cheaper unit. Therefore, the average effect in the learning scenarios is that generator 4 is dispatched to some extent in each hour and has a steep increase in power output during the peak hour as it does in the no-learning scenario.
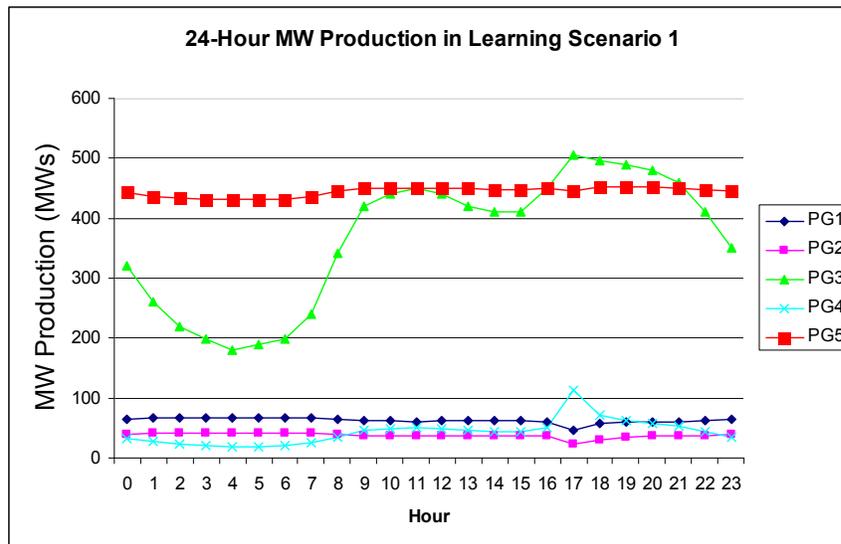

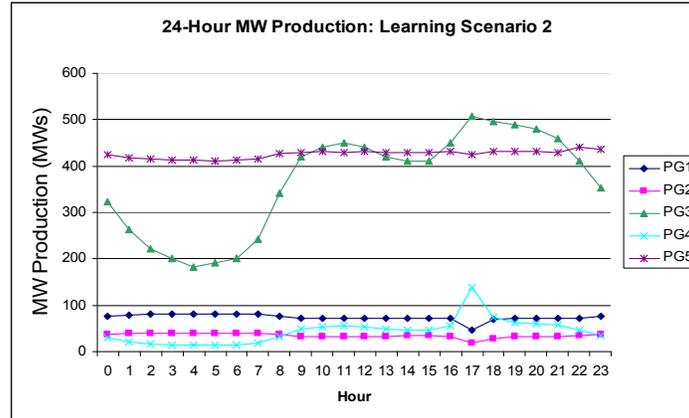**Figure 6. 5-Bus Simulation Results of 24-Hour MW Production (Learning Scenario 1)**

Figure 7. 5-Bus Simulation Results of 24-Hour MW Production (Learning Scenario 2)

The 24-hour mean LMP values for the learning scenarios 1 and 2 are shown in Fig. 8 and Fig. 9 along with the 24-hour LMP values for the no-learning scenario. In the no-learning scenario, the price spike at hour 17 is obvious. Although the LMPs in the learning scenarios 1 or 2 are substantially higher than for no-learning, the price fluctuation around the peak hour is much less. This finding is similar to the finding of Sun and Tesfatsion [2], who used reactive reinforcement learning to model the learning process of generators. However, since the sets of actions are different, one cannot draw a definitive conclusion about the learning techniques used in the two studies.
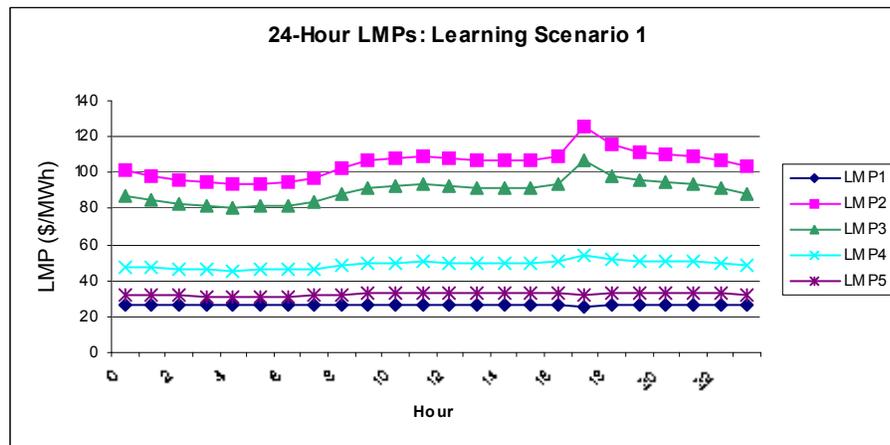

Figure 8. 5-Bus Simulation Results of 24-Hour LMPs (Learning Scenario 1)
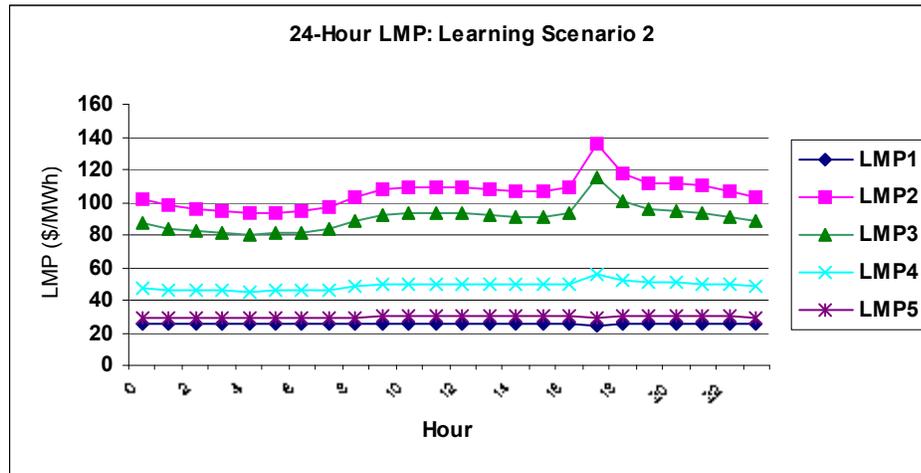
24-Hour LMP: Learning Scenario 2

Figure 9. 5-Bus Simulation Results of 24-Hour LMPs (Learning Scenario 2)

Figure 10 shows that the mean of the total profit gained by the generators in each learning scenario is much higher than what they made in the no-learning scenario. In fact, in the no-learning scenario the generators are not able to recover their fixed cost because they only covered their variable costs in their supply offers. This fact demonstrates that Q-Learning helps the generators to learn to exercise their potential market power to maximize their profits. It can be observed in Fig 10 that, during peak hour 17, the generators are making more profits in learning scenario 2 than they are in learning scenario 1. The high level of tolerance for price spikes of the LSEs in learning scenario 2 gives the generators more opportunities to manipulate the market.
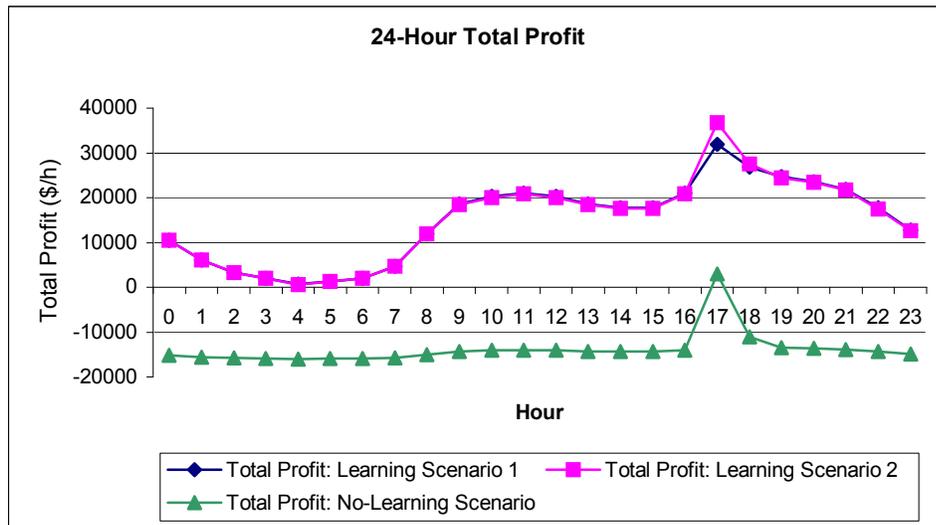
Figure 10. 5-Bus Transmission Grid Simulation Results of 24-Hour Total Profits (No-Learning Compared with Learning Scenarios 1 and 2)

4.3 Summary

In this chapter the simulation results of a 5-bus test case in no-learning scenario and both of the learning scenarios are demonstrated and compared. It has been shown that the 24-hour LMPs in the learning scenarios are significant higher than that in the no-learning scenarios due to strategic gaming of the supplier agents. However, the volatility of the 24-hour LMPs in the learning scenarios are lower than that in the no-learning scenario. In addition, the MW output of peak unit 4 in the learning scenarios is higher than that in the no-learning scenario, which leads to market inefficiency. Simulation results show that Q-Learning helps electricity suppliers learn how to bid strategically under the condition of a simple demand-side response model. With Q-Learning capabilities, electricity suppliers find their way to make more profits in the long term by sacrificing their immediate profits.

CHAPTER 5 CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

This research provides a method to model the Day-Ahead electricity market by a multi-agent system. The multi-agent system includes three types of interacting agents: supplier agents, LSEs and MO. The supplier's objective is to maximize its profit over a planning horizon. The learning behavior of the supplier agent is modeled by Q-Learning. LSEs in the model are assumed to have a simple demand-side response.

Simulation results on a 5-bus transmission grid show that Q-Learning help electricity suppliers learn how to bid strategically in a market environment with a simple demand-side response model. With Q-Learning capabilities, electricity suppliers find a way to make more profits in the long term by sacrificing their immediate profits. Without communicating with one another, every supplier agent chooses to bid higher than their marginal cost, which in turn yields significantly higher LMPs at each bus. In addition, the volatility of the 24-hour LMPs in the learning scenarios is lower than that in the no-learning scenario, which makes it harder for the MO to mitigate the potential use of market power.

Q-Learning has some limitations. It assumes a finite domain of actions. Also, the Q-Learning model developed in this research assumes that electricity suppliers do not explicitly take into account the presence of other electricity suppliers in the environment. Therefore the results may not be accurate when multiple agents interact in a realistic market environment. These limitations should be relaxed in the future research.

5.2 Future Work

In the future work, the following extensions will give useful results and insights into the design of market rules and the strategic bidding behaviors of market participants:

1. The actual system's demand data will be used instead of the time invariant load profile. In this case, the way Q-Learning is implemented for supplier agents' should be reexamined. The forecasted demand should also be considered in the state domain. Larger power system models will be used as a test system to study the network effects on the market.

2. The ancillary service market will be incorporated into the current market framework. Market participants will also be able to bid into the ancillary service market to provide regulation up, regulation down, spinning reserve, and non-spinning reserve services. Different designs of the ancillary service market will be tested.

3. Different learning algorithms with different parameters will be tested on the supplier agents. Simulation results will be available on which learning algorithm is better suited for what type of supplier agents. The effect of market power can be further investigated, if the supplier agents are assumed to own multiple plants.

4. The price sensitive demand could be included to test its impact on the market efficiency. In addition, the market clearing algorithm should be modified to minimize the 24-hour power purchasing cost instead of optimizing each hours' purchasing cost separately.

5. The Transmission Company (TRANSCO) may be included as another type of agent to study the long term effects of transmission planning and expansion on the electricity market. The supplier agents will be assumed to be able to invest in new power plants based on the long-term economic signals.

REFERENCES

[1]   S. Borenstein, J. Bushnell, and F. A. Wolak, "Measuring Market Inefficiencies in California's Restructured Wholesale Electricity Market," Center for the Study of Energy Markets. Paper CSEMWP-102, June. 200.

[2]   J. Sun and L. Tesfatsion, "Dynamic Testing of Wholesale Power Market Designs: An Open-Source Agent-Based Framework," ISU Economics Working Paper No. 06025, Revised April 2007. http://www.econ.iastate.edu/tesfatsi/DynTestAMES.JSLT.pdf

[3]   V. Nanduri and T. K. Das, "A Reinforcement Learning Model to Assess Market Power under Auction-Based Energy Pricing," *IEEE Transactions on Power Systems*, vol. 22, No.1, Feb. 2007.

[4]   C. W. Richter, G. B. Sheblé, and D. Ashlock, "Comprehensive Bidding Strategies with Genetic Programming/Finite State Automata," *IEEE Transactions on Power Systems*, vol. 14, pp. 1207-1212, Nov. 1999.

[5]   M. Wooldridge and N. Jennings, "Intelligent Agents: Theory and Practice," *The Knowledge Engineering Review*, vol. 10, pp. 115-152, 1995.

[6]   N. Jennings and M. Wooldridge (Ed.), "Agent Technology: Foundations, Applications, and Markets," Springer-Verlag, 1998.

[7]   L, Padgham and M, Winikoff, "Developing Intelligent Agent Systems: A Practical Guide," John Wiley & Sons, July. 2004.

[8]   H, Oh and T. Mount, "Testing the Effects of Holding Forward Contracts on the Behavior of Suppliers in an Electricity Auction," *Proc. 38th Hawaii Intl. Conf. on System Sciences*, 2005.

[9]   R. J. Thomas and T. D. Mount, "Using Software Agents to Test Electric Markets and Systems," *IEEE Power Eng. Soc. Summer Meeting*, Vol. 3, pp. 2808-2812, 2005

[10] P. Wei, Y. Yan, Y. Ni, J. Yen, and F. Wu, "A Decentralized Approach for Optimal Wholesale Cross-Border Trade Planning Using Multi-Agent Technology," *IEEE Transactions on Power Systems*, vol. 16, pp. 833-838, Nov 2001.

[11] J. H. Yuan, J. Wang, and Z. G. Hu, "Simulation of Large Customer Price Response under Time-of-Use Electricity Pricing Based on Multi-Agent System", *Intl. Conf. on Power System Technology*, 2006.

[12] E. Gnansounou, J. Dong, S. Pierre, and A. Quintero, "Market Oriented Planning of Power Generation Expansion using Agent-Based Model," *IEEE Power Eng. Soc. Summer Meeting*, Oct. 2004

[13] S. D. J. McArthur, C. D. Booth, J. R. McDonald, and I. T. McFadyen, "An Agent-Based Anomaly Detection Architecture for Condition Monitoring," *IEEE Transactions on Power Systems*, vol. 20, pp. 1675-1682, Nov. 2005.

[14] S. D. J. McArthur and E. M. Davidson, "Multi-Agent Systems for Diagnostic and Condition Monitoring Applications," *IEEE Power Eng. Soc. Summer Meeting*, Vol. 1, pp. 50-54, June. 2004.

[15] L, Liu, K. P. Logan, D. A. Cartes, and S. K. Srivastava, "Fault Detection, Diagnostics, and Prognostics: Software Agent Solutions", *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 1613-1622, July. 2007.

[16] T. Nagata and H. Sasaki, "A Multi-Agent Approach to Power System Restoration," *IEEE Transactions on Power Systems*, vol. 17, pp. 457-462, May. 2002.

[17] J. A. Momoh, "NAVY Ship Power System Restoration using Multi-Agent Approach," *IEEE Power Eng. Soc. Summer Meeting*, June. 2006.

[18] A. L. Dimeas and N.D. Hatziargyriou, "Operation of a Multiagent System for Microgrid Control," *IEEE Transactions on Power Systems*, vol. 20, pp. 1447-1455, Aug. 2005.

[19] J. Jung, C. C. Liu, S. L. Tanimoto, and V. Vittal, "Adaptation in Load Shedding under Vulnerable Operating Conditions," *IEEE Transactions on Power Systems*, vol. 17, pp. 1199-1205, Nov. 2002.

[20] F. Bellifemine, G. Caire, and D. Greenwood, "Developing Multi-Agent Systems with JADE", John Wiley & Sons, Feb. 2007.

[21] FIPA, FIPA 2000 Specification: FIPA SL Content Language Specification, SC00008. [Online]. Available: http://www.fipa.org/.

[22] FIPA, FIPA 2000 Specification: FIPA Abstract Architecture Specification, SC00001. [Online]. Available: http://www.fipa.org/.

[23] R. D. Zimmerman and D. Gan, "MATPOWER: A MATLAB Power System Simulation Package (Version 2.0)," Cornell University, New York 1997. http://www.pserc.cornell.edu/.

[24] Y. Shoham, R. Powers, and T. Grenager, "If Multi-Agent Learning is the Answer, What is the Question?" *Artificial Intelligence*, vol. 171, pp. 365-377, Mar. 2007.

[25] G. Brown, "Iterative Solution of Games by Fictitious Play," *Activity Analysis of Production and Allocation*, John Willey and Sons, New York, 1951.

[26] D. Fudenberg and D. Kreps, "Learning Mixed Equilibria," *Games and Economic Behavior*, vol 5, pp. 320-367, 1993.

[27] D. Fudenberg and D. Levine, "Universal Consistency and Cautious Fictitious Play," *Journal of Economic Dynamics and Control*, vol. 19, pp. 1065-1089, 1995.

[28] C. J. C. H. Watkins, "Learning from Delayed Rewards," Ph.D. Thesis, University of Cambridge, England, 1989.

[29] C. J. C. H. Watkins and P. Dayan, "Q-Learning," *Machine Learning*, 3:279-292, 1992.

[30] S. Hart and A. Mas-Colell, "A Simple Adaptive Procedure Leading to Correlated Equilibrium," *Econometrica*, vol. 68, No.5, pp. 1127-1150, Sep. 2000.

ACKNOWLEDGEMENTS