

University of Strathclyde  
Department of Electronic and Electrical Engineering

# Learning to Trade Power

by

Richard W. Lincoln

A thesis presented in fulfilment of the  
requirements for the degree of

*Doctor of Philosophy*

2010

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.51. Due acknowledgement must always be made of the use of any material contained in, or derived from, this thesis.

## Acknowledgements

The research reported in the present thesis was funded by the United Kingdom Engineering and Physical Research Council under grant GR/T28836/01.

# Abstract

Connectionist reinforcement learning methods approximating value-functions offer few convergence guarantees, even in simple systems. Reinforcement learning has been applied previously to agent-based simulation of energy markets using only discrete action and sensor domains. If learning algorithms are to deliver on their potential for application in operational settings, modelling continuous domains is necessary. The contribution of this thesis is to show that policy-gradient reinforcement learning algorithms can be applied to continuous representations of energy trading problems and that their superior use of sensor data results in improved performance over previously applied value-function methods. From this it follows that algorithms which search directly in the policy space will be better suited to decision support applications and automated energy trade.

# Contents

<b>Abstract</b>	<b>iv</b>
<b>List of Figures</b>	<b>vii</b>
<b>List of Tables</b>	<b>viii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Electric Power Systems . . . . .	2
1.3 Electricity Markets . . . . .	3
1.3.1 British Electricity Transmission and Trading Arrangements . . . . .	3
1.3.2 The England & Wales Electricity Pool . . . . .	4
1.4 Electricity Market Simulation . . . . .	5
1.4.1 Agent-Based Modelling . . . . .	6
1.5 Problem Statement . . . . .	6
1.6 Research Contributions . . . . .	7
1.7 Outline . . . . .	7
<b>2 Background</b>	<b>8</b>
2.1 Optimal Power Flow . . . . .	8
2.1.1 Interior-Point Methods . . . . .	9
2.2 Reinforcement Learning . . . . .	9
2.2.1 Introduction . . . . .	9
2.2.2 Value Function Methods . . . . .	9
2.2.3 Policy Gradient Methods . . . . .	12
<b>3 Related Work</b>	<b>14</b>
3.1 Agent-Based Simulation . . . . .	14
3.1.1 Learning Method Comparisons . . . . .	14
3.1.2 Heuristic Approaches . . . . .	14
3.1.3 Simulations Applying Genetic Algorithms . . . . .	14
3.1.4 Learning Classifier Systems . . . . .	14
3.1.5 Simulations Applying Q-learning . . . . .	14
3.1.6 Simulations Applying Roth-Erev . . . . .	15
3.2 Closed-Form Equilibrium . . . . .	16
3.3 Policy-Gradient Reinforcement Learning . . . . .	16

3.4	Open Source Power Engineering Software . . . . .	16
<b>4</b>	<b>Modelling</b>	<b>17</b>
4.1	Electricity Market Model . . . . .	17
4.1.1	Power System Model . . . . .	18
4.1.2	AC Power Flow Equations . . . . .	19
4.1.3	DC Power Flow Equations . . . . .	21
4.1.4	AC OPF Formulation . . . . .	23
4.1.5	DC OPF Formulation . . . . .	24
4.1.6	OPF Solution . . . . .	24
4.1.7	Unit Decommitment . . . . .	24
4.1.8	Auction Interface . . . . .	25
4.2	Multi-Agent System . . . . .	26
4.2.1	Agent, Task & Environment . . . . .	26
4.2.2	Simulation Event Sequence . . . . .	28
<b>5</b>	<b>Learning to Trade Power</b>	<b>30</b>
5.1	Aims & Objectives . . . . .	30
5.2	Method of Simulation . . . . .	30
5.3	Results . . . . .	31
5.4	Discussion . . . . .	31
5.5	Critical Analysis . . . . .	31
<b>6</b>	<b>Competitive Power Trade</b>	<b>32</b>
6.1	Aims & Objectives . . . . .	32
6.2	Method of Simulation . . . . .	32
6.3	Results . . . . .	33
6.4	Discussion . . . . .	33
6.5	Critical Analysis . . . . .	33
<b>7</b>	<b>System Constraint Exploitation</b>	<b>34</b>
7.1	Aims & Objectives . . . . .	34
7.2	Results . . . . .	34
7.3	Discussion . . . . .	34
7.4	Critical Analysis . . . . .	34
<b>8</b>	<b>Further work</b>	<b>35</b>
8.1	AC Optimal Power Flow . . . . .	35
8.2	Decentralised Trade . . . . .	35
8.3	Standarisation . . . . .	35
8.4	Blackbox optimisation . . . . .	35
<b>9</b>	<b>Summary conclusions</b>	<b>36</b>
	<b>Bibliography</b>	<b>37</b>

# List of Figures

4.1	Acceptable price range . . . . .	26
-----	----------------------------------	----

# List of Tables



# Chapter 1

## Introduction

This thesis compares methods from the field of artificial intelligence in their ability to learn from reinforcement when trading electricity in simulated competitive marketplaces. An introduction to electric energy and the associated markets is provided by the present chapter. An explanation of the motivation for the research presented in this thesis follows, along with a statement of the principle contributions that have been made. A outline of the remaining chapters is finally provided.

### 1.1 Motivation

The average total demand for electricity in the UK is now approximately 50GW. and the cost of buying 1MW for one hour is around £40. This equates to yearly transaction values of £17.52 billion. The importance of electricity to society is clearly evident when supply fails. The New York black-out in August 2003 involved 61.8GW of lost power supply to approximately 50 million consumers. The majority of supplies were restored within two days and the event is estimated to have cost \$7-10 billion.

Quality of life for a person is directly proportional to that person's electricity consumption[ref]. The world population is currently 6.7 billion and forecast to pass 9 billion by the year 2050. Electricity production currently demands over 1/3 of the annual primary energy extracted. As people endeavour to improve their quality of life, primary energy fuels will become increasingly scarce. Competitive markets are a proven device for allocation of scarce resources.

Commercialisation of electricity supply industries is a new practice, relative to other commodities, having begun in the early 1990s. The inability to store electricity, once generated, in a commercially viable quantity prevents trade as a conventional commodity. Trading mechanisms must be created to allow shortfalls in electric energy to be purchased a short notice from quickly dispatchable generators. Different implementations have been

tried in various countries and states and how best to structure electricity markets is an open field of research. Electricity market designs are further complicated by the need to manage complex dynamic constraints in the power systems that deliver the traded product.

## 1.2 Electric Power Systems

Generation and bulk movement of electricity takes place in a three-phase alternating current power system. The three phases are high voltage waveforms,  $120^\circ$  offset in time and oscillating at 50Hz. Alternators or synchronous generators, typically rotating at 3600rpm or 1800rpm, generate apparent power  $S$  and inject current  $I_l$  at a line voltage  $V_l$  typically between 11kV and 25kV. One of the principal reasons that alternating current, and not direct current, systems are used to supply electricity is that the power can be transformed between voltages with very high efficiency. The apparent power conducted by a line is the product of the line voltage and the line current,

$$S = \sqrt{3}V_l I_l \quad (1.1)$$

and the ohmic heating losses are proportional to the square of the line current,

$$P_r = 3I^2 R \quad (1.2)$$

where  $R$  is the resistance of the line. Transmitting power at ultra-high voltages reduces the current flow, resulting in substantially reduced losses. One consequence of higher voltages is the larger extent and integrity of conductor insulation between one another, neutral and earth. This results in large transmission towers and high costs for undergrounding systems.

The UK transmission systems operate at 400kV, 275kV or 132kV and systems upto and beyond 1000kV are in operation in larger countries. For transmission over very long distances or undersea, high voltage direct current systems have become economically viable in recent years. The ability to transmit large amounts of power over long distances allows generation at high capacity stations, located away from large load centres, which offer economies of scale and lower operating costs. It also allows electricity to be transmitted across country borders and from renewable energy plant in remote locations.

From the transmission system, electric energy is transferred at a substation to the grid supply point of a distribution system for delivery to consumers. Distribution networks are also three-phase AC systems, but operate at lower voltages and differ in their general structure or topology from transmission networks. Transmission networks are typically highly interconnected, providing several paths for power to flow in, whereas distribution networks, in rural areas, typically consist of long radial feeder (usually overhead lines) or, in urban areas, contain many ring circuits. Three-phase transformers that step the voltage

down to levels more convenient for general use (typically from 11kV or 33kV to 400V) are spaced along the branches/rings. All three-phases may be provided for industrial loads or individual phases at 230V supply typical domestic and commercial loads. Splitting of phases is usually planned so that each is loaded equally as this produces a balanced symmetrical system that may be analysed as a single phase circuit (See Section 4.1.1).

## **1.3 Electricity Markets**

The United Kingdom was the first large country to privatise its electricity supply industry and the market structures that have since been adopted encapsulate the main principles behind electricity markets.

The England & Wales Electricity Pool was created to break apart the monolithic Central Electricity Generating Board and gradually introduced competition in generation and retail supply (See Section 1.3.2, below). As a result of early adoption, the United Kingdom is host to the main European power and gas exchanges and boasts a relatively high degree of consumer switching, which deemed an essential aspect of a competitive marketplace. The Pool has since been replaced with the British Electricity Transmission and Trading Arrangements in which market outcomes are no longer centrally determined, but arise largely from bilateral agreements between producers and suppliers.

### **1.3.1 British Electricity Transmission and Trading Arrangements**

Concerns over exploitation of market power in The England & Wales Electricity Pool and its effectiveness in encouraging reduced consumer electricity prices prompted introduction of the New Electricity Trading Arrangements (NETA) in March 2001 (?). A very similar nationwide scheme was formed in April 2005 with the inclusion of the Scottish electricity industry in the British Electricity Transmission and Trading Arrangements (BETTA). The aim was to provide greater choice to participants and improve trading efficiency. While The Pool operated a single daily auction and plant was dispatched centrally, under the new arrangements, market positions became determined through continuous bilateral trading between generators, suppliers, traders and consumers and participants became self-dispatching.

Under BETTA, the majority of power is traded through long-term contracts that are customised to the requirements of each party. These suit participants responsible for large power plants or those needing to purchase large volumes of power. A relatively large amount of time is required for long-term contracts to be agreed upon and this has an associated transaction cost. However, they reduce risk for large players and can provide a degree of flexibility through option contracts.

Power is also traded directly between participants through over-the-counter (OTC) contracts that are usually of a standardised form. Such contracts typically concern smaller volumes of power and have much lower associated transaction costs. Often they are used by participants to refine their market position ahead of delivery time.

Trading facilities, such as power exchanges, provide a means for participants to fine-tune their positions further through short-term transactions for relatively small quantities. Modern exchanges are computerised and accept anonymous offers and bids submitted electronically. A submitted offer/bid will be paired with any outstanding bids/offers in the system with compatible prices and quantities and the details are displayed for traders to observe.

All bilateral trading must be completed before “gate-closure”, which is point set before real-time that give the system operator the opportunity to balance supply and demand and mitigate breaches of system limits. In keeping with the free market philosophy of the UK, a competitive spot market is used in the balancing process also. A generator that is not fully loaded may offer a price at which it is willing to increase its output by a specified quantity. The speed at which it is capable of doing so must be stated with the offer. Certain loads may also offer demand reductions at a price and can typically be implemented very quickly. Longer-term contracts for balancing services are also struck between the system operator and generators/suppliers in order to avoid the price volatility often associated with spot markets.

### **1.3.2 The England & Wales Electricity Pool**

The Electric Lighting Act 1882 began the development of the UK’s electricity supply industry by allowing persons, companies and local authorities to set up supply systems, principally at the time for the purposes of street lighting and trams. Under The Electricity Supply Act 1926 the Central Electricity Board started operating the first grid of regional networks interconnected and synchronised at 132kV, 50Hz in 1933. This began operation as a national system five years later in 1938 and was nationalised under The Electricity Act 1947 with the merger of over 600 electricity companies and the creation of the British Electricity Authority. This was then dissolved and replaced with the Central Electricity Generating Board (CEGB) and the Electricity Council under The Electricity Act 1957. The CEGB was responsible for planning the network and generating sufficient electricity until the start of privatisation in 1990.

Privatisation of the UK electricity supply industry under Prime Minister Margaret Thatcher created The England & Wales Electricity Pool in March 1990. The control of the transmission system was transferred from the CEGB to The National Grid Company which was originally owned by twelve regional electricity companies and is now publically listed.

The Pool was a multilateral contractual arrangement between generators and suppliers and did not itself buy or sell electricity. Competition in generation was introduced gradually, by only entitling customers with consumption greater than or equal to 1MW to purchase electricity from any listed supplier. This limit was lowered in April 1994 to include customers consuming 100kW or more and finally to include all customers in 1998.

Scheduling of generation was on a merit order basis (cheapest first) at a day ahead stage and set a wholesale electricity price for each half-hour period of the schedule day.

Forecasts of total demand in MW, based on historic data and adjusted for factors such as the weather, for each settlement period were used by generating companies and organisations with interconnects to the England & Wales grid to formulate bids that had to be submitted to the grid operator by 10AM on the day before the schedule day.

Bids consisted of five price parameters as illustrated in Figure and represented the avoidable cost of generation. A start-up price would be included, representing the cost of turning on the generator from cold. A no-load price  $c_{noload}$  would equal the cost in pounds of keeping the generator running regardless of output. Three incremental prices  $c_1$ ,  $c_2$  and  $c_3$  specify the cost per MWh of generation between set-points  $p_1$ ,  $p_2$  and  $p_3$ .

A settlement computer program would calculate an unconstrained schedule (not accounting for the physical limitations of the transmission system), meeting the forecast demand and requirements for reserve while minimising cost. Cheapest bids up to the marginal point get accepted first and generally the bid price from the marginal generator is used to determine the system marginal price for each settlement period. The system marginal price determines the prices paid by consumers and paid to the generators and the prices paid by the consumers that get adjusted such that the costs of transmission are covered by the market and that the availability of capacity is encouraged at certain times.

Variations in demand and changes in plant availability would be adjusted for by the grid operator, producing a constrained schedule. Generators having submitted bids would be instructed to increase or reduce production as appropriate. Alternatively, the grid operator could instruct large customers with contracts to curtail their demand to do so or instruct generators contracted to provide ancillary services to adjust production.

## 1.4 Electricity Market Simulation

The previous section have explained the dependence of modern societies on on electric energy and how its supply is trusted to unadministered bilateral trading arrangements. Electricity supply involves technology, money, people, natural resources and the environment. These aspects are all changing and the discipling must be constantly researched in order that system such as electricity markets are fit for purpose. The value of electricity to society means that it is not feasible to experiment with radical changes to trading ar-

rangements on real systems. A practical alternative is to model them mathematically and simulate using a computer program.

Game theory is a branch of applied mathematics in which behaviour in strategic situations is captured mathematically. A common approach to doing this is to model the system and players as a numerical optimisation problem. Section 2.1 defines the optimal power flow problem which is a classical optimisation problem from the field of Power Engineering. Electricity markets are commonly modelled using variations on the optimal power flow problem with player strategies integrated. The present thesis presents electricity market research using agent-based modelling, which is an alternative approach to the mathematics of games.

#### **1.4.1 Agent-Based Modelling**

Social systems, such as electricity markets, are inherently complex. They involve many interactions between different types of individual and between individuals and collective entities, such as organisations or groups, the behaviour of which is itself the result of individual interactions. This degree of complexity drives classical closed-form equilibrium models to their limits. Trading environments must of be highly stylised and limited to artificially small numbers of players with strong constraining assumptions on their behaviour.

Agent-based simulation, involves modelling interacting entities as separable, adaptive, purposeful objects that seek to act in their own perceived interests. Macro-level regularities arise from micro-level inter-play between agents as they adapt to their view of other agents.

### **Reinforcement Learning**

## **1.5 Problem Statement**

Research into electricity markets using reinforcement learning typically involves discretization the action domain, often into incremental markups on marginal cost. Also, either sensor domains are discretized or state information is disregarded altogether. Despite these simplifications, authors have been drawn many practical conclusions from this approach (Weidlich & Veit, 2008). If learning algorithms are to deliver on their potential for application in operational electricity market settings, modelling continuous domains will be necessary.

In the field of robotic control, environment state, action and observation spaces are often continuous or mixed. Traditional reinforcement learning methods such as Sarsa and Q-learning (See Section 2.2, below) can be applied to systems with continuous domains by using connectionist systems for value function approximation (Barto, Sutton, & Anderson, 1983). However, feedback between policy updates and value function changes can result in

oscillations or divergence in these algorithms even when applied to simple systems (Peters & Schaal, 2008). In response to this, policy-gradient methods, pioneered by Williams (Williams, 1992), which search directly in the policy space have been developed and applied in many real-life settings (Sutton, Mcallester, Singh, & Mansour, 2000; Peters & Schaal, 2006; Moody & Saffell, 2001; Peshkin & Savova, 2002).

The proposal is to combine policy-gradient reinforcement learning methods with artificial neural networks for continuous policy function approximation and apply to simulations electricity trade. The hypothesis being that superior use of sensor data results in improved performance over previously applied value-function methods.

## 1.6 Research Contributions

The research presented in this thesis pertains to the academic fields of power engineering, artificial intelligence and economics. The principle contributions in these areas are

- The proof that policy gradient reinforcement learning algorithms outperform value-function algorithms when applied to the power trade problem,
- A novel coupling of power system models and optimal power flow algorithm results with agents capable of handling discrete and continuous sensor and action spaces,
- Implementations of Roth-Erev reinforcement learning algorithms and continuous versions of Q-learning and  $Q(\lambda)$  for the open source PyBrain library,
- Open source implementations of power flow and optimal power flow algorithms in the Python programming language, preserving sparsity throughout the optimisation using the open source CVXOPT library.

## 1.7 Outline

This thesis is focussed on the application of standard and advanced reinforcement learning algorithms to a particular problem domain. The reader will require a certain degree of prior knowledge, or must be willing to read much of the referenced material, to fully understand the methodology taken. The intended audience is engineering and economics researchers interested in the application of reinforcement learning algorithms to the problem of trading energy in electric power systems.

## Chapter 2

# Background

This chapter provides an introduction to optimal power flow and reinforcement learning. Different formulations of the optimal power flow problem are explained along with the principles behind interior-point methods, common used to find their solutions. Reinforcement learning is also a generic term and Section 2.2 introduces computational approaches to learning to maximise long term reward. Definitions are provided for the valued-based and direct search algorithms compared in subsequent chapters.

### 2.1 Optimal Power Flow

Optimal power flow is a term used to describe a broad class of problems in which an objective function is optimised while constraints on the optimisation variables, which represent physical characteristics of the electric power system, are satisfied. Optimisation techniques typically solve problems of the form

$$\min_x f(x) \tag{2.1}$$

subject to

$$g(x) = 0 \tag{2.2}$$

$$h(x) \leq 0 \tag{2.3}$$

$$x_{min} \leq x \leq x_{max} \tag{2.4}$$



### 2.1.1 Interior-Point Methods

## 2.2 Reinforcement Learning

This section describes agent’s policies that represent a store of experience and the learning algorithms that its modify parameters. Together these components form models of individual behavior which are used to determine the actions to be performed in the agent’s environment and to learn from received rewards.

For a comprehensive introduction to reinforcement learning with evaluations of algorithm designs through mathematical analysis and computational experiments the interested reader is directed to the seminal work by Barto and Sutton

### 2.2.1 Introduction

The problem of learning how best to interact with an environment so as to maximise some long-term reward is one that arises in many aspect of life. Reinforcement learning is a term that is typically applied to understanding, automating and solving this problem through computational approaches. Unlike with the majority of Machine Learning techinques, the algorithms are not instructed as to which actions to take, but must learn to maximise the long-term reward through trial-and-error.

Reinforcement learning starts with an interactive, goal-seeking individual and an associated environment. The individuals require the ability to sense aspects of their environment, perform actions that influence the state of their environment and be assigned rewards as a response to their chosen action. An agent is said to follow a particular *policy* when mapping the perceived state of its environment to an action choice.

Value-based methods attempt to find the optimal policy by approximating a *value-function* which returns the total reward an agent can expect to accumulate, given an initial state and following the current policy thereafter.

Policy-gradient methods are an alternative to this which represent a policy using a learned function approximator with its own parameters The function approximator is updated according to the gradient of expected reward with respect to these parameters.

### 2.2.2 Value Function Methods

#### Basic Roth-Erev

The Roth-Erev reinforcement learning algorithm uses a stateless policy to select actions from a discrete domain(Roth et al., 1995; Erev & Roth, 1998). The dataset stored by each agent,  $j$ , contains an array of length  $K$ , where  $K$  is the number of feasible actions  $k$ . Each value in the array represents the propensity for selection of the associated action

in all states of the environment. Following interaction  $t$  in which agent  $j$  performed on the environment action  $k'$ , for arbitrary positive  $t$ , a reward,  $r_{jk'}(t)$ , is calculated. The propensity for agent  $j$  to select action  $k$  for interaction  $t + 1$  is

$$q_{jk}(t+1) = \begin{cases} (1 - \phi)q_{jk}(t) + r_{jk'}(t)(1 - \epsilon), & k = k' \\ (1 - \phi)q_{jk}(t) + r_{jk'}(t)(\frac{\epsilon}{K-1}), & k \neq k' \end{cases} \quad (2.5)$$

where  $\phi$  and  $\epsilon$  denote *recency* and *experimentation* parameters, respectively. The recency (forgetting) parameter degrades the propensity for all actions and prevents the value from going unbounded. It is intended to represent the tendency for players to forget older action choices and to prioritise more recent experience. The experimentation parameter prevents the probability of choosing an action from going to zero and thus encourages exploration of the action space.

Erev and Roth proposed that actions be selected according to a discrete probability distribution function where action  $k$  is selected for interaction  $t + 1$  with probability:

$$p_{jk}(t+1) = \frac{q_{jk}(t+1)}{\sum_{l=0}^K q_{jl}(t+1)} \quad (2.6)$$

Since  $\sum_{l=0}^K q_{jl}(t+1)$  increases with  $t$ , a reward  $r_{jk}(t)$  for performing action  $k$  will have a greater effect on the probability  $p_{jk}(t+1)$  during early interactions while  $t$  is small. This is intended to represent Psychology's *Power Law of Practice* in which it is qualitatively stated that, with practice, learning occurs at a decaying exponential rate and that a learning curve will eventually flatten out.

This algorithm may alternatively use a form of the *softmax* method (Sutton & Barto, 1998) using the Gibbs, or Boltzmann, distribution to select action  $k$  for the  $t + 1$ th interaction with probability

$$p_{jk}(t+1) = \frac{e^{q_{jk}(t+1)/\tau}}{\sum_{l=0}^K e^{q_{jl}(t+1)/\tau}} \quad (2.7)$$

where  $\tau$  is the *temperature* parameter. This parameter may be decreased in value over the course of an experiment since high values give all actions similar probability and encourage exploration of the action space, while low values promote exploitation of past experience.

### Variant Roth-Erev

Two shortcomings of the basic Roth-Erev algorithm (§2.2.2) have been identified and a variant formulation proposed (Nicolaisen, Petrov, & Tesfatsion, 2002). The problems are that the values by which propensities are updated can be zero or very small for certain

combinations of the experimentation parameter  $\epsilon$  and the total number of feasible actions  $K$ . Also, all propensity values are decreased by the same amount when the reward,  $r_{jk'}(t)$  is zero. Under the variant algorithm the propensity of agent  $j$  to select action  $k$  for interaction  $t + 1$  becomes:

$$q_{jk}(t+1) = \begin{cases} (1 - \phi)q_{ik}(t) + r_{jk'}(t)(1 - \epsilon), & k = k' \\ (1 - \phi)q_{ik}(t) + q_{jk}(t)(\frac{\epsilon}{K-1}), & k \neq k' \end{cases} \quad (2.8)$$

As with the basic Roth-Erev algorithm, the propensity for the action that the reward is associated with is adjusted by the experimentation parameter. All other action propensities are adjusted by a small proportion of their current value.

## SARSA

The SARSA algorithm is an on-policy Temporal Difference control method, similar to Q-learning. The action-value update for agent  $j$  is defined by

$$Q_j(s_{jt}, a_{jt}) + \alpha[r_{jt+1} + \gamma Q_j(s_{jt+1}, a_{jt+1}) - Q_j(s_{jt}, a_{jt})]. \quad (2.9)$$

While the Q-learning algorithm updates action-values using a greedy policy, which is a different policy to that being followed, SARSA uses the discounted future reward of the next state-action observation following the original policy.

## Q-Learning

The formulation of the Q-learning algorithm used is that of the original off-policy Temporal Difference algorithm developed by Watkins(Watkins, 1989). The action-value function,  $Q(s, a)$ , returns values from a  $M \times N$  matrix where  $M$  and  $N$  are arbitrary positive numbers equal to the total number of feasible states and actions, respectively. Each value represents the *quality* of taking a particular action,  $a$ , in state  $s$ . Actions are selected using either the  $\epsilon$ -greedy or softmax (See section 2.2.2) methods. The  $\epsilon$ -greedy method either selects the action (or one of the actions) with the highest estimated value or it selects an action at random, uniformly, independently of the estimated values with, typically small, probability  $\epsilon$ .

Agent  $j$  will observe a reward,  $r_{jt}$ , and a new state,  $s_{jt+1}$ , after taking action  $a_{jt}$  at step  $t$  when in state  $s_{jt}$ . The state-action value,  $Q_j(s_{jt}, a_{jt})$ , is updated according to the maximum value of available actions in state  $s_{t+1}$  and becomes

$$Q_j(s_{jt}, a_{jt}) + \alpha[r_{jt+1} + \gamma \max_a Q_j(s_{jt+1}, a_{jt}) - Q_j(s_{jt}, a_{jt})] \quad (2.10)$$

where  $\alpha$  and  $\gamma$  are the learning rate,  $0 \leq \alpha \leq 1$ , and discount factor,  $0 \leq \gamma \leq 1$ , respectively. The learning rate determines the extent to which new rewards will override the effect of older rewards. The discount factor allows the balance between maximising immediate rewards and future rewards to be set.

## Q( $\lambda$ )

With the Q-learning formulation, described in equation 2.10, only the quality associated with the previous state,  $s_{jt}$ , is updated. However, the preceding states can also, in general, be said to be associated with the reward  $r_{jt+1}$ . Eligibility traces are a mechanism for facilitating this effect and in algorithms such as Q( $\lambda$ ), the  $\lambda$  refers to it. The eligibility trace for a state  $e(s)$  represents how eligible the state  $s$  is to receive credit or blame for the error. The term “trace” refers to fact that only recently visited states become eligible. The eligibility value for the current state is increased, while for all other states it is attenuated by a factor  $\lambda$ .

The off-policy nature of Q-learning requires special care to be taken when implementing eligibility traces. While the algorithm may learn a greedy policy, in which the action with the maximum value would always be taken, typically a policy with some degree of exploration will be followed when choosing actions. If an exploratory (pseudo-random) step is taken the preceding states can no longer be considered eligible for credit or blame. Setting  $\lambda$  to 0 for non-greedy actions removes much of the benefit of using eligibility traces if exploratory actions are frequent. A solution to this has been developed, but requires a very complex implementation (Peng & Williams, 1996). A naïve approach can be taken, where the effect of exploratory actions is ignored, but the results of this are unexplored.

### 2.2.3 Policy Gradient Methods

#### REINFORCE

The previously defined learning methods typically rely upon discretisation of the sensor and action spaces so the associated values may be stored in tables. The memory requirements for this restrict the application of these methods to only small environments. Many environments, particularly from real applications, exhibit continuous sensor and/or action spaces and require generalisation techniques to be employed to provide a more compact policy representation.

REINFORCE is an associative reinforcement learning algorithm that determines a policy by modifying the parameters of a policy function approximator, rather than approximating a value function (Williams, 1992). Commonly, feedforward artificial neural networks are used to represent the policy, where the input is a representation of the state and the output is action selection probabilities. In learning, a *policy gradient* approach

is taken where the weights of the network are adjusted in the direction of the gradient of expected reinforcement.

Defining the network, let  $\mathbf{x}^i$  denote the vector of inputs to the  $i$ th unit and  $y_i$  denote output of the unit. In the input layer of the network the elements  $x_j$  of  $\mathbf{x}^i$  are normalised sensor values from the environment and in the output layer, or in any hidden layers, they are outputs from the  $j$  unit in the preceding layer. Let  $\mathbf{w}^i$  denote the vector of the weights,  $w_{ij}$ , on the connections to the  $i$ th unit. The output of the  $i$ th unit is dependant on the vector of inputs,  $\mathbf{x}^i$ , and the associated weights,  $\mathbf{w}^i$ .

For each interaction of the agent with the environment, each parameter  $w_{ij}$  is incremented by

$$\Delta w_{ij} = \alpha_{ij}(r - b_{ij}) \frac{\partial \ln \rho_i}{\partial w_{ij}} \quad (2.11)$$

where  $\alpha_{ij}$  is the *learning factor*,  $b_{ij}$  is the *reinforcement baseline* and  $\rho_i$  is the performance of the policy (e.g., the average reward per interaction).

## ENAC

ToDo: Episodic Natural Actor Critic(Peters & Schaal, 2008).

## Chapter 3

# Related Work

### 3.1 Agent-Based Simulation

Relative to the traditional closed-form equilibrium approaches, agent-based simulation of (electricity) markets is a new field of research. For comprehensive reviews and surveys of the many different techniques that have been applied in recent years the interested reader is directed to (Weidlich & Veit, 2008; Tesfatsion & Judd, 2006; Visudhiphan, 2003). This section will focus on reviewing literature from the field in which reinforcement learning techniques were applied in combination with explicit power system models. A short review is also provided of some more general applications of reinforcement learning with connectionist systems and policy-gradient methods.

#### 3.1.1 Learning Method Comparisons

#### 3.1.2 Heuristic Approaches

#### 3.1.3 Simulations Applying Genetic Algorithms

#### 3.1.4 Learning Classifier Systems

#### 3.1.5 Simulations Applying Q-learning

Krause et al. have published agent-based energy market research in which Q-learning methods were applied while considering physical system properties. In a comparison between Nash equilibrium analysis and agent-based simulation, the suitability of bottom-up modelling for the assessment of market evolution was assessed (Krause et al., 2004). This is built upon in subsequent publications which evaluate the influence on market power and social welfare of three congestion management schemes and which analyse strategic behavior in combined gas and electricity markets (Krause & Andersson, 2006; Kienzle, Krause, Egli, Geidl, & Andersson, 2007). Power Transmission Distribution Factors (PTDF) are

used in place of explicit power flow equations in determining line flows. The action domain of generating agents is limited to 0%, 5% and 10% markups on true marginal costs. The implementation of the Q-learning method used does not differentiate between environment states when selecting actions. This is a modification to the traditional formulation that still results in convincing conclusions, as with the popular Roth-Erev method.

There are similar applications of Q-learning in which states *are* defined, but none model the AC transmission system. A common approach is to use categorised market price from the previous period as state information (Bakirtzis & Tellidou, 2006; Xiong, Okuma, & Fujita, 2004).

### 3.1.6 Simulations Applying Roth-Erev

The AMES (Agent-based Modeling of Electricity Systems) power market test bed is a software package that models core features of the Wholesale Power Market Platform (WPMP) – a market design proposed by the US Federal Energy Regulatory Commission (FERC) in April 2003 for common adoption in regions of the US (Sun & Tesfatsion, 2007a). The design features:

- a centralised structure managed by an independent market operator,
- parallel day-ahead and real-time markets and
- locational marginal pricing.

Learning agents may represent load serving entities or generating companies and learn using implementations of the Roth-Erev method (See sections 2.2.2 and 2.2.2, above) built on the Repast agent simulation toolkit (Gieseler, 2005). The permissive license under which the source code for these algorithms has been released allowed direct translation of them for use in this study. Agents learn from the solutions of hourly bid/offer based DC-OPF problems formulated as quadratic programs and solved using QuadProgJ (Sun & Tesfatsion, 2007b).

The ability of generator agents to learn particular supply offers has been demonstrated along with the plotting and data handling capabilities of AMES using a 5-bus network model (Li & Tesfatsion, 2009). The same network has been used to investigate strategic capacity withholding in FERC wholesale power market design (Hongyan & Tesfatsion, 2009). Generator agents report linear marginal cost functions to the market operator and supply functions are formed through linear interpolation between the prices at minimum and maximum production limits. Load serving agents submit combinations of fixed demand bids and price-sensitive bid functions, the ratio between which is varied between 0.0 and 1.0 to test physical and economic capacity withholding potential. Comparing results from

a benchmark case (in which true production costs are reported, but higher than marginal cost functions may be reported) and cases in which reported production limits may be less than the true values, the authors find, that with sufficient capacity reserve, no evidence to suggest potential for inducing higher net earnings through capacity withholding in the WPMP.

### **3.2 Closed-Form Equilibrium**

### **3.3 Policy-Gradient Reinforcement Learning**

### **3.4 Open Source Power Engineering Software**



## Chapter 4

# Modelling

The present chapter concerns the approach taken in comparing methods for learning to trade power. Transmission system constraints are accounted for and the power system model used is defined. Generator costs are added to this model and used to form an optimal power flow problem. An auction interface to the optimal power flow is used to provide a representation of a realistic electricity market. This market is then combined with a multi-agent system to create a simulation platform for competitive energy trade. Finally, three experiments are defined to test different aspects of the methods learning abilities.

### 4.1 Electricity Market Model

Computation of the generator dispatch points is executed using parts of the of the optimal power flow formulation from MATPOWER (R. Zimmerman, Murillo-Sánchez, & Thomas, 2009). In order that the optimal power flow routine could be coupled with agents from the machine learning library PyBrain, the MATLAB<sup>TM</sup> source code from MATPOWER was translated to the Python programming language. With the permission of the MATPOWER developers the resulting package has been released under the terms of version 2 of the GNU General Public License as a project named PYLON (Lincoln, Galloway, & Burt, 2009). Sparse matrix objects from the convex optimisation library CVXOPT were used to allow the implementation to scale well to solving for very large systems.

This section describes parts of the optimal power flow formulation, unit-decommitment algorithm and auction interface from MATPOWER that were used to represent a centralised electricity market. Notable components of the full optimal power flow formulation (available in (R. D. Zimmerman & Murillo-Sánchez, 2007)) that have been ignored are shunt capacitors and inductors, generator P-Q capability curves and dispatchable loads.

The power system model is described by defining the bus, branch and generator objects. The power flow equations associated with a network of these components are subsequently defined. The constrained cost variable approach to modelling generator cost functions from (Wang, Murillo-Sanchez, Zimmerman, & Thomas, 2007) is introduced, from which the optimal power flow formulation follows.

The experiments described in Section 4 require an optimal power flow problem to be solved at each step. To accelerate the simulation process for certain experiments the option to use a linearised DC formulation is used, the formulation of which is provided also. The tradeoffs between using DC models over AC have been examined in (Overbye, Cheng, & Sun, 2004) and found reasonable for locational marginal price calculations.

Since the optimal power flow formulations do not facilitate shutting down expensive generators, the unit-decommitment algorithm from MATPOWER is defined. Finally, to provide an interface to agent participants that resembles that of real electricity market, MATPOWER's auction wrapper for the optimal power flow routine is described.

#### 4.1.1 Power System Model

The power system is assumed to be a three-phase AC system operating in the steady-state and under balanced conditions in which it may be represented by a single phase network of busbars connected by branch objects.

##### Branches

Each branch is modelled as a medium length transmission line in series with a transformer at the *from* end. A nominal- $\pi$  model with total series admittance  $y_s = 1/(r_s + jx_s)$  and total shunt capacitance  $b_c$  is used to represent the transmission line. The transformer is assumed to be ideal and both phase-shifting and tap-changing, with the ratio between primary winding voltage  $v_f$  and secondary winding voltage  $N = \tau e^{j\theta_{ph}}$  where  $\tau$  is the tap ratio and  $\theta_{ph}$  is the phase shift angle. From Kirchhoff's current law the current in the series impedance is

$$i_s = \frac{b_c}{2} v_t - i_t \quad (4.1)$$

and from Kirchhoff's voltage law the voltage across the secondary winding of the transformer is

$$\frac{v_f}{N} = v_t + \frac{i_s}{y_s} \quad (4.2)$$

Substituting  $i_s$  from (4.1), gives

$$\frac{v_f}{N} = v_t - \frac{i_t}{y_s} + v_t \frac{b_c}{2y_s} \quad (4.3)$$

and rearranging in terms of  $i_t$ , gives

$$i_t = v_s \left( \frac{-y_s}{\tau e^{\theta_{ph}}} \right) + v_r \left( y_s + \frac{b_c}{2} \right) \quad (4.4)$$

The current through the secondary winding of the transformer is

$$N^* i_f = i_s + \frac{b_c}{2} \frac{v_f}{N} \quad (4.5)$$

Substituting  $i_s$  from (4.1), gives

$$N^* i_f = \frac{b_c}{2} v_t - i_t + \frac{b_c}{2} \frac{v_f}{N} \quad (4.6)$$

Substituting  $\frac{v_f}{N}$  from (4.3) and rearranging, gives

$$i_s = v_s \left( \frac{1}{\tau^2} \left( y_s + \frac{b_c}{2} \right) \right) + v_r \left( \frac{y_s}{\tau e^{-j\theta}} \right) \quad (4.7)$$

## Generators

Each generator  $i$  is modelled as an apparent power source  $s_g^i = p_g^i + jq_g^i$  at a specific bus  $k$ , where  $p_g^i$  is the active power injection and  $q_g^i$  the reactive power injection, each expressed in per-unit to the system base MVA. Upper and lower limits on  $p_g^i$  are specified by  $p_{max}^i$  and  $p_{min}^i$ , respectively, where  $p_{max}^i > p_{min}^i \geq 0$ . Similarly, upper and lower limits on  $q_g^i$  are specified by  $q_{max}^i$  and  $q_{min}^i$ , respectively, where  $q_{max}^i > q_{min}^i$ .

## Buses and Loads

At each bus  $k$ , constant active power demand is specified by  $p_d^k$  and reactive power demand by  $q_d^k$ . Upper and lower limits on the voltage magnitude at the bus are defined by  $v_m^{k,max}$  and  $v_m^{k,min}$ , respectively. For one bus with an associated generator, designated the *reference* bus, the voltage angle is  $\theta_k^{ref}$  and typically valued zero. Dispatchable loads are modelled as generators with negative  $p_g^i$ , where  $p_{min}^i < p_{max}^i = 0$ .

### 4.1.2 AC Power Flow Equations

For a network of  $n_b$  buses,  $n_l$  branches and  $n_g$  generators, let  $C_g$  be the  $n_b \times n_g$  bus-generator connection matrix such that the  $(i, j)^{th}$  element of  $C_g$  is 1 if generator  $j$  is connected to bus  $i$ . The  $n_b \times 1$  vector of complex power injections from generators at all buses is

$$S_{g,bus} = C_g \cdot S_g \quad (4.8)$$

where  $S_g = P_g + jQ_g$  is the  $n_g \times 1$  vector with the  $i^{th}$  element equal to  $s_g^i$ .

Combining (4.7) and (4.4), the *from* and *to* end complex current injections for branch  $l$  are

$$\begin{bmatrix} i_f^l \\ i_t^l \end{bmatrix} = \begin{bmatrix} y_{ff}^l & y_{ft}^l \\ y_{tf}^l & y_{tt}^l \end{bmatrix} \begin{bmatrix} v_f^l \\ v_t^l \end{bmatrix} \quad (4.9)$$

where

$$y_{ff}^l = \frac{1}{\tau^2} \left( y_s + \frac{b_c}{2} \right) \quad (4.10)$$

$$y_{ft}^l = \frac{y_s}{\tau e^{-j\theta_{ph}}} \quad (4.11)$$

$$y_{tf}^l = \frac{-y_s}{\tau e^{j\theta_{ph}}} \quad (4.12)$$

$$y_{tt}^l = y_s + \frac{b_c}{2} \quad (4.13)$$

Let  $Y_{ff}$ ,  $Y_{ft}$ ,  $Y_{tf}$  and  $Y_{tt}$  be  $n_l \times 1$  vectors where the  $l$ -th element of each corresponds to  $y_{ff}^l$ ,  $y_{ft}^l$ ,  $y_{tf}^l$  and  $y_{tt}^l$ , respectively. Furthermore, let  $C_f$  and  $C_t$  be the  $n_l \times n_b$  branch-bus connection matrices, where  $C_{f,i,j} = 1$  and  $C_{t,i,k} = 1$  if branch  $i$  connects *from* bus  $j$  *to* bus  $k$ . The  $n_l \times n_b$  branch admittance matrices are

$$Y_f = \mathbf{diag}(Y_{ff})C_f + \mathbf{diag}(Y_{ft})C_t \quad (4.14)$$

$$Y_t = \mathbf{diag}(Y_{tf})C_f + \mathbf{diag}(Y_{tt})C_t \quad (4.15)$$

and relate the complex bus voltages  $V$  to the branch *from* and *to* end current vectors

$$I_f = Y_f V \quad (4.16)$$

$$I_t = Y_t V \quad (4.17)$$

The  $n_b \times n_b$  bus admittance matrix is

$$Y_{bus} = C_f^T Y_f + C_t^T \quad (4.18)$$

and it relates the complex bus voltages to the nodal current injections

$$I_{bus} = Y_{bus} V \quad (4.19)$$

The complex power losses from all branches are expressed as a non-linear function of  $V$

$$\begin{aligned} S_{bus}(V) &= \mathbf{diag}(V) I_{bus}^* \\ &= \mathbf{diag}(V) Y_{bus}^* V^* \end{aligned} \quad (4.20)$$

The complex power injections at the *from* and *to* ends of all branches are also expressed

as a non-linear functions of  $V$

$$\begin{aligned} S_f(V) &= \mathbf{diag}(C_f V) I_f^* \\ &= \mathbf{diag}(C_f V) Y_f^* V^* \end{aligned} \quad (4.21)$$

$$\begin{aligned} S_t(V) &= \mathbf{diag}(C_t V) I_t^* \\ &= \mathbf{diag}(C_t V) Y_t^* V^* \end{aligned} \quad (4.22)$$

### 4.1.3 DC Power Flow Equations

The same power system model is used in the formulation of the linearised DC power flow equations, but the following additional assumptions are made:

- The resistance  $r_s$  and shunt capacitance  $b_c$  of all branch can be considered negligible.

$$y_s \approx \frac{1}{jx_s}, b_c \approx 0 \quad (4.23)$$

- Bus voltage magnitudes  $v_{m,i}$ , are all approximately 1 per-unit.

$$v_i \approx 1e^{j\theta_i} \quad (4.24)$$

- The voltage angle difference between bus  $i$  and bus  $j$  is small enough that

$$\sin \theta_{ij} \approx \theta_{ij} \quad (4.25)$$

Applying the assumption that branches are lossless from (4.23), the quadrants of the branch admittance matrix, (4.10), (4.11), (4.12) and (4.13), approximate to

$$y_{ff}^l = \frac{1}{jx_s \tau^2} \quad (4.26)$$

$$y_{ft}^l = \frac{-1}{jx_s \tau e^{-j\theta_{ph}}} \quad (4.27)$$

$$y_{tf}^l = \frac{-1}{jx_s \tau e^{j\theta_{ph}}} \quad (4.28)$$

$$y_{tt}^l = \frac{1}{jx_s} \quad (4.29)$$

Applying the uniform bus voltage magnitude assumption from 4.24 to (4.9), the branch *from* end current approximates to

$$i_f \approx \frac{e^{j\theta_f}}{jx_s\tau^2} - \frac{e^{j\theta_t}}{jx_s\tau e^{-j\theta_{ph}}} \quad (4.30)$$

$$= \frac{1}{jx_s\tau} \left( \frac{1}{\tau} e^{j\theta_f} - e^{j(\theta_t + \theta_{ph})} \right) \quad (4.31)$$

The branch *from* end complex power flow  $s_f = v_f i_f^*$  therefore approximates to

$$s_f \approx e^{j\theta_f} \cdot \frac{j}{x_s\tau} \left( \frac{1}{\tau} e^{-j\theta_f} - e^{j(\theta_t + \theta_{ph})} \right) \quad (4.32)$$

$$= \frac{1}{x_s\tau} \left[ \sin(\theta_f - \theta_t - \theta_{ph}) + j \left( \frac{1}{\tau} - \cos(\theta_f - \theta_t - \theta_{ph}) \right) \right] \quad (4.33)$$

Applying the voltage angle difference assumption from 4.25 yields the approximation

$$p_f \approx \frac{1}{x_s\tau} (\theta_f - \theta_t - \theta_{ph}) \quad (4.34)$$

Let  $B_{ff}$  and  $P_{f,ph}$  be the  $n_l \times 1$  vectors where  $B_{ff_i} = \frac{1}{x_s^i \tau^i}$  and  $P_{f,ph_i} = \frac{-\theta_{ph}^i}{x_s^i \tau^i}$ . If the system  $B$  matrices are

$$B_f = \mathbf{diag}(B_{ff})(C_f - C_t) \quad (4.35)$$

$$B_{bus} = (C_f - C_t)^\top B_f \quad (4.36)$$

then the real power bus injections are

$$P_{bus}(\Theta) = B_{bus}\Theta + P_{bus,ph} \quad (4.37)$$

where  $\Theta$  is the  $n_b \times 1$  vector of bus voltage angles and

$$P_{bus,ph} = (C_f - C_t)^\top + P_{f,ph} \quad (4.38)$$

The active power flows at the branch *from* ends are

$$P_f(\Theta) = B_f\Theta + P_{f,ph} \quad (4.39)$$

and  $P_t = -P_f$  since branches are assumed to be lossless.

#### 4.1.4 AC OPF Formulation

Generator active and, optionally, reactive power output costs are defined by convex  $n$ -segment piecewise linear cost functions.

$$c^{(i)}(x) = m_i p + c_i \quad (4.40)$$

for  $p_i \leq p \leq p_{i+1}$ ,  $i = 1, 2, \dots, n$  where  $m_{i+1} \geq m_i$  and  $p_{i+1} > p_i$ . Since these costs are non-differentiable the constrained cost variable approach from (Wang et al., 2007) is used to make the optimisation problem smooth. For each generator  $i$  a helper cost variable  $y_i$  added to the objective function. Inequality constraints

$$y_i \geq m_{i,j}(p - p_j) + c_j, \quad j = 1 \dots n \quad (4.41)$$

require  $y_i$  to lie on the epigraph of  $c^{(i)}(x)$ . The objective of the optimal power flow problem is to minimise the sum of the cost variables for all generators.

$$\min_{\theta, V_m, P_g, Q_g, y} \sum_{i=1}^{n_g} y_i \quad (4.42)$$

Equality constraints enforce the balance between generator complex power injections  $S_g$  and the sum of apparent power demand  $S_d$  and the branch losses expressed in (4.20).

$$S_{bus}(V) + S_d - S_g = 0 \quad (4.43)$$

Branch complex power flow limits  $S_{max}$  are enforced by the inequality constraints

$$|S_f(V)| - S_{max} \leq 0 \quad (4.44)$$

$$|S_f(V)| - S_{max} \leq 0 \quad (4.45)$$

The reference bus voltage angle  $\theta_i$  is fixed by the equality constraint

$$\theta_i^{ref} \leq \theta_i \leq \theta_i^{ref}, \quad i \in \mathcal{I}_{ref} \quad (4.46)$$

Upper and lower limits on the optimisation variables  $V_m$ ,  $P_g$  and  $Q_g$  are enforced by the inequality constraints

$$v_m^{i,min} \leq v_m^i \leq v_m^{i,max}, \quad i = 1 \dots n_b \quad (4.47)$$

$$p_g^{i,min} \leq p_g^i \leq p_g^{i,max}, \quad i = 1 \dots n_g \quad (4.48)$$

$$q_g^{i,min} \leq q_g^i \leq q_g^{i,max}, \quad i = 1 \dots n_g \quad (4.49)$$

#### 4.1.5 DC OPF Formulation

Piecewise linear cost functions are also used to define generator active power costs in the DC optimal power flow formulation. Since the power flow equations are linearised, following assumptions (4.23), (4.24) and (4.25), the optimal power flow problem simplifies to a linear program. The voltage magnitude variables  $V_m$  and generator reactive power set-point variable  $Q_g$  are eliminated following assumption (4.25) since branch reactive power flows depend on bus voltage angle differences. The objective function reduces to

$$\min_{\theta, P_g, y} \sum_{i=1}^{n_g} y_i \quad (4.50)$$

Combining the nodal real power injections, expressed as a function of  $\Theta$ , from (4.37) with active power generation  $P_g$  and active demand  $P_d$ , the power balance constraint is

$$B_{bus}\Theta + P_{bus,ph} + P_d - C_g P_g = 0 \quad (4.51)$$

Limits on branch active power flows  $B_f\Theta$  and  $B_t\Theta$  are enforced by inequality constraints

$$B_f\Theta + P_{f,ph} - F_{max} \leq 0 \quad (4.52)$$

$$-B_f\Theta + P_{f,ph} - F_{max} \leq 0 \quad (4.53)$$

The reference bus voltage angle equality constraint from (4.46) and the  $p_g$  limit constraint from (4.48) are applied also.

#### 4.1.6 OPF Solution

##### 4.1.7 Unit Decommittment

In the OPF formulation above (See section 2.1) the solver must attempt to dispatch generators within their minimum and maximum power limits. Expensive generators can not be completely shutdown even if doing so would result in a lower total system cost. To achieve this an implementation of the *unit decommitment* algorithm (See Algorithm 1, below) from MATPOWER was used (R. D. Zimmerman & Murillo-Sánchez, 2007, p. 20). The algorithm finds the least cost dispatch by solving repeated OPF problems with different combinations of generating units at their minimum active power limit deactivated. The lowest cost solution is returned when no further improvement can be made and no candidate generators remain.



---

**Algorithm 1** Unit decommitment

---

```
1: initialise  $N \leftarrow 0$ 
2: solve initial OPF
3:  $L_{tot} \leftarrow$  total load capacity
4: while total min gen. capacity  $> L_{tot}$  do
5:    $N \leftarrow N + 1$ 
6: end while
7: repeat
8:   for  $c$  in candidates do
9:     solve OPF
10:  end for
11: until done = True
```

---

#### 4.1.8 Auction Interface

Solving the optimisation problem defined above (See section 2.1) is intended to represent the function of a pool market operator. To present agents participating in this market with a simpler interface, more representative of a pool market an implementation of the “smart market” auction clearing mechanism from MATPOWER was used (R. D. Zimmerman & Murillo-Sánchez, 2007, p. 31). Using this interface the OPF problem is formulated from a list of offers to sell and bids to buy power.

An offer/bid specifies a quantity of power in MW and a price for that power in \$/MWh, to be traded over a particular period of time. The market accepts sets of offers and bids and uses the solution of the unit decommitment algorithm to return sets of *cleared* offer and bids. The cleared offers/bids can then be used to produce dispatch orders from which values of revenue and earnings/losses may be determined.

The market features the ability to set maximum offer price limits and minimum bids price limits. The process of clearing the market begins by withholding offers/bids outwith these limits, along with those specifying non-positive quantities. Valid offers/bids for each generator are then sorted into non-decreasing/increasing order and used to form new piecewise-linear cost functions and adjust the generator’s active power limits.

The dispatch points and nodal prices from solving a unit decommitment OPF with the newly configured generators as input are used in the auction clearing mechanism to determine the proportion of each offer/bid block that should be cleared and the associated price for each.

Different pricing options arise from the fact that a gap in which any price is acceptable to all participants may exist between the last accepted offer price and the last accepted bid price (See Figure X). This allows, for example, the prevention of bids setting the price,

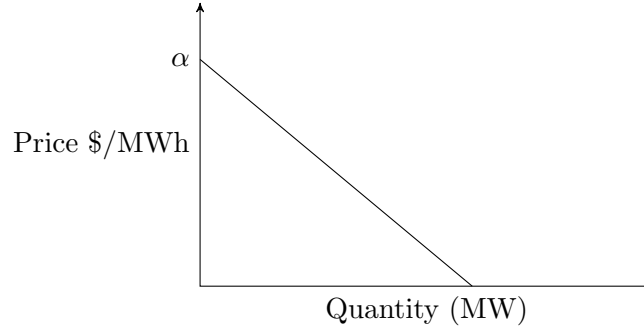


Figure 4.1: Acceptable price range

even when they are marginal, by selecting the *last accepted offer* auction type.

## 4.2 Multi-Agent System

This section describes the implementation of agents and the coordination of their interactions in multi-agent systems. A generic market environment, with which agents interact regardless of the learning method employed, is defined along with tasks that associate a purpose with an environment. The design of connectionist systems and tables, used to represent agent policies, are given and the process by which they are modified by the agent's learning algorithm is explained. Finally, the collection of agents and tasks into a multi-agent system and the sequence of interactions is illustrated.

### 4.2.1 Agent, Task & Environment

#### Environment

Each generator/dispatchable load in the power system model (See Section 4.1.1, above) is associated with an agent<sup>1</sup> via the agent's environment. Each environment maintains an association with a singular market instance for submission of offers/bids. Two main operations are supported by an agent's environment.

For a power system with  $n_b$  buses,  $n_l$  and  $n_g$  generators, the `getSensors` method returns a  $n_s \times 1$  vector of sensor values  $s_e^i$  for generator  $i$  where  $n_s = 2n_b + 2n_l + 3n_g$ .  $s_g^i$  represents the visible state of the environment for the agent associated with generator  $i$ .

---

<sup>1</sup>Management of a portfolio of generators is also supported by the architecture used, but this feature has not been exploited.

$s_e^i$  is composed of sensor values for all buses, branches and generators.

$$s_{e,l}^i = \begin{bmatrix} P_f \\ Q_f \\ P_t \\ Q_t \\ \mu_{S_f} \\ \mu_{S_t} \end{bmatrix}, \quad s_{e,b}^i = \begin{bmatrix} V_m \\ V_a \\ \lambda_P \\ \lambda_Q \\ \mu_{v_{min}} \\ \mu_{v_{max}} \end{bmatrix}, \quad s_{e,g}^i = \begin{bmatrix} P_g \\ \mu_{p_{min}} \\ \mu_{p_{max}} \\ \mu_{q_{min}} \\ \mu_{q_{max}} \end{bmatrix}, \quad s_e^i = \begin{bmatrix} s_{e,b}^i \\ s_{e,b}^i \\ s_{e,g}^i \end{bmatrix} \quad (4.54)$$

Not all values are used by the agent and the filtration is done according to the agent's task.

The **performAction** method takes  $n_a \times 1$  vector of action values  $a_e$  if  $s_{bid} = 0$ , otherwise a  $2n_a \times 1$  vector. If  $s_{bid} = 0$ , the  $i$ -th element of  $a_e$  is the offered/bid price in \$/MWh, where  $i = 1, 2, \dots, n_{in}$ . If  $s_{bid} = 1$ , the  $j$ -th element of  $a_e$  is the offered/bid price in \$/MWh, where  $j = 1, 3, 5, \dots, n_{in} - 1$  and the  $k$ -th element of  $a_e$  is the offered/bid quantity in MW where  $j = 2, 4, 6, \dots, n_{in}$ . The action vector is separated into offers/bids and submitted to the market. If  $s_{bid} = 0$ , then  $qty = p_{max}/n_{in}$ .

## Task

An agent does not interact directly with its environment, but is associated with a particular task. A task associates a purpose with an environment and defines what constitutes a reward. Regardless of the learning method employed, the goal of an agent participant is to make a financial profit and the rewards are thus defined as the sum of earnings from the previous period  $t$  as calculated by the market. Sensor data from the environment is filtered according to the task being performed. Agents using the value-function methods under test have a tabular representation of their policy with one row per environment state. Thus, observations consist of a single integer value  $s_v$ , where  $s_v \leq n_s$  and  $s_v \in \mathbb{Z}^+$ . Agents using the policy-gradient methods under test have policy functions represented by connectionist systems that use an input vector  $w_i$  of arbitrary length where the  $i$ -th element  $\in \mathbb{R}$ . Before input to the connectionist policy function approximator, sensor values are scaled to be between  $-1$  and  $1$ . Outputs from the policy are denormalised using action limits before the action is performed on the environment.

## Agent

Agent  $i$  is defined as an entity capable of producing an action  $a_i$  based on previous observations of its environment  $s_i$ , where  $a_i$  and  $s_i$  are vectors of arbitrary length. As illustrated in Figure X, each agent is associated with a *module*, a *learner* and a *dataset*. The module represents the agent's policy for action selection and returns an action vector  $a_m$  when

activated with observation  $s_t$ . The value-function methods under test use modules which represent a  $N \times M$  table, where  $N$  is the total number of states and  $M$  is the total number of actions.

$$\begin{bmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,m} \\ v_{2,1} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ v_{n,1} & \cdots & \cdots & v_{n,m} \end{bmatrix} \quad (4.55)$$

Whereas for the policy gradient methods, the module is a connectionist network of other modules as illustrated in Figure X. The learner can use any reinforcement learning algorithm and modifies the values/parameters of the policy module to increase expected future reward. The dataset stores state-action-reward tuples for each interaction between the agent and its environment. The stored history is used by value-function learners when computing updates to the policy values. Policy gradient learners search directly in the space of the policy network parameters.

Value-function learners have an association with an explorer module which returns an explorative action  $a_e$  when activated with the current state  $s_t$  and action  $a_m$  from the policy module. For example, the  $\epsilon$ -greedy explorer has a randomness parameter  $\epsilon$  and a decay parameter  $d$ . When the  $\epsilon$ -greedy explorer is activated, a random number  $x_r$  is drawn where  $0 \leq x_r \leq 1$ . If  $x_r < \epsilon$  then a random vector of the same length as  $a_e$  is returned, otherwise  $a_e = a_m$ .

#### 4.2.2 Simulation Event Sequence

In each simulation of a system consisting of one or more task-agent pairs a sequence of interactions is coordinated, as illustrated in Figure X.

At the beginning of each step/period the market is initialised and all offers/bids removed. From each task-agent tuple  $(T, A)$  an observation  $s_t$  is retrieved from  $T$  and integrated into agent  $A$ . When an action is requested from  $A$  its module is activated with  $s_t$  and the action  $a_e$  is returned.  $a_e$  is performed on the environment of  $A$  via its associated task  $T$ . Recall, this process involves the submission of offer/bids to the market. Once all actions have been performed the offer/bids are cleared using the auction mechanism. Each task  $T$  is requested to return a reinforcement reward  $r_t$ . All cleared offers/bids associated with the generator in the environment of  $T$  are retrieved from the market and  $r_t$  is computed from the difference between revenue and cost values.

$$r_t = \text{revenue} - (c_{fixed} + c_{variable}) \quad (4.56)$$

The reward  $r_t$  is given to agent  $A$  and the value is stored under a new sample is the

dataset, along with the last observation  $s_t$  and the last action performed  $a_e$ . Each agent is instructed to learn from its actions using  $r_t$ , at which point the values/parameters of the module of  $A$  are updated according to the algorithm of the learner.

This constitutes one step of the simulation and the process is repeated until the specified number of steps are complete. Unless agents are reset, the complete history of states, actions and received rewards is stored in the dataset of each agent.

## Chapter 5

# Learning to Trade Power

To the best of the author’s knowledge, this thesis presents the first case of policy gradient reinforcement learning methods being applied to electricity trading problems. It must first be proven that these methods are capable of learning a basic power trading policy. This section describes the method used to compare methods in their ability to do so.

### 5.1 Aims & Objectives

The purpose of this first experiment is to compare the relative abilities of value-function and policy gradient methods in learning a basic policy for trading power. The objective of the exercise is to examine:

- Speed of convergence to an optimal policy,
- Magnitude and variance of profit and,
- Sensitivity to algorithm parameter changes.

### 5.2 Method of Simulation

Each learning method is tested individually using a range of parameter configurations. A power system model with one bus, one generator  $k$  and one dispatchable load  $l$ , as illustrated in Figure X is used. In this context, the market clearing process is equivalent to creating offer and bids stacks and finding the point of intersection. A passive agent is associated with the dispatchable load. This agent bids for  $-p_{g,l}^{min}$  at marginal cost each period regardless of environment state or reward signal. A dispatchable load is used instead of a constant load to allow a price to be set. Generator  $k$  is given sufficient capacity to supply the demand of the dispatchable load,  $p_{g,k}^{max} > -p_{g,l}^{min}$ , and the marginal of the  $k$  is half that of the load  $l$ . The generator and dispatchable load attributes are given in Table

X. A price cap for the market is set to twice the marginal cost of the  $l$  at full capacity,  $p_{g,l}^{min}$ . The DC optimal power flow formulation (See Section 2.1, above) is used to clear the market and reactive power trade is omitted. The Python code used to conduct the simulations is provided in Listing X.

### **5.3 Results**

### **5.4 Discussion**

### **5.5 Critical Analysis**

## Chapter 6

# Competitive Power Trade

Having compared the learning methods in a one-player context, this section describes the method used to pit them against one and other and compare their performance.

### 6.1 Aims & Objectives

Competition is fundamental to markets and this experiment aims to compare learning methods in a complex dynamic market environment with multiple competing participants. The objective is to compare:

- Performance, in terms of profitability, over a finite number of periods,
- Profitability when trading both active and reactive power.
- Consistency of profit making and,
- Sensitivity to algorithm parameter changes.

### 6.2 Method of Simulation

Figure X illustrates the structure of the six bus power system model, from (Wood & Wollenberg, 1996), with three generators and fixed demand at three of the buses used to provide a dynamic environment with typical system values. Bus, branch and generator attribute values are stated in Tables X, Y, Z, respectively. Three learning methods are compared in six simulations encapsulating all method-generator combinations.

A price cap  $c_{cap}$  of twice the marginal cost of the most expensive generator at full capacity is set by the market. The simulations are repeated for with agents actions composing both price and quantity and with just price. For the value-function methods, the state is defined by the market clearing price from the previous period, divided equally into



$x_s$  discrete states between 0 and  $c_{cap}$ . The state vector  $s_t$  for the policy gradient methods consists of the market clearing price and generator set-point from the previous period.

$$s_t = \begin{bmatrix} c_{mcp} \\ p_g \end{bmatrix} \quad (6.1)$$

The script used to conduct the simulation is provided in Listing X.

### **6.3 Results**

### **6.4 Discussion**

### **6.5 Critical Analysis**

## Chapter 7

# System Constraint Exploitation

One of the main features of agents using policy gradient learning methods and artificial neural networks for policy function approximation is their ability to accept many signals of continuous sensor data. This section describes an experiment in which the power system is severely constrained for certain periods, resulting in elevated nodal marginal prices in particular areas. The methods are tested in their ability to exploit these constraints and improve their total accumulated reward.

### 7.1 Aims & Objectives

### 7.2 Results

### 7.3 Discussion

### 7.4 Critical Analysis

## Chapter 8

# Further work

8.1 AC Optimal Power Flow

8.2 Decentralised Trade

8.3 Standardisation

8.4 Blackbox optimisation

## Chapter 9

# Summary conclusions

# Bibliography

- Bakirtzis, A., & Tellidou, A. (2006, November). Agent-based simulation of power markets under uniform and pay-as-bid pricing rules using reinforcement learning. In *Power systems conference and exposition, 2006. psce '06. 2006 ieee pes* (p. 1168-1173). 15
- Barto, A. G., Sutton, R. S., & Anderson, C. W. (1983). Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-13*, 834-846. 6
- Erev, I., & Roth, A. E. (1998). Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. *The American Economic Review*, 88(4), 848-881. 9
- Gieseler, C. (2005). *A Java reinforcement learning module for the Repast toolkit: Facilitating study and implementation with reinforcement learning in social science multi-agent simulations*. Unpublished master's thesis, Department of Computer Science, Iowa State University. 15
- Hongyan, L., & Tesfatsion, L. (2009, March). Capacity withholding in restructured wholesale power markets: An agent-based test bed study. In *Power systems conference and exposition, 2009* (p. 1-11). 15
- Kienzle, F., Krause, T., Egli, K., Geidl, M., & Andersson, G. (2007, September). Analysis of strategic behaviour in combined electricity and gas markets using agent-based computational economics. In *1st european workshop on energy market modelling using agent-based computational economics* (p. 121-141). Karlsruhe, Germany. 14
- Krause, T., & Andersson, G. (2006). Evaluating congestion management schemes in liberalized electricity markets using an agent-based simulator. In *Power engineering society general meeting, 2006. ieee*. 14
- Krause, T., Andersson, G., Ernst, D., Beck, E., Cherkaoui, R., & Germond, A. (2004). Nash Equilibria and Reinforcement Learning for Active Decision Maker Modelling in Power Markets. In *Proceedings of 6th IAEE European Conference 2004, modelling in energy economics and policy*. 14
- Li, H., & Tesfatsion, L. (2009, July). The ames wholesale power market test bed: A computational laboratory for research, teaching, and training. In *Ieee proceedings*,

- power and energy society general meeting*. Alberta, Canada. 15
- Lincoln, R., Galloway, S., & Burt, G. (2009, May). Open source, agent-based energy market simulation with Python. In *Energy market, 2009. eem 2009. 6th international conference on the european* (p. 1-5). 17
- Moody, J., & Saffell, M. (2001, July). Learning to trade via direct reinforcement. *IEEE Transactions on Neural Networks*, 12(4), 875-889. 7
- Nicolaisen, J., Petrov, V., & Tesfatsion, L. (2002, August). Market power and efficiency in a computational electricity market with discriminatory double-auction pricing. *Evolutionary Computation, IEEE Transactions on*, 5(5), 504-523. 10
- Overbye, T., Cheng, X., & Sun, Y. (2004, Jan.). A comparison of the ac and dc power flow models for lmp calculations. In *System sciences, 2004. proceedings of the 37th annual hawaii international conference on* (p. 9-). 18
- Peng, J., & Williams, R. J. (1996). Incremental multi-step q-learning. In *Machine learning* (pp. 226-232). Morgan Kaufmann. 12
- Peshkin, L., & Savova, V. (2002). Reinforcement learning for adaptive routing. In *Neural networks, 2002. ijcnn '02. proceedings of the 2002 international joint conference on* (Vol. 2, p. 1825-1830). 7
- Peters, J., & Schaal, S. (2006, October). Policy gradient methods for robotics. In *Intelligent robots and systems, 2006 ieee/rsj international conference on* (p. 2219-2225). 7
- Peters, J., & Schaal, S. (2008). Natural actor-critic. *Neurocomputing*, 71(7-9), 1180-1190. 7, 13
- Roth, A. E., Erev, I., Fudenberg, D., Kagel, J., Emilie, J., & Xing, R. X. (1995). Learning in extensive-form games: Experimental data and simple dynamic models in the intermediate term. *Games and Economic Behavior*, 8(1), 164-212. 9
- Sun, J., & Tesfatsion, L. (2007a). Dynamic testing of wholesale power market designs: An open-source agent-based framework. *Computational Economics*, 30(3), 291-327. 15
- Sun, J., & Tesfatsion, L. (2007b, June). Open-source software for power industry research, teaching, and training: A dc-opf illustration. In *Power engineering society general meeting, 2007. ieee* (p. 1-6). 15
- Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction (adaptive computation and machine learning)*. Mit Press. Gebundene Ausgabe. 10
- Sutton, R. S., Mcallester, D., Singh, S., & Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. In *In advances in neural information processing systems 12* (Vol. 12, pp. 1057-1063). 7
- Tesfatsion, L., & Judd, K. L. (2006). *Handbook of computational economics, volume 2: Agent-based computational economics (handbook of computational economics)*. Amsterdam, The Netherlands: North-Holland Publishing Co. 14
- Visudhiphan, P. (2003). *An agent-based approach to modeling electricity spot markets*. Un-

- published doctoral dissertation, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology. 14
- Wang, H., Murillo-Sanchez, C., Zimmerman, R., & Thomas, R. (2007, Aug.). On computational issues of market-based optimal power flow. *Power Systems, IEEE Transactions on*, 22(3), 1185-1193. 18, 23
- Watkins, C. (1989). *Learning from delayed rewards*. Unpublished doctoral dissertation, University of Cambridge, England. 11
- Weidlich, A., & Veit, D. (2008, July). A critical survey of agent-based wholesale electricity market models. *Energy Economics*, 30(4), 1728-1759. 6, 14
- Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. In *Machine learning* (pp. 229-256). 7, 12
- Wood, A. J., & Wollenberg, B. F. (1996). *Power Generation Operation and Control* (second ed.). New York: Wiley, New York. 32
- Xiong, G., Okuma, S., & Fujita, H. (2004). Multi-agent based experiments on uniform price and pay-as-bid electricity auction markets. In *Electric utility deregulation, restructuring and power technologies, 2004. (drpt 2004). proceedings of the 2004 ieee international conference on* (Vol. 1, pp. 72-76). 15
- Zimmerman, R., Murillo-Sánchez, C., & Thomas, R. J. (2009, July). Matpower's extensible optimal power flow architecture. In *Ieee pes general meeting*. Calgary, Alberta, Canada. 17
- Zimmerman, R. D., & Murillo-Sánchez, C. E. (2007, September). Matpower: A matlab<sup>TM</sup> power system simulation package (Version 3.2 ed.) [Computer software manual]. School of Electrical Engineering, Cornell University, Ithaca, NY 14853. 17, 24, 25