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.

Signed: Date: September 5, 2010

Acknowledgements

I wish to thank Professor Jim McDonald for giving me the opportunity to study at The Institute for Energy and Environment and for giving me the freedom to pursue my own research interests. I also wish to thank my supervisors, Professor Graeme Burt and Dr Stuart Galloway, for their guidance and scholarship. I wish to offer very special thanks to my parents, my big brother and my little sister for all of their support throughout my PhD.

This thesis makes extensive use of open source software projects developed by researchers from other institutions. I wish to thank Dr Ray Zimmerman from Cornell University for his work on optimal power flow, researchers from the Dalle Molle Institute for Artificial Intelligence (IDSIA) and the Technical University of Munich for their work on reinforcement learning algorithms and artificial neural networks and Charles Gieseler from Iowa State University for his implementation of the Roth-Erev reinforcement learning method.

This research was funded by the United Kingdom Engineering and Physical Sciences Research Council through the Supergen Highly Distributed Power Systems consortium under grant GR/T28836/01.

Abstract

In Electrical Power Engineering, learning algorithms can be used to model the strategies of electricity market participants. The objective of this work is to establish if *policy gradient* reinforcement learning algorithms can be used to create participant models superior to those using previously applied *value function* based methods.

Supply of electricity involves technology, money, people, natural resources and the environment. All of these aspects are changing and electricity market designs must be suitably researched to ensure that they are fit for purpose. In this thesis electricity markets are modelled as non-linear constrained optimisation problems, which are solved using a primal-dual interior point method. Policy gradient reinforcement learning algorithms are used to adjust the parameters of multi-layer feed-forward artificial neural networks that approximate each market participant's policy for selecting power quantities and prices that are offered in the simulated marketplace.

Traditional reinforcement learning methods that learn a value function have been previously applied in simulated electricity trade, but are largely restricted to discrete representations of a market environment. Policy gradient methods have been proven to offer convergence guarantees in continuous environments and avoid many of the problems that mar value function based methods.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis examines reinforcement learning algorithms in the domain of electric power trade. In this chapter the motivation for research into electricity trade is explained, the problem under consideration is defined and the principle research contributions are stated.

1.1 Research Motivation

Quality of life for a person is directly proportional to his or her electricity usage (Alam, Bala, Huo, & Matin, 1991). The world population is currently 6.7 billion and forecast to exceed 9 billion by 2050 (United Nations, 2003). Electricity production currently demands over one third of the annual primary energy extracted (The International Energy Agency, 2010) and as people endeavour to improve their quality of life, finite fuel resources will become increasingly scarce. Market mechanisms, such as auctions, where the final allocation is based upon the claimants' willingness to pay for the goods, provide a device for efficient allocation of resources in short supply.

Commercialisation of large electricity supply industries began two decades ago in the UK. The inability to store electricity, once generated, in a commercially viable quantity prevents trade as a conventional commodity. Trading mechanisms must allow shortfalls in electric energy to be purchased at short notice from quickly dispatchable generators. Designed correctly, a competitive electricity market promotes efficiency and drives down costs to the consumer, while design errors can allow market power abuse and elevated market prices.

The average total demand for electricity in the United Kingdom (UK) is approximately 45GW and the cost of buying 1MW for one hour is around £40 (Department of Energy and Climate Change, 2009). This equates to yearly transaction values of £16 billion. The value of electricity to society is particularly apparent when supply fails. The New York black-out in August 2003 involved a loss of 61.8GW of power supply to approximately 50 million consumers. The majority of supplies were restored within two days, but the event is estimated to have cost more than $6 billion (Minkel, 2008; ICF Consulting, 2003).

The value of electricity to society makes it impractical to experiment with radical changes to trading arrangements on real systems. An alternative is to study abstract mathematical models with sets of simplifying approximations and assumptions and, where possible, to find analytical solutions using digital computer programs. Competition is a fundamental part of all markets, but the strategies of human participants are difficult to model. Reinforcement learning methods can be used to represent adaptive behaviour in competing players and are capable of learning complex strategies (Tesauro, 1994).

1.2 Problem Statement

Individuals participating in an electricity market (be they representing generating companies, load serving entities, firms of traders etc.) must utilise noisy, mostly continuous, multi-dimensional data to their advantage. Certain types of data, e.g. demand forecasts, are uncertain and other types, e.g. the bids of competitors, are hidden. Reinforcement learning algorithms must operate with data of this kind if they are to successfully model participant strategies.

Traditional reinforcement learning methods attempt to find the *value* of each available action in a given state. When discrete state and action spaces are defined, these methods become restricted by Bellman's Curse of Dimensionality (Bellman, 1961) and can not be applied to highly complex problems. When used with function approximation techniques (e.g. artificial neural networks) they can be applied to continuous representations of an environment. However, the greedy updates used by most techniques have been shown to cause algorithms approximating a value function to not converge or even diverge (Tsitsiklis & Roy, 1994; Peters & Schaal, 2008; Gordon, 1995; Baird, 1995).

Policy gradient reinforcement learning methods do not attempt to approximate a value function, but to approximate a *policy-function* that, given the current perceived state of the environment, returns an action. They do not suffer from many of the problems that mar value-function based methods in high-dimensional problems. They have strong convergence properties, do not require that all states be continuously visited and work with state and action spaces that are continuous, discrete or mixed. Policy performance may be degraded by uncertainty in state data, but the learning methods do not need to be altered. They have been successfully applied in many operational settings, including: robotic control (Peters & Schaal, 2006), financial trading (Moody & Saffell, 2001) and network routing (Peshkin & Savova, 2002) applications.

It is proposed in this thesis that agents which learn using policy gradient methods may outperform those using value function based methods in simulated competitive electricity trade. It is further proposed that policy gradient methods may operate better under dynamic electric power system conditions, achieving greater profit by exploiting constraints to their financial benefit. This thesis will use electricity market simulation techniques to compare value function based and policy gradient learning methods to explore these proposals.

1.3 Research Contributions

The research presented in this thesis pertains to the academic fields of Electric Power Engineering, Artificial Intelligence and Economics. The principle contributions made by this thesis are:

- The first application of policy gradient reinforcement learning methods in simulated electricity trade.
- The first application of a non-linear optimal power flow formulation in agent based electricity market simulation.
- A new Stateful Roth-Erev reinforcement learning method.
- Simulation results comparing the convergence to a Nash equilibrium of policy gradient and value function based reinforcement learning methods.
- Simulation results that examine the exploitation of electric power system constraints by policy gradient reinforcement learning methods.
- An implementation of a power exchange auctions market model and multi-learning-agent system for simulating electricity trade.
- The idea of applying Neuro-Fitted Q-Iteration and GQ(λ) in simulations of competitive energy trade.
- A model of the UK transmission system derived from data in the National Grid Seven Year Statement.

The publications that have resulted from this thesis are: Lincoln, Galloway, and Burt (2009, 2007); Lincoln, Galloway, Burt, and McDonald (2006).

1.4 Thesis Outline

The presentation of this thesis is organised into nine chapters. Chapter 2 provides background information on electricity supply, wholesale electricity markets and reinforcement learning. It describes how optimal power flow formulations can be used to model electricity markets and defines the reinforcement learning algorithms that are later compared.

In Chapter 3 the research in this thesis is described in the context of previous work that is related in terms of application field and methodology. Publications on agent based electricity market simulation are reviewed with emphasis on the reinforcement learning methods used. Previous applications of policy gradient learning methods in other types of market setting are reviewed also.

Chapter 4 describes the power exchange auction market model and the multi-agent system used to simulate electricity trade. It defines the association of learning agents with portfolios of generators, the process of offer submission and the reward process.

Simulations that examine the convergence to a Nash equilibrium of systems of multiple electric power trading agents is reported in Chapter 5. A six bus test case is used and results for four learning algorithms under two cost configurations are presented and analysed.

Chapter 6 examines the ability of agents to learn policies for exploiting constraints in simulated power systems. The 24 bus model from the IEEE Reliability Test System provides a complex environment with dynamic loading conditions.

The primary conclusions drawn from the results in this thesis are summarised in Chapter 7. Shortcomings of the approach are noted and the broader implications are addressed. Some ideas for further work are also outlined, including alternative reinforcement learning methods and uses for a model of the UK transmission system.

Bibliography

Alam, M. S., Bala, B. K., Huo, A. M. Z., & Matin, M. A. (1991). A model for the quality of life as a function of electrical energy consumption. Energy, 16(4), 739-745.

Aleksandrov, V., Sysoyev, V., & Shemeneva, V. (1968). Stochastic optimization. Engineering Cybernetics, 5, 11-16.

Amerongen, R. van. (1989, May). A general-purpose version of the fast decoupled load flow. Power Systems, IEEE Transactions on, 4(2), 760-770.

Application of Probability Methods Subcommittee. (1979, November). IEEE reliability test system. Power Apparatus and Systems, IEEE Transactions on, PAS-98(6), 2047-2054.

Auer, P., Cesa-Bianchi, N., Freund, Y., & Schapire, R. E. (2003). The non-stochastic multiarmed bandit problem. SIAM Journal of Computing, 32(1), 48-77.

Baird, L. (1995). Residual algorithms: Reinforcement learning with function approximation. In Proceedings of the Twelfth International Conference on Machine Learning (p. 30-37). Morgan Kaufmann.

Bellman, R. E. (1961). Adaptive control processes – A guided tour. Princeton, New Jersey, U.S.A.: Princeton University Press.

Bertsekas, D. P., & Tsitsiklis, J. N. (1996). Neuro-dynamic programming. Belmont, MA: Athena Scientific.

Bhatnagar, S., Sutton, R. S., Ghavamzadeh, M., & Lee, M. (2009). Natural actor-critic algorithms. Automatica, 45(11), 2471–2482.

Bishop, C. M. (1996). Neural networks for pattern recognition (1st ed.). Oxford University Press, USA. Paperback.

Bower, J., & Bunn, D. (2001, March). Experimental analysis of the efficiency of uniform-price versus discriminatory auctions in the england and wales electricity market. Journal of Economic Dynamics and Control, 25(3-4), 561-592.

Bower, J., Bunn, D. W., & Wattendrup, C. (2001). A model-based analysis of strategic consolidation in the german electricity industry. Energy Policy, 29(12), 987-1005.

Boyd, S., & Vandenberghe, L. (2004). Convex optimization. Cambridge University Press. Hardcover.

Bunn, D., & Martoccia, M. (2005). Unilateral and collusive market power in the electricity pool of England and Wales. Energy Economics.

Bunn, D. W., & Oliveira, F. S. (2003). Evaluating individual market power in electricity markets via agent-based simulation. Annals of Operations Research, 57-77.

Carpentier, J. (1962, August). Contribution à l'étude du Dispatching Economique. Bulletin de la Society Francaise Electriciens, 3(8), 431-447.

Cole, S. (2010, February 4). MatDyn [Computer software manual]. Katholieke Universiteit Leuven.

Department of Energy and Climate Change. (2009). Digest of United Kingdom Energy Statistics 2009. In (chap. 5). National Statistics – Crown.

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.

Ernst, D., Minoia, A., & Ilic, M. (2004, June). Market dynamics driven by the decision-making of both power producers and transmission owners. In Power Engineering Society General Meeting, 2004. IEEE (p. 255-260).

Fausett, L. (Ed.). (1994). Fundamentals of neural networks: architectures, algorithms, and applications. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

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.

Glimn, A. F., & Stagg, G. W. (1957, April). Automatic calculation of load flows. Power Apparatus and Systems, Part III. Transactions of the American Institute of Electrical Engineers, 76(3), 817-825.

Glynn, P. W. (1987). Likelilood ratio gradient estimation: an overview. In Wsc '87: Proceedings of the 19th conference on winter simulation (p. 366-375). New York, NY, USA: ACM.

Goldfarb, D., & Idnani, A. (1983). A numerically stable dual method for solving strictly convex quadratic programs. Mathematical Programming, 27, 1-33.

Gordon, G. (1995). Stable function approximation in dynamic programming. In

Proceedings of the Twelfth International Conference on Machine Learning (p. 261-268). Morgan Kaufmann.

Grainger, J., & Stevenson, W. (1994). Power system analysis. New York: McGraw-Hill.

Guo, M., Liu, Y., & Malec, J. (2004, October). A new Q-learning algorithm based on the metropolis criterion. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 34(5), 2140-2143.

ICF Consulting. (2003, August). The economic cost of the blackout: An issue paper on the northeastern blackout. (Unpublished)

IEEE Working Group. (1973, November). Common format for exchange of solved load flow data. Power Apparatus and Systems, IEEE Transactions on, 92(6), 1916-1925.

Kallrath, J., Pardalos, P., Rebennack, S., & Scheidt, M. (2009). Optimization in the energy industry. Springer.

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.

Kietzmann, T. C., & Riedmiller, M. (2009). The neuro slot car racer: Reinforcement learning in a real world setting. Machine Learning and Applications, Fourth International Conference on, 0, 311-316.

Kirschen, D. S., & Strbac, G. (2004). Fundamentals of power system economics. Chichester: John Wiley & Sons.

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.

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.

Krause, T., Beck, E. V., Cherkaoui, R., Germond, A., Andersson, G., & Ernst, D. (2006). A comparison of Nash equilibria analysis and agent-based modelling for power markets. International Journal of Electrical Power & Energy Systems, 28(9), 599-607.

Li, H., & Tesfatsion, L. (2009a, 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.

Li, H., & Tesfatsion, L. (2009b, March). Capacity withholding in restructured wholesale power markets: An agent-based test bed study. In Power systems conference and exposition, 2009 (p. 1-11).

Lincoln, R., Galloway, S., & Burt, G. (2007, May 23-25). Unit commitment and system stability under increased penetration of distributed generation. In Proceedings of the 4th International Conference on the European Energy Market, 2007. EEM 2007. Cracow, Poland.

Lincoln, R., Galloway, S., & Burt, G. (2009, May). Open source, agent-based energy market simulation with Python. In Proceedings of the 6th International Conference on the European Energy Market, 2009. EEM 2009. (p. 1-5).

Lincoln, R., Galloway, S., Burt, G., & McDonald, J. (2006, 6-8). Agent-based simulation of short-term energy markets for highly distributed power systems. In Proceedings of the 41st international universities power engineering conference, 2006. UPEC '06. (Vol. 1, p. 198-202).

Maei, H. R., & Sutton, R. S. (2010). Gq(λ): A general gradient algorithm for temporal-difference prediction learning with eligibility traces. In In proceedings of the third conference on artificial general intelligence. Lugano, Switzerland.

McCulloch, W., & Pitts, W. (1943, December 21). A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biology, 5(4), 115-133.

Micola, A. R., Banal-Estañol, A., & Bunn, D. W. (2008, August). Incentives and coordination in vertically related energy markets. Journal of Economic Behavior & Organization, 67(2), 381-393.

Micola, A. R., & Bunn, D. W. (2008). Crossholdings, concentration and information in capacity-constrained sealed bid-offer auctions. Journal of Economic Behavior & Organization, 66(3-4), 748-766.

Minkel, J. R. (2008, August 13). The 2003 northeast blackout–five years later. Scientific American.

Momoh, J., Adapa, R., & El-Hawary, M. (1999, Feb). A review of selected optimal power flow literature to 1993. I. Nonlinear and quadratic programming approaches. Power Systems, IEEE Transactions on, 14(1), 96-104.

Momoh, J., El-Hawary, M., & Adapa, R. (1999, Feb). A review of selected optimal power flow literature to 1993. II. Newton, linear programming and interior point methods. Power Systems, IEEE Transactions on, 14(1), 105-111.

Moody, J., & Saffell, M. (2001, July). Learning to trade via direct reinforcement. IEEE Transactions on Neural Networks, 12(4), 875-889.

Moody, J., Wu, L., Liao, Y., & Saffell, M. (1998). Performance functions and reinforcement learning for trading systems and protfolios. Journal of Forecasting, 17, 441-470.

Naghibi-Sistani, M., Akbarzadeh-Tootoonchi, M., Javidi-D.B., M., & Rajabi-Mashhadi, H. (2006, November). Q-adjusted annealing for Q-learning of bid selection in market-based multisource power systems. Generation, Transmission and Distribution, IEE Proceedings, 153(6), 653-660.

Nash, J. F. (1950, January). Equilibrium points in n-person games. Proceedings of the National Academy of Sciences of the United States of America, 36(1), 48-49.

Nash, J. F. (1951, September). Non-cooperative games. The Annals of Mathematics, 54(2), 286-295. Available from `http://dx.doi.org/10.2307/1969529`

National Electricity Transmission System Operator. (2007, September). Large combustion plant directive (Tech. Rep.). National Grid Electricity Transmission plc. (GCRP 07/32)

National Electricity Transmission System Operator. (2010, May). 2010 National Electricity Transmission System Seven Year Statement (Tech. Rep.). National Grid Electricity Transmission plc.

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.

Nicolaisen, J., Smith, M., Petrov, V., & Tesfatsion, L. (2000). Concentration and capacity effects on electricity market power. In Evolutionary Computation. Proceedings of the 2000 Congress on (Vol. 2, p. 1041-1047).

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-).

Peshkin, L., & Savova, V. (2002). Reinforcement learning for adaptive routing. In Neural Networks, 2002. IJCNN 2002. Proceedings of the 2002 International Joint Conference on (Vol. 2, p. 1825-1830).

Peters, J. (2010). Policy gradient methods. (Available online: www.scholarpedia.org)

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).

Peters, J., & Schaal, S. (2008). Natural actor-critic. Neurocomputing, 71(7-9), 1180-1190.

Rastegar, M. A., Guerci, E., & Cincotti, S. (2009, May). Agent-based model of the Italian wholesale electricity market. In Energy Market, 2009. 6th International Conference on the European (p. 1-7).

Riedmiller, M. (2005). Neural fitted Q iteration - first experiences with a data efficient neural reinforcement learning method. In In 16th European conference on machine learning (pp. 317–328). Springer.

Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: the rprop algorithm.

Robbins, H. (1952). Some aspects of the sequential design of experiments. Bulletin American Mathematical Society, 58(5), 527-535.

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.

Schaul, T., Bayer, J., Wierstra, D., Sun, Y., Felder, M., Sehnke, F., et al. (2010). PyBrain. Journal of Machine Learning Research, 11, 743-746.

Schweppe, F., Caramanis, M., Tabors, R., & Bohn, R. (1988). Spot pricing of electricity. Dordrecht: Kluwer Academic Publishers Group.

Sharpe, W. F. (1966, January). Mutual fund performance. Journal of Business, 119-138.

Sharpe, W. F. (1994). The Sharpe ratio. The Journal of Portfolio Management, 49-58.

Stott, B., & Alsac, O. (1974, May). Fast decoupled load flow. Power Apparatus and Systems, IEEE Transactions on, 93(3), 859-869.

Sun, J., & Tesfatsion, L. (2007a). Dynamic testing of wholesale power market designs: An open-source agent-based framework. Computational Economics, 30(3), 291-327.

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).

Sutton, R. S., & Barto, A. G. (1998). Reinforcement learning: An introduction. MIT Press. Gebundene Ausgabe.

Sutton, R. S., McAllester, D., Singh, S., & Mansour, Y. (2000). Policy gradient methods for reinforcement learning with function approximation. In Advances in neural information processing systems (Vol. 12, p. 1057-1063).

Tellidou, A., & Bakirtzis, A. (2007, Novemeber). Agent-based analysis of capacity withholding and tacit collusion in electricity markets. Power Systems, IEEE Transactions on, 22(4), 1735-1742.

Tesauro, G. (1994). TD-Gammon, a self-teaching backgammon program, achieves master-level play. Neural Computation, 6(2), 215-219.

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.

The International Energy Agency. (2010, Septemeber). Key world energy statistics 2010. Paris.

Tinney, W., & Hart, C. (1967, Novemeber). Power flow solution by Newton's method. Power Apparatus and Systems, IEEE Transactions on, 86(11), 1449-1460.

Tsitsiklis, J. N., & Roy, B. V. (1994). Feature-based methods for large scale dynamic programming. In Machine learning (p. 59-94).

United Nations. (2003, December 9). World population in 2300. In Proceedings of the United Nations, Expert Meeting on World Population in 2300.

U.S.-Canada Power System Outage Task Force. (2004, April). Final report on the august 14, 2003 blackout in the united states and canada: Causes and recommendations (Tech. Rep.). North American Electric Reliability Corporation.

Veit, D., Weidlich, A., Yao, J., & Oren, S. (2006). Simulating the dynamics in two-settlement electricity markets via an agent-based approach. International Journal of Management Science and Engineering Management, 1(2), 83-97.

Vengerov, D. (2008). A gradient-based reinforcement learning approach to dynamic pricing in partially-observable environments. Future Generation Computer Systems, 24(7), 687-693.

Visudhiphan, P. (2003). An agent-based approach to modeling electricity spot markets. Unpublished doctoral dissertation, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology.

Visudhiphan, P., & Ilic, M. (1999, February). Dynamic games-based modeling of electricity markets. In Power Engineering Society 1999 Winter Meeting, IEEE (Vol. 1, p. 274-281).

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.

Wang, J., Koritarov, V., & Kim, J.-H. (2009, July). An agent-based approach to modeling interactions between emission market and electricity market. In Power Energy Society General Meeting, 2009. PES 2009. IEEE (p. 1-8).

Weidlich, A., & Veit, D. (2006, July 7-10). Bidding in interrelated day-ahead electricity markets - insights from an agent-based simulation model. In Proceedings of the 29th IAEE International Conference.

Weidlich, A., & Veit, D. (2008, July). A critical survey of agent-based wholesale electricity market models. Energy Economics, 30(4), 1728-1759.

WG 31.04. (1983). Electric power transmission at voltages of 1000 kV and above plans for future AC and DC transmission. Electra. (ELT_091_3)

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. In Machine learning (p. 229-256).

Wood, A. J., & Wollenberg, B. F. (1996). Power Generation Operation and Control (second ed.). New York: Wiley, New York.

Yao, J., Adler, I., & Oren, S. S. (2008). Modeling and computing two-settlement oligopolistic equilibrium in a congested electricity network. Operations Research, 56(1), 34-47.

Yao, J., Oren, S. S., & Adler, I. (2007). Two-settlement electricity markets with price caps and cournot generation firms. European Journal of Operational Research, 181(3), 1279-1296.

Zimmerman, R. (2010, March 19). MATPOWER 4.0b2 User's Manual [Computer software manual]. School of Electrical Engineering, Cornell University, Ithaca, NY 14853.

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.

Appendix A

Open Source Electric Power Engineering Software

For the purposes of this thesis the Matlab source code from MATPOWER was translated into the Python programming language and released as a project named Pylon[1] (Lincoln et al., 2009). It was translated to allow existing implementations of policy gradient reinforcement learning methods, from the PyBrain machine learning library (Schaul et al., 2010), to be coupled with MATPOWER's scalable and extensible optimal power flow formulations. With permission from the MATPOWER developers, the resulting code was released under the terms of the Apache License, version 2.0, and this section describes the project in the context of other open source Electrical Power Engineering software tools to illustrate the contribution made.

A.1 MATPOWER

Since 1996, a team of researchers from the Power Systems Engineering Research Center (PSERC) at Cornell University have been developing MATPOWER: a package of Matlab[2] workspace files for solving power flow and optimal power flow problems (Zimmerman, Murillo-Sánchez, & Thomas, 2009). Initial development was part of the PowerWeb project in which the team created a power exchange auction market simulator that could be accessed by multiple users simultaneously through a web-browser interface. MATPOWER was originally available under a custom license that permitted use for any purpose providing the project and authors were cited correctly, but since version 4.0b3 it has been released under the

[1] http://packages.python.org/Pylon/

[2] Matlab is a registered trademark of The Mathworks, Inc.

Package	Language	Licence	PF	MPF	DCOPF	ACOPF	CPF	SSSA	TDS	SE	SP	GUI	RL
AMES	Java	GPL			•							•	•
DCOPFJ	Java	GPL			•								
GridLab-D	C++	BSD	•	•							•		
MatDyn	Matlab								•		•		
MATPOWER	Matlab	GPL	•		•	•	•			•	•		
OpenDSS	Pascal	BSD	•	•							•	•	
PSAT	Matlab	GPL	•			•	•	•	•		•	•	
Pylon	Python	Apache	•		•	•				•	•	•	•
TEFTS	C						•		•		•		
VST	Matlab		•				•	•	•		•	•	
UWPFLOW	C						•				•		

Table A.1: Open source electric power engineering software feature matrix.

less permissive Gnu General Public License (GPL), version 3. Matpower has become very popular in education and research and has an active mailing list that is moderated by Dr Ray Zimmerman of PSERC.

Matpower includes five solvers for AC and DC power flow. The default solver uses Newton's method (Tinney & Hart, 1967) with the full Jacobian matrix updated at each iteration. Two variations on the fast decoupled method (Stott & Alsac, 1974) described in Amerongen (1989) provide quicker convergence for certain networks. The standard Gauss-Seidel method (Glimn & Stagg, 1957) is provided for academic purposes and the DC solver provides non-iterative solutions. The properties of Matlab sparse matrices are exploited to allow solvers to scale well with very large systems. All functions are run from the Matlab command-line or from within users programs and no graphical user interface is provided.

Starting with version 4.0, Matpower includes the Matlab Interior Point Solver (MIPS) that can be used for solving DC and AC optimal power flow problems (H. Wang, Murillo-Sanchez, Zimmerman, & Thomas, 2007). Previously, FMINCON from the Matlab Optimization Toolbox[3] was required or one of a suite of high performance closed-source solvers: TSPOPF is a collection of three AC optimal power flow solvers, implemented in C and released as Matlab MEX files. It includes the original implementation of the step-controlled interior point method from which MIPS was derived. MINOPF provides an interface to the Fortran based MINOS[4] solver, developed at the Systems Optimization Laboratory at Stanford University, and is available only for educational and research purposes. Since version 4.0b4 Matpower has also included an interface to IPOPT from the COIN-OR project that provides an alternative open source solution to MIPS. DC optimal power flow problems can be solved with a Quadratic Programming interface to MIPS or using a MEX interface to BPMPD: a commercial interior point method for linear and quadratic programming.

Matpower has an extensible optimal power flow formulation that allows users to introduce additional optimisation variables and problem constraints. It is used internally to extend the standard DC and AC formulations to support piecewise linear cost functions, dispatchable loads, generator PQ capability curves and branch angle difference limit constraints. Examples of possible additional extensions include: reserve requirements, environmental costs and contingency constraints.

Matpower currently runs on Matlab, a commercial software product from

[3]Optimization Toolbox is a registered trademark of The Mathworks, Inc.
[4]MINOS is trademark of Stanford Business Software, Inc.

The Mathworks that is supported on all major platforms, or on GNU/Octave, a free program for numerical computation with strong Matlab compatibility.

A.2 MATDYN

MATDYN is an extension to MATPOWER developed by Stijn Cole from the Katholieke Universiteit Leuven for dynamic analysis of electric power systems. It was first released in 2009 under MATPOWER's custom license. It uses the same programming style and extends the MATPOWER case format with structs for dynamic generator and event data. MATDYN uses MATPOWER to obtain a power flow solution that is then used in solving a system of differential algebraic equations representing the power system. Results from MATDYN have been validated by Cole (2010) against those obtained from PSS/E[5] and the Power System Analysis Toolbox (See Section A.3, below) and show good correspondence.

A.3 PSAT

The Power System Analysis Toolbox (PSAT) is a Matlab toolbox for static and dynamic analysis of electric power systems developed by Federico Milano of the University of Castilla. It is released under the terms of the GNU GPL version 2 and offers routines for:

- Power flow,
- Bifurcation analysis,
- Optimal power flow,
- Small signal stability analysis,
- Time domain simulation and
- Phasor measurement unit placement.

A large number of input data formats are supported through Perl scripts and simulation reports can be exported as plain text, Excel spreadsheets or $\LaTeX\, 2_\varepsilon$ code. PSAT may be run from the Matlab command-line or through a Matlab based graphical user interface. The interface can be used with Simulink[6] to construct

[5] PSS/E is a registered trademark of Siemens Power Transmission & Distribution, Inc. Power Technologies International.

[6] Simulink is a registered trademark of The Mathworks, Inc.

cases such as the UK Generic Distribution System network shown in Figure A.1. A slightly modified version of PSAT that can be run from the GNU/Octave command-line is also available.

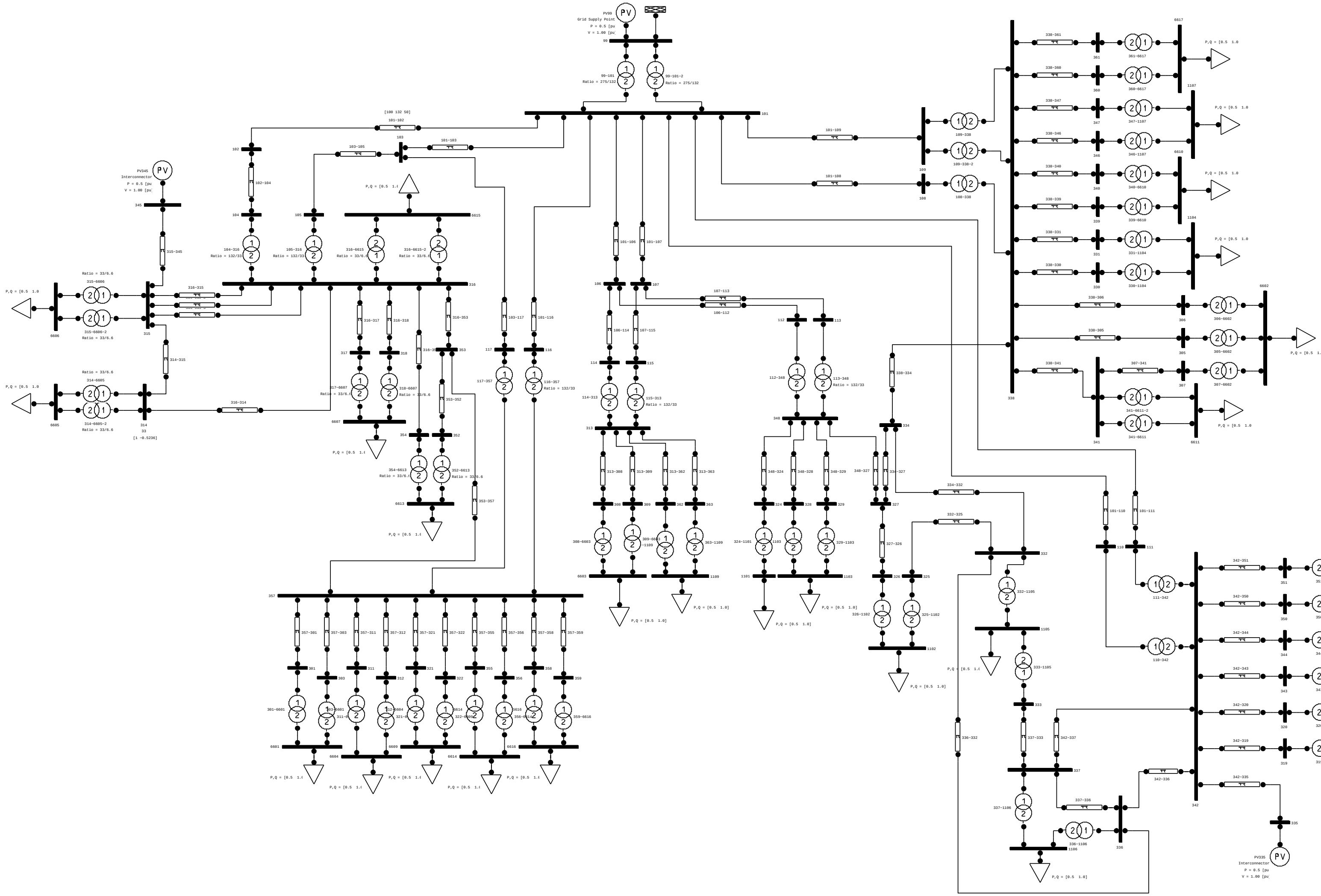

Figure A.1: UKGDS EHV3 model in PSAT Simulink network editor.

Optimal power flow problems are solved via an interface to the General Algebraic Modeling System (GAMS). GAMS defines optimisation problems using a high-level modelling language and has a large solver portfolio that includes all of the major commercial and academic solvers. The interface can be used for solving single period optimal power flow problems where the objective function can model maximisation of social benefit, maximisation of the distance to the maximum loading condition or a multi-objective combination of these. Multi-period optimal power flow is formulated as a mixed integer problem with linearised power balance constraints. The objective function models maximisation of social welfare, but is extended to include start-up and shutdown costs.

Power flow and dynamic data are often separated in electric power simulation tools, but in PSAT they are integrated. This combined with the large number of routines supported by PSAT can make the code base difficult to understand and modify. However, comprehensive documentation is included with PSAT and the mailing list is very active. The price of GAMS licenses and the need for optimal power flow problems to be converted to the GAMS language before being solved may be considered barriers to its selection for certain projects.

A.4 UWPFLOW

UWPFLOW is a research tool for voltage stability analysis developed at the University of Waterloo, Ontario, and the University of Wisconsin-Madison. It is written in ANSI-C and is available as open source for research purposes only. The program can be run with the terminal command

```
$ uwpflow [-options] input_file
```

where `input_file` is the path to a data file in the IEEE common data format (CDF) (IEEE Working Group, 1973), that may contain High-Voltage Direct Current (HVDC) and Flexible Alternating Current Transmission System (FACTS) device data. Output is also in the CDF and can include additional data for post-processing, including values for nose curve plots. An interface to UWPFLOW is provided with PSAT and can be used for bifurcation analysis.

A.5 TEFTS

The University of Waterloo also hosts TEFTS – a transient stability program for studying energy functions and voltage stability phenomena in AC/HVDC dynamic power system models. It too is written in ANSI-C and is licensed for research purposes only. An executable file for DOS is provided and the source package contains a simple example.

A.6 VST

The Voltage Stability Toolbox (VST) is a Matlab toolbox, developed at the Center for Electric Power Engineering at Drexel University in Philadelphia, for investigating stability and bifurcation issues in power systems. The source is available for any purpose providing that the authors are suitably cited. VST features routines for:

- Power flow,
- Time domain simulation,
- Static and dynamic bifurcation analysis,
- Singularity analysis and
- Eigenvalue analysis.

The feature matrix in Table A.1 shows the similar capabilities of VST and PSAT. It was developed around the same time and has the same goals for educational and research applications. However, VST does not have the same quality of documentation, graphical user interface or such an active community of users and developers.

A.7 OpenDSS

In November 2008, the Open Distribution System Simulator (OpenDSS) was released by the Electric Power Research Institute (EPRI) as open source. Development of OpenDSS began in April 1997 and it has been used extensively in studies of distribution systems including distributed generation impact assessments.

OpenDSS supports steady-state analysis in the frequency domain, including power flow, harmonics and dynamics. Arbitrary n-phase unbalanced circuit analysis is supported using an object orientated data model. Circuit elements are defined in Object Pascal and solutions are obtained using KLUSolve: a linear sparse matrix solver written in C and C++and developed specifically for solving electrical circuits. The OpenDSS Pascal code is available under the Berkeley

Software Distribution (BSD) license, which allows use for almost any purpose. KLUSolve, is available under the GNU Lesser GPL. Circuits are defined in scripts, using a domain specific language, that may be executed through a graphical user interface or a Common Object Model (COM) interface. The user interface also provides circuit data editing, plotting and power flow visualisation tools.

The power flow solver is fast and can be configured for repeated studies using daily, yearly or duty-cycle data. The multi-phase circuit model allows complex transformer models and fault conditions to be defined and three short-circuit analysis methods are provided. The heritage of OpenDSS is in harmonics and dynamics analysis and it does not support system optimisation.

A.8 GridLAB-D

GridLAB-D is an energy system simulation and analysis tool designed to investigate the latest energy technologies. The project was initiated by the U.S. Department of Energy in 2004 and developed at Pacific Northwest National Laboratory. It was released under a BSD-style license in September 2009 and has since been developed in collaboration with industry and academia.

A distributed simulation architecture is used to coordinate energy system component interactions over short and long timescales. The core of GridLAB-D is made up of modules for simulating: distribution and transmission systems, commercial and residential buildings, energy markets, power system faults and meteorological systems. GridLAB-D is written in C++ and uses a domain specific language to define models. Additional modules can be written in C++ or Java and Python is under consideration. It is designed for multicore/multiprocessor parallelism and the developers intend to use it simulate large areas of the U.S. on supercomputers. The source code includes reports and data from the Olympic Peninsula Project: a futuristic energy pricing experiment that provides a practical demonstration of GridLAB-D in operation.

GridLAB-D is a unique simulation tool that has the potential to play an important role in future energy system development. Its size and complexity can make for a steep learning curve, but extensive documentation is provided and training courses are run periodically. Activity on the mailing lists is low, suggesting poor uptake, but the software is actively supported and a new version is under development.

A.9 AMES

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

- A centralised structure managed by an independent market operator,
- Parallel day-ahead and real-time markets and
- Locational marginal pricing.

Learning agents represent load serving entities or generating companies and learn using Roth-Erev reinforcement learning methods, implemented using the Repast agent simulation toolkit (Gieseler, 2005). Agents learn from the solutions of hourly bid/offer based DC-OPF problems formulated as quadratic programs using the DCOPFJ package (Sun & Tesfatsion, 2007b) (See Section A.10, below).

The capabilities of AMES are demonstrated using a 5-bus network model in Li and Tesfatsion (2009a). The model is provided with AMES and a step-by-step tutorial describes how it may be used. AMES comes with a Swing-based graphical user interface with plotting and table editor tools and is released under the GNU GPL, version 2.

A.10 DCOPFJ

To solve market problems defined in AMES, researchers at Iowa State University developed a stand-alone DC optimal power flow solver in Java named DCOPFJ. It formulates optimal power flow problems as convex quadratic programs which are solved using QuadProgJ. The same researcher developed QuadProgJ as an independent solver that uses a dual active set strictly convex quadratic programming algorithm (Goldfarb & Idnani, 1983). DCOPFJ requires generator costs to be modelled as polynomial functions, of second order or less, and does not exploit sparse matrix features.

A.11 PYLON

Pylon is a translation of MATPOWER v4.0b2 and MATDYN v1.2 to the Python programming language. It has extensions for agent-based electricity market simulation that provide features similar to those of AMES. Both the DC and AC formulations of the extensible optimal power flow model from MATPOWER are implemented (Zimmerman et al., 2009). Either a Python version of MIPS or an interface to IPOPT from COIN-OR can be used to compute solutions. The sparsity of the problems is exploited throughout the solution process using matrix packages from SciPy and bindings to SuperLU or UMFPACK for LU decomposition and solving sparse sets of linear equations. Scripts are provided for reading and writing data files in PSS/E, MATPOWER and PSAT format. A wide variety of learning methods are available in Pylon due to its use of the PyBrain machine learning library (Schaul et al., 2010). PyBrain also provides the artificial neural network models used for policy function approximation, that may be accelerated using C extension modules from the ARAC sub-project.

In addition to its market simulation capabilities, Pylon also features solvers for power flow problems (using fast decoupled or Newton's method), state estimation, continuation power flow and time domain simulation. Pylon includes both a text interface and a graphical user interface (GUI) based on TkInter: which is included with Python and imposes no additional dependencies. A feature rich GUI is provided by plug-ins for Puddle: an extensible, GUI toolkit independent integrated development environment, developed for the purposes of this thesis also.

The use of matrix libraries from NumPy and SciPy has allowed Pylon (with the permission of the MATPOWER developers) to be released under the Apache license, version 2.0. This allows Pylon to be used as a library in proprietary software as well as free and open source tools since derivatives of the source code may be made available under more restrictive terms than the original Apache license. This is in contrast to strong "copyleft" licenses, such as the GNU GPL, that require the same rights to be preserved in modified versions of the work.

A.12 Summary

A diverse range of open source Electric Power Engineering tools are available. Implementations of most of the traditional power system analysis routines can be found and many offer performance comparable with proprietary offerings. Various programming languages are used, but Matlab is the most popular choice.

Several projects are licensed under the GNU GPL and it ensures that all users have access to the full source code. This does impose restrictions on the redistribution of projects that use the routines and this can be a barrier to use in certain types of project. To encourage commercial use and promote industrial involvement, two large code bases (OpenDSS and GridLAB-D) have recently been released under weak copyleft licenses. Pylon was developed using specific scientific computing libraries and permission was obtained from the developers of MATPOWER to allow release under a similarly permissive license. Most of the projects described above are led and developed by one individual and contributions from the user community are typically minimal. It is hoped that Pylon's use of a popular free programming language and its liberal licensing conditions will encourage community involvement and lead to inventive combinations of simulation routines and web technologies in the development of intelligent electric power grids.

Appendix B

Case Data

This appendix provides data for the electric power system models used in Chapters 5 and 6.

B.1 6-Bus Case

The data for the six bus case adapted from Wood and Wollenberg (1996, pp. 104, 112, 119, 123-124, 549) is presented in this section. The data was imported from the "case6ww.m" case file provided with MATPOWER. Table B.1 lists the bus data, Table B.2 lists the generator data and Table B.3 lists the branch data.

Table B.1: 6-bus case bus data.

Bus	P_d	Q_d	G_s	B_s	V_{base}	V_{max}	V_{min}
1	0	0	0	0	230	1.05	1.05
2	0	0	0	0	230	1.05	1.05
3	0	0	0	0	230	1.07	1.07
4	70	70	0	0	230	1.05	0.95
5	70	70	0	0	230	1.05	0.95
6	70	70	0	0	230	1.05	0.95

Table B.2: 6-bus case generator data.

Bus	P_{max}	P_{min}	V_g	Q_{max}	Q_{min}
1	1.05	200	50	100	-100
2	1.05	150	37.5	100	-100
3	1.07	180	45	100	-100

Table B.3: 6-bus case branch data.

From	To	r	x	b_c	S_{max}	τ	θ_{ph}
1	2	0.1	0.2	0.04	40	0	0
1	4	0.05	0.2	0.04	60	0	0
1	5	0.08	0.3	0.06	40	0	0
2	3	0.05	0.25	0.06	40	0	0
2	4	0.05	0.1	0.02	60	0	0
2	5	0.1	0.3	0.04	30	0	0
2	6	0.07	0.2	0.05	90	0	0
3	5	0.12	0.26	0.05	70	0	0
3	6	0.02	0.1	0.02	80	0	0
4	5	0.2	0.4	0.08	20	0	0
5	6	0.1	0.3	0.06	40	0	0

B.2 IEEE Reliability Test System

This section provides data for the modified IEEE Reliability Test System that was imported from the "case24_ieee_rts.m" case file, provided with MATPOWER and was originally contributed by Bruce Wollenberg. Table B.4 lists the bus data, Table B.5 lists the generator data, Table B.6 lists the branch data and Table **??** lists the generator cost data provided by Georgia Tech Power Systems Control and Automation Laboratory.

Table B.4: IEEE RTS bus data.

Bus	P_d	Q_d	G_s	B_s	V_{base}	V_{max}	V_{min}
1	108	22	0	0	138	1.05	0.95
2	97	20	0	0	138	1.05	0.95
3	180	37	0	0	138	1.05	0.95
4	74	15	0	0	138	1.05	0.95
5	71	14	0	0	138	1.05	0.95
6	136	28	0	-100	138	1.05	0.95
7	125	25	0	0	138	1.05	0.95
8	171	35	0	0	138	1.05	0.95
9	175	36	0	0	138	1.05	0.95
10	195	40	0	0	138	1.05	0.95
11	0	0	0	0	230	1.05	0.95
12	0	0	0	0	230	1.05	0.95
13	265	54	0	0	230	1.05	0.95
14	194	39	0	0	230	1.05	0.95
15	317	64	0	0	230	1.05	0.95
16	100	20	0	0	230	1.05	0.95
17	0	0	0	0	230	1.05	0.95
18	333	68	0	0	230	1.05	0.95
19	181	37	0	0	230	1.05	0.95
20	128	26	0	0	230	1.05	0.95
21	0	0	0	0	230	1.05	0.95
22	0	0	0	0	230	1.05	0.95
23	0	0	0	0	230	1.05	0.95
24	0	0	0	0	230	1.05	0.95

Table B.5: IEEE RTS generator data.

Bus	P_{max}	P_{min}	V_g	Q_{max}	Q_{min}	Type
1	152	30.4	1.035	60	-50	U76
2	152	30.4	1.035	60	-50	U76
7	300	75	1.025	180	0	U100
13	591	207	1.02	240	0	U197
14	0	0	0.98	200	-50	SynCond
15	155	54.3	1.014	80	-50	U155
16	155	54.3	1.017	80	-50	U155
18	400	100	1.05	200	-50	U400
21	400	100	1.05	200	-50	U400
22	300	60	1.05	96	-60	U50
23	310	108.6	1.05	160	-100	U155
23	350	140	1.05	150	-25	U350

Table B.6: IEEE RTS branch data.

From	To	r	x	b_c	S_{max}	τ	θ_{ph}
1	2	0.0026	0.0139	0.4611	175	0	0
1	3	0.0546	0.2112	0.0572	175	0	0
1	5	0.0218	0.0845	0.0229	175	0	0
2	4	0.0328	0.1267	0.0343	175	0	0
2	6	0.0497	0.192	0.052	175	0	0
3	9	0.0308	0.119	0.0322	175	0	0
3	24	0.0023	0.0839	0	400	1.03	0
4	9	0.0268	0.1037	0.0281	175	0	0
5	10	0.0228	0.0883	0.0239	175	0	0
6	10	0.0139	0.0605	2.459	175	0	0
7	8	0.0159	0.0614	0.0166	175	0	0
8	9	0.0427	0.1651	0.0447	175	0	0
8	10	0.0427	0.1651	0.0447	175	0	0
9	11	0.0023	0.0839	0	400	1.03	0
9	12	0.0023	0.0839	0	400	1.03	0
10	11	0.0023	0.0839	0	400	1.02	0
10	12	0.0023	0.0839	0	400	1.02	0
11	13	0.0061	0.0476	0.0999	500	0	0
11	14	0.0054	0.0418	0.0879	500	0	0
12	13	0.0061	0.0476	0.0999	500	0	0
12	23	0.0124	0.0966	0.203	500	0	0
13	23	0.0111	0.0865	0.1818	500	0	0
14	16	0.005	0.0389	0.0818	500	0	0
15	16	0.0022	0.0173	0.0364	500	0	0
15	21	0.0063	0.049	0.103	500	0	0
15	21	0.0063	0.049	0.103	500	0	0
15	24	0.0067	0.0519	0.1091	500	0	0
16	17	0.0033	0.0259	0.0545	500	0	0
16	19	0.003	0.0231	0.0485	500	0	0
17	18	0.0018	0.0144	0.0303	500	0	0
17	22	0.0135	0.1053	0.2212	500	0	0
18	21	0.0033	0.0259	0.0545	500	0	0
18	21	0.0033	0.0259	0.0545	500	0	0
19	20	0.0051	0.0396	0.0833	500	0	0
19	20	0.0051	0.0396	0.0833	500	0	0
20	23	0.0028	0.0216	0.0455	500	0	0
20	23	0.0028	0.0216	0.0455	500	0	0
21	22	0.0087	0.0678	0.1424	500	0	0