



Graduate Theses, Dissertations, and Problem Reports

2002

Operating market based regulation service using software agents compliant with NERC's control performance standards

Ramanujam Srinivasan
West Virginia University

Follow this and additional works at: <https://researchrepository.wvu.edu/etd>

Recommended Citation

Srinivasan, Ramanujam, "Operating market based regulation service using software agents compliant with NERC's control performance standards" (2002). *Graduate Theses, Dissertations, and Problem Reports*. 1899.

<https://researchrepository.wvu.edu/etd/1899>

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.

**OPERATING MARKET BASED REGULATION SERVICE
USING SOFTWARE AGENTS COMPLIANT WITH
NERC'S CONTROL PERFORMANCE STANDARDS**

Ramanujam Srinivasan

**Thesis submitted to
The College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of**

**Master of Science
In
Electrical Engineering**

**Ali Feliachi, Ph.D., Chair
Srinivas Kankanahalli, Ph.D.
Bojan Čukić, Ph.D.**

Lane Department of Computer Science and Electrical Engineering

**Morgantown, West Virginia
2002**

**Keywords:
Software Agents, Load Frequency Control, Power Market, Ancillary Services
Market Clearing Price, Bidding Strategy**

Copyright 2002 Ramanujam Srinivasan

ABSTRACT
OPERATING MARKET BASED REGULATION SERVICE
USING SOFTWARE AGENTS COMPLIANT WITH
NERC'S CONTROL PERFORMANCE STANDARDS

Ramanujam Srinivasan

With the changing scenario for procurement of energy it becomes necessary to understand the process of obtaining energy from a diverse set of suppliers capable of providing substantial amounts of electric power at competitive prices. Sufficient insight has been gained in the energy brokerage system design and planning owing to experiences in the recently established markets especially the California market. It becomes contextual to analyze and understand the procurement of ancillary services, which are generally bundled as part of the wholesale energy supply chain, using a similarly competitive environment having a number of players that provide electric power for such services.

The objectives of this thesis are: (1) to provide a simulation package for conducting competitive auctions using software agents for the regulation service auction market, and (2) to demonstrate the compliance of a power system, employing Automatic Generation Control with parameters obtained from such a market, with North American Electric Reliability Council's performance standards. The package employs a flexible and extensible Java-based agent development environment, MADKIT, to simulate the auctions for regulation service, and MATLAB/SIMULINK models with a fuzzy controller to simulate the power system. The framework is tested using a sample three-area power system, where the parameters for regulation service in the second area are obtained from a competitive auction market. A bidding strategy based on fuzzy logic is also designed and tested for ensuring good profit for a bidding supplier in the auctions.

ACKNOWLEDGEMENTS

First I would like to express my gratitude to my advisor Professor Ali Feliachi for providing me the opportunity to work in this project. His continuous encouragement, positive attitude, critical examination, and motivational remarks have greatly helped me over the past two years.

I would like to thank Professors Bojan Čukić and Srinivas Kankanahalli for serving in my examining committee, and also providing timely help through discussions. Special thanks to Dulpichet Rerkpreedapong, Amer AL Hinai, Amer Hasanović, Karl Schoder, and the rest of my friends in the project team for providing valuable suggestions and performing peer reviews of my work.

Finally, my deepest gratitude and love goes to my parents, my fiancée, and my entire family, for providing unconditional support, understanding, patience and encouragement. This work is dedicated to them. My humble pranams to the lotus feet of His Holiness Bhagwan Sairam for being with me as a guiding light through this endeavor and my entire life.

Funding for this project was provided by the National Science Foundation under grant ECS-9870041 and a DOE/EPSCoR WV State Implementation Award.

TABLE OF CONTENTS

LIST OF FIGURES vii

LIST OF TABLES vii

CHAPTER 1 INTRODUCTION

1.1 Introduction and Background 1
1.2 Outline..... 3

CHAPTER 2 LITERATURE SURVEY

2.1 Work related to Energy Markets and Ancillary Services..... 5
2.2 Work related to Software Agents in Power Systems..... 9

CHAPTER 3 PROCURING ENERGY USING AN AUCTION MARKET

3.1 Market Structure and the CalPX..... 13
3.2 Bidding Strategies..... 15
3.3 Auction market for Ancillary Services 16
 3.3.1 Capacity Reservation and Energy Prices..... 17
 3.3.2 Participation Factors for the bidders..... 19

CHAPTER 4 SOFTWARE AGENTS BASED AUCTION MARKET FOR REGULATION SERVICE

4.1 Agent Architecture for Collaboration..... 21
4.2 Madkit and its Interface with Matlab..... 23
4.3 Auction Market for Regulation Service..... 25

4.3.1 The Auction Process	25
4.3.2 Run-Time Dispatch and Actual Payment	27

CHAPTER 5
FUZZY SYSTEM DESIGN FOR BIDDING STRATEGY AND COMPLIANCE WITH NERC STANDARDS

5.1 Uniform Random Number based Bidding Strategy	30
5.2 Fuzzy Inference System based Bidding Strategy	30
5.3 Fuzzy controller design for NERC standards compliance.....	34

CHAPTER 6
SIMULATION AND RESULTS

6.1 FIS based Regulation Service Compliant with NERC Standards.....	37
6.1.1 CPS1 and CPS2	38
6.2 Sample Three Area System and Market Interaction	40
6.3 Auction Market Simulation	41
6.4 Power System Simulation and NERC Compliance	47

CHAPTER 7
CONCLUSIONS AND FUTURE WORK

7.1 Conclusions.....	49
7.2 Recommendations for future work	49

APPENDIX A

SIMULINK MODELS AND RESULTS OF SIMULATION.....	51
---	-----------

APPENDIX B

GLOSSARY.....	55
----------------------	-----------

APPENDIX C

SOURCE CODE FOR AGENT PROGRAMS	57
C.1 Program ISO.JAVA.....	57
C.2 Program SUPPLIER.JAVA.....	67
C.3 Program BIDPRICE.JAVA	73

REFERENCES

REFERENCES.....	76
------------------------	-----------

LIST OF FIGURES

Figure 3.1: Organizational Structure of the power market.....	14
Figure 3.2: Participation Factors obtained using a Competitive Market.....	20
Figure 4.1: Agent/Group/Role model.....	22
Figure 4.2: Madkit Architecture for Agent Development.....	24
Figure 4.3: Auction Market for Regulation Service	26
Figure 5.1: Snapshot of simulation using Madkit platform.....	29
Figure 5.2: Input Membership Functions.....	31
Figure 5.3: Output Membership Functions.....	32
Figure 5.4: Surface Plot of the Fuzzy Inference System.....	33
Figure 5.5: Time period for determining compliance.....	34
Figure 5.6: Operation of the fuzzy controller.....	35
Figure 5.7: Input membership functions for the fuzzy controller	35
Figure 5.8: Output membership functions for the fuzzy controller	35
Figure 5.9: Surface plot for the fuzzy controller.....	36
Figure 6.1: Sample Three-Area System	40
Figure 6.3: Plot of number of successful bidders in First Sample Run	43
Figure 6.4: Plot of number of successful bidders in Second Sample Run	43
Figure 6.5: Plot of Rewards for Bidders in First Sample Run.....	45
Figure 6.6: Plot of Rewards for Bidders in Second Sample Run	45
Figure 6.7: Plot of price changes for bidder-2 in sample runs.....	46
Figure 6.8: Plot of price changes for bidder-2 in sample runs.....	47
Figure 6.9: Model showing area-2 with the parameters from the market	48
Figure A.1: Simulink model of the sample three area system	51
Figure A.2: Simulink model of the fuzzy controller in the system	51
Figure A.3: NERC CPS1 compliance of area-2.....	52
Figure A.4: NERC CPS2 compliance of area-2.....	52
Figure A.5: Area Control Error (ACE) signal of area-2.....	53
Figure A.6: Fuzzy Controller Output signal for area-2	53

LIST OF TABLES

Table 5.1: Fuzzy Logic Rules for Bid Change.....	33
Table 5.2: Fuzzy Logic Rules for controller gain inference	36
Table 6.1: Parameters for the units in the simulation	41

CHAPTER 1

INTRODUCTION

1.1 Introduction and Background

The power industry has been undergoing numerous changes over the past decade. Many of the changes are due to the modification of the “way things happen”. The unified and regulated power sector has been transformed into a diversified and deregulated environment, where power can be traded like a commodity in the market. In the recent days, this scenario is effectively seen in the California power exchange market [1]. The previously vertically integrated utility system is now split into three major components: Generation companies (Gencos), Transmission companies (Transcos) and Distribution companies (Discos) enabling a competitive environment for provision of each of these services. Competitive markets for energy trading are being formed as part of the restructuring of the electricity supply industries. Since electricity is delivered instantaneously, it is obvious that a day-ahead market requires a balancing service with an associated real-time market [2]. In this context, the Ancillary Services market derives importance owing to the Independent System Operator’s responsibility to provide reliable power to the consumer along with maintaining system stability.

Software agents have been employed for a wide variety of applications involving distributed computing and decision-making. An agent is a computer system situated within and as a part of an environment that senses that environment and acts on it, over time, in pursuit of its design objectives [3, 4]. In general, they possess the following characteristics:

- Agents are *autonomous*, that is they act on behalf of the user, and *goal-oriented*.

- They contain some level of *intelligence*, from fixed rules to learning engines that allow them to adapt to changes in the environment.
- They not only act *reactively*, but sometimes also *proactively*.
- They can *communicate* and *cooperate* with other agents in a collaborative environment to solve complex tasks.

Distributed control and decision-making is mainly useful when the components are geographically dispersed and there is more logic in keeping certain control decisions localized. The auction set up for the power market would be an ideal situation for application of distributed processing involving software agents to accomplish the decision tasks and design objectives of the participants in the market. Though sufficient scope and research exists in applying agents for the energy supply and demand market [5], the real-time energy balancing market for supply of Ancillary Services, such as operating reserves, regulation service and reactive power supply, has not matured enough to ensure perfect competition, and hence requires further investigation for the provision of such services.

The objective of this thesis work aims at providing a framework for the ancillary services auction market for investigating the behavior of software agents representing the various participants including the auction coordinator or ISO (Independent System Operator) and simulation of the influence of such market behavior on the power system Load Frequency Control to ensure compliance with NERC (North American Electric Reliability Council) control performance standards. A simulation environment *Madkit* (Multi-Agent Development Kit), which simulates the agents in the Java virtual machine, is used to study the dynamics of the auction market as well as model the bidding

strategies for the agents competing for providing regulation service. The results of the hourly auctions are provided to the power system as participation factors for the winning bidders to provide regulation service. A fuzzy-logic based controller [6] is used along with a conventional PID controller for ensuring compliance with performance standards for the power system Automatic Generation Control (AGC).

1.2 Outline

The thesis is organized as follows.

In chapter 2, a literature survey is presented, discussing the work done by other researchers in defining and elaborating the procurement process for energy and ancillary services in auction markets. The application of software agents to power systems stability as well as their use in the competitive procurement of energy from auction markets is also presented.

In chapter 3, a description of the power market auction mechanism is given with emphasis on the need for such a market for ancillary services. An account of the techniques developed for energy procurement from the suppliers by the governing authority is provided. The concept of strategic bidding employed by suppliers to maximize profit from the market together with optimal use of resources is explained and an account of the costs incurred by bidding suppliers in the regulation service auction market is also given.

In chapter 4, the software agents paradigm and its application to the regulation service scenario is presented. The environment details of the Madkit platform and its interface with Matlab are given. A detailed algorithm for conducting the auction for regulation

service using agents to represent the controller, who conducts the auction market, and the participants, who maximize their gain by bidding in the market, is provided.

In chapter 5, a fuzzy logic based bidding strategy that could be adopted by the agents for success in the auction market is discussed and compared with a simple random number bidding strategy to complement its effectiveness in ensuring good profits for the bidding supplier.

In chapter 6, simulation results of the proposed system on the *Madkit* platform for the agents, and Matlab/Simulink platform model for power systems compliance with the NERC standards using market-based data are presented.

In chapter 7, the research conclusion is presented and recommendations are made for further work.

CHAPTER 2

LITERATURE REVIEW

In this chapter a literature survey of topics related to procurement of energy and ancillary services via auction markets, and application of software agents to power systems stability and energy marketplace is given.

2.1 Work related to Energy Markets and Ancillary Services

Over the past decade, significant amount of research work has been done towards forming rules and procedures for deregulation and implementation of profound know-how in the economic and technical areas of power systems. Different countries adopt different models of the electricity market applying various operating mechanisms. Sheblé [7] introduces various auction mechanisms for the deregulated power industry with detailed treatment of methodologies followed towards ensuring competition in all aspects of the industry. An introduction to power market design with significant emphasis on economic theory as applicable to energy trading markets is provided by Stoft [8]. The plan of deregulation is to achieve efficiency through competition. In a competitive market, suppliers adjust their output until marginal cost equals the market price and adjust the price until the market clears i.e., supply equals demand.

Goran Strbac [9] explains the electricity trading set up in England and Wales, the basis for selecting the appropriate market or the combination of markets, and the types of ancillary services markets like reactive power service and frequency regulation service. There is a stress on the need for generation companies to quantify the value of various energy and ancillary services products in order to optimally allocate their activities in a number of competing markets. California [10], on the other hand, has established a

system with both pool operations, where all producers of power pool their energy into a central dispatching agency, which ensures dispersal of energy and revenue to the producers, and bilateral contracts, where producers and consumers of power come to mutual agreements about transfer of electric power and corresponding revenues. The California Power Exchange (CalPX) serves as a good example for understanding the decision process facing producers in the deregulated marketplace. Illic and Skantze [11] present a simplified account of the operating rules of the new energy marketplace, the challenges faced by the producers towards reliable operation and profit making, and the feasible risk management by the producers in the associated financial markets. From the perspective of the power systems stability and operability, Alvarado [12] examines the dynamics of power system markets considering single and multiple supplier and consumer scenarios. It is stressed that the stable well-behaved markets for electric power require consideration of the dynamics of those markets. The stabilizing price signal reflecting the energy error and effects of congestion on the equilibrium point for the market are discussed with simplifying assumptions on participant behavior and elasticity of the power demand.

A simple, intuitive dynamic model for consumer/supplier response is presented by Meng et al [13] with the energy imbalance hypothesized as the key driving term for updating price. Numeric tests with various market update models, with detailed generator/turbine/ governor dynamics, are developed. Eigen-values are examined and stability regions are given for market-only dynamics and coupled system dynamics. However, network losses are ignored and a simple approximation is made equating the energy imbalance with the weighted sum of frequency errors. Chattopadhyay et al [14]

and Weber et al [15] consider inclusion of pricing scheme into the economic dispatch scheduling while using the price signal as a feedback (demand pricing and generation pricing) to both the importing side and exporting side, and only on the load side using Optimal Power Flow algorithm respectively. Load Frequency Control (LFC) technical issues with deregulation are examined by Christie et al [16] emphasizing the operational structures likely to result from deregulation.

Papalexopoulos et al. [17] describe some challenging issues in the design of real-time competitive market for electricity with deregulation. Two major forms of dispatch of energy are examined namely, Merit-Order dispatch, where power dispatch is based on a merit-order energy stack obtained by arranging the prices from the power suppliers in order, and Optimal Power Flow (OPF), where an Optimal Power Flow algorithm is used to determine power dispatched to the supplier units according to a Total Cost-Minimization objective. Ancillary Services auction mechanisms, operational concerns, compliance of the participants towards committed deliverables, and various options for settlement are presented. It gives a good understanding of the underlying principles and procedures for competitive procurement of energy and support services. Ilic et al. [18] provide the basic structure for a power exchange market structure for frequency control to enable purchasing and selling of frequency control as a well-defined commodity.

Most of the above work has been focused on the basic structure for the primary energy market, while the ancillary services have been given less focus mainly due to the lesser quantity of sale and amount of profit that could be obtained from such a market. There are other researchers who have worked exclusively on bidding strategies in the competitive market to maximize supplier or consumer gains. In the auction-market based

bidding environment, the participant's freedom to price away from the marginal production costs is utilized for increasing profit from the market. The notion of each participant about the system and other participants in the market is considered as estimates Lamont [19] and a sub-optimal bidding strategy is investigated for a two-player case. In a similar case, Shrestha et al. [20] present a strategic bidding with respect to ramp up rate of the power supplier where the power balance and spinning reserve requirement are considered simultaneously. A method is proposed to optimize the benefit of the supplier by fine-tuning its bids for ramp up rate and price while ensuring its success in the competition with other suppliers.

Xiong et al [21] present an improved evolution based approach developed to evolve the bidding strategy and maximize the supplier's profit in the long run. Based on a number of simple assumptions and simulations the authors have shown the feasibility of the proposed bidding strategy. Stochastic and ordinal optimization based formulations with assumptions of perfect competition are done by Wen, Zhang and Guan et al [22,23,24] where bidding strategies are developed considering expectations of other suppliers and the presence of a market regulator along with the application of Lagrangian relaxation technique for solving optimal power generation. Optimal bidding strategy considering probability of winning is examined by Hao [25] by solving differential equations that reflect the conditions for bidders to maximize their expected payoffs. A supplier would construct his bid as a function of his private cost and cost distributions of other bidders. Otero-Novas et al. [26] employ mixed integer programming to simulate a real wholesale electricity market. The expected bid prices and quantities, system hourly prices and generation schedules are estimated, taking into account the bidding strategies

of generators and the structure of the market. Application of fuzzy logic towards bidding strategies in a continuous double auction is presented by Leung and He [27], where the authors give an account of the heuristic fuzzy rules, prevalent techniques for changing the bidding strategy and a reasoning mechanism to find the best ask/bid for a hypothetical agent.

Building on the effort described in [2], [6], [17], [18] for Load Frequency Control and market structure and [21], [27] for bidding strategies, a market structure for the procurement of Regulation Service in a competitive one-sided auction market with suppliers employing simple bidding strategies for success in supplying regulation service to an anticipated demand is presented in this thesis.

2.2 Work related to Software Agents in Power Systems

Software agents have been in the power systems domain for quite some period owing to their inherent qualities that tie-up well with the distributed nature of the electrical system. Russell and Norvig [4] provide an excellent textual as well as simulation oriented explanation of agent technology with possible application scenarios and behavioral patterns of multi-agent systems. Ferber, Knapik, Weiss and Bond et al [28,29,30,31] provide an account of the prevalent techniques for analyzing, designing, and implementing multi-agent systems for a variety of applications including telecommunications, air-traffic control, telemetry in enterprise systems, and supply chain management. Reasoning about the state of the individual agent and its neighbors, interaction languages like KQML (Knowledge Querying and Manipulation Language), FIPA (Foundation of Intelligent Physical Agents) and KIF (Knowledge Interchange Format), implementation using object-oriented techniques, and typical structures and

applications are discussed. The realization of Distributed Artificial Intelligence or DAI to a problem at hand is visualized as an application of intelligent software agents. Warwick et al. [32] present many of the prevalent artificial intelligence techniques in power systems. An account of applications of evolutionary programming, fuzzy logic rule-based systems, neural network systems and interestingly hybrid systems, which are a combination of any/all of the aforementioned schemas of DAI is given with experiences from specific project environments along with relevant simulation results.

Woolridge [33] considers the problem of building a multi-agent system as a software engineering enterprise. His article focuses on specification, implementation and verification of such systems with reference to a number of case studies. Jennings et al. [34, 35] present a methodology for agent-oriented analysis and design considering the system to be developed as a computational organization with various interacting roles. Analysis is performed using a Roles model and Interaction model for the agents at the micro-level, while design uses the Agent model, Services model and Acquaintance model at the macro-level. The key concepts are roles, which have associated with them responsibilities, permissions and protocols. The methodology is illustrated with an example business process management system. At the website agentbuilder.com [36] a complete and comprehensive list of agent-building environments in the academic and commercial areas, along with a description and link to the web page of the corresponding project is provided. Amin [37, 38] explains the inherent complexity in both system maintenance and stable operation that has resulted from the effects of deregulation and restructuring of the energy industry. He illustrates the Electric Power Research Institute's (EPRI) initiative to develop new tools and techniques to enable the power system

infrastructure to self-stabilize and maintain reliable operation when subject to failures and destabilizing occurrences, and suggests that agent-oriented technology will be well suited for such an application owing to its inherent distributed nature.

Application of intelligent software agents to stability analysis of power systems has also been prevalent in the literature. Kundur et al [39, 40, 41] provide good accounts of the various techniques like eigen-value analysis and transient stability analysis with loss of a line and cascading failures. The majority of the work for applying software agents to power systems stability analysis has been done by Talukdar and his team using Collaborative Networks [42, 43, 44]. Here, C-Nets are employed to solve complex problems by breaking them up into simpler problems and solving them using agents operating asynchronously. The Rolling Horizon Formulation (Model Predictive Control) is employed to break up the overall dynamic control task into a series of localized static optimization problems, each approximating a part of the total problem task. The agents communicate using a Collaboration protocol to find solutions to the overall problem. Prototypical networks illustrated include the array of pendulums and standard electric grid (IEEE 14 bus and 30 bus networks were chosen for the experiments).

Getting to the field of power system simulations using software agents, Harp et al. [45] developed a proof-of-concept prototype tool, Simulator for Electric Power Industry Agents or SEPIA as an agent-oriented approach to solve complex tasks arising in the new power system operation. They illustrate the use of a simulation engine with a multi-agent system that takes care of business processes as well as control objectives. The system is based on component software technologies and runs on windows 32-bit platforms. In [46] the authors Lucas and Jahromi present a framework for agent-based operation of the

power market with buyers, sellers and a coordinator. The main emphasis in this work is finding the bidder agent commitment period, and the bidding time in the decision process of an agent submitting bids in a Dutch auction scenario. An algorithm is presented to determine bidder prices and amounts to enable market intelligent agents. Lam and Wu [47, 48] suggest use of Java technology for implementing multi-agent system for simulating electricity markets owing to object-orientation, platform-independence, support for distributed decision-making, communication among participants and interface with various software tools. A market simulator based on the California system has been designed and developed using the JATLite multi-agent template environment. Java API, JNI (Java Native Interface) is used to invoke native methods executing power system analysis and market analysis programs.

Building on the effort described in [33], [34], [46], [47] for Multi-Agent Systems (MAS), an agent-based single-sided auction market controlled by an Independent System Operator (ISO) responsible for procuring Ancillary Services including regulation service to an anticipated demand is presented in this thesis. The Multi-Agent Development Kit or *Madkit* [49] environment developed using Java Beans™ Technology is used for creating the supplier and ISO agents and running the auction for regulation service every hour for a 24-hour period. The participation factors acquired by the suppliers are then utilized for maintaining the Area Control Errors of a three-area example system. MATLAB®/SIMULINK® [50] models are used for illustration of the compliance with North American Electric Reliability Council's (NERC) performance standards using a Fuzzy-Logic based controller.

CHAPTER 3

PROCURING ENERGY USING AN AUCTION MARKET

In this chapter, focus is given on the procurement of energy from an auction market, the terms associated with such a market, and the need for a similar auction scenario for providing ancillary services. The California ISO model is considered here as a primary reference for explaining the methodology for energy procurement as well as ancillary services procurement. A summary of the terms and expressions can be obtained from the glossary in order to understand their usage throughout the material.

3.1 Market Structure and the CalPX

Most of the market structures used for trading power in the United States involve an auction mechanism to ascertain which generation units should be used to meet the demand. Some areas like California also allow for bilateral trades between load and generation. The California system consists of a Power Exchange (CalPX) to facilitate trading between GENCOs and ESCOs (Energy Service Companies that may be large industrial customers or customer pools acting to purchase power at the least cost when needed by the consumers) and serves as a good example for understanding the process of acquiring the required power from an auction market. A supplier wishing to sell power supplies a *bid curve* to the exchange that reflects the power delivering capacity of that supplier as a function of market price. Such bid prices are submitted on an hourly and day-ahead basis and a different bid curve may be submitted for each of the 24 hours of future operation. Similarly bid curves are obtained from consumers wishing to utilize power at corresponding prices.

The bids from either side are used to compile aggregate supply and demand curves for each hour. The intersection of the supply and demand curves determines the Market Clearing Price (MCP). Now, all supply bids with a price less than the MCP are accepted and those bidders are paid the MCP, while all demand bids higher than MCP are accepted and bidders are charged the MCP. Thus, the balance is ensured between supply and demand and the PX remains revenue neutral. Figure 3.1 provides a pictorial illustration of the relationship between the bidding entities in the market and the governing entities. With this organizational structure in place there exist other variations and types of markets for procuring energy.

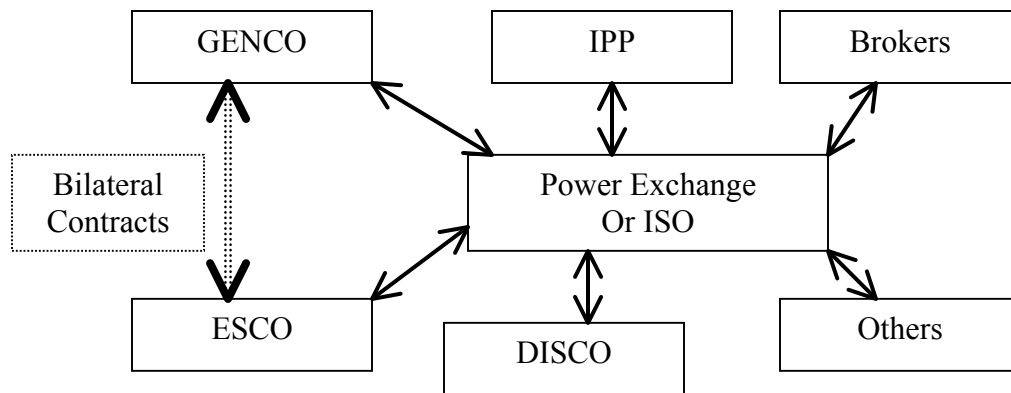


Figure 3.1: Organizational Structure of the power market

Among the various market structures are the markets that include cash, futures, planning and swap markets. The cash market undertakes trading in spot market that reflects real-time power trading and forward contracts that reflect trading for power interchange at a future time. The futures market facilitates trading of what are termed as futures and options, which are a means of risk management in power trading. While the planning market is intended towards trading of contracts aimed at system expansion, the swap market establishes trading between markets when the participants desire conversion

from one type of contract to another. Further explanation and investigation of these markets and the relations between them is beyond the scope of this work.

3.2 Bidding Strategies

There exist a number of ways in which the participating suppliers of energy namely Independent Power Producers (IPP), Generation Companies (Gencos) or any other producers capable of selling their power can maximize their net worth or profit from the auction market. Also, in a dual auction environment, there exist consumers, who can be large industrial entities, Distribution Companies (Discos) or any other consumers capable of buying their required power from the market. These consumers of electric power can also implement simple, intuitive bidding strategies or intelligent, elite strategies based on gaming theory and optimization to increase the benefit they would obtain from the auction market.

The ISO or PX would implement an optimization algorithm to dispatch the bids in an efficient manner based on maximum social benefit, least cost formulation or least loss formulations. Among the various methods employed by the participants in the auction the most prominent would be optimization-based techniques that aim at maximizing or minimizing an objective function that characterizes the cost or benefit function of each individual participant in the auction. This function could very well be one that reflects the variable costs associated with fuel procurement and O & M (Operation and Maintenance) costs. These methods are generally termed as bidding strategies aimed at maximizing the benefit to the bidder from the competitive set up.

Microeconomic theory holds that the optimal bidding strategy [5] for a supplier is to simply bid marginal cost. When a generator tries to exploit imperfection in the market

to increase its profit from the market, it is said to employ strategic bidding. Among the various ways to employ strategic bidding, probabilistic methods and fuzzy methods of estimating bid values or behaviors of the other participants in the market are among the prominent techniques.

The market operator generally employs a merit-order based dispatch procedure, or an optimal power flow (OPF) based dispatch procedure, to procure power from the participants in the supply side, which is the side with greater competition among the bidders. Several citations [15, 19, 22] can be offered from the literature describing the nuances and procedures involved in these mechanisms, which have proven quite effective in achieving their primary purposes of procuring electric power in a competitive manner. The main idea here is to understand the existence of several methods to procure power from bidding suppliers and briefly preview the objective function based techniques. Similar techniques, often simpler in nature, are employed to procure power for balancing services such as regulation, spinning reserves and replacement reserves using competitive auction markets.

3.3 Auction market for Ancillary Services

Since electricity is delivered instantaneously, it is obvious that a day-ahead market requires a balancing service with an associated real-time market. In this context, the Ancillary Services market derives importance owing to the ISO or Independent System Operator's responsibility to provide reliable power to the consumer along with maintaining system stability. The ISO must ensure that there is sufficient capacity among available generators to participate in the real-time market and secure enough reserve capacity to keep the system operating reliably. Though sufficient scope and research

exists in applying agents for the energy supply and demand market, the real-time energy balancing market for supply of Ancillary Services, such as operating reserves, regulation service and reactive power supply, is not mature enough to ensure perfect competition and hence requires further investigation for the provision of such services. . Such a separate reserve auction is better in the market structure where the ISO does not have any direct bondage with the Power Exchange (PX), which takes care of the functioning of the forward energy market.

Considering a one-sided auction for regulation service the following section examines the costs and payments the bidding suppliers are entitled for setting aside a portion of their capacity for such service.

3.3.1 Capacity Reservation and Energy Prices

Generators provide Automatic Generation Control (AGC) services by reserving a portion of their unloaded capacity for use in real-time by the ISO, and these can be made available within a 10-minute period. There are two types of costs associated with such markets where capacity is reserved in advance and later dispatched in real time. These include the opportunity costs associated with providing the service and the actual cost incurred during real-time dispatch of power.

The general cost function used for studies is a quadratic approximation of the total costs involved in providing power by the bidder 'i' that would include fixed costs for infrastructure as well as variable costs for fuel procurement and Operation and Maintenance costs.

This function is given by:

$$\left[C_i(P_{gi}) = \alpha_i + \beta_i * P_{gi} + \gamma_i * (P_{gi})^2 \right] \quad (3.1)$$

$\alpha_i, \beta_i, \gamma_i$: are constants that reflect approximated cost parameters.

An infra-marginal unit incurs an Opportunity Cost (cost of an opportunity lost to bid in the real-time market for energy) for providing regulation service since it is constrained by ramp-rate limitations. Hence, there occurs a need to provide for the opportunity costs in the bids submitted by such units in the auction in terms of a Capacity Reservation Bid (R_i). In the case of AGC, a unit is required to operate above its minimum output in order to allow movement in upward and downward directions. Since no unit would prefer to operate at higher levels when the market price is lower than its costs, it is entitled to bid for a capacity payment to cover the actual costs the unit may incur during operation.

The value of reservation capacity bid, R_i is given by

$$R_i = \beta_i + 2 * \gamma_i * (r_i * T); \quad (i = 1 \dots n) \quad (3.2)$$

Where

n: the number of participating bidders

r_i : the ramping rate of the unit (MW/min)

T: the ramping time given by ISO (Minutes)

β_i, γ_i : The values in the cost function $C_i(P_{gi})$

This R_i reflects the opportunity cost incurred by the unit in providing regulation service limited by its ramp-rate.

The other cost associated with the bidding is the Energy Bid (E_i), which is the general cost function of the bidding entity. Successful bidders from the auction are paid a capacity reservation payment that reflects their opportunity costs, and also paid for power if they provide it during real-time.

The value of the energy bid, E_i

$$E_i = \beta_i + 2 * \gamma_i * Pgi ; \quad (i = 1 \dots n) \quad (3.3)$$

This actually reflects the Marginal Cost incurred by the unit in providing power Pg_i .

With these two costs incurred by the bidders, the auction considered here takes into account a two-part bid for deciding the successful bidders and the MCP.

3.3.2 Participation Factors for the bidders

During the operation of AGC (Automatic Generation Control), each generator is required to provide some amount of power for the service in a particular control area to maintain the frequency within the required limits. In other words, each generator is required to participate in the frequency regulation service according to a pre-determined level along with the other generators in the control area. This is referred to as the participation factor of the generator in providing regulation service. As illustrated in Figure 3.2, an auction market, where the GENCOs compete to procure more participation in the service by strategic bidding, could determine the participation factors.

In an auction set up, the participation factor would be determined for each bidding hour by dividing the allocation done for each successful supplier from the outcome of the corresponding hour-auction by the forecasted demand for regulation for the particular

period. The forecasted demand is generally determined by examining historical data for the hour of operation and also by studying the weather trends and consumer behaviors.

For each hour in the auction market, corresponding participation factors are obtained for each the successful bidders who are paid the Market Clearing Price. These factors are later used during the actual dispatch hour of the next day when there is a need to actually provide power for regulation of frequency in the control area.

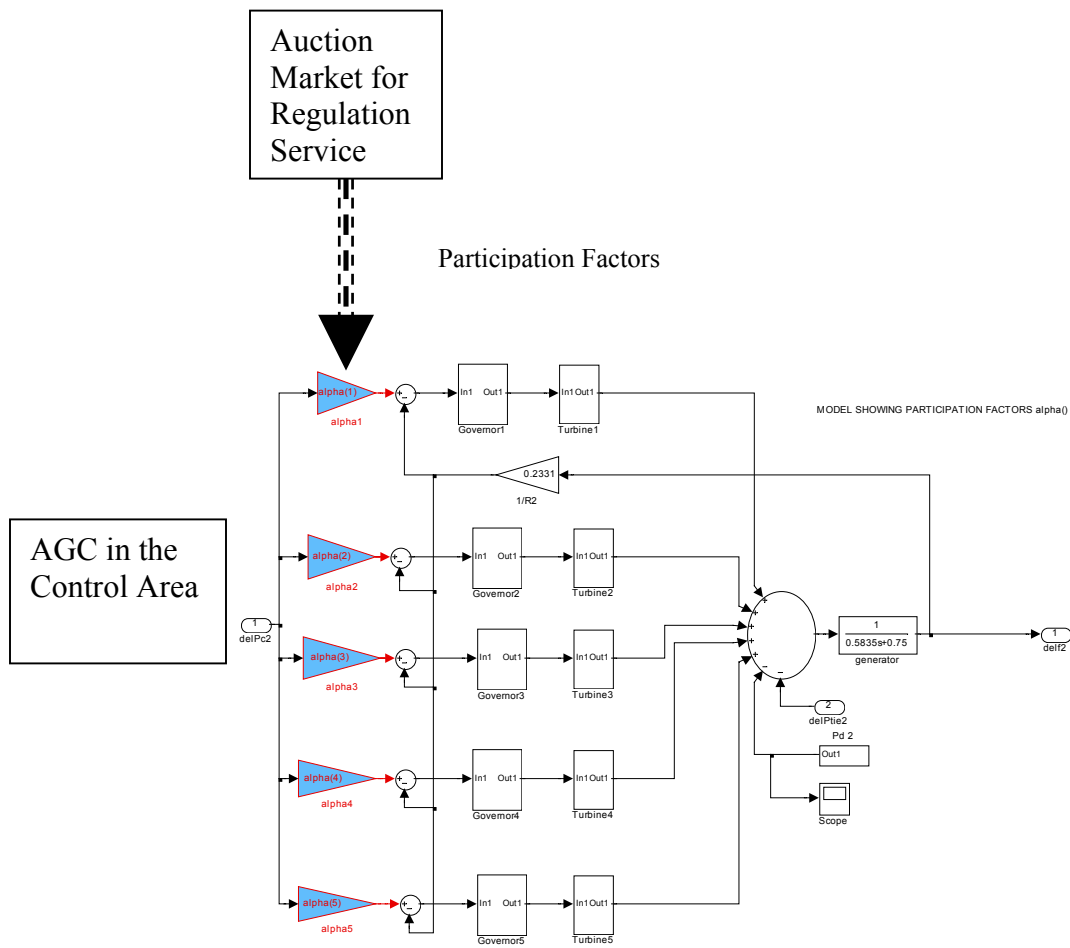


Figure 3.2: Participation Factors obtained using a Competitive Market

CHAPTER 4

SOFTWARE AGENTS BASED AUCTION MARKET FOR REGULATION SERVICE

In this chapter, the important details of the solution environment for simulating regulation service auctions using software agents are provided. While the first section explains the Agent Group/Role model and the agent-based software engineering approach for this market, the ensuing sections explain the algorithm executed for running the auction and procuring service by the controller agent.

4.1 Agent Architecture for Collaboration

Some of the qualities to be present in an agent development platform for agent collaboration and cooperation are:

- Need for a robust communication environment for ease of message transfer among the agents.
- Need to work together to achieve some predefined global objectives while completing their local operations.
- Need to use a generic, customizable and scalable platform.
- Modular construction in order to define and simulate working of the agent quickly.
- Ease of work with respect to using Agent Communication Languages – No emphasis on a specific ACL.

An auction market implementation would be close to a Decision Making System composed of agents performing corresponding roles in the automated system for the entities in the real world system [33]. The agent specifications should be capable of capturing at least the following information:

- The *beliefs* the agents have;
- The ongoing *interactions* the agents have with their environment and among themselves;
- The *goals* that agent will try to achieve;
- The *actions* that agents perform and the effects of these actions.

Auctions are similar to a *Contract-Net Protocol* application, where a manager agent with a task to perform puts it out to tender to a number of contractors agents. The contractors willing to perform the task return bids to the manager that are evaluated by the manager according to some criteria, and the contractor with the best bid is allocated the task. It is however important to identify the flow of information among the agents. It would be useful to adopt the Agent/Group/Role model [34] as shown in figure 4.1 to define the various participating agents and their roles in the regulation market. Object-oriented analysis and design can be performed identifying each agent as an *actor* performing tasks to achieve its design objectives.

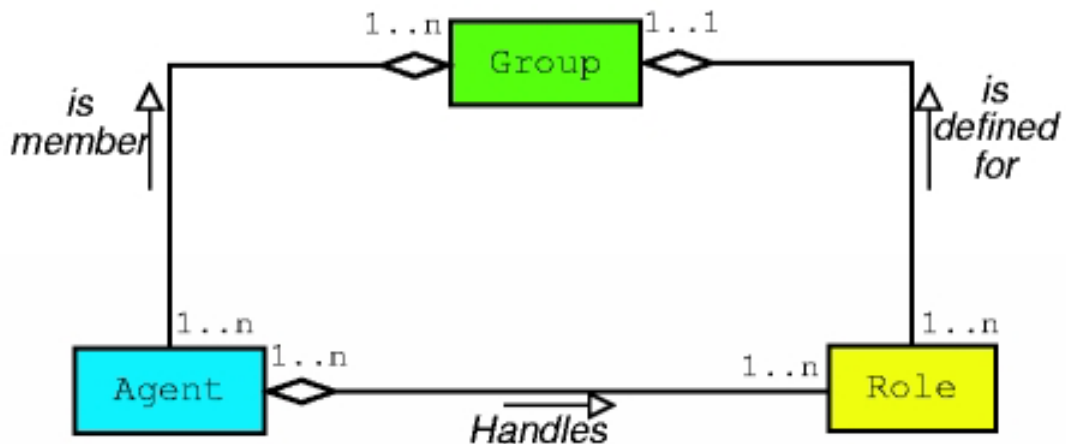


Figure 4.1: Agent/Group/Role model

The Group is an identifiable entity that consists of Agents, which are members handling one or more Roles defined within the group. While there can be more than one agent performing the same role within a group, there cannot be two roles for the agent with identical nomenclature and definitions.

There is a perception to view an organization as a collection of roles that have specified relationships with one another and hold systematic interactions with other roles. Collections of roles are aggregated into groups and an agent can hold distinct roles in many groups. While an agent is considered as an active communicating entity, the group is a means of agent aggregation and the role is an abstraction of an agent function, service or identification within a group.

4.2 Madkit and its Interface with Matlab

Madkit [49] is a versatile agent development platform built using the Java Beans specification and the agent/group/role paradigm. It is a generic, highly customizable and scalable with a goal of providing a foundation layer for various agent models, and to make the basic services extensible and replaceable. It is a set of Java classes that implements the agent kernel, the various libraries of messages, probes and agents. It includes a graphical environment and many system and demonstration agents.

The platform has three basic design principles (Figure 4.2):

- Micro-kernel architecture
- Agentification of services
- Component model for graphical interface

The idea is to provide a modular development environment with minimum resource requirements for building customized agent applications with no work from the developer towards system activities. The developer need only to concentrate on the architecture, goals, and other application aspects for the agents without having to look into communication, addressing and other services.

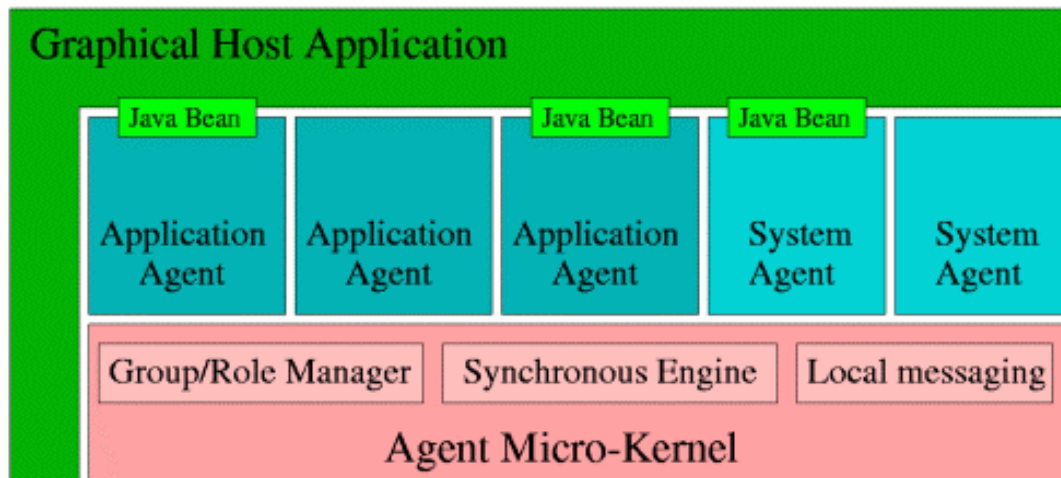


Figure 4.2: Madkit Architecture for Agent Development

The interface between Madkit and Matlab is established using the JMatLink software Java class library [51] that provides a *.dll* file to call Matlab routines. The basis of this function is the Java Native Interface (JNI) method declaration within the JMatLink class and the corresponding C language implementation calling Matlab routines. The ‘mex’ file utility is used to enable C language syntax commands to call Matlab commands and execute them inside the Matlab engine previously invoked using the *engOpen()* method. With calls to Matlab computational scripts and functions from the agent bean in Madkit, a lot of additional functionality can be implemented into the agents.

4.3 Auction Market for Regulation Service

The reserve auctions differ in a number of ways from the regular energy auction market. While the main aim of the PX auction is to facilitate trade of electricity between buyers and sellers, the procurement of ancillary services is the sole responsibility of the ISO, which “reserves in advance” a portion of the unloaded capacity of the participating bidders. The ISO pays the successful bidders for the reserved capacity as well as for the energy used during run-time for energy balancing.

4.3.1 The Auction Process

The implemented market structure assumes that there exists a separate market for regulation service. The ISO procures regulation service from bidding suppliers in the following steps:

- Each of the bidders submits an Energy Bid (E_i) in \$/MWh and a separate Reservation Capacity Bid (R_i) in \$/MW to the ISO at the beginning of each auction for the corresponding hour during the next day.
- After receiving all the bids, the ISO builds what is called the “Energy Stack” in which the bids are arranged in ascending order of bid prices. In this implementation the sum of both parts of the bid is used while building the energy stack.
- The forecasted demand or required capacity is filled starting from the lowest price point on the stack working upwards until the capacity needed is satisfied.
- The last price at which the required capacity is filled is termed as the “Market Clearing Price (MCP)” in \$/MW.

- The successful bidders are the ones below the MCP in the energy stack and are paid their capacity bids initially to compensate for their opportunity costs. Their energy bids are the amounts paid to them when they are called upon to provide energy at run-time.

A pictorial representation of the auction process is shown in Figure 4.3. The bidders compete against each other in the open market to secure a position in the dispatch stack and deliver as much power as possible as a profitable price. In this context, the bidding strategy employed by the bidder plays a significant role in ensuring success for the bidder.

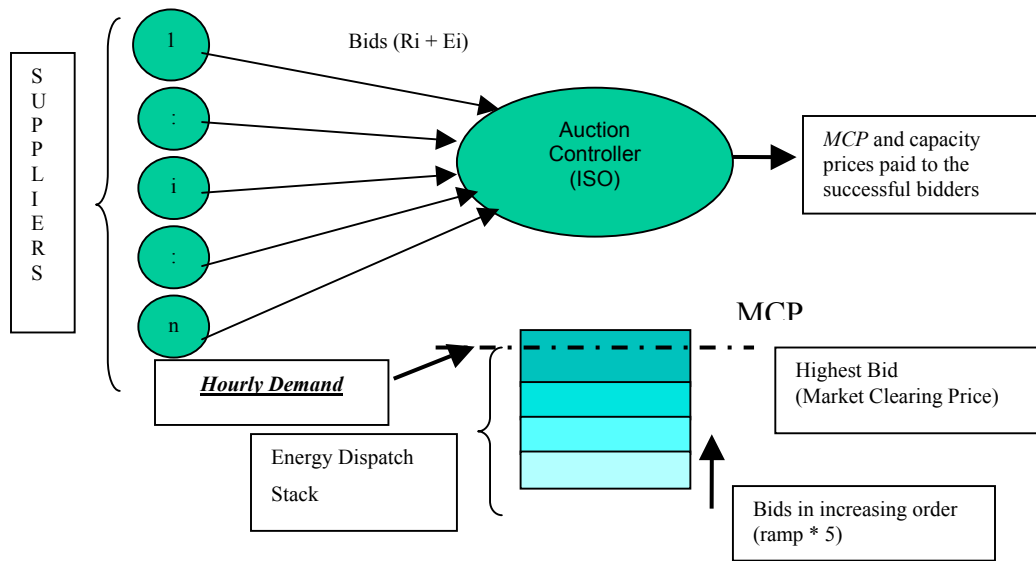


Figure 4.3: Auction Market for Regulation Service

Among the different ways in which payments are paid to the suppliers we have considered payment for the reserved capacity and run-time dispatch. The outputs of the auction are:

- The Market Clearing Price.
- The Reservation Capacity Price C_i given by

$$C_i = MCP - E_i \quad (4.1)$$

where

MCP: the Market Clearing Price from the energy stack for the bid-hour

E_i : the Energy Bid of the successful bidder for the bid-hour

- The participation factor for each successful bidder unit in the regulation service during that bid-hour of operation during the next day given by:

$$p.f = \frac{\text{Capacity allocated for the bidder for the hour}}{\text{Total forecasted regulation demand for the hour}} \quad (4.2)$$

In this implementation, there occur two payments made by the ISO to the successful bidders, one for the regulation reserved capacity, which is the MCP from the auction results, and another paid in real-time when the unit is called for supply of regulation power, which is nothing but the energy bid of the bidder during the auction. The reserved capacity price is C_i , and the real-time price is the energy ‘strike’ price E_i .

The auction market participants can be implemented as *Supplier* agents and *ISO* agent over an Internet Protocol (IP) network. The computation complexity can be embedded into the agents to reveal the actual calculations done by the suppliers in providing bids to the ISO controller. This approach would be more pragmatic since bidders could be geographically distributed in a real-life situation.

4.3.2 Run-Time Dispatch and Actual Payment

It would be a good assumption to consider that the suppliers are price takers in the auction market, where they cannot set the real-time of their commodity and have to accept the resultant price from the auction. The successful bidders in the auction market

are called for supplying power for regulation service at run-time according to the order of the bids in the dispatch stack, i.e., the lower the energy bid of the bidder the better is the chance for calling upon that bidder to supply at run-time.

For the sake of simplicity of agent implementations a competitive auction scenario for the suppliers has been considered and the agents do not deal with issues such as uninstructed deviations and subsequent settlements. The successful bidders in each hour auction are paid the MCP from that auction for their capacity reservation initially. When called upon for dispatch at run-time the generators receive additional strike price payment according to their energy bid during the auction. This is the form of implementation in the California market [10] and has been adopted for the agent simulation and auction automation process in this work.

CHAPTER 5

FUZZY SYSTEM DESIGN FOR BIDDING STRATEGY AND COMPLIANCE WITH NERC STANDARDS

In this chapter, the two types of bidding strategies implemented to improve the profit obtained by the supplier agents in the auction market are explained. While the first section explains the Fuzzy Logic based Bidding Strategy, which proves to be effective in ensuring good profits for its agent, the second section explains the design of the Fuzzy Controller that ensures compliance with NERC's performance standards for the operating power system. Figure 5.1 shows a snapshot of the simulation performed on Madkit. With the strategies in place, it provides an opportunity to study the working of the market when each of the bidders chooses a different strategy for making more profit from the market or enters the dispatch stack after a failed attempt.

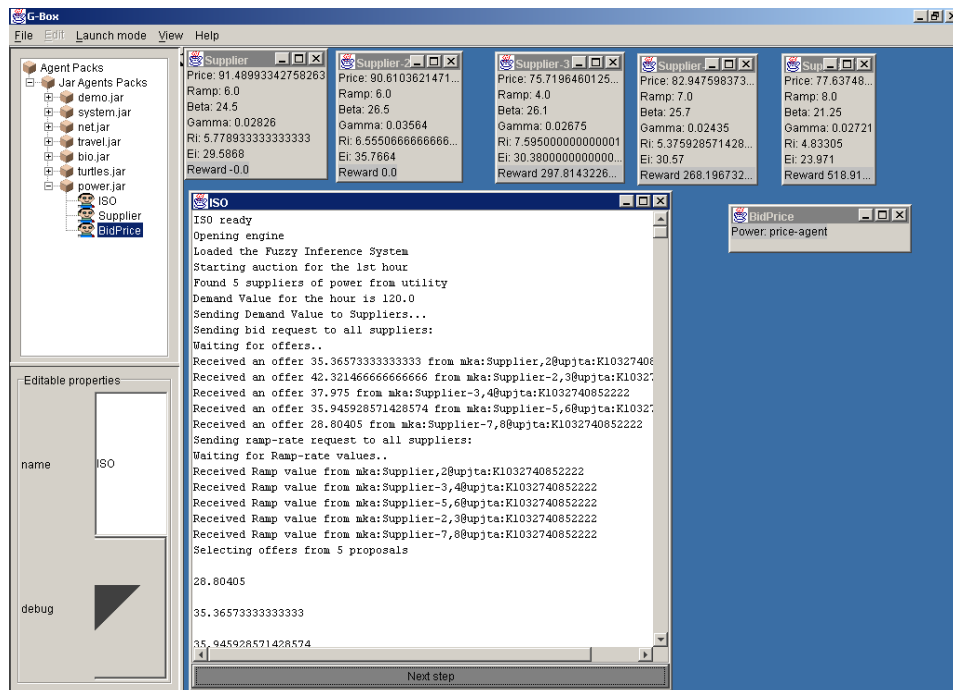


Figure 5.1: Snapshot of simulation using Madkit platform

5.1 Uniform Random Number based Bidding Strategy

The profit obtained by a successful supplier agent depends upon the extent to which its bid price is lower in the dispatch stack. In the random number strategy a random number is initially generated that follows a uniform gaussian distribution [21]. This number is then added or subtracted from the previous value of the bid amount depending upon the corresponding success or failure of the supplier agent to find a place in the final dispatch stack formulated by the ISO. The simple formula is given by:

$$P_{new} = P_0 \pm e \quad (5.1)$$

P_{new} : the new bid price (\$/MW)

P_0 : the original bid price (\$/MW), value over which bid price is changed

e : the uniform random component

The *randn(1)* function in Matlab is used to generate the random number and add it to the original bid price if the agent has some profit or reward from the previous auction and subtract it from the original bid price if the agent did not have any profit in the previous auction. The *randn(1)* function generates a random number that follows a uniform distribution with mean 0, variance $\sigma^2 = 1$, and standard deviation $\sigma = 1$. This strategy, however, has a static nature with a random number being simply added or subtracted without utilizing the extent of profit made by the agent.

5.2 Fuzzy Inference System based Bidding Strategy

This strategy employs a Fuzzy Inference System developed in Matlab using the Mamdani-type fuzzy system. There is one input (profit value) and one output (price change value). The membership functions for the input reward and output price change

function are given in Figure 5.2 and Figure 5.3 respectively. The output is varied according to the extent to which the input profit characterizes over the input membership functions.

The membership functions and rule base are created using the fuzzy inference system editor in Matlab, which produces an output structure having all the attributes to execute the fuzzy bidding strategy.

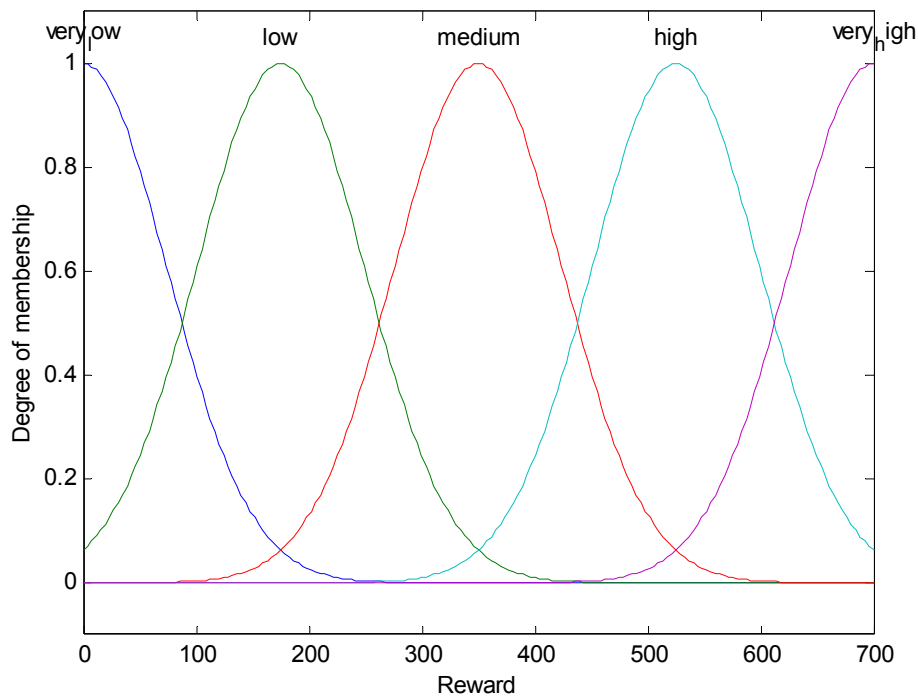


Figure 5.2: Input Membership Functions

The input membership functions describe the extent of spread of the reward or profit obtained by the successful supplier over the input space spanning until a value of 700. The behavior of the fuzzy inference system would be altered when the input space is changed to span a different set of values providing scope for further investigation of the performance of the system over different sets of input spaces.

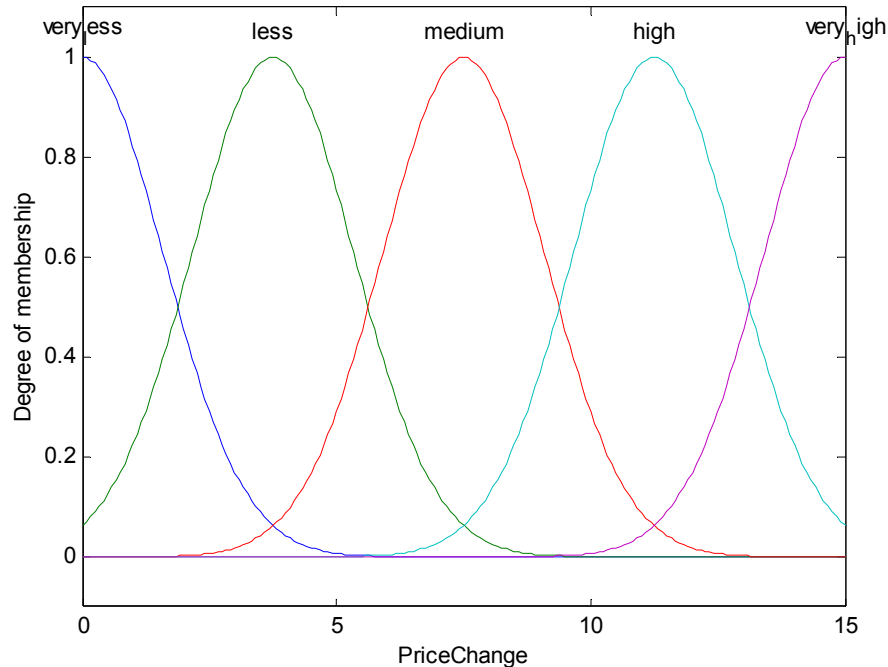


Figure 5.3: Output Membership Functions

The rule-base for computing the fuzzy rules for processing the input to produce the output is given in Table 5.1. This explains the rules used to transform the extent of profit obtained by the bidder into a corresponding extent of price change to ensure success as well as sustained profit for the bidder. Similar to the input membership functions, the output membership functions describe the extent of spread of the price change made for the successful supplier over the output space spanning until a value of 15. Hence, the behavior of the fuzzy inference system would be altered when the output space is changed to span a different set of values influencing the performance of the fuzzy system.

Table 5.1: Fuzzy Logic Rules for Bid Change

Input (profit)		Output (price change)
Very low	→	Very high
Low	→	High
Medium	→	Medium
High	→	Low
Very high	→	Very low

Finally, figure 5.4 shows the surface plot of the fuzzy system, which pictorially represents the relationship between the input and the output of the FIS.

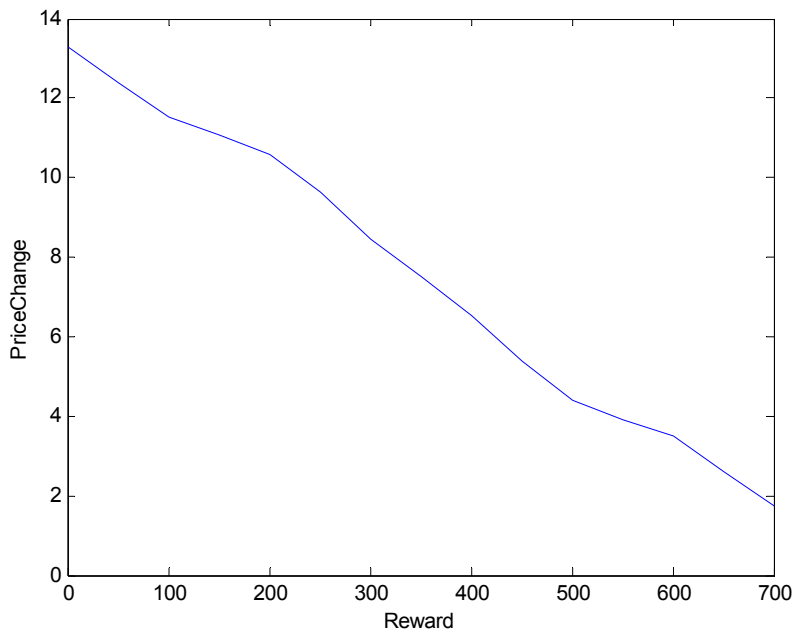


Figure 5.4: Surface Plot of the Fuzzy Inference System

5.3 Fuzzy controller design for NERC standards compliance

Each control area is required to monitor its control performance and report its compliance with CPS1 and CPS2 to NERC at the end of each month [2]. The control structure implemented is a PID type controller whose control parameter is tuned by the fuzzy rules. Specifically, when there is high compliance with CPS1, the control parameter is gradually reduced to diminish high-frequency movement in the speed governor's equipments. Once there is low compliance the control parameter is raised up accordingly. This ensures compliance with standards as well as reduced wear and tear of equipment.

The input for the fuzzy logic rules is the information that reflects compliance with CPS1 and CPS2. The FIS system will lower the control parameter of the PID controller of the secondary loop of load frequency control when the value of either input rises. On the other hand, the parameter will be increased when the value of either input reduces.

The first input to the fuzzy system is the cumulative average compliance factor (CFac) used to find compliance with CPS1 given by:

$$CF_{ac} = AVG_{X \rightarrow Y} [CF_1] \quad (5.2)$$

$$CF = AVG_{X \rightarrow Z} [CF_1] \quad (5.3)$$



Figure 5.5: Time period for determining compliance

The other input is the 10-minute window average of ACE divided by L10 ($AVG_{10} [ACE]/L_{10}$) used to find compliance with CPS2. Both the inputs are calculated every one-minute for firing the fuzzy logic output, which is the fuzzy gain (α) used to tune the gain control parameter of the PID controller as shown in figure 5.6.

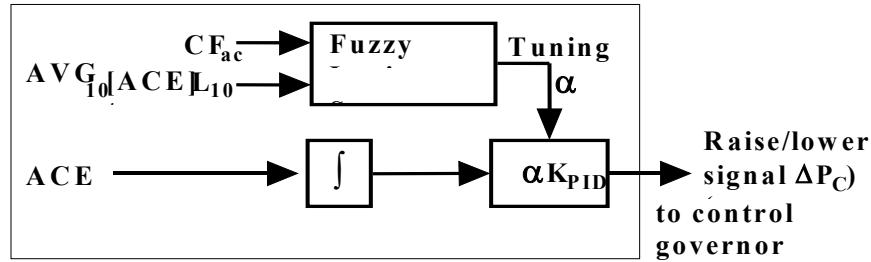


Figure 5.6: Operation of the fuzzy controller

The input membership functions for the fuzzy controller are shown in figure 5.7

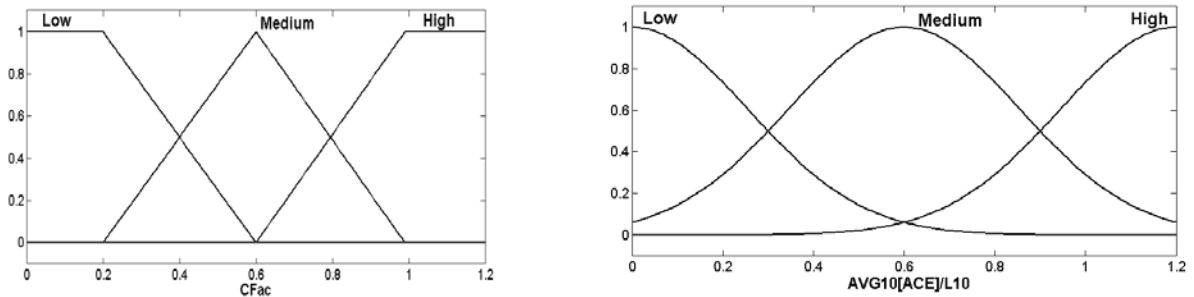


Figure 5.7: Input membership functions for the fuzzy controller

The input membership functions for the fuzzy controller are shown in figure 5.8

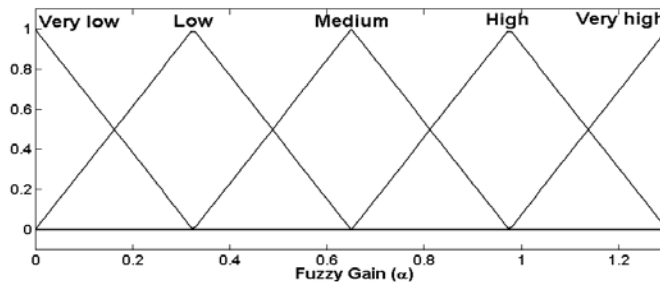


Figure 5.8: Output membership functions for the fuzzy controller

The Fuzzy Rules for inferring the output from the input is shown in table 5.2.

Table 5.2: Fuzzy Logic Rules for controller gain inference

CF_{ac}	Operator	$AVG_{10}[ACE]/L_{10}$	Fuzzy gain (α)
High			Very High
Medium	and	High	High
Medium	and	Medium	Medium
Medium	and	Low	Low
Low	and	High	Medium
Low	and	Medium	Low
Low	and	Low	Very Low

Finally, figure 5.9 shows the surface plot for the fuzzy controller

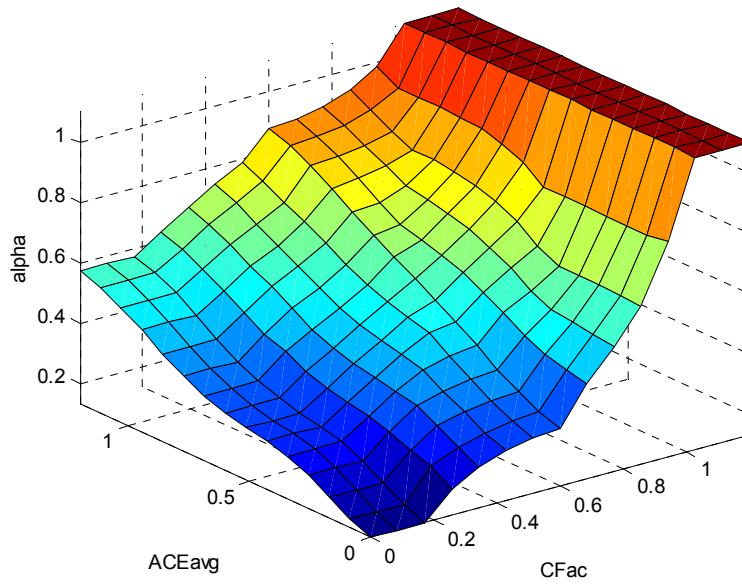


Figure 5.9: Surface plot for the fuzzy controller

CHAPTER 6

SIMULATION AND RESULTS

In this chapter, the details of the NERC performance standards and the Matlab/Simulink model for a three area power system with a fuzzy logic controller for ensuring compliance is presented followed by interaction between the market and the power system. Simulations and results are next presented that show bidder performance during two sets of auctions with the fuzzy strategy. The simulation results of the power system and plots of CPS1 and CPS2 that ensure compliance with NERC standards are later presented.

6.1 FIS based Regulation Service Compliant with NERC Standards

Regulation service provided by the generation companies (Gencos) is required to track the fluctuations in power demand of distribution companies (Discos) to keep good reliability, and to meet the North American Electric Reliability Council, NERC's performance standards namely CPS1 and CPS2 [6]. Fuzzy logic based controller is developed for ensuring compliance with the performance standards for a test system with three control areas and corresponding regulation loads. The signals, which determine compliance, are used as input to the fuzzy controller, which adjusts the output of a PID controller by a factor to administer compliance.

Each control area is required to monitor its control performance and report its compliance with CPS1 and CPS2 to NERC at the end of each month. The control structure implemented is a PID type controller whose control parameter is tuned by the fuzzy rules. Specifically, when there is high compliance with CPS1, the control parameter is gradually reduced to diminish high-frequency movement in the speed

governor's equipments. Once there is low compliance the control parameter is raised up accordingly. This ensures compliance with standards as well as reduced wear and tear of equipment.

6.1.1 CPS1 and CPS2

CPS1 is defined as follows: Over a sliding 12-month period, the average of the “clock-minute averages” of a control area’s ACE divided by “ten times its area frequency bias” times the corresponding “clock-minute averages of the interconnection frequency error” shall be less than the square of a given constant, ε_1 , representing a targeted frequency bound.

This is expressed by:

$$CF_{12\ months} = AVG_{12\ months} \left[\left(\frac{ACE}{-10\ B} \right)_1 \times \Delta F_1 \right] \leq \varepsilon_1^2 \quad (6.1)$$

Where ε_1 : the targeted frequency bound

(.)₁: refer to 1-min. averages

ΔF : interconnection frequency error

B_i : frequency bias of the i^{th} control area in MW/0.1Hz

To calculate the frequency-related parameter, CPS1 a compliance factor (CF) and 1-minute average compliance factor are introduced namely:

$$CF = AVG_{12\text{-month}} [CF_1] \quad ; \quad CF_1 = \left[\left(\frac{ACE}{-10\ B} \right)_1 * \left(\frac{\Delta F}{\varepsilon_1^2} \right)_1 \right] \quad (6.2)$$

Then CPS1 is obtained from the following equation:

$$CPS1 = (2 - CF) * 100\% \quad (6.3)$$

To comply with NERC, CPS1 should be less than 100%.

CPS2 requires the 10-minute averages of a control area's ACE (Area Control Error) to be less than a constant (L_{10}) given in the equation:

$$AVG_{10 \text{ min}} (ACE) \leq L_{10} \quad (6.4)$$

Where

$$L_{10} = 1.65 * \varepsilon_{10} \sqrt{(-10 B_i)(-10 B_s)} \quad (6.5)$$

$$B_s = \sum_{i=1}^n B_i \quad (6.6)$$

B_s : the sum of the frequency bias settings of all control areas

ε_{10} : the targeted frequency bound for CPS2

The targeted frequency bounds considered here are $\varepsilon_1 = 6.20mHz$ and $\varepsilon_{10} = 0.64mHz$.

CPS2, which should be greater than or equal to 90%, is given by:

$$CPS 2 = \left[1 - \frac{Violations_{month}}{Total Periods_{month} - Unavailable Periods_{month}} \right] \times 100\% \quad (6.7)$$

Where $Violations_{month}$ are a count of the number of periods that the clock-10-minute averages of ACE are greater than L_{10} in one month.

6.2 Sample Three Area System and Market Interaction

We consider a sample three-area power system as shown in figure 6.1 with the second area consisting of bidding agents implementing the market structure for regulation service. The outcome of each hour auction would be participation factors for each of the generators to provide regulation service.

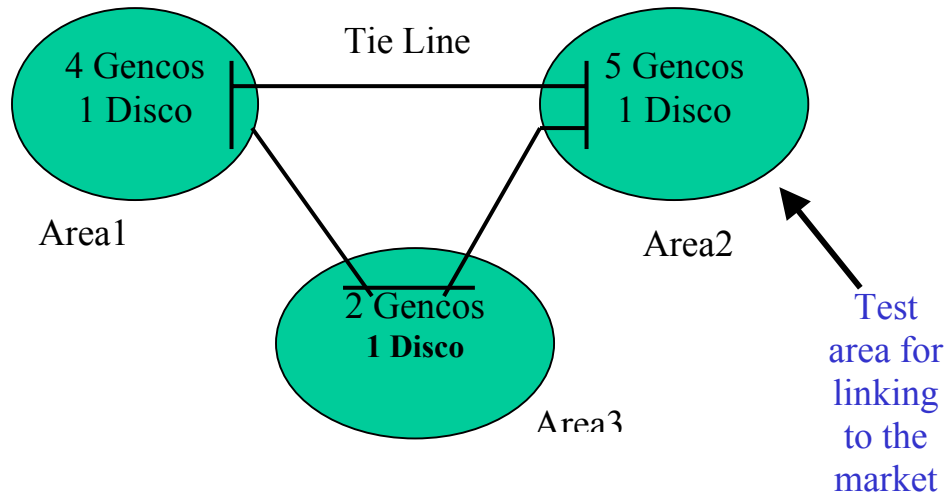


Figure 6.1: Sample Three-Area System

The synchronizing power coefficients of the lines are

$$T_{12} = 50 \text{ MW/rad}, T_{13} = 200 \text{ MW/rad}, \text{ and } T_{23} = 120 \text{ MW/rad}.$$

The random components of the load are the entities that need to be compensated by the regulation service and hence are simulated in the model using controlled random noise signals. A vector alpha consists of participating factors of the 5 Gencos in area-2 for regulation service during any particular hour of operation. Appendix A provides a snapshot of the Simulink models and fuzzy controller.

6.3 Auction Market Simulation

In this section, the agent based auction system is simulated for regulation service for the 24-hour period of the next day. An auction consisting of 5 Supplier agents is assumed to be present along with an ISO Controller agent. An array consisting of 24 elements is assumed to provide the forecasted regulation requirement for each hour in the 24-hour operation period for the next day. The sum of the capacity bid and energy bid is considered for building the energy stack by the ISO. Table 6.1 shows the parameters of the five bidders, their Reservation Capacity Bids (R_i) and Energy Bids (E_i). With these parameters for the individual supplier agents, the Madkit platform is used to simulate the auction market.

The reward for each successful supplier is given by:

$$reward = (MCP - Bidprice) * (Allocation\ for\ bidder) \quad (6.8)$$

Table 6.1: Parameters for the units in the simulation

Unit	Ramp-Rate MW/min	R_i \$/MW	E_i \$/MWh
1	10	8.552	10.01
2	9	7.895	8.198
3	10	8.552	10.01
4	6	8.795	9.147
5	8	8.565	9.513

The bar chart in figure 6.2 shows the forecasted demand values for the 24 hours considered for regulation auction.

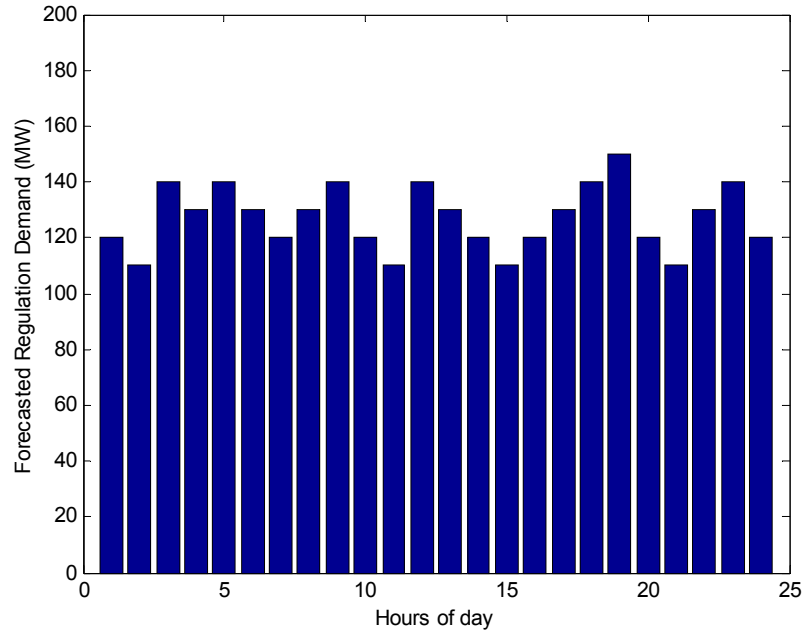


Figure 6.2: Plot of forecasted regulation demand

For the purpose of simulation, at the end of each hour auction, the ISO agent calls a *BidPrice* agent, which invokes the bidding strategy routines in Matlab for each of the agents. The changed bid prices from Matlab are again fetched by the BidPrice agent and sent to the Supplier agents, who participate in the next hour auction. This cycle continues until auctions for the 24 hours. In the first example, the second supplier agent is chosen to implement the fuzzy bidding strategy while the other agents implement the random number bidding strategy.

The bar chart in figure 6.3 and 6.4 show the number of successful bidders during the first and second sample runs of 24-hour auctions respectively.

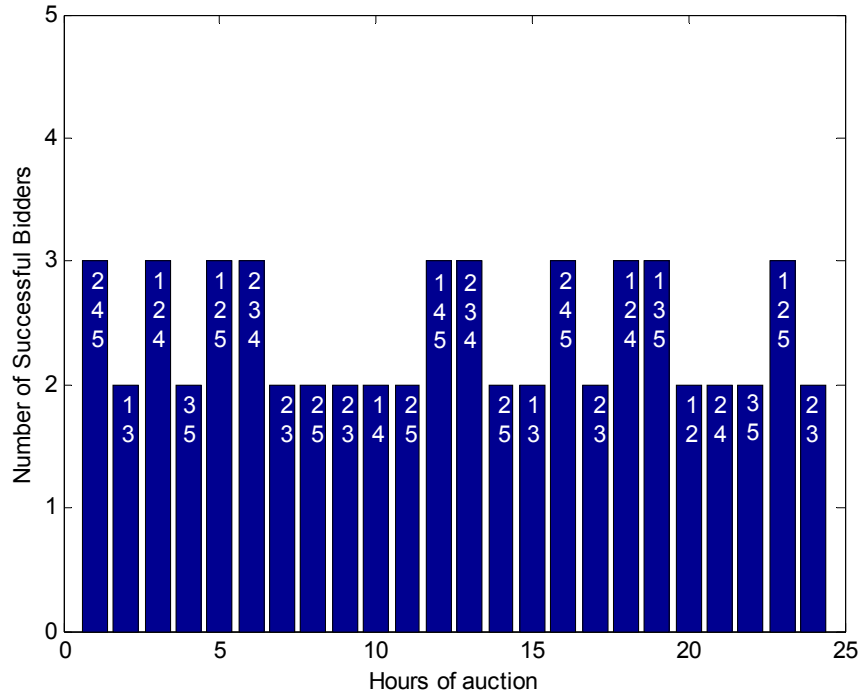


Figure 6.3: Plot of number of successful bidders in First Sample Run

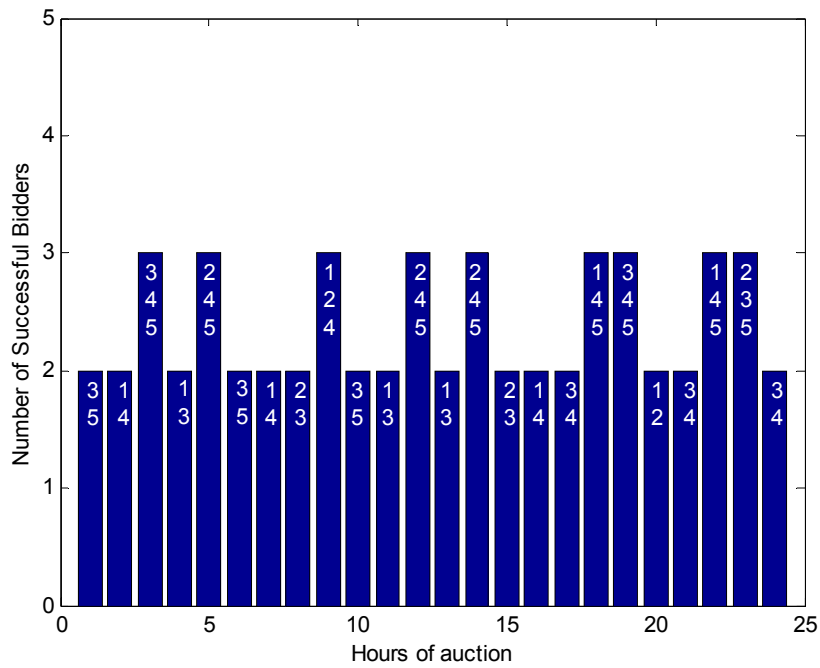


Figure 6.4: Plot of number of successful bidders in Second Sample Run

The rewards for each of the five agents in the first sample simulation are plotted in figure 6.5. The areas under the reward plots for each of the individual agents are \$3389.3, \$6654.2, \$2598.0, \$1108.6 and \$1571.6 respectively in the order supplier agent1 to supplier agent5. It is observed that the second supplier agent with the fuzzy bidding strategy has been able to make maximum profit from the auction by adjusting its bid price value being sensitive to the extent of change in the reward value. This ensured that the bidder was not only made good profit but was also in the stack of successful bidders for the maximum number of times.

For the second sample simulation, the bidding strategy of Supplier 3 was changed to be FIS strategy and that of Supplier 2 changed to the other strategy to evaluate the effectiveness of the FIS strategy. A second 24-hour auction was conducted following the first one with the changed bidding strategy options. The rewards for each of the five agents in the second sample simulation are plotted in figure 6.6. The areas under the reward plots for each of the individual agents are \$1595.8, \$1752.8, \$3909.2, \$1670.9 and \$2327.9 respectively in the order supplier agent1 to supplier agent5.

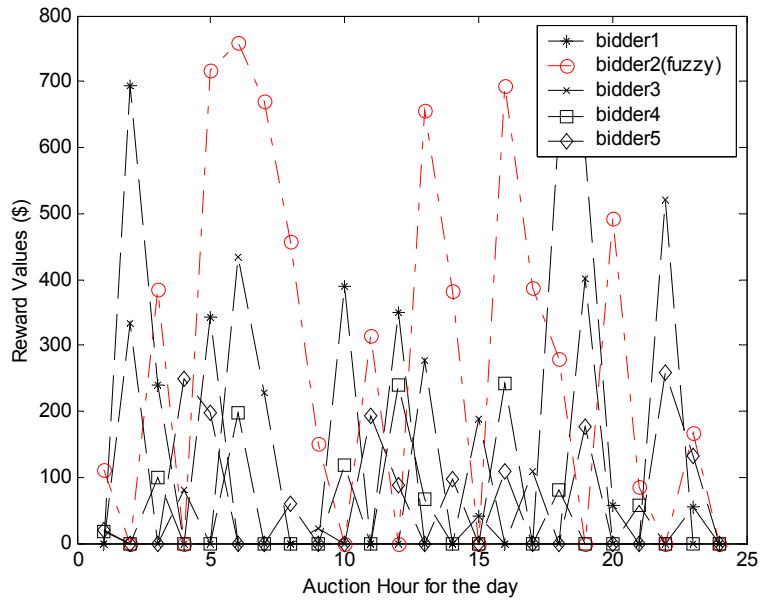


Figure 6.5: Plot of Rewards for Bidders in First Sample Run

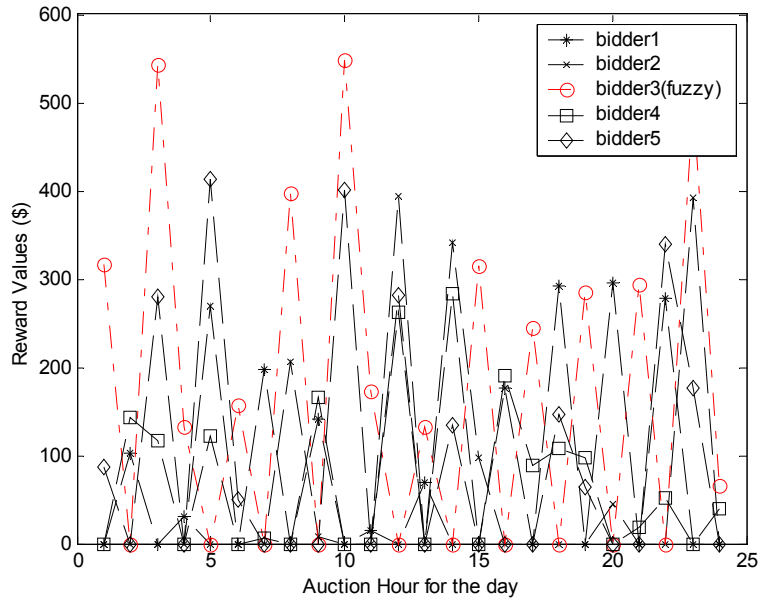


Figure 6.6: Plot of Rewards for Bidders in Second Sample Run

This shows that the fuzzy bidding strategy is more effective than the other strategy. It is assumed that the regulation bidders would bid their true energy generation

costs, and that their capacity reservation bids R_i would reflect the corresponding opportunity costs.

Figure 6.7 shows the variation of the price values for Bidder-2 with application of the fuzzy strategy in the first sample run, and without the fuzzy strategy in second sample run.

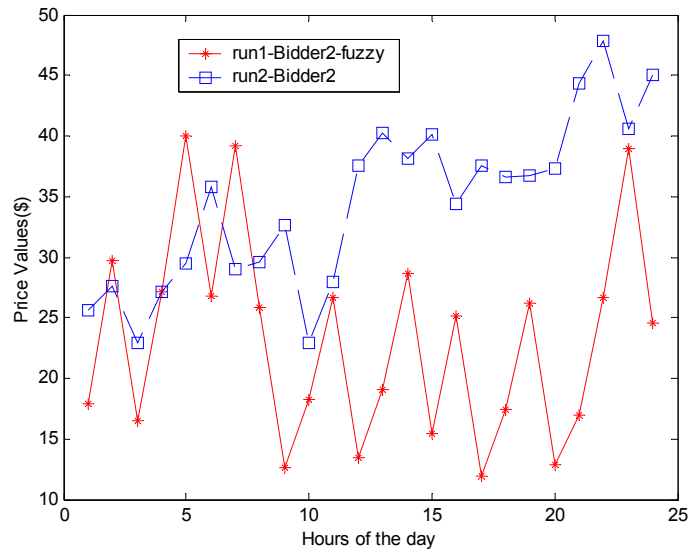


Figure 6.7: Plot of price changes for bidder-2 in sample runs

Figure 6.8 shows similar variation for Bidder-3 without application of fuzzy strategy in first sample run, and with application of fuzzy strategy in second sample run.

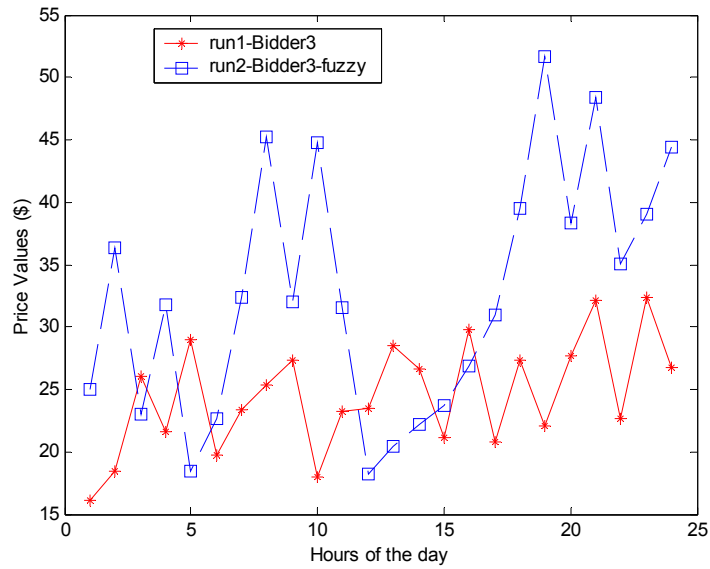


Figure 6.8: Plot of price changes for bidder-2 in sample runs

6.4 Power System Simulation and NERC Compliance

The test system adopted is a three-area system as shown in figure 6.1 earlier with the given Gencos and Discos. Control area-2 with five Gencos is considered for the regulation auction market. They are equivalent with the five Supplier Agents in the auction market with corresponding parameters. The participation of the generators in area-2 is shown in figure 6.9. The Simulink model for the power system and the results of the simulation are shown in Appendix A. It is observed that the participating factors of the generators enabled them to contribute correspondingly to the regulation service and the system is compliant with NERC's performance standards as shown by the plots for CPS1 and CPS2 in Appendix A.

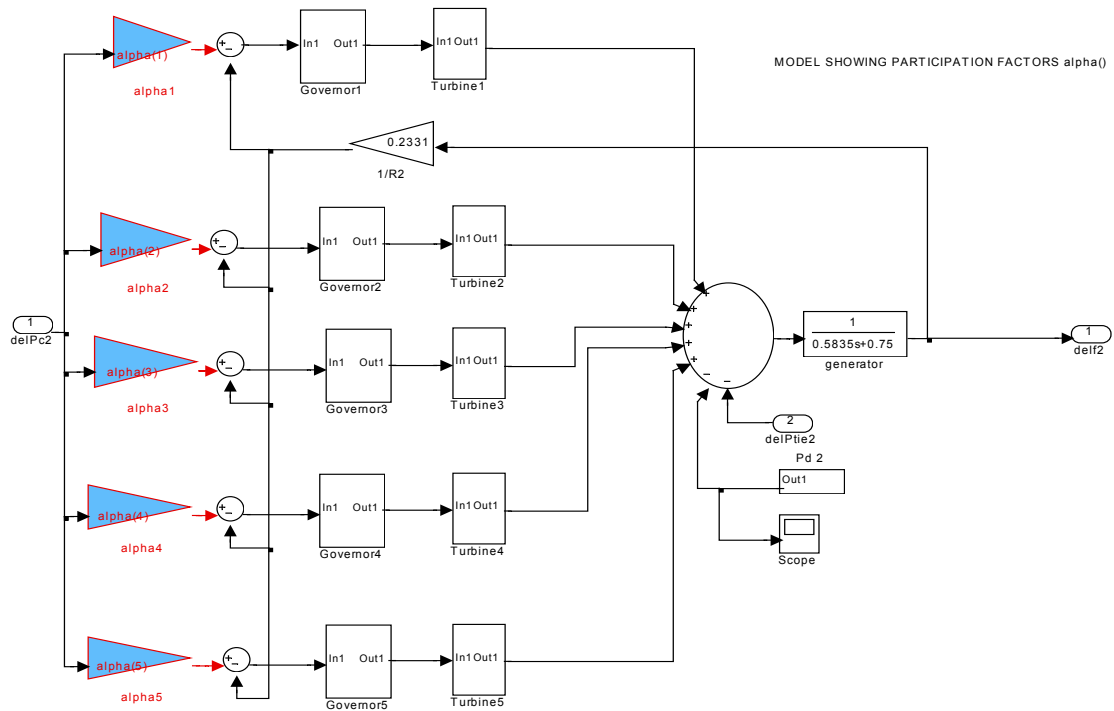


Figure 6.9: Model showing area-2 with the parameters from the market

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

An implementation has been done for the regulation service auction market using software agent technology. The Java-based multi-agent platform Madkit has been successfully used to implement competitive procurement of regulation power by the ISO from the suppliers. A test auction set up having five bidders has been considered for study of the auction mechanism as well as a couple of bidding strategies employed by the agents to increase their profit from the market. Simulation results indicate that the fuzzy bidding strategy provides more effective market catch for the bidder than the random number strategy.

The participation factors of the generators obtained from this market have been used for simulating a sample three-area power system compliant with NERC control performance standards using a fuzzy logic controller that modifies the gain of a PID controller. Further study in terms of auction mechanism and bidding strategies would provide better insight into the techniques and market behavior of such automated multi agent-based systems.

7.2 Recommendations for future work

The initial enhancement over this work would be to implement more bidding strategies. The fuzzy inference system could be expanded with another variable reflecting the time of the day during which the auction takes place viz., peak time or off-peak time. This would enable more precise decisions from the fuzzy system since it would be wise to increase the bid price when there would be good demand for the commodity. The next

step would be to implement optimization techniques to improve the functioning of the ISO as well as bidding agents. The ISO could work on minimizing a cost function and/or maximizing a social welfare function while the supplier agents work on maximizing their profit functions.

Also, double auctions could be implemented with Consumer agents trying to buy regulation power from the ISO as well as trying to enter into bilateral contracts with Supplier agents thereby reducing the available capacity. Also, similar auctions can be implemented for the other ancillary services like spinning reserves and replacement reserves.

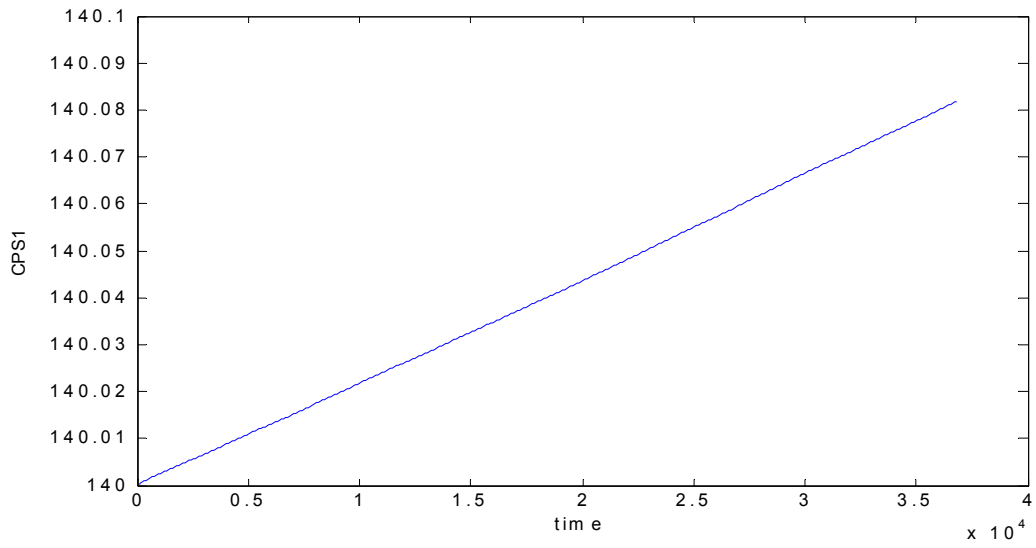


Figure A.3: NERC CPS1 compliance of area-2

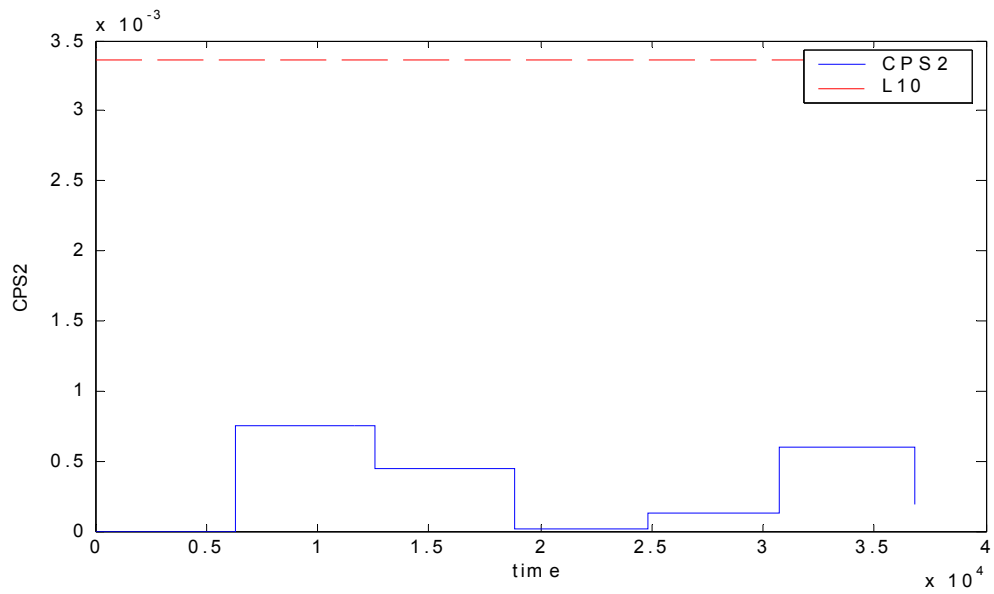


Figure A.4: NERC CPS2 compliance of area-2

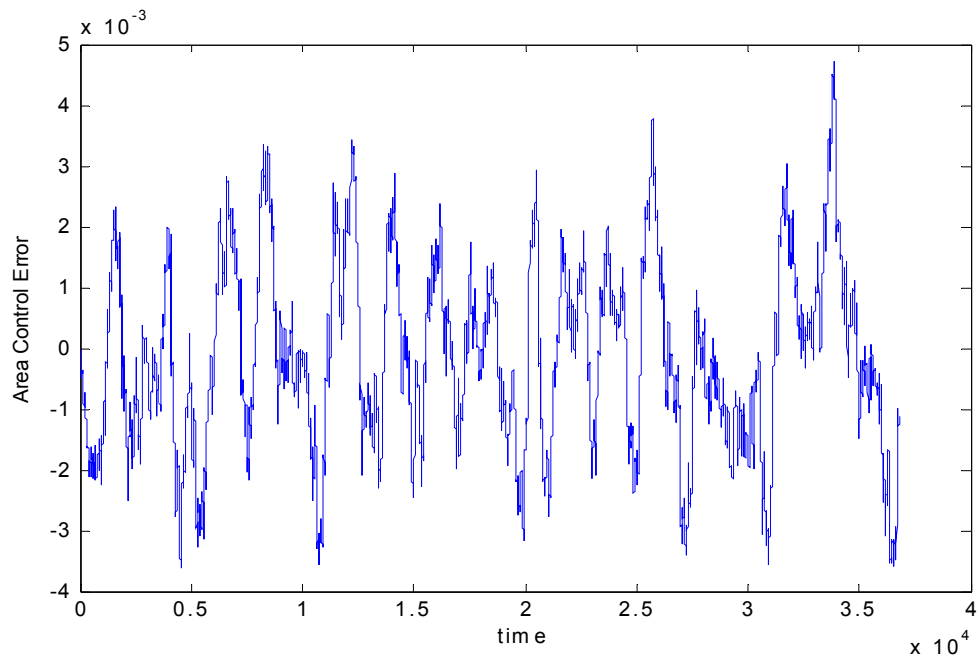


Figure A.5: Area Control Error (ACE) signal of area-2

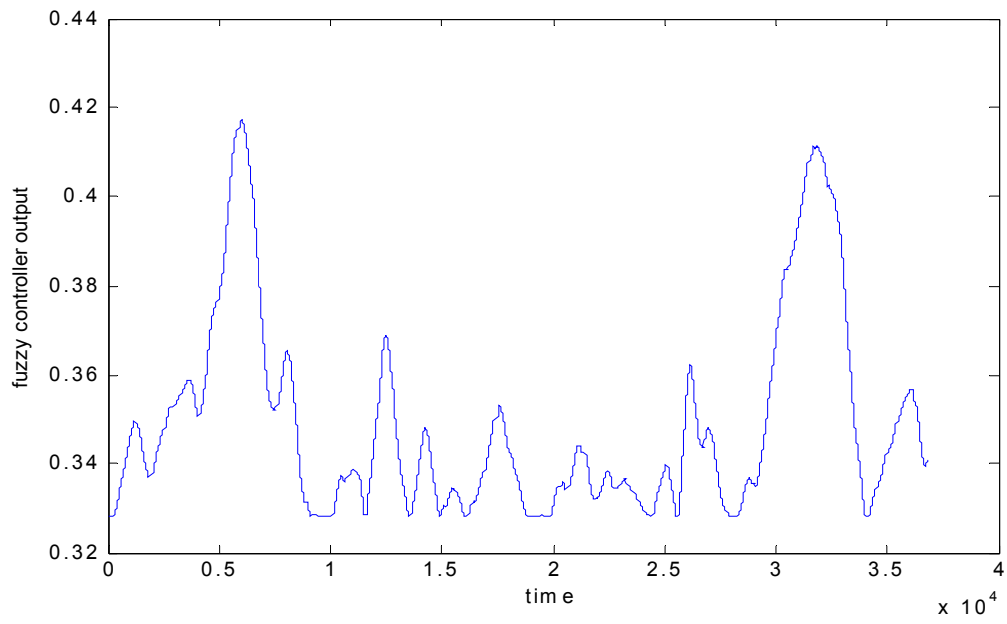


Figure A.6: Fuzzy Controller Output signal for area-2

APPENDIX A SIMULINK MODELS AND RESULTS OF SIMULATION

% INITIALIZATION FOR THE PARAMETERS IN THE THREE AREA SYSTEM IN MATLAB

%MACHINE PARAMETERS

beta1 = 1.68;

beta2 = 1.9155;

beta3 = 1.6934;

betaSUM = 5.2885;

H1 = 6;

H2 = 3.5;

H3 = 10;

% NERC PARAMETERS

e10 = 0.64/1000;

e1 = 6.2/1000;

L10(1)= sqrt(beta1*betaSUM)*1.65*e10;

L10(2)= sqrt(beta2*betaSUM)*1.65*e10;

L10(3) = sqrt(beta3*betaSUM)*1.65*e10;

tg1 = 35 ; % time constant for supplier i

tg2 = 30 ;

tg3 = 25 ;

%FUZZY SYSTEM INITIALIZATION

proj1matrix = readfis('proj1.fis');

fprintf('\n Initialization Completed!');

return;

APPENDIX B

GLOSSARY

1. **Energy Auction System** is defined as a central auction mechanism, which provides trading opportunities to the market participants and maintains reliability of operation by coordinating generation, transmission, and distribution functions.
2. **Supportive Services** are the services used for supporting a reliable delivery of electric energy in power system operation.
3. **Regulation Reserve** is the unused capacity of generation that can be made available for use within a period of 10 minutes.
4. **Load Following** is defined as the amount of electric energy provided to maintain the contracted tie line flow and frequency in power system operation during the trading period. The load following energy is required to respond in 5-10 minutes of time frame.
5. **Forward Market** is the trading place where short-term transactions (hourly or daily transactions) take place.
6. **Futures Market** is the trading place where long-term transactions (monthly or yearly transactions) take place.
7. **Planning Market** is the market that would underwrite the usage of assets (such as transmission lines) to very long-term commitments (15-20 years or more).
8. **Swap Market** is the clearinghouse that allows contracts to be terminated with an exchange of physical or financial substitution.
9. **Market Clearing Price** is a price at which all valuations of the commodity are in equilibrium. In the regulation auction, it is the highest price in the energy stack that clears the forecasted demand value.
10. **Control Area** is an electric system or systems, bound by interconnection metering and telemetry, capable of controlling generation to maintain its

interchange schedule with other control areas and contributing to frequency regulation of the interconnection.

11. **Day-ahead Market** is the market involving trading of multi-hour energy blocks for delivery during the following day. There can be off-peak market and on-peak market for trading to supply during the off-peak and on-peak hours respectively during the next day.
12. **Operating Reserve** is that generating capability over and above the system demand required to provide for regulation, load forecasting error, forced and unscheduled outages and local area protection. It consists of spinning and non-spinning reserve. It relates to the short-term ability to meet a given load.
13. **Spot Market** is a market where good are traded for immediate delivery.
14. **Strike Price** is the price at which the underlying options contract is bought and sold in the event the option is exercised.
15. **Installed Capacity Reserve** relates to the long-term ability of the system to meet the expected demand requirements. It is the capacity that must be planned and constructed in advance to provide for uncertainties in the forecast of demand, and unplanned or unscheduled generation outages.
16. **Area Control Error (ACE)** is a measure of instantaneous imbalance between actual and scheduled generation and demand taking into account energy exchanges with other control areas.
17. **Outage** refers to the removal of a generating unit from service in order to be replaced or serviced as a result of a failure. Such events can compromise the ability of the system to supply the load and affect system reliability. A forced outage results from emergency conditions requiring the unit to be taken out of service immediately, while a scheduled outage results when a unit is deliberately taken out of service for service or preventive maintenance.

APPENDIX C**SOURCE CODE FOR AGENT PROGRAMS****C.1 Program ISO.JAVA**

THIS PROGRAM IS USED TO SIMULATE THE FUNCTIONING OF THE ISO CONTROLLER IN THE COMPETITIVE AUCTION FOR REGULATION SERVICE. IT OBTAINS THE BID PRICES FROM THE PARTICIPATING BIDDERS, BUILDS THE ENERGY STACK AND DETERMINES THE MARKET CLEARING PRICE AND SUBSEQUENT ALLOCATIONS TO SUCCESSFUL BIDDERS. IT ALSO INTERFACES WITH THE MATLAB ENVIRONMENT.

```

*/
//package demo.agents.power;

import madkit.kernel.*;
import madkit.lib.graphics.*;
import madkit.lib.messages.ACLMessage;

import java.awt.*;
import java.io.*;
import java.awt.event.*;
import java.util.*;

import javax.swing.*;

/* CONTAINS THE BID OF A SUPPLIER. THIS CLASS IS USED BY THE ISO
   TO STORE THE PARAMETERS OF THE BIDDERS AND TO OBTAIN THE MCP
   FROM THE BIDS OF THE SUPPLIERS */
class BidAnswer {
    AgentAddress bidder;
    double value;
    double ramp;
    double alloc;
    double reward;
    BidAnswer(AgentAddress b, double v)
    {
        bidder = b;
        value = v;
        ramp = 0.0;
        alloc=0.0;
        reward = 0.0;
    }
    double getValue(){return value;}
    void setRamp(double r){ramp = r;}

```

```

        void setAlloc(double s){alloc = s;}
        void setReward(double r){reward = r;}
        double getRamp(){return ramp;}
        double getAlloc(){return alloc;}
        double getReward(){return reward;}
        AgentAddress getBidder(){return bidder;}
    }
    /* THE ISO CLASS PERFORMS THE ROLE OF THE ISO CONTROLLER IN THE
    REGULATION SERVICE AUCTION. IT RECEIVES THE BID PRICES FROM
    THE BIDDERS FOR EACH HOUR, BUILDS THE ENERGY STACK AND FINDS
    THE MARKET CLEARING PRICE. IT ALSO DISPATCHES THE ALLOCATIONS
    TO THE UNITS AND INITIATES THE INTERFACE WITH MATLAB.
    */
    public class ISO extends Agent implements ActionListener
    {
        int flag = 0;
        String product = "utility";
        int auction_count;
        BidAnswer[] answers;
        BidAnswer[] sorted_answers;
        static AgentAddress[] bidders,price_agent;
        int contract = 0;
        JButton b;
        int cpt=0;
        int cptramp = 0;
        AgentAddress client;
        FileWriter fw,reward_file;
        double reward24[][];
        double hour_demand;
        double demand_for_day[] = {120, 110, 140, 130, 140, 130, 120, 130, 140, 120,
                                    110, 140, 130, 120, 110, 120, 130, 140, 150, 120,
                                    110, 130, 140, 120};

        static JMatLink engine;
        double bidprices[][];

        boolean pause = false;

        public void enterPause()
        {
            pause = true;
            b.setBackground(Color.red);
            b.repaint();

            while (pause)
                pause(200);
        }
    }

```

```

public void initGUI()
{
    JPanel p = new JPanel(new BorderLayout());
    OPanel o = new OPanel();
    p.add("Center",o);
    b = new JButton("Next step");
    b.addActionListener(this);
    p.add("South",b);
    setGUIObject(p);
    setOutputWriter(o.getOut());
}

public void actionPerformed(ActionEvent e)
{
    pause=false;
    b.setBackground(Color.gray);
}

public void activate()
{
    println("ISO ready");
    enterPause();

    joinGroup("power-suppliers");
    requestRole("power-suppliers", "ISO");
}

public void handleMessage(ACLMessage m)
{
    if (m.getAct().equals("ramp"))
    {
        receiveRamp(m);
    }
    else if (m.getAct().equals("bid"))
    {
        receiveBid(m);
    }
};
}

public void live()
{
    println("Opening engine");
}

```

```

engine = new JMatLink();
engine.engOpen();
engine.engEvalString("fismat=readfis('fuzzybid')");
println("Loaded the Fuzzy Inference System");
pause(3000);

reward24 = new double[5][24];
auction_count = 0; AgentAddress aa2 = null;
while (auction_count < 24)
{
    if ( ((auction_count+1) == 1 | ((auction_count+1) == 21) )
        {
            println("Starting auction for the " + (auction_count+1) + "st hour");
        }
    else if ( ((auction_count+1) == 2) | ((auction_count+1) == 22) )
        {
            println("Starting auction for the " + (auction_count+1) + "nd hour");
        }
    else if ( ((auction_count+1) == 3) | ((auction_count+1) == 23) )
        {
            println("Starting auction for the " + (auction_count+1) + "rd hour");
        }
    else if ( ((auction_count+1) >= 4) | ((auction_count+1) <= 20)
              | ((auction_count+1) == 24) )
        {
            println("Starting auction for the " + (auction_count+1) + "th hour");
        };
    pause(10000);
    //enterPause();

    bidders=getAgentsWithRole("power-suppliers",
                              product+"-supplier");

    if (bidders.length==0)
    {
        println("Not found providers of " + product);
        return;
    }
    println ("Found "+ bidders.length+" suppliers of power from "+product);
    cpt = bidders.length;
    cptramp = cpt;
    answers = new BidAnswer[cpt];
    sorted_answers = new BidAnswer[cpt];

// SEND DEMAND VALUE FOR THAT HOUR TO SUPPLIERS

```

APPENDIX C SOURCE CODE FOR AGENT PROGRAMS

```

hour_demand = demand_for_day[auction_count];
println("Demand Value for the hour is " + hour_demand);
println("Sending Demand Value to Suppliers...");
for(int t1=0; t1<bidders.length;t1++)
{
    aa2 = bidders[t1];
    sendMessage(aa2, new
    ACLMessage("demand",String.valueOf(hour_demand)));
}
pause(2000);

// SEND BID REQUESTS TO SUPPLIERS
println ("Sending bid request to all suppliers:");

ACLMessage req=new ACLMessage("request-for-bid",product);
broadcastMessage("power-suppliers", product+"-supplier",req);

println ("Waiting for offers..");
pause(8000);
while (cpt > 0)
{
    Message m = waitNextMessage();

    if (m instanceof ACLMessage)
        handleMessage((ACLMessage)m);
    else
        System.err.println("ERROR: invalid message type: "+ m);
}

requestRamp();

dispatch();
auction_count++;

// ACTIVATE THE BIDPRICE AGENT TO SEND CHANGED
// BID VALUES TO THE SUPPLIERS

if (auction_count < 24)
{
    println("Sending message to BidPrice agent...");
    ACLMessage req1=new ACLMessage("send-prices",product);
    broadcastMessage("power-suppliers", product+"-priceagent",req1);
};

if ((auction_count == 23) & (flag == 0))

```

```

        {
            engine.engPutArray("reward24",reward24);
            println("Reward24 written into Matlab");
            println("Thank you for using the agent based auction system.\n");
            println("\nRunning the system again for another day.\n");
            auction_count = 0;
            flag = 1;
            enterPause();
        }

    }// FOR THE WHILE AUCTION_COUNT<24 LOOP
    // CODE FOR WRITING THE REWARD24 DATA TO MATLAB FOR
    // PLOTTING FOR RESULTS
    engine.engPutArray("reward24",reward24);
    println("Reward24 written into Matlab");

    println("Thank you for using the agent based auction system.\n");
    println("\nRunning the system again for another day.\n");
} // END BRACES FOR THE LIVE() METHOD

protected void receiveBid(ACLMessage m)
{
    println("Received an offer " + m.getContent() + " from " + m.getSender());
    cpt--;
    answers[cpt]=new BidAnswer(m.getSender(),((Double) new
    Double(m.getContent().toString())).doubleValue());
}

protected void requestRamp()
{
    println ("Sending ramp-rate request to all suppliers:");

    ACLMessage req1=new ACLMessage("request-for-ramp",product);
    broadcastMessage("power-suppliers", product+"-supplier",req1);

    println ("Waiting for Ramp-rate values..");
    pause(8000);
    while (cptramp > 0)
    {
        Message m1 = waitNextMessage();
        if (m1 instanceof ACLMessage)
            handleMessage((ACLMessage)m1);
        else
            System.err.println("ERROR1: invalid message type: "+ m1);
    }
}
}

```

```

protected void receiveRamp(ACLMessage m)
{
    println("Received Ramp value from " + m.getSender());
    cptramp--;
    for (int q = 0; q < answers.length; q++)
    {
        if (answers[q].getBidder() == m.getSender())
        {
            double d1 = ((Double) new
                Double(m.getContent().toString())).doubleValue();
            answers[q].setRamp(d1);
        }
    };
}

void dispatch()
{
    AgentAddress aa = null, aa1 = null;
    sorted_answers = answers;
    BidAnswer temp;
    println("Selecting offers from " + sorted_answers.length + " proposals");
    pause(4000);
    // SORT THE BID ANSWERS OBJECTS ACCORDING TO BID PRIX VALUE
    for (int i=0; i<sorted_answers.length;i++)
    {
        for (int j=(i+1); j<sorted_answers.length;j++)
        {
            double v1 = sorted_answers[i].getValue();
            double v2 = sorted_answers[j].getValue();
            if (v1 > v2)
            {
                temp = sorted_answers[j];
                sorted_answers[j] = sorted_answers[i];
                sorted_answers[i] = temp;
            }
        }
    }
    // PRINT OUT THE SORTED BID VALUES
    for(int t=0; t<sorted_answers.length;t++)
    {
        double f = sorted_answers[t].getValue();
        AgentAddress add = sorted_answers[t].getBidder();
        double f1 = sorted_answers[t].getRamp();
        println("\n" + String.valueOf(f));
    };
}

```

```

double Pdsum = 0.0;
int k1 = 0;

// NOW WE HAVE THE SORTED BIDANSWERS NEXT IS ALLOCATION
println("Building the Load Stack");
for (int k=0; k < sorted_answers.length; k++)
{
    double v3 = sorted_answers[k].getRamp();
    // calculate supply in 5 min.s
    double s = v3 * 5;
    Pdsum = Pdsum + s;

    if (Pdsum < hour_demand)
    {
        // set the amount of dispatch given for that unit
        sorted_answers[k].setAlloc(s);
        // populate the allocation in the unsorted list of bidders
        aa = sorted_answers[k].getBidder();
        for(int q1=0; q1<answers.length;q1++)
        {
            aa1 = answers[q1].getBidder();
            if (aa == aa1)
                answers[q1].setAlloc(s);
        }
    }
    else if (Pdsum == hour_demand)
    {
        sorted_answers[k].setAlloc(s);
        aa = sorted_answers[k].getBidder();
        // populate the allocation in the unsorted list od bidders
        for(int q1=0; q1<answers.length;q1++)
        {
            aa1 = answers[q1].getBidder();
            if (aa == aa1)
                answers[q1].setAlloc(s);
        }
        k1 = k;
        break;
    }
    else if (Pdsum > hour_demand)
    {
        double d = Pdsum - hour_demand;
        sorted_answers[k].setAlloc((s-d));
        aa = sorted_answers[k].getBidder();
        // POPULATE THE ALLOCATION IN THE UNSORTED LIST OF BIDDERS
        for(int q1=0; q1<answers.length;q1++)

```



```

        {
            aa1 = answers[q1].getBidder();
            if (aa == aa1)
                answers[q1].setAlloc((s-d));
        }
        k1 = k;
        break;
    }
}

// DETERMINE THE MARKET CLEARING PRICE
// AND SEND IT TO THE SUPPLIERS
double MCP = sorted_answers[k1].getValue();
println("MCP value is " + MCP);
for(int t1=0; t1<sorted_answers.length;t1++)
{
    aa = sorted_answers[t1].getBidder();
    sendMessage(aa, new ACLMessage("mcp",String.valueOf(MCP)));
}
pause(3000);

// SEND THE ALLOCATION TO EACH BIDDER
for(int t2=0; t2<answers.length;t2++)
{
    aa = answers[t2].getBidder();
    double d2 = answers[t2].getAlloc();
    double v2 = answers[t2].getValue();
    println("Allocation sent for bidder " + aa + "Amount= " + d2);
    sendMessage(aa, new ACLMessage("alloc",String.valueOf(d2)));
}

// FIND THE REWARD FOR THE ALLOCATED BIDDERS FROM MCP,
// ALLOCATION, AND PRICE VALUES OF THE BIDDERS
// REWARD = (MCP // - PRICE) * ALLOC
double r = (MCP - v2) * d2;
if (r < 0.0) r = 0.0;
answers[t2].setReward(r);
};
bidprices = new double[3][5];
// WRITE THE DATA AS AN ARRAY IN MATLAB
// NOW WE OUTPUT THE BID PRICES OF THE BIDDERS
// FIRST ROW OF BIDPRICES IS PRICE VALUES
// SECOND ROW IS REWARD VALUES
// third row is the choice of the bidder for strategy
for (int t3=0; t3<bidders.length; t3++)
{
    for(int t2=0; t2<answers.length;t2++)

```

```

        {
        aa = answers[t2].getBidder();
        if (bidders[t3] == aa)
        {
            double d2 = answers[t2].getReward();
            double d3 = answers[t2].getValue();
            reward24[t3][auction_count] = d2;
            bidprices[0][t3] = d3;
        };
    };
};

// NOW OUTPUT THE REWARD VALUES OF THE BIDDERS
for (int t3=0; t3<bidders.length; t3++)
{
    for(int t2=0; t2<answers.length;t2++)
    {
        aa = answers[t2].getBidder();
        if (bidders[t3] == aa)
        {
            double d2 = answers[t2].getReward();
            bidprices[1][t3] = d2;
        };
    };
};

// Now Output The Selected Bidding Strategy Of The Bidders
for (int t4=0; t4<bidders.length; t4++)
{
    bidprices[2][t4] = 0.0;
}
// implement the FIS strategy for the second bidder
//bidprices[2][1] = 1.0;
if (flag == 0)
{
    bidprices[2][1] = 1.0;
}
else
{
    bidprices[2][2] = 1.0;
}

// PUT BIDPRICES ARRAY INTO MATLAB WORKSPACE
engine.engPutArray("bidprices",bidprices);
println("Matlab written with array");
} // FOR DISPATCH() METHOD
} // FOR ISO.JAVA
*****

```

C.2 Program SUPPLIER.JAVA

THIS PROGRAM IS USED FOR SIMULATING THE SUPPLIER AGENT IN THE REGULATION AUCTION. IT GENERATES THE CAPACITY RESERVATION BID AND THE ENERGY BID, ADDS THEM AND SUBMITS THE RESULT TO THE ISO AGENT AS THE BID PRICE FOR EVERY HOUR OF AUCTION. IT PROCURES THE NEW PRICE FROM THE BID PRICE AGENT BEFORE THE NEXT AUCTION EVERYTIME AFTER THE FIRST HOUR.

```

*/
//package demo.agents.power;
import java.awt.*;
import java.net.*;
import java.io.*;
import madkit.kernel.*;
import java.util.*;
import javax.swing.*;

import madkit.lib.messages.ACLMessage;

public class Supplier extends Agent
{
    double prix = 0.0;
    //alpha value
    double alpha = 0.0;
    // beta value
    double beta = 0.0;
    // gamma value
    double gamma = 0.0;
    // ramp-rate value for the supplier
    double ramp = 0.0;
    // unloaded capacity available with the supplier
    double unload = 0.0;
    double Ei = 0.0;
    double Ri = 0.0;
    double rand = 0.0;
    double rand1 = 0.0;
    double MCP = 0.0;
    double demand;
    String competence;
    JLabel message,price1;

    public Supplier()
    {
        // generate random number and populate
        // supplier parameters

```

```

rand = Math.random();
competence = "utility";
if (rand >= 0.0 & rand < 0.1)
{
    alpha = 78.0;
    beta = 7.97;
    gamma = 0.005820;
    ramp = 6;
    unload = 124.4;
};
if (rand >= 0.1 & rand < 0.2)
{
    alpha = 75.9;
    beta = 8.69;
    gamma = 0.001750;
    ramp = 8;
    unload = 165;
};
if (rand >= 0.2 & rand < 0.3)
{
    alpha = 76.9;
    beta = 7.92;
    gamma = 0.001562;
    ramp = 7;
    unload = 155.53;
};
if (rand >= 0.3 & rand < 0.4)
{
    alpha = 75.5;
    beta = 7.68;
    gamma = 0.002394;
    ramp = 9;
    unload = 108.19;
};
if (rand >= 0.4 & rand < 0.5)
{
    alpha = 77.8;
    beta = 7.85;
    gamma = 0.008940;
    ramp = 8;
    unload = 93.02;
};
if (rand >= 0.5 & rand < 0.6)
{
    alpha = 75.9;
    beta = 6.34;

```

APPENDIX C SOURCE CODE FOR AGENT PROGRAMS

```

        gamma = 0.002394;
        ramp = 9;
        unload = 135.0;
    };
    if (rand >= 0.6 & rand < 0.7)
    {
        alpha = 78.0;
        beta = 8.88;
        gamma = 0.05572;
        ramp = 7;
        unload = 98.0;
    };
    if (rand >= 0.7 & rand < 0.8)
    {
        alpha = 76.8;
        beta = 9.71;
        gamma = 0.002820;
        ramp = 8;
        unload = 180.0;
    };
    if (rand >= 0.8 & rand < 0.9)
    {
        alpha = 75.5;
        beta = 7.97;
        gamma = 0.005820;
        ramp = 10;
        unload = 175.53;
    };
    if (rand >= 0.9)
    {
        alpha = 77.8;
        beta = 8.69;
        gamma = 0.001750;
        ramp = 6;
        unload = 130.5;
    };

// MARGINAL COST PER ENERGY UNIT
// MCI = EI = B + C * PG
// THIS IS THE ENERGY BID EI
Ei = beta + ( 2 * gamma) * unload);

// POWER VALUE FOR OPPORTUNITY GENERATION FOR 5 MINS
double opp_Pg = ramp * 5;

// CAPACITY BID PRICE RI FOR THE BIDDING SUPPLIER

```

APPENDIX C SOURCE CODE FOR AGENT PROGRAMS

```

//RI = BETA + (2 * (GAMMA * OPP_PG);
Ri = beta + (2 * (gamma * opp_Pg));

// PUT THE VALUE TO THE INITIAL VALUE FOR PRODUCTION
// BY THE SUPPLIER
prix = Ri + Ei;

}

public void initGUI()
{
    JPanel p= new JPanel();
    p.setLayout(new BorderLayout(p,BoxLayout.Y_AXIS));

    // add the bid price to the GUI
    p.add(price1 = new JLabel("Price: "+ prix));
    p.add( new JLabel("Ramp: " + ramp));
    p.add( new JLabel("Ei: " + Ei));
    p.add( new JLabel("Ri: " + Ri));

    p.add(message = new JLabel("                "));
    message.setOpaque(true);
    p.setSize(170,100);
    message.setBackground(Color.lightGray);

    setGUIObject(p);
}

public void activate()
{
    joinGroup("power-suppliers");
    requestRole("power-suppliers",competence+"-supplier");
    display("Power: "+competence);
}

public void display(String s)
{
    if (hasGUI())
        message.setText(s);
    else
        println(s);
}

public void display_price(String s)
{
    if (hasGUI())

```

```

    price1.setText(s);
else
    println(s);
}

protected void handleMessage(ACLMessage m)
{
    Color bkg=message.getBackground();
    // RESPONSE FOR REQUEST FOR BID MESSAGE FROM ISO
    if (m.getAct().equals("request-for-bid"))
    {
        debug("s"+m.getSender());
        debug("r"+m.getReceiver());
        display("Sending bid");
        pause(4000);
        message.setBackground(Color.yellow);

        // send the bid value to the ISO
        sendMessage(m.getSender(),
            new ACLMessage("bid",String.valueOf(prix)));
        display("Bid value sent");
        pause(2000);

        display_price("Price: "+ prix);
        message.setBackground(Color.lightGray);
        display("Power: "+competence);
    }
    // RESPONSE FOR REQUEST FOR RAMP MESSAGE FROM ISO
    else if (m.getAct().equals("request-for-ramp"))
    {
        debug("s"+m.getSender());
        debug("r"+m.getReceiver());
        display("Sending ramp");
        pause(4000);
        message.setBackground(Color.yellow);

        // send the ramp value to the ISO
        sendMessage(m.getSender(),
            new ACLMessage("ramp",String.valueOf(ramp)));
        display("Ramp value sent");
        pause(2000);
        message.setBackground(Color.lightGray);
        display("Power: "+competence);
    }
}
// RESPONSE FOR MARKET CLEARING PRICE MESSAGE FROM ISO
else if (m.getAct().equals("mcp"))

```

```

    {
        MCP = ((Double) new Double(m.getContent().toString())).doubleValue();
        display("MCP value received: " + MCP);
    }
// RESPONSE FOR MESSAGE WITH NEW PRICE
// OBTAINED USING THE BIDDING STRATEGY
// USING MATLAB FROM BIDPRICE AGENT
else if (m.getAct().equals("newprice"))
{
    // Check for condition for no loss to supplier in
    // the auction
    double temp;
    temp = ((Double) new Double(m.getContent().toString())).doubleValue();
    if (temp < Ri)
    {
        prix = Ri;
    }
    else
    {
        prix = temp;
    }
    display_price("Price: "+ prix);
display("New price received");
    message.setBackground(Color.yellow);
    pause(2000);
    message.setBackground(Color.lightGray);
display("Power: "+competence);
}
// RESPONSE FOR MESSAGE WITH FORECASTED
// DEMAND VALUE FROM ISO
else if (m.getAct().equals("demand"))
{
    demand = ((Double) new
Double(m.getContent().toString())).doubleValue();
    display("Demand value received: " + demand);
}
// response for message with allocation from ISO
else if (m.getAct().equals("alloc"))
{
    double a = ((Double) new
Double(m.getContent().toString())).doubleValue();
    // Calculate the participation
    double p = a/demand;

    // calculate and display the reward
    pause(1000);
}

```



```

        double reward = (MCP - prix) * a;
        display("Reward " + reward);
    }
} // end of handleMessage() method

public void live()
{
    while (true) {
        Message m = waitNextMessage();
        if (m instanceof ACLMessage)
            handleMessage((ACLMessage)m);
    }
}
}
}

```

C.3 Program BIDPRICE.JAVA

THIS PROGRAM IS USED TO PROCURE THE NEW BID PRICES FROM MATLAB THE PARTICIPATING BIDDERS ACCORDING TO THE STRATEGY EMPLOYED BY EACH OF THE PARTICIPATING BIDDERS, AND PASSES THE VALUES AS MESSAGES TO THE RESPECTIVE SUPPLIER AGENTS.

*/

```

//package demo.agents.power;
import java.awt.*;
import java.net.*;
import java.io.*;
import madkit.kernel.*;
import java.util.*;
import javax.swing.*;

import madkit.lib.messages.ACLMessage;

public class BidPrice extends Agent
{
    double rand = 0.0;
    double rand1 = 0.0;
    double MCP = 0.0;
    double demand;
    double new_prices[][];
    String competence;
    JLabel message,price1;
    FileReader fr;
}

```

```

BufferedReader br;

public BidPrice()
{
    competence = "utility";
}

public void initGUI()
{
    JPanel p1= new JPanel();
    p1.setLayout(new BorderLayout(p1,BoxLayout.Y_AXIS));
    p1.add(message = new JLabel(""));
    message.setOpaque(true);
    p1.setSize(170,100);
    message.setBackground(Color.lightGray);

    setGUIObject(p1);
}

public void activate()
{
    joinGroup("power-suppliers");
    requestRole("power-suppliers",competence+"-priceagent");
    display("Power: "+"price-agent");
}

public void display(String s)
{
    if (hasGUI())
        message.setText(s);
    else
        println(s);
}

protected void handleMessage(ACLMessage m)
{
    Color bkg=message.getBackground();
    String s="";
    // OBTAIN REQUEST FOR BID MESSAGE FROM ISO
    if (m.getAct().equals("send-prices"))
    {
        debug("s"+m.getSender());
        debug("r"+m.getReceiver());
        message.setBackground(Color.yellow);

        // INTERFACE WITH MATLAB AND RUN ROUTINE
    }
}

```

APPENDIX C SOURCE CODE FOR AGENT PROGRAMS

```

// price_change.m
    ISO.engine.engEvalString("price_change");
    display("Finding new prices...");
    pause(500);

// OBTAIN THE NEW PRICES ARRAY FROM MATLAB
    new_prices = ISO.engine.engGetArray("curr_prices");
    display("current prices received");
    pause(500);

// SEND THE NEW BID VALUES TO THE SUPPLIERS
    AgentAddress aa2;
    for(int t1=0; t1<ISO.bidders.length;t1++)
    {
        aa2 = ISO.bidders[t1];
        sendMessage(aa2, new
        ACLMessage("newprice",String.valueOf(new_prices[0][t1]]));
        pause(500);
    }
    display("New Bid values sent");
    pause(2000);

    message.setBackground(Color.lightGray);
    display("Power: "+"price-agent");
}
}

public void live()
{
    while (true) {
        Message m = waitNextMessage();
        if (m instanceof ACLMessage)
            handleMessage((ACLMessage)m);

    }
}
}
}
*****

```

REFERENCES

- [1] M. Ilic, F. Galina and L. Fink, *Power Systems Restructuring: Engineering and Economics*, Kluwer Academic Publishers, 1998.
- [2] H. Singh, A. Papalexopoulos, “Competitive Procurement of Ancillary Services by an Independent System Operator”, *IEEE Trans. on Power Systems*, Vol. 14, No.2, May 1999.
- [3] M. Woolridge for “Cooperating and Collaborating Intelligent Agents”. And N. Jennings (1995). “Intelligent Agents: Theory and Practice”. *Knowledge Engineering Review*, 10(2), 115-152.
- [4] S. J. Russel, P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall series in Artificial Intelligence, 1st Edition, 1995.
- [5] A.K. David and F.S. Wen, “Strategic bidding in competitive electricity markets: A Literature survey”, *Proceedings of IEEE PES 2000 Summer Power Meeting*, vol.4, Seattle, USA, July 2000, pp. 2168-2173.
- [6] D. Rerkpreedapong, A. Feliachi, “Fuzzy Rule Based Load Frequency Control In Compliance with NERC’s Standards”, *Proceedings of IEEE PES 2002 Summer Meeting*, Chicago, USA, July 2002.
- [7] G. Sheblé, *Computational Auction Mechanisms for Restructured Power Industry Operation*, Power Electronics and Power Systems Series, Kluwer Academic Publishers, 1999. ISBN 0-7923-8475-X
- [8] S. Stoft, *Power System Economics*, IEEE/WILEY, ISBN- 0-471-15040-1, February 2002.
- [9] G. Strbac, “Trading Electricity and Ancillary Services in the Reformed England and Wales Electricity Market”, *Proceedings of IEEE PES 2000 Winter Meeting*, Vol.1, USA, pp. 41-44.
- [10] <http://www.caiso.com>; <http://www.calpx.com>
- [11] M. Ilic, P. Skantze, “Electric Power Systems Operation by Decision and Control”, *IEEE Control Systems Magazine*, August 2000, pp. 25-39.
- [12] F.L. Alvarado, “*The dynamics of power system markets*”, Report Number PSERC-97-01. © F.Alvarado, March 12, 1997.
- [13] F.L. Alvarado, J. Meng, C.L. DeMarco, W.S. Mota, “Stability Analysis of Interconnected Power Systems Coupled with Market Dynamics”, *IEEE Power Engineering Society Winter Meeting*, 2002. Volume: 2, 2002, pp. 1009 –1009
- [14] K. Bhattacharya, D. Chattopadhyay, and J. Parikh, “Real-Time Adaptive Pricing for Load-Frequency Control in an Interconnected Power System”, *International Journal of Power and Energy Systems*, Vol. 18, No. 2, 1998.
- [15] J.D. Weber, J.T. Overbye, Christopher L. DeMarco, “Inclusion of Price Dependent Load Models in the Optimal Power Flow”, *IEEE Proceedings of Hawaii International Conference On System Sciences*, January 1998, Kona, Hawaii.

REFERENCES

- [16] R. Christie, A. Bose, "Load Frequency Control Issues In Power Systems Operation After Deregulation", *IEEE Trans. on Power Systems*, Vol. 11, No. 3, August 1996.
- [17] A.D. Papalexopoulos, A.G. Angelidis, "On the Operation and Pricing of Real-Time Competitive Electricity Markets", *IEEE Power Engineering Society Winter Meeting*, 2002, Volume 1, pp. 420-427.
- [18] M. Ilic, P. Skantze, C.N. Yu, L. Fink, J. Cardell, "Power Exchange for Frequency Control (PXFC)", *Proceedings in Bulk Power System Dynamics and Control IV: Restructuring*, Santorini, Greece, 1998.
- [19] J.W. Lamont, S. Rajan, "Strategic Bidding in an Energy Brokerage", *IEEE Trans. on Power Systems*, Vol. 12, No.4, November 1997, pp. 1729-1733.
- [20] S. Kai, G.B. Shrestha, L. Goel, "Strategic Bidding Strategies in Power Market: Ramp Rate Considerations", *IEEE Power Engineering Society*, Summer Meeting, 2000, Volume 4, pp. 2144-2149.
- [21] G. Xiong, T. Hashiyama, S. Okuma, "An Evolutionary Computation for Supplier Bidding Strategy in Electricity Auction Market", *IEEE Power Engineering Society*, Winter Meeting, 2002, Vol. 1, pp. 83-88.
- [22] F. Wen, A.K. David, "Optimal Bidding Strategies and Modeling of Imperfect Information among Competitive Generators", *IEEE Trans. on Power Systems*, Vol. 16, No. 1, February 2001, pp. 15-21.
- [23] D. Zhang, Y. Wang, P.B. Luh, "Optimization Based Bidding Strategies in the Deregulated Market", *Proceedings of IEEE PES 21st Power Industry Computer Applications Conference (PICA'99)*, 1999, pp. 63-69.
- [24] X. Guan, Y. Ho, F. Lai, "An Ordinal Optimization Based Bidding Strategy for Electric Power Suppliers in the Daily Energy Market", *IEEE Trans. on Power Systems*, Vol. 16, No.4, November 2001, pp. 788-797.
- [25] S. Hao, "A Study of basic bidding strategy in clearing pricing auctions", *Proceedings of IEEE PES 21st Power Industry Computer Applications Conference (PICA'99)*, 1999, pp. 55-60.
- [26] I.O. Novas, C. Meseguer, C. Battle, and J. J. Alba, "A Simulation Model for a Competitive Generation market", *IEEE Trans. on Power Systems*, Vol. 15, No. 1, February 2000.
- [27] M. He, H. Leung, "An Agent Bidding Strategy Based on Fuzzy Logic in A Continuous Double Auction", *IEEE Intl. Conference on Systems, Man and Cybernetics: E-systems and E-man for Cybernetics in Cyberspace*, Tucson, Arizona, 2001, pp 583-588.
- [28] J. Ferber, *Multi-Agent Systems - An Introduction to Distributed Artificial Intelligence*, Addison Wesley Paris, 1995, ISBN: 0-201-36048-9.
- [29] M. Knapik, J. Johnson, *Developing Intelligent Agents for Distributed Systems*, McGraw-Hill, ISBN: 0-07-035011-6.
- [30] G. Weiss, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, 1999 - MIT Press, ISBN: 0-262-23203-0.

REFERENCES

- [31] A.H. Bond, L. Gasser, *Readings in Distributed Artificial Intelligence*, Morgan Kaufmann Publishers, 1988, ISBN: 0-934613-63-X.
- [32] K. Warwick, A. Ekwus, R. Aggarwal, *Artificial Intelligence Techniques In Power Systems*, 1997: Institute of Electrical Engineers, London ISBN:0-85296-897-3.
- [33] M. Woolridge, "Agent-Based Software Engineering", *IEE Proceedings on Software Engineering*, 144(1): pp. 26-37, February 1997.
- [34] M. Woolridge, N.R. Jennings, D. Kinny, "A Methodology for Agent-Oriented Analysis and Design", *Proceedings of the Third International Conference on Autonomous Agents (Agents'99)*, 1999 ACM Press, Seattle, WA, USA, pp. 69-76. <http://citeseer.nj.nec.com/wooldridge99methodology.html>
- [35] F. Brazier, B.D. Keplicz, N.R. Jennings, and J. Treur, "Formal Specification of Multi-Agent Systems: a Real-World Case", *ICMAS 95, First International Conference on Multi-Agent Systems: proceedings*, June 1995, San Francisco, California, MIT Press, pp. 25-32.
- [36] Agent Environments, <http://www.agentbuilder.com/AgentTools/index.html>
- [37] M. Amin, "Modeling and Control of Electric Power Systems and Markets", Guest Editorial, *IEEE Control Systems Magazine*, 2000, pp. 20-24.
- [38] Massoud Amin, "Toward Self-Healing Energy Infrastructure Systems", *IEEE Computer Applications in Power*, 2001, pp. 20-28.
- [39] P. Kundur, *Power System Stability and Control*, N. J. Balu, M. G. Lauby: McGraw- Hill, Inc., 1993.
- [40] R.Graham, *Power System Oscillations*, M. A. Pai, Editor, Norwell: Kluwer Academic Publishers, 2000
- [41] P.M. Anderson, A.A. Fouad, *Power System Control and Stability*, IEEE Press, IEEE Power Systems Engineering Series, 1994.
- [42] S. N. Talukdar, "Autonomous Cyber Agents: Rules for Collaboration", *Proceedings of the Thirty-First Hawaii International Conference on Systems Science (HICSS-31)*, Hawaii, January 1998
- [43] E. Camponogara, S. N. Talukdar, "Agent Cooperation: Distributed Control Applications", *Proceedings of the International Conference on Intelligent System Application to Power Systems*, April 1999.
- [44] Eduardo Camponogara, "Controlling Networks with Collaborative Nets", *Dissertation*, Carnegie Mellon University, September 2000.
- [45] S.A. Harp, S. Brignone, B.F. Wollenberg, and T. Samad, "SEPIA, A Simulator for Electric Power Industry Agents", *IEEE Control Systems Magazine*, August 2000, pp.53-68.
- [46] C. Lucas, A.G. Jahromi, "Commitment in Bidding Agents", *IEEE International Conference on System, Man and Cybernetics*, 2001, pp. 3107-3112.
- [47] Y. C. Lam, Felix F. Wu, "Simulating Electricity Markets with Java", *Proceedings of Power Engineering Society 1999 Winter Meeting*, Vol. 1, pp. 406-410.
- [48] H. Liu, B. Yuan, H. Dai, J. Lin, Y.X. Ni, F.F. Wu, "Framework Design of a General-purpose Power Market Simulator based on Multi-Agent Technology",

REFERENCES

- Power Engineering Society Summer Meeting*, 2001, Volume: 3, 2001, pp. 1478 – 1482.
- [49] MADKIT, Multi-Agent Development Kit, <http://www.madkit.org>, 2001.
- [50] Fuzzy Logic Toolbox, MATLAB[®], High Performance Numeric Computation and Visualization Software, and SIMULINK[®], Dynamic System Simulation Software, The Mathworks Inc., 2001.
- [51] JMatLink, Software Package for Dynamic Interfacing between Java and Matlab, <http://www.held-mueller.de/JMatLink>, 2001.
- [52] A. Papalexopoulos, H. Singh, “On the Various Design Options for Ancillary Services Markets”, *Proceedings of the 34th Hawaii International Conference on System Sciences*, Hawaii, 2001. <http://www.hicss.hawaii.edu/diglib.htm>
- [53] S.S. Oren, “Design of Ancillary Service Markets”, *Proceedings of the 34th Hawaii International Conference on System Sciences*, Hawaii, 2001. <http://www.hicss.hawaii.edu/diglib.htm>