ELECTRICITY MARKET SIMULATOR

by

ZIHAN NING

Submitted in partial fulfillment of the requirements

For the degree of Master of Science

Thesis Adviser: Dr. Mingguo Hong

Department of Electrical Engineering and Computer Science

CASE WESTERN RESERVE UNIVERSITY

January, 2017

CASE WESTERN RESERVE UNIVERSITY

SCHOOL OF GRADUATE STUDIES

We hereby approve the thesis of

Zihan Ning

Candidate for the degree of Master of Science*

Committee Chair

Dr. Mingguo Hong

Committee Member

Dr. Vira Chankong

Committee Member

Dr. Kenneth Loparo

Committee member

Dr. Evren Gurkan-Cavusoglu

Committee member

Dr. Vincenzo Liberatore

Date of Defense

October 17, 2016

*We also certify that written approval has been obtained for any proprietary material contained there

TABLE OF CONTENTS

List of Figures

Acknowledgement

I would like to thank my advisor, Prof. Mingguo Hong. He always gives me quite helpful advice and guidance on my research area with his excellently professional knowledge and experience. I do feel so lucky and honored to be one of his students, because he is a very responsible and great teacher who not only teaches me knowledge but also provides good training to help me gain the capability for doing research in a professional way. I really appreciate the process doing research with his patient guidance and encourage.

And I would like to express my thank to Prof. Kenneth Loparo, Prof. Vincenzo Liberatore, Prof. Vira Chankong and Prof. Evren Gurkan Cavusoglu for attending my maser thesis defense. I really appreciate your time and helps with my graduation.

In addition, I want to thank Case Western Reserve University and the Department of Electrical Engineering and Computer Science for providing me good studying resources and education.

Finally, I really thank my family and friends for their support and encourage during the three years in the United States. I feel so proud and lucky to have them in my life.

Electricity Market Simulator

Abstract

by

ZIHAN NING

Electricity market enables the purchase, sale and trading of electricity through an economic market mechanism. Most North American electricity markets adopt the standard Locational Marginal Price (LMP) design that embeds physical system operational reliability requirements in the market clearing algorithms. This thesis describes the design and implementation of an Electricity Market Simulator based on the LMP market design for supporting classroom teaching and professional research. Among the key functions, the simulator provides a web-based interactive environment for students to witness the end-to-end electricity market clearing process. The Electricity Market Simulator demonstrates both power system operation and economic market competition through intuitive graphic animation of realistic scenarios. The users will be able to conduct market scenario research experiments, either in the roles of a market participant or the Independent System Operator (ISO) who is the market administrator, and study market bidding strategies.

Chapter 1. Introduction

This thesis presents the design and implementation of the Electricity Market Simulator, as a part of the author's effort to fulfill her Master degree requirement at Case Western Reserve University (CWUR). The Electricity Market Simulator has been developed for supporting classroom teaching and professional training, with an interactive web access to demonstrate the end-to-end electricity market clearing process. The users will also be able to use the simulator tool to conduct market experiments, either in the roles of the market participants or an Independent System Operator (ISO), to test market bidding strategies.

Before the design and implementation details are presented, the basic concepts and theories of electricity market design are first introduced as the following.

Electricity, which is a special kind of commodity, can be bought and sold in the electricity market. The electricity market provides a competition platform to enable the electricity purchase and sale among different market participants. Similar to the wholesale and retail markets of other common commodity products, electricity markets can be categorized as wholesale electricity markets and retail electricity markets.

Generally speaking, electricity is first produced and sold in bulk quantities in the wholesale markets before it is sold to end-use consumers in the retail markets [1]. The electricity market being discussed in this thesis is the wholesale electricity market.

1.1 Wholesale Electricity Market

The wholesale electricity market facilitates trading between the generation companies (GENCOs) and different types of resellers. As illustrated in [1], resellers mainly include electricity utility companies, competitive power providers and electricity marketers. As in the markets of other common commodities, electricity market participants compete with each other to maximize their own economic benefits. The wholesale electricity market is regulated by the Federal Energy Regulatory Commission (FERC) and operated by an Independent System Operator (ISO) or Regional Transmission Organization (RTO).

1.1.1 Restructuring of U.S. Electricity Market

In the early days, electric utilities initially were known as vertically integrated electric utilities that handle all the functions of generation, transmission and distribution in their domain of business areas. In power system operation, one critical task is to guarantee electric power supply and demand remain in precise balance at every instant over a wide area. Under the vertically integrated operation scheme, this challenging task requires a central authority to govern both the supply and demand of electricity within a control area of the power grid [2]. Moreover, in the vertically integrated electric utilities, large utilities had the authority over all activities in generation, transmission and distribution of power [3] and served as the only electricity provider in a region. As a result, only the local electric utility can produce and sell electric power and the utility must provide service to all electric consumers in its service territory, not just those that would be profitable [4]. In addition, vertically integrated utilities would also prefer using their own generation resources to meet the load demand, even if it may be cheaper to buy from the neighboring generation resources. As a result, high production costs would occur as well as high charge rates for consumers.

Therefore, due to the lack of competition in the traditional vertically regulated structure, a single monopoly supplier may occur and overcharge consumers to maximize its own profits [5]. The objectives of electric utility restructuring are to increase competition, decrease regulation, and in the long run lower consumer prices [6]. The restructuring of power market in U.S. has resulted in the decomposition of the three major components of power system: generation, transmission and distribution. The independent system operators (ISO) or the regional transmission organizations (RTO) are established to regulate and administer the operation of power grid.

1.1.2 Electricity Market Types

Based on the traded commodity, there are four different wholesale market types in current deregulated power market, including Energy Market, Capacity Market, Ancillary Service Market and Transmission Market. The energy market is a centralized mechanism to facilitate the energy trading between buyers and sellers. The capacity market and ancillary market are operated unbundled from the energy market to guarantee the reliability of power system. In the capacity market, the auctions for capacity typically look at least 6 months and up to a few years ahead [7]. Moreover, typical ancillary service markets include regulation, spinning reserve, non-spinning reserve and replacement reserve. The commodity traded in transmission market is the transmission rights which have significant economic values when congestions occurs.

In the restructured energy markets, energy is traded through a two-settlement system that consists of the day-ahead market and the real-time market. On one hand, market participants in the day-ahead market commit to buy or sell electricity one day before the operating day to help avoid price volatility. On the other hand, market participants in the real-time market buy and sell electricity during the real-time operation of the day. Day-ahead market is cleared based on the forecasted day-ahead hourly load and price information one day before the operation day, while the real-time market is usually cleared every five or fifteen minutes. In the real time market, the market participants can respond to the imbalances when random fluctuations of real time load occur. The market simulator of this thesis work can simulate both the day-ahead and the real time market.

1.1.3 Electricity Market Entities

As stated in section 1.1.1, the restructuring of power market has changed the role of traditional entities in the vertically integrated utility. The key entities of the current power market can be categorized into market operators and market participants.

ISO/RTOs are the leading market operators. The critical responsibilities of ISO/RTOs include 1) maintain and improve the security and reliability of power system, 2) promote and operate a fair and competitive electric wholesale markets, 3) coordinate the long-term planning and 4) provide technical information on energy issues.

The key market participants include GENCOs, TRANSCOs (Transmission Company) and DISCOs (Distribution Company). In electricity markets, a GENCO operates the existing generating plants and its objective is to maximize the power generation profits. A TRANSCO builds and maintains the high-voltage transmission system to deliver power from the GENCOs to DISCOs. TRANSCOs will recover the investments by the access charges for the transmission systems. DISCOs such as electric utilities purchase power from power generation providers and distribute the power to their customers through its network facilities. In competitive electricity markets, customers expect the least-cost and high-quality supply of electric energy.

In a competitive electricity market, the ISO coordinates with market participants and solve the market clearing problem by maximize the social welfare while satisfying the hourly load demand, transmission security and operation requirements [8]. Briefly speaking, the day-ahead market is operated as follows. Market participants such as GENCOs submit their hourly and block generation offers to the ISO. ISO will solve the market optimization problem based on market participants' energy and price bids to get the hourly generation and dispatch schedule for supplying hourly loads. The generation schedule is made available by the ISO to corresponding market participants. The market participants could use the ISO's market signals such as Locational Marginal Prices (LMPs) and transmission congestion for reconsidering their proposed offers on generating resources. And ISO will resolve the market problem after receiving the updated biding information, in what is called the rebid period.

1.2 Power System Operation and Economics

The electric power production and consumption are rapidly becoming market-driven for better economic efficiency as electric energy plays an increasingly intimate role in the U.S. national economy. In the meantime, power system security remains to be the most important aspect of operation. The main objective in power system operation, therefore, is to maintain low production cost while ensuring the security of the system. Power system Security is defined as the ability of the interconnected system to provide electricity with the appropriate quality under normal and contingency conditions [9] and often introduced to the economic study of an electric system, in the form of constraints to be satisfied.

1.2.1 Optimization in Power Systems Operation

To achieve the economic operation of electric power systems, the optimization models such as Economic Dispatch (ED), Security Constrained Economic Dispatch (SCED), Unit Commitment (UC), and Security Constrained Unit Commitment (SCUC) are commonly used. Among the models, SCUC and SCED are two essential mathematical engines used to perform market clearing in the day-ahead and real-time markets.

ED determines the most efficient, least-cost generation schedule by minimizing the total production cost of all committed generating units while satisfying the operational constraints of the available generation resources, such as power balance and unit generation capacity limits. ED problem can be extended to an advanced SCED problem by also considering power flow equations and transmission constraints under base case and contingencies.

Different from ED which only schedules the economic dispatch of generators already online, the UC problem is to optimally decide hourly schedule of generating units' online status while satisfying all operation requirements/constraints for generating units and power systems. The unit and system level constraints usually include power balance limit, spinning and operating reserve requirements, minimum On/Off time and ramping limits of generating units, generating capacity limits, must-on and reliability requirements, and etc.. Mathematically, UC in practical power system is a large scale Mixed Integer Linear Problem (MILP). Lagrangian relaxation and mixed integer programming are two widely used method to solve the UC problem. In addition, the UC problem is the base problem of SCUC where the power transmission security is modeled for both the base case and contingencies.

In our designed simulator, the market clearing engine can simulate both SCUC and SCED problem efficiently for the day-ahead 24 hourly intervals or 5 minutes' real time intervals (details in section 2.1.4).

1.2.2 Electricity Pricing

In restructured electricity market, the price of electricity is always the focus of all activities. In general, supply and demand determines the price of a commodity. In the case of electricity markets, the most basic pricing concept is the market clearing price (MCP) which is calculated according to the intersection of generation supply and load bidding curves. Generally, when there is no transmission congestion, MCP is a single price for the entire power system. However, when there is congestion on the transmission system, the electricity market cannot be cleared uniformly due to transmission line limits. Therefore, the locational marginal price (LMP) is designed to determine the nodal price of electricity and the market will be cleared at the bus level. In physical definition, LMP is the cost of supplying the next MW of load at a specific nodal location after considering costs associated with generation, transmission, and losses. Mathematically, LMP at any bus equals to the dual variable for the equality constraint at that bus. In addition, when there is no congestion or loss, LMP is the same with each other at any bus.

LMP reinforces the efficiency and reliability of the electric grid by providing price signals by which market participants can see when and where the system is stressed. The price signals also tell market participants when congestion or supply shortages will occur and allow them to react quickly to this.

As the most important pricing information, LMP information of the system can be efficiently calculated and clearly shown in our designed simulator (details in section 2.1.5 and 3.2).

1.3 A sReview on Electricity Market Simulators

In the ISO/RTOs, power market operation and economic analysis are operated by the commercially available energy market simulation tools. However, electricity market simulation is also required to support training and education, as well as research studies to examine the electricity market performance with both realistic and hypothetical scenarios. There are plenty of electricity market simulators including commercial closed-source software and open-source software. This section gives a brief review of the current popular commercial market simulators and academic simulators.

1.3.1 Commercial Market Simulators

The commercial closed-source software used by different ISOs including PLEXOS, PROMOD, GridView, PROSYM, PROBE are introduced in this section.

Energy Exemplar is the developer of the class-leading power market simulation software called PLEXOS for Power Systems [10]. PLEXOS is a successful fundamental power market simulation product that is used for power market modelling, simulation and analysis. It can simulate various applications in power system such as price forecasting, power market simulation and analysis, strategic modelling and decision support, co-optimization of ancillary services and energy dispatch, transmission analysis and congestion management, risk management and stochastic optimization. All of these functions can be performed for any time horizon from minutes to 40 years.

ABB’s product PROMOD is another software package that has been widely used in the power industry. PROMOD has been implemented in PJM for simulation modeling [11].

GridView is another software application developed by ABB, Inc. and it can efficiently simulate the generation and transmission planning, operational decision making and risk management [12].

PROSYM is a multi-area electric energy production simulation model developed by Henwood energy Inc. It is an hourly simulation engine for least-cost optimal production dispatch based on the resources’ marginal costs, with full representation of generating unit characteristics, network area topology and electrical loads. PROSYM also considers and respects operational and chronological constraints; such as minimum up and down times, random forced outages and transmission capacity. It is designed to determine the station generation, emissions and economic transactions between interconnected areas for each hour in the simulation period [13].

The PROBE software developed by PowerGEM, in general, is an advanced market simulation tool designed for efficient analysis of electricity markets, utilizing advanced security constrained unit commitment and economic dispatch algorithms to arrive at a least-cost solution and set energy market LMPs. PROBE’s proprietary algorithms are specifically designed for energy market optimization and do not rely on expensive third-party “black box” solvers. The result is a unique product that solves with extraordinary speed and accuracy while providing unprecedented transparency in commitment and dispatch decisions [14].

In general, power industries rely on these commercial simulators and although they have been licensed typically, the codes in those commercial simulators are still closed-source files. As a result, these commercial simulators may prevent usage by customers who do not have the financial resources to buy a full access license and also prevent university researchers and students to efficiently study the power system market. In addition, these commercial simulators which always require intensive experimentation and sensitivity analyses can be cumbersome to the research, teaching, and training purposes.

1.3.2 Open Source Market Simulators

Two open source simulators for power market application are developed to encourage market stakeholders, universities researchers and students to study restructured electricity markets in a more interactive way. These two simulators including Agent-based Modeling of Electricity Systems (AMES) and Tools for Assessment of Bidding into Electricity Auctions are briefly introduced and analyzed in this section.

The AMES Wholesale Power Market Test Bed, developed entirely in Java by an interdisciplinary team of researchers at Iowa State University, is a free open-source tool suitable for research, teaching, and training applications [15]. It is suited for the design of small and medium-sized systems (2-500 nodes). Through the graphical user interface in the software, users can easily perform functions such as the scenarios creation, modification, storage and analysis, parameter initialization and editing, the specification of behavioral rules for market participants, etc. [15]. In addition, it is worth to mention that AMES can also perform the operation and simulation of AC transmission grid. However, before one can successfully use AMES, an installation of AMES Market Package and the JAVA environment are required. In addition, AMES does not support web access and is therefore not friendly for users of moveable device such as phones and tablets. AMES may not be able to support multiple users who may want to study the same market case scenario with their own interfaces simultaneously.

Another open-source simulator is called Tools for Assessment of Biding into Electricity Auctions which is developed by the Power Systems Engineering Research Center (PSERC) as a computational tool for analyzing offers into auctions [16]. The tool is also implemented in Java and require input information include offers into the market, a generation firm's costs, and the transmission constraints. This tool can only simulate the power market through DC-OPF solvers and is specifically designed to ERCOT's zonal market.

In conclusion, very few free open source simulators are available and these simulators usually only support single computer users. This is also one of the motivations for the author to design and implement the Electricity Market Simulator with web access to allow multiple simultaneous access, for supporting classroom teaching and professional training.

Chapter 2. Electricity Market Simulator Design and Implementation

The Electricity Market Simulator is a software tool developed at the Case Western Reserve University (CWRU), for teaching college students who are studying the electricity market in a one-semester course, or training professionals who are at various levels of knowledge about the electricity markets. It integrates the economic principles of supply and demand and the physical operation constraints of electric power transmission, and provides concise explanations about the market clearing and settlement principles.

The current market simulators are either commercial software for power industries to run their business or research oriented to train the market operators. They lack the necessary pedagogy features to educate or train beginners who need to learn from hands-on experiments and step-by-step visual explanations. Therefore, a more intuitive electricity market simulator tool would be greatly valuable for supporting classroom education and professional training.

The Electricity Market Simulator is integrated with the Energy Information Dashboard (EIDA) [17], a software system that is developed at CWRU, to train electricity market participants in managing generation portfolios and schedule generation in response to the electricity markets price information. The integration of the market simulator with EIDA can significantly enrich the function of the original EIDA by allowing market participants submit offers and bids to the Independent System operator (ISO), and witness the market clearing and settlement process in an interactive manner. As a result, beginners will receive jump-starts in learning about electricity market and the seasoned professionals will gain further insights into previous work experiences.

This chapter will first provide the illustration of the communication mechanism among different components and followed by the description of the design and implementation of the electricity market simulator with detailed explanation of each component. The whole picture of how to operate this simulator is also provided in this chapter.

2.1 Components of Electricity Market Simulator

As illustrated in Fig. 1, the Electricity Market Simulator comprises of 1) the Market Database that stores supply offers and demand bids, load forecasts and transmission constraints data; 2) Market Clearing Engines that perform market clearing and settlement; 3) Market Operator Interface (MOI) that allows the operator to access the market database information and control the executions of the market clearing engine. 4) Web Service that supports the web interface to perform functions such as reading data from and writing data to the market database.

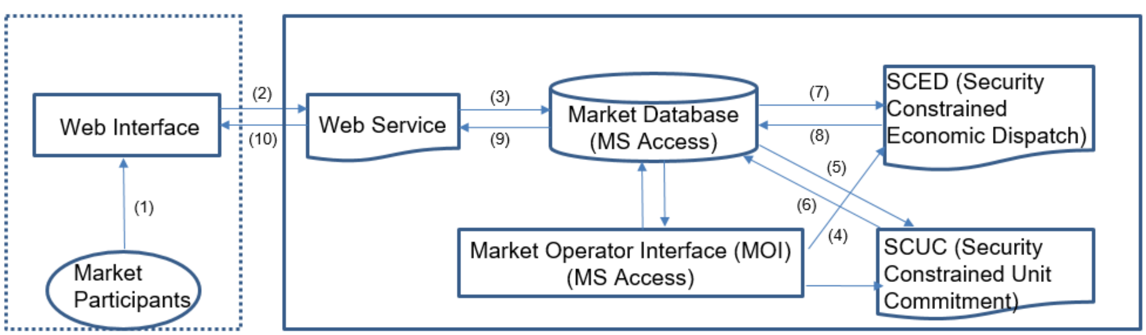


Fig. 1. Electricity Market Simulator Data Flow and Software Architecture

The electricity market simulator interfaces with the web interface through web service. The web interface allows market participants to submit offers into the market database and after the market clearing engines have finished the market clearing functions, those market participants will receive the output data via the web interface.

Each components of the electricity market simulator and the operation mechanism of the whole system are explained in details in the following sections.

2.1.1 Communications among Different Components

This section will conclude the whole picture of the data exchange process among different components in our power market simulator.

(1) Market participants submit their supply offers into the web interface of EIDA. The web interface will use these input data to call the web service and web service will transfer the data from the web interface to the market database.

(2) After the market database is updated with the newly input data in the web interface, the market operator can control the operation of market clearing through the MOI.

(3) The market clearing process includes two procedures. The first one is performed by one of the two market clearing engines to clear the market and return the generators' commitment results. The second one is performed by the other market clearing engine and this market clearing engine will use solved generators' commitment results and system data and return the generators' dispatched output and LMPs. After these two procedures, all of the output data will be written into the related database tables where the results data are stored.

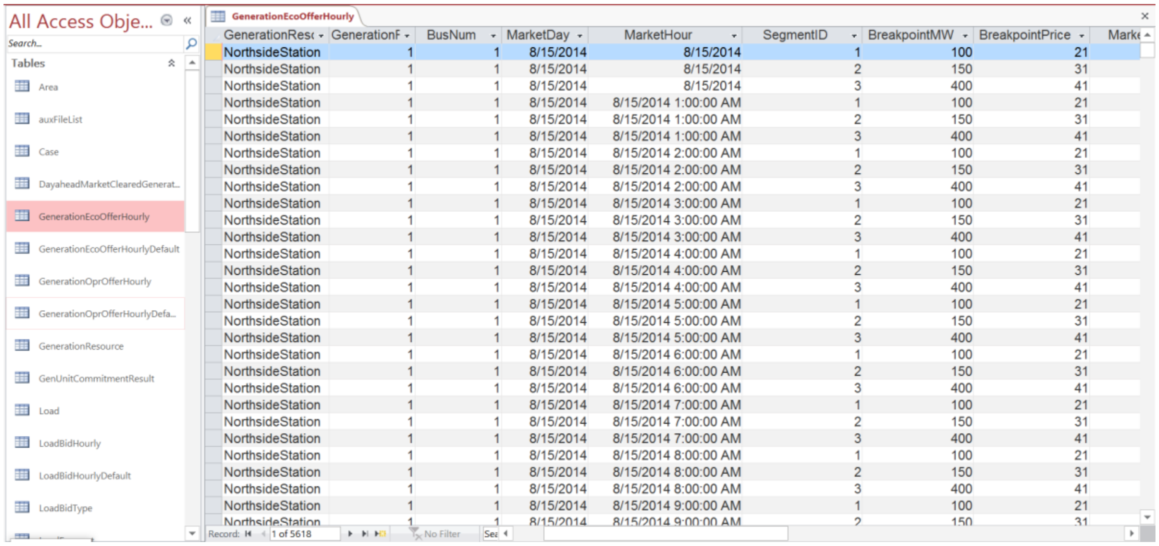
(4) Finally, when users query the data from the web interface, it will start to call the web service to read results data from the market database.

2.1.2 Market Database

Generally speaking, three kinds of databases including desktop, web enabled and server database have been widely used. Among these three databases, desktop database is the smallest and least expensive one. Microsoft (MS) Access is one of the most widely used desktop database system with powerful database engine and has become an industry standard in the desktop use of database system. In addition, MS Access can provide user-friendly graphical interface and software-development tools. Due to these advantages, the market database component of our market simulator is designed based on MS Access.

Tables are the basic objects that store the data throughout MS Access database. MS Access can also import or link directly to data stored in other applications and databases [29]. Specifically, in our designed market database, different tables are created to store the data of generators, loads, buses and branch which are the source data files for the market clearing engine. In addition, the results solved by the market clearing engine will also be sent back and stored in the designed database. In our market database, the stored data of generators include generator names, IDs of buses where generators are located, operational offer parameters, economic offer parameters and generation quantities and LMPs after market is cleared. The operational offer parameters include energy offer data and economic offer parameters which can be further divided into no load cost, generators' maximum and minimum generation limit for both economic and emergency, generators' minimum run time and minimum down time. The data for loads include loads names, IDs of buses where loads are located, bid offers and LMPs after market is cleared. Moreover, the data for buses indicate the bus names and bus IDs. Last but not the least, the data for branches include IDs of buses where branch are located and the transmission line limits.

A typical table in our market database containing the data of generators is illustrated in Fig. 2.



GenerationRes	GenerationF	BusNum	MarketDay	MarketHour	SegmentID	BreakpointMW	BreakpointPrice	Marke
NorthsideStation	1	1	8/15/2014	8/15/2014	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 1:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 1:00:00 AM	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014 1:00:00 AM	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 2:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 2:00:00 AM	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014 2:00:00 AM	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 3:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 3:00:00 AM	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014 3:00:00 AM	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 4:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 4:00:00 AM	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014 4:00:00 AM	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 5:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 5:00:00 AM	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014 5:00:00 AM	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 6:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 6:00:00 AM	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014 6:00:00 AM	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 7:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 7:00:00 AM	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014 7:00:00 AM	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 8:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 8:00:00 AM	2	150	31	
NorthsideStation	1	1	8/15/2014	8/15/2014 8:00:00 AM	3	400	41	
NorthsideStation	1	1	8/15/2014	8/15/2014 9:00:00 AM	1	100	21	
NorthsideStation	1	1	8/15/2014	8/15/2014 9:00:00 AM	2	150	31	

Fig. 2. Market Database Tables

MS Access use Structured Query Language (SQL) to store, manipulate and retrieve data stored in the database. SQL allows users to access data by defining the data in database and manipulate that data. For instance, in this Electricity Market Simulator project, the most frequently used manipulations include selection, insertion and updating or deleting records. When a SQL statement is executed, it will be passed to Jet Database Engine and Jet will return the appropriate data required by the SQL statement.

The execution of SQL statement is controlled by Visual Basic for Application (VBA). VBA is a programming language applied in MS Office Application to help users to automatically implement tasks instead of doing them manually and repeatedly.

The following is an example in the Electricity Market Simulator project to illustrate how SQL is executed in the VBA code to manipulate data in database.

The function of the code in Fig. 3 is to update the "*LMP*" column in a predefined table named "*MarketClearedLoadDA*". The SQL statement which is established as a text string (codes inside the "") is executed by the VBA function *dbs.Execute*. The part of the code in Fig. 3 after *WHERE* is the criteria to filter records. Specifically, after the execution of this example code, the LMP information will be updated by the information solved from Power World (e.g. *GenReords2*).

```
dbs.Execute "UPDATE MarketClearedLoadDA SET [LMP] = " & GenRecords2(1)(2)(i) & " " & _
            "WHERE (ScenarioID = " & ScenarioID & " AND LoadID = " & GenRecords2(1)(1)(i) & " " & _
            "AND BusNum = " & GenRecords2(1)(0)(i) & " AND MarketDay=#" & mktDay & "# " & _
            "AND MarketHour= #" & mktHour & "#)"
```

Fig. 3. Example of SQL usage in VBA

In addition to the *UPDATE* function illustrated in Fig. 3, other frequently used SQL statements include *INSERTION* and *SELECT* in this electricity market simulator project, which represent the function of inserting new records and reading existing records in database respectively.

2.1.3 Market Operator Interface

The MOI is an interface designed for operators to provide them an access to the market database information and control the executions of the market clearing engine. MOI is implemented in MS Access in our market simulator and Fig. 4 is an example form of this interface.

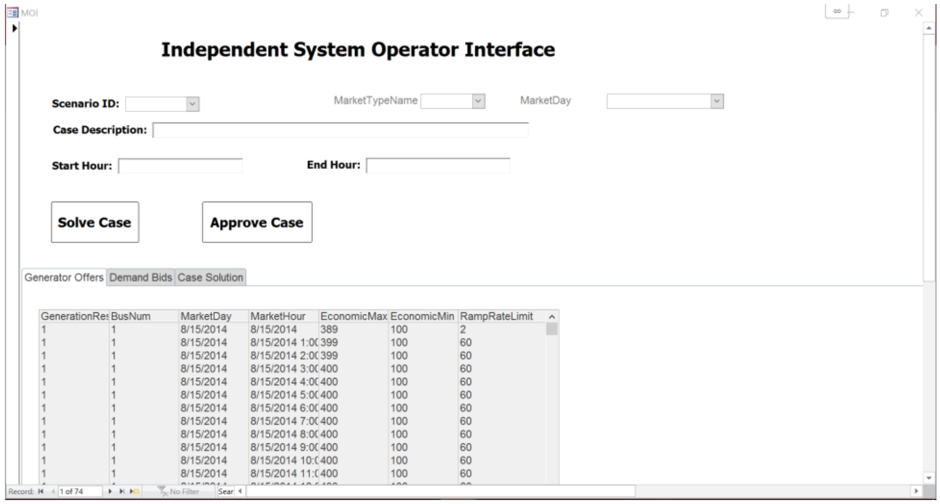


Fig. 4. Market Operator Interface

The purpose of the illustrated MOI in Fig. 4 is to execute market clearing function and settlement. The market operator could select scenario ID, market type and market day through this interface to execute the market clearing of a specific day. Different scenarios which represent different transmission network systems can be designed in our simulator. In our project, two scenarios including a three bus system and a five bus system are mainly designed and tested. Provided market types in this interface include the day-ahead and real time market. In addition, in this MOI, there is also a sub form displaying generator offers, demand bids and case solution which could help market operator have a quick view of these data.

In order to perform the market clearing functions through this MOI, market operator needs to click the "*Solve Case*" button. Then the connected VBA function will response to this click event and control the execution of market clearing process. The market clearing process will be performed by two market clearing engines which will be described in the next section.

2.1.4 Market Clearing Engines

There are two market clearing engines in the electricity market simulator which separately solve SCUC and SCED. In this section, the market clearing engine which solves SCUC will first be introduced following with the introduction of the other market clearing engine which solves the SCED. The market clearing engine which solves SCUC is built in the optimization software platform Aimms (an acronym for "Advanced Interactive Multidimensional Modeling System"). Aimms [18] is a powerful platform for the modeling and simulation of large-scale optimization problems [19] [20]. Aimms can provide convenient graphical end-user environment and algebraic modeling language [21].

Other advantages of Aimms are concluded as follows. (1) We can have direct access to the state of the art solvers such as CPLEX and Gurobi in Aimms. (2) Aimms provide a mixture of declarative and imperative programming style. Optimization models in Aimms are formulated through declarative language elements such as sets and indices, as well as scalar and multidimensional parameters, variables and constraints which is very convenient for the establishing the models. (3) Procedures and control flow statements in Aimms support the exchange of data with external data sources such as databases. (4) Aimms can solve a wide range of mathematical optimization problem types efficiently such as the mixed-integer programming problem which the mathematic essence of the basic SCUC problem in power market.

A simple description of SCUC has been illustrated in Chapter 1. In this section, the detailed explanation of the model of SCUC procedures in our market clearing engine is provided. The market clearing engine utilizes SCUC to optimize over the 24 hourly intervals of the operation day for day-ahead market.

In our established SCUC model, the total system operation cost is minimized while satisfying the generation and system level constraints.

The mathematic formulation of SCUC problem in our market clearing engine is represented in equation (1) - (16):

$$\begin{aligned}
\min \quad & \sum_i \sum_t C(i,\ t) * P(i,t) + \sum_i \sum_t ST(i,\ t) * C_{st}(i) + \sum_i \sum_t On(i,\ t) * C_{NL}(i) \\
& + \sum_i \sum_t C_{RegUp}(i,t) *\ P_{RegUp}(i,t) + \sum_i \sum_t C_{RegDn}(i,t) *\ P_{RegDn}(i,t) \\
& + \sum_i \sum_t C_{SpinRsv}(i,t) *\ P_{SpinRsv}(i,t) + \sum_i \sum_t C_{SuppRsv}(i,t) *\ P_{SuppRsv}(i,t)
\end{aligned}$$

$$s.t.\ \ P(i,t)\ +\ P_{RegUp}(i,t)\ +\ P_{SpinRsv}(i,t) \le P_{\max}(i,\ t) * On(i,t) \tag{1}$$

$$P(i,t)\ -\ P_{RegDn}(i,t) \ge P_{\min}(i,t) * On(i,t) \tag{2}$$

$$ST(i,\ t)\ -\ SD(i,\ t)\ =\ On(i,\ t)\ -\ On(i,\ t\ -1) \tag{3}$$

$$ST(i,\ t)\ +\ SD(i,\ t) \le 1 \tag{4}$$

$$ST(i,t) \le \frac{\sum_{t1=t}^{t+T_{\min\_run}(\mathrm{i})-1} On(i,t1)}{T_{\min\_run}(i)} \tag{5}$$

$$SD(i,t) \le \frac{\sum_{t1=t}^{t+T_{\min\_down}(i)-1} (1 - On(i,t1))}{T_{\min\_down}(i)} \tag{6}$$

$$\sum_n P(n,\ t)\ = \sum_n D(n,\ t)\ +\ Loss \tag{7}$$

$$\sum_i PTDF(n,\ k,\ t) * P(i,\ t) \le F_k^{\max}(MW) \tag{8}$$

$$P(i,t+1) - P(i,t) \le RRL(i,t) \tag{9}$$

$$P_{\mathrm{Re}\,gUp}(i,t) \le 5\ *\ RRL(i,t) \tag{10}$$

$$P_{SpinRsv}(i,t) \le 10\ *\ \mathrm{RRL}(i,t) \tag{11}$$

$$P_{SuppRsv}(i,t) \le MaxOffLineLimit(i,t)\ *\ RRL(i,t) \tag{12}$$

$$\sum_i P_{RegUp}(i,t) \ge SystemRegUpRequirement(t) \tag{13}$$

$$\sum_i P_{RegDn}(i,t) \ge SystemRegDnRequirement(t) \tag{14}$$

$$\sum_i P_{RegUp}(i,t) + \sum_i P_{SpinRsv}(i,t) \ge SystemRegSpinRsvRequirement(t) \tag{15}$$

$$\sum_i P_{RegUp}(i,t) + \sum_i P_{SpinRsv}(i,t) + \sum_i P_{SuppRsv}(i,t) \ge SystemOprRsvRequirement(t) \tag{16}$$

Where

t	*Time index*
T	*Total number of time intervals in the time horizon*
i	*Resource index*
n	*Bus index*
k	*Network branch index*
$P(i,t)$	*Power output from resource i* at *time t*
$P(n,t)$	*Power output from bus n* at *time t*
$D(n,t)$	*Power consumed from bus n* at *time t*
$ST(i,t)$	*Start – up provided by resource i* at *time t*
$SD(i,t)$	*Shut – Down provided by resource i* at *time t*
$On(i,t)$	*Online status of resource i* at *time t*
$C(i,t)$	*Energy bid cost of output* (MW) *for resource i* at *time t*
$C_{st}(i)$	*Cost of starting up for resource i* at *time t*
$C_{NL}(i)$	*Cost of no load for resource i* at *time i*
$C_{RegUp}(i,t)$	*Cost of regulating up reserve for resource i at time t*
$C_{RegDn}(i,t)$	*Cost of regulating* down *reserve for resource i at time t*
$C_{SpinRsv}(i,t)$	*Cost of spinning reserve for resource i at time t*
$C_{SuppRsv}(i,t)$	*Cost of supplemental reserve for resource i at time t*
$P_{max}(i,\ t)$	*Maximum output of resource i* at *time t*
$P_{min}(i,t)$	*Minimum output of resource i* at *time t*
$P_{RegUp}(i,t)$	*Regulating up reserve for resource i at time t*
$P_{\mathrm{Re}gDn}(i,t)$	*Regulating* down *reserve for resource i at time t*
$P_{SpinRsv}(i,t)$	*Spinning reserve for resource i at time t*
$P_{SuppRsv}(i,t)$	*Supplenmental reserve for resource i at time t*
$PTDF$	*Power transfer Distributeion factor*
F_k^{max}	*Transmission line capacity limits of branch k*
$T_{\min\_run}(i)$	*Minimum run time limit of resource i*
$T_{\min\_down}(i)$	*Minimum down time limit of resource i*
$RRL(i,t)$	*Ramp rate limit of resource i at time t*

Specifically, the objective function is to minimize the total system production cost. Constraints (1) and (2) are generators' output limits constraints; constraints (3) and (4) are additional constraints relating the binary variables of *ST*, *SD* and *On*; constraints (5) and (6) are the minimum run time and minimum down time limit of generators respectively; constraint (7) is system power balance constraint; constraint (8) is transmission line capacity constraint established by the power transfer distribution factor [34]; constraint (9) is resource ramp rate constraint for limiting the maximum change in resource generation output from hour to hour; constraints from (10) to (16) are operating reserve constraints including the regulating up reserve (10), spinning reserve (11) and supplemental reserve (12) [22]; constraints (13) to (16) ensure the total reserves procured on all qualifying resources must meet operation requirements to maintain reliability.

In the above SCUC model, energy bid costs and start-up/no-load costs for generators, capacity limits of generators, demand bid for load, transmission line capacity limits, Minimum run/down time limits of generators, ramp rate limits of generators and all of the indexes are taken as input data. The output data after solving this model by the market clearing engine are generators' commitment status, committed generators' power output. Load is assumed fixed

This model has been implemented in our market simulator where the input data is extracted from market database. Moreover, the output data will be written into the market database automatically after solving the SCUC model by the market clearing engine with Aimms.

The second market clearing engine is built in the software tool -PowerWorld [23], that runs SCED algorithm. As illustrated in section 2.1.4, the online/offline status of different generators have been first solved by SCUC that is built in Aimms. After receiving the unit commitment decisions solved by the first market clearing engine SCUC, the second market clearing engine SCED in our simulator determines the output of the generators as well as the LMP information for each bus. The PowerWorld tool is also utilized in our simulator to make graphical illustrations of the market and power system operations.

Through the Simulator Automation Server (SimAuto) interface, the PowerWorld software can be controlled by an external programs to build the power system model, initiate the optimal SCED solution and retrieving the solution data. Also with the help of SimAuto, market operators can obtain essential transmission model information such as network topology, the Y-bus matrix and generation shift factors easily from PowerWorld.

In this project, SimAuto is accessed from VBA in MS Access where the market database is built. With this capability, PowerWorld Simulator is externally controlled by MS Access. As a result, MS Access can 1) read and change data from the Simulator case, 2) control the execution of functions, 3) write the solution data back into database and 4) write solution auxiliary files. The market clearing results solved by the market clearing engine can also be imported into PowerWorld to be displayed an the one-line diagram of PowerWorld.

The followings are example codes to help illustrate how to use SimAuto to control the Power World and how PowerWorld Simulator can help perform the graphical display.

```
Public MySimAuto As SimulatorAuto
Public Sub ConnectSimAuto()
    Set MySimAuto = New SimulatorAuto
End Sub
Public Function OpenCase() As Boolean
    Dim output As Variant
    Dim ErrString As String

    CaseName = pwPathName + "\" + pwCaseName
    If Len(CaseName) > 0 And LCase(Right(CaseName, 4)) <> ".pwb" Then
        CaseName = CaseName + ".pwb"
    End If

    output = MySimAuto.OpenCase(CaseName)

    ErrString = output(0)

    If ErrString = "" Then
        OpenCase = True
    Else
        OpenCase = False

    End If
End Function
```

Fig. 5. Example of connection to SimAuto Server

Fig. 5 is the example code to enable the connection between MS Access and SimAuto Server. In addition, if the variable *OpenCase* equals to true, a case will be opened in the PowerWorld.

```
Public Function SetLoadMW(mktDay As Date, mktHour As Date) As Boolean
    Dim i As Long
    Dim ParamList As Variant
    Dim ValueList() As Variant
    Dim output As Variant
    Dim ObjType As String
    ObjType = "Load"
    ParamList = Array("BusNum", "LoadID", "LoadMW")

    If LoadVectorIndex < LoadVectorCount Then
        ReDim ValueList(0)
        Do While LoadVector(LoadVectorIndex, 2) = mktDay And LoadVector(LoadVectorIndex, 3) = mktHour
            Dim IndValueList(2) As Variant
            IndValueList(0) = LoadVector(LoadVectorIndex, 5)
            IndValueList(1) = LoadVector(LoadVectorIndex, 1)
            IndValueList(2) = LoadVector(LoadVectorIndex, 4)

            ValueList(UBound(ValueList)) = IndValueList

            LoadVectorIndex = LoadVectorIndex + 1
            If LoadVectorIndex < LoadVectorCount Then
                ReDim Preserve ValueList(UBound(ValueList) + 1)
            Else
                Exit Do
            End If

        Loop
        output = MySimAuto.ChangeParametersMultipleElement(ObjType, ParamList, ValueList)
    End If
    SetLoadMW = True
End Function
```

Fig. 6. Example code to write data into PowerWorld using SimAuto Server

In Fig. 6, the function *SetLoadMW* defined by VBA in MSAcess can use command "*ChangeParametersMultipleElement*" to write the data into Power World through SimAuto.

```
Public Sub SetUnitCommitment(mktDay As Date, mktHour As Date)
    Dim rst As ADODB.RecordSet
    Dim i As Long
    Dim SQLCommand As String
    Dim ParamList As Variant
    Dim ValueList() As Variant
    Dim output As Variant
    Dim ObjType As String

    ObjType = "GEN"
    ParamList = Array("BusNum", "GenID", "GenStatus")
    SQLCommand = "SELECT MarketDay, MarketHour,BusNum,GenID, GenName,GenDispatch,GenOn,GenST,GenSD " _
             + "FROM GenUnitCommitmentResult " _
             + "WHERE MarketDay =#" & mktDay & "# And MarketHour = #" & mktHour & "# And MarketTypeID = "
             " & marketTypeID & " And ScenarioID = " & ScenarioID & ""

    Set rst = New ADODB.RecordSet
    rst.CursorLocation = adUseClient
    rst.Open SQLCommand, CurrentProject.Connection
    rst.MoveFirst

    GenMWVectorCount2 = rst.RecordCount
    ReDim GenMWVector2(GenMWVectorCount2 - 1, 8)

    For i = 0 To rst.RecordCount - 1
        GenMWVector2(i, 0) = rst.Fields("MarketDay")
        GenMWVector2(i, 1) = rst.Fields("MarketHour")
        GenMWVector2(i, 2) = rst.Fields("BusNum")
        GenMWVector2(i, 3) = rst.Fields("GenID")
        GenMWVector2(i, 4) = rst.Fields("GenName")
        GenMWVector2(i, 5) = rst.Fields("GenDispatch")
        GenMWVector2(i, 6) = rst.Fields("GenOn")
        GenMWVector2(i, 7) = rst.Fields("GenST")
        GenMWVector2(i, 8) = rst.Fields("GenSD")
        rst.MoveNext
    Next i


    rst.Close
    Set rst = Nothing

    GenMWVectorIndex2 = 0

    If GenMWVectorIndex2 < GenMWVectorCount2 Then
        ReDim ValueList(0)
        Do While GenMWVector2(GenMWVectorIndex2, 0) = mktDay And GenMWVector2(GenMWVectorIndex2, 1) = mktHour
            Dim IndValueList(2) As Variant
            IndValueList(0) = GenMWVector2(GenMWVectorIndex2, 2)
            IndValueList(1) = GenMWVector2(GenMWVectorIndex2, 3)
            'dispatch only status of generators
            IndValueList(2) = GenMWVector2(GenMWVectorIndex2, 6)

            ValueList(UBound(ValueList)) = IndValueList

            GenMWVectorIndex2 = GenMWVectorIndex2 + 1
            If GenMWVectorIndex2 < GenMWVectorCount2 Then
                ReDim Preserve ValueList(UBound(ValueList) + 1)
            Else
                Exit Do
            End If
        Loop
        output = MySimAuto.ChangeParametersMultipleElement(ObjType, ParamList, ValueList)
    End If
End Sub
```

Fig. 7. Example code to write data into market database

Fig. 7 is an example code to illustrate how to import the generators' commitment statuses which are solved by the first market clearing engine SCUC into the PowerWorld

Simulator. The generators commitmet statuses are firstly solved by SCUC in Aimms and sent to the market database. MS Access calls the function *SetUnitCommitmet* to input the commitment results into the second market clearing engine SCED in PowerWorld.

```
Public Function RunUCSimulation()
  Dim output As Variant
  SetGenMaxMW mktDay, mktHour
  SetGenMinMW mktDay, mktHour
  SetLoadMW mktDay, mktHour
  setGenPiecewiseLinearCosts
  setBranchLimits
  SetUnitCommitment mktDay, mktHour
  SetGen1AGC

  SolvePWOPF mktDay, mktHour
  GetPWGenOutput mktDay, mktHour
  GetPWLoadOutput mktDay, mktHour

End Function
```

Fig. 8. Example code to run PowerWorld

Fig. 8 illustrates the whole process to run PowerWorld with the results from the first market clearing engine SCUC. Different set functions are called to write different data into Power World. After this input data process is completed, the function *SolvePWOPF* is called for the Power World to solve Optimal Power Flow (or SCED). After solution, the generator output and LMPs information can be read from the PowerWorld through calling SimAuto function *GetParametersMultipleElement*.

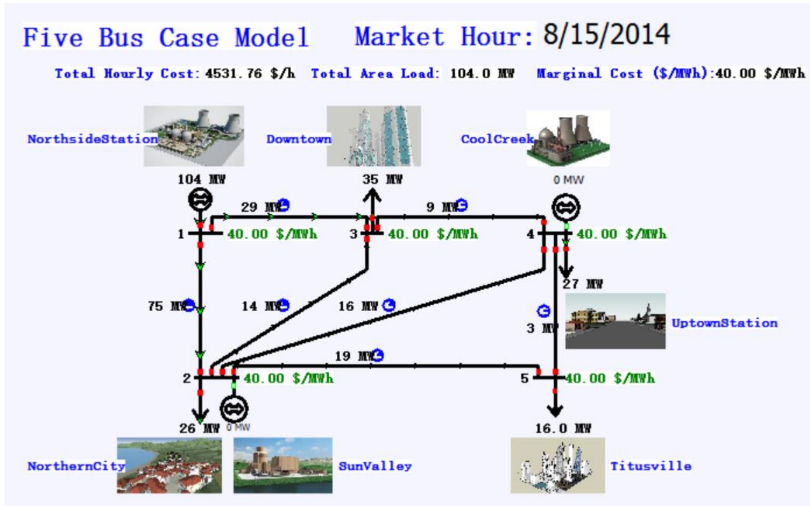


Fig. 9 An example of the graphical illustration of PowerWorld

Fig. 9 is an example of the graphical illustration built by PowerWorld for the first hour of the five-bus system. As illustrated in Fig. 9, magnitude and direction of power flow on each transmission line, LMP information on each bus, marginal cost and the total system cost are clearly shown by the graphical one-line display of the cleared market.

2.1.6 Web Service

The web interface is designed for the market participants to submit their biding information to the system. Web service is the technique support of this web interface.

In our project, we use Microsoft Visual Studio and Visual Basic.NET to develop the web service. The functions in web service will be called by the web portal by providing the input data and retrieving the output data. The functions written with Visual Basics.NET and SQL statements can be executed to read or write data from and into the market database.

The followings are two simplified examples from the web service to show how web service can exchange data with the market database.

Example 1: How to use web service to write maximum output limits of generators into the database.

```
Public Class WebService
    Inherits System.Web.Services.WebService
    <WebMethod()> _
    Public Function SetGenerationEconomicOffer(GenName, mktDay, mktHour, economicMax) As String
        Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\EIDA.accdb;Persist Security Info=False;")
        Dim insertcmd As New OleDbCommand
        Dim i As Integer
        i = 0
        While i < marketHour.Count()
            insertcmd.CommandText = "UPDATE GenerationOprOfferHourly SET [EconomicMax] = " + economicMax(i).ToString + " " &_
                                    "where (GenerationResourceName = '" + GenName + "' AND MarketDay=#" + mktDay + "# AND MarketHour=#" + mktHour(i))"
            i = i + 1
            con.Open()
            insertcmd.Connection = con
            insertcmd.ExecuteNonQuery()
            con.Close()
        End While
        Return "Successful"
        con.Close()
    End Function
```

Fig. 10. Example 1 of web service code

In the example code of Fig. 10, web interface will call the function "*SetGenerationEconomicOffer*" and provide the input data including generator names, operating day, operating hour and generators' maximum output limit. The output of this function will only be a string to show the status of the function calling. Inside the function, a SQL statement is executed to update the maximum output limit of the input generator. After calling this function, this generators' maximum output limits in market database will be updated for the whole operating day.

Example 2: how to use web service to read maximum output limits of generators from the database.

The function *GetGenerationOprOffer* in Fig. 11 is defined in the web interface to read the maximum limits data of generators. When the web interface calls this function and provides the input data including the generator's name and operating day, a SQL statement will be executed to retrieve the data from the market database as the function's output data.

```
Public Class WebService
    Inherits System.Web.Services.WebService
        <WebMethod()> _
    Public Function getGenerationOprOffer(GenName, mktDay) As Integer()
        End While
        Dim con As New OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\EIDA.accdb;Persist Security Info=False;")
        Dim insertcmd As New OleDbCommand
        Dim getcmd As New OleDbCommand
        Dim ds As New DataSet
        Dim result() As Integer
        con.Open()
        getcmd.Connection = con
        getcmd.CommandText = "SELECT * From GenerationOprOfferHourly where (GenerationResourceName = '" + GenName + "' AND MarketDay=#" + mktDay + ")"
        Dim db As New OleDbDataAdapter(getcmd)
        db.Fill(ds, "GenerationOprOfferHourly")
        con.Close()
        Dim t2 As DataTable = ds.Tables("GenerationOprOfferHourly")
        Dim j = 0
        For Each row As DataRow In t2.Rows
            result(j) = row.Field(Of Integer)(7)
            j = j + 1
        Next
        Return result
        con.Close()
    End Function
```

Fig. 11. Example 2 of web service code

2.2 User Experience

At the beginning of the simulation, users will form groups to represent various market participants such as generator companies (GENCO) and the ISO as the market administrator. Each group will receive a generation or load portfolio to begin with.

The market participants will then submit their offers through the EIDA web portal, to the ISO (i.e., the market simulator). These offers will update the market simulator database. The participants can confirm the update status of their offers by querying the database through the web interface. Once the offers are submitted, the market closes and the market operator proceeds to clear the market by executing SCUC through the Market Operator Interface. The market clearing outcomes in terms of cleared MW quantities and locational marginal prices (LMP) are then sent to both the database, and to PowerWorld through SimAuto for display on the one-line diagram. Figure [18] shows the PowerWorld one-line diagram with both market clearing MW and LMP. In this Figure, a 5-bus model has been chosen for demonstration.

Upon the market clearing, the market participants will also be notified and they can individually query, through the EIDA web interface the market clearing results on their submitted offers.

The Electricity Market Simulator will allow market participants to interact with the ISO in manners that emulate the real electricity market operation. For the market participants, their screens will be pre-filled with default offers, representing their actual physical generation capacity. These offers can be changed to reflect their market decisions. Guided by the instructor, students will be able to formulate decision strategies on their own, and witness how their decisions will impact the market clearing outcomes. For instructional purpose, the ISO's step-by-step market clearing process will be demonstrated in public to all market participants. The students will be informed that this process is kept strictly confidential by the ISO in reality. The Electricity Market Simulator provides an interactive environment for students to witness the end-to-end wholesale electricity market operation and to experiment different market bidding strategies and gain insight into the economic behaviors of the LMP market. As an outstanding feature of the simulator, the graphic animation of PowerWorld serves as a powerful visual aid to illustrate the interactions between economic decisions and physical system operation.

Chapter 3. Case Studies of the Electricity Market Simulator

Practical case studies are analyzed in this chapter to illustrate the characteristic of the designed electricity market simulator. Three case studies including a three-bus system simulation for one hour, five-bus system simulation for the day ahead market and five-bus system simulation for the real-time market are tested in this chapter. In Section 3.4, features that will be implemented on the future version of this market simulator is also analyzed.

3.1 Scenario A: Three Bus Model Simulation

The simple and intuitive three bus system is solved by our simulator for one hour. The web interface for this three bus model is illustrated in Fig. 12 and Fig. 13 is the graphical illustration of this system in through PowerWorld. As shown in Fig. 13, this three bus system includes two generators, one load and three transmission lines.

The web interface in Fig. 12 can automatically show up when users select the three bus model after they login in. In this web interface, two generation supplier should submit their supply offers under the generation offer tab and the load consumer should submit its load demand under the “MW” tab. After the users fill out the offers and bids, they need to click the submit button to allow the web service write the offers and bids data into the market database. The “Solve Scenario” button is designed for market operator to allow them to clear the market via Market Operator Interface. In this case study, the market operator will call PowerWorld to simulate the market by solving SCED. After the market is cleared, the solved results including the generation schedule, power follow on the transmission lines and LMP information on each bus will be written into market database.

Users can view the market clearing results by clicking the “query” button in the web interface. The results for generators are designed under the “Market Clearing” tab and the results for load is under the “Price” tab.

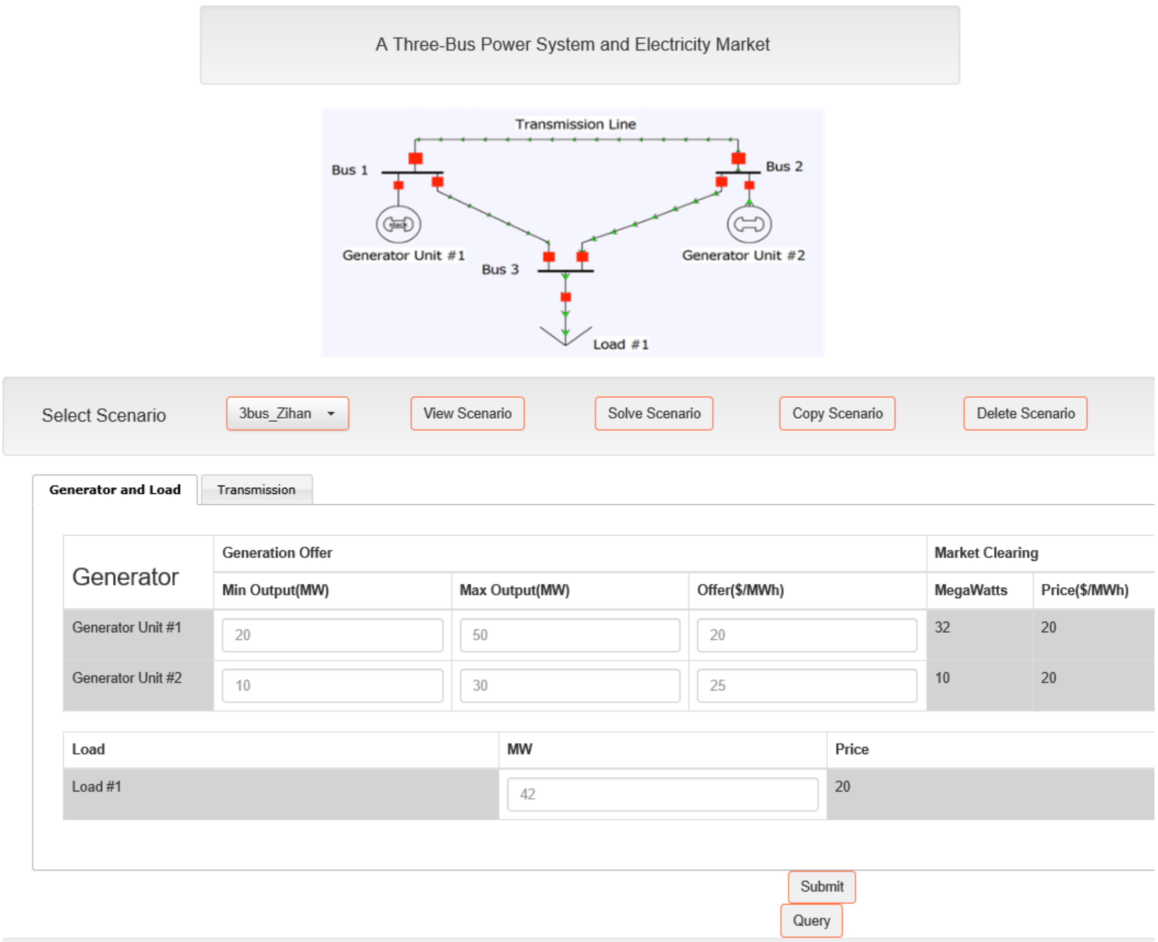


Fig. 12. Web interface for three bus case

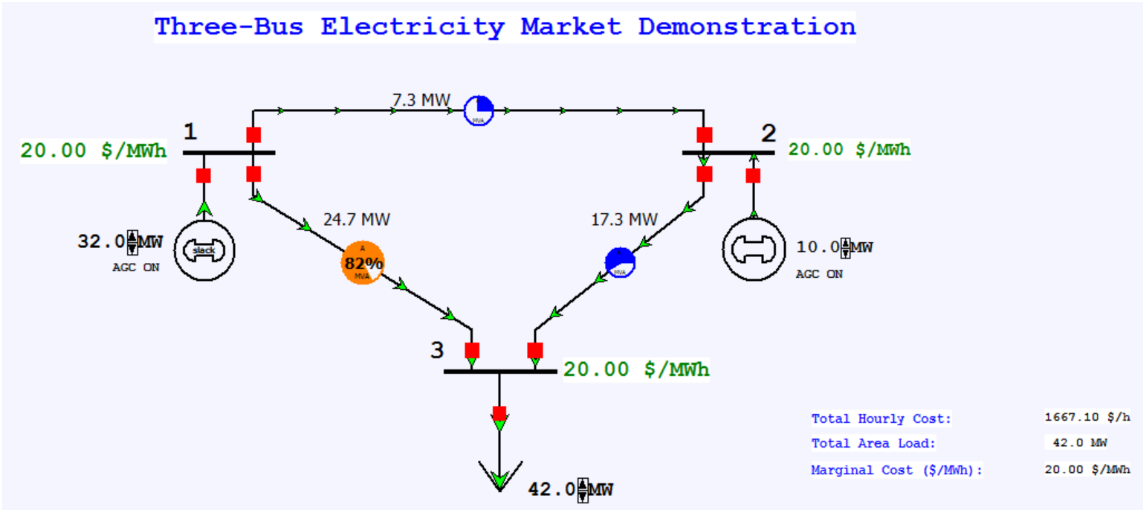


Fig. 13. Transmission network model of three bus case

In Fig. 13, no congestion occurs in each transmission line and the LMP results on three buses are the same and equals to 20 $/MW. Fig. 14 is the graphical illustration of this three bus system when congestion occurs. As we can see in Fig. 14, the power transferred on transmission line connecting bus 1 and bus 3 has reached the maximum capacity limit. To check the capacity limit of each transmission line, the users can click the "*Transmission*" tab in Fig. 15 and the web interface will be changed into Fig. 16. What's more, as shown in Fig. 14, the LMPs at each node are different as there is a congestion after the market is cleared.

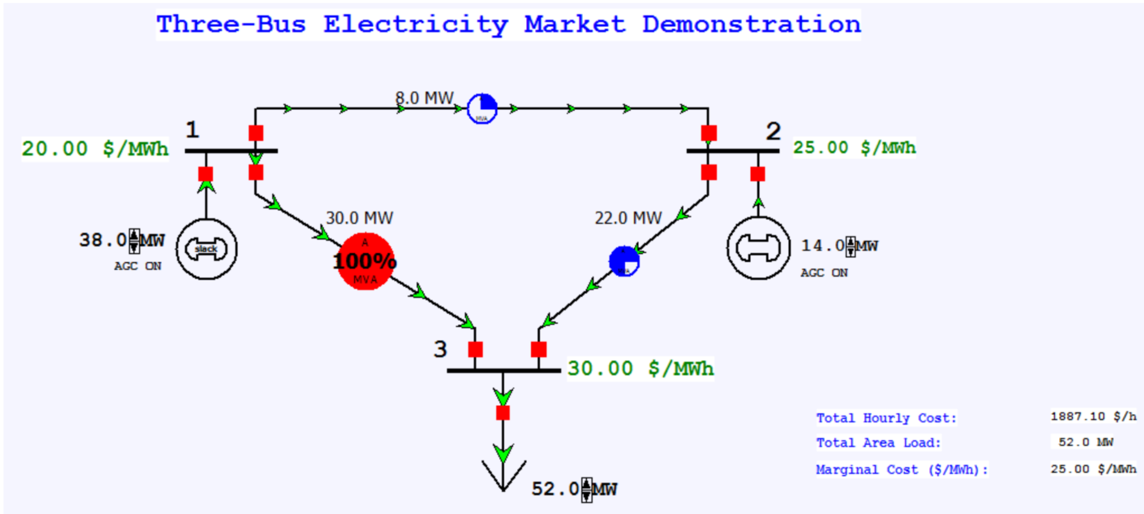


Fig. 14. Transmission network model of three bus case with a congestion

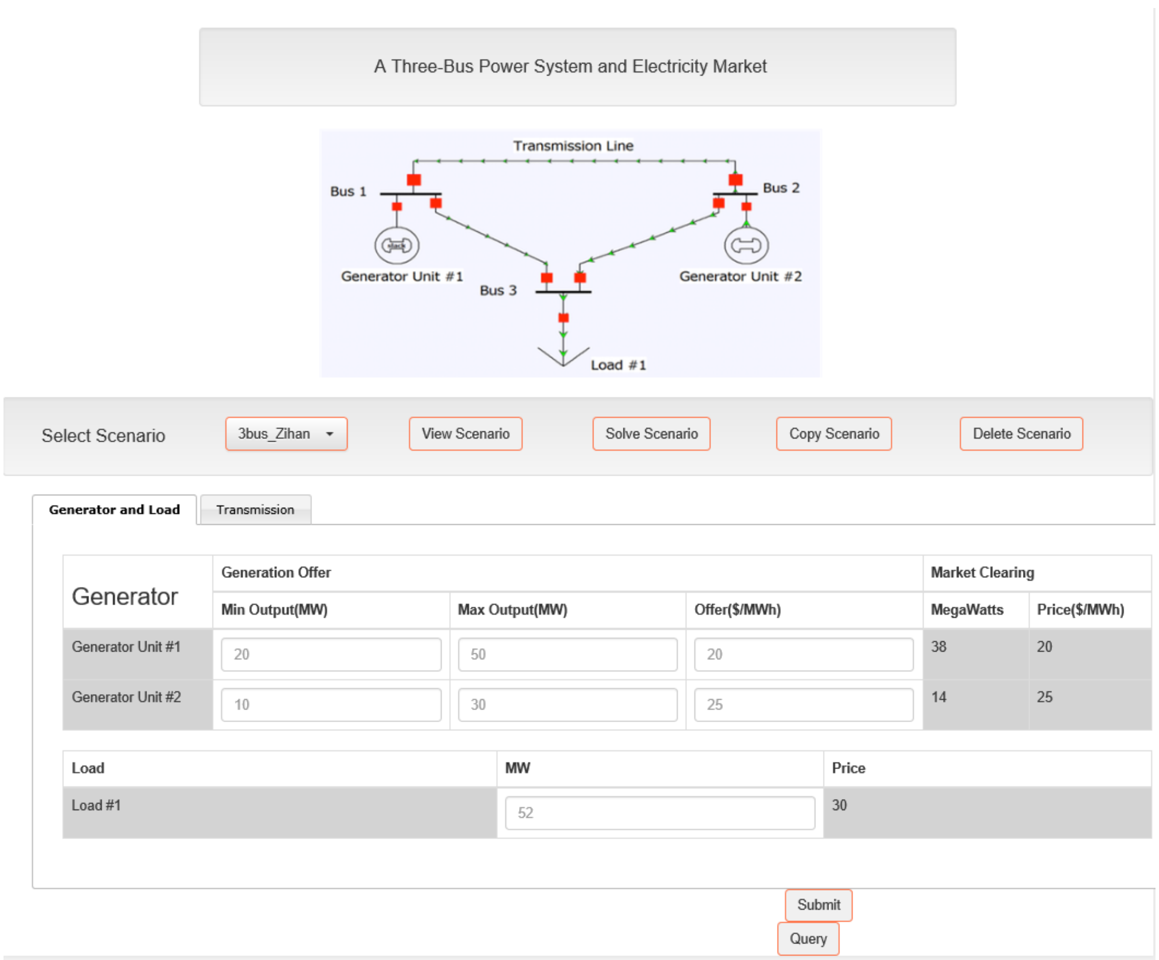


Fig. 15. Web Interface of three bus case with a congestion

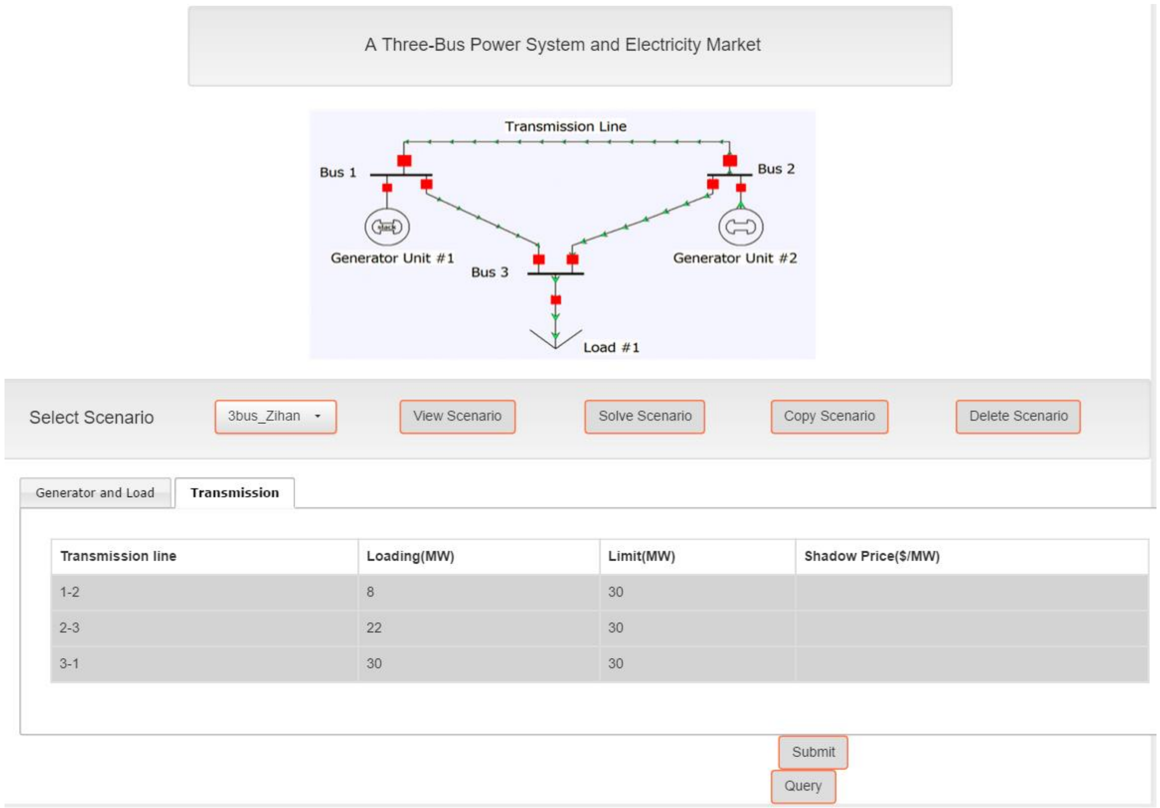


Fig. 16. Transmission line limits of three bus case

3.2 Scenario B: Five Bus Model Simulation for DA Market

The case study illustrated in this section is a five bus system simulated for the day-ahead market. For this case study, we will firstly run the first market clearing engine to solve the SCUC problem and secondly call the second market clearing engine to solve the SCED problem and calculate the output of generators and LMP information using the solved commitment decisions by SCUC. The details of the simulation of this five bus model simulation are explained as follows.

Fig. 17 is the graphical transmission line model of the five bus model and as shown in this figure the five bus system contains three generators, two load consumers and seven transmission lines.

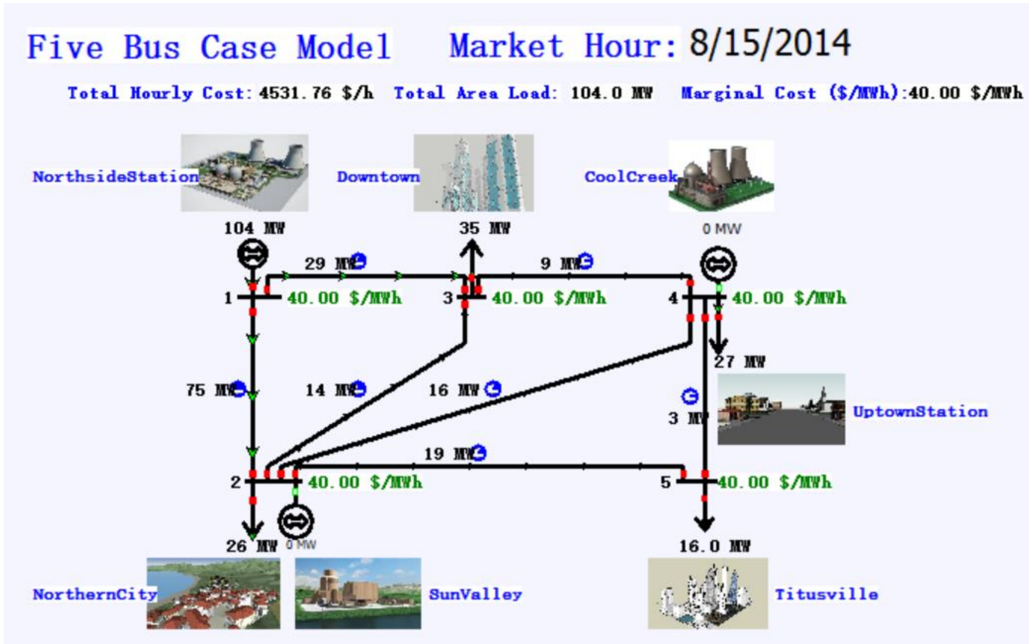


Fig. 17. Transmission network model of five bus case in PowerWorld

Fig. 18 is the designed web interface of this case study. For different market participants, different interfaces are designed. As shown in Fig. 18, users can select their roles in market and have access to their own interfaces. The interface of market operator is shown in Fig. 19 and the interfaces for the GENCOs are illustrated in Fig. 20-Fig. 23.

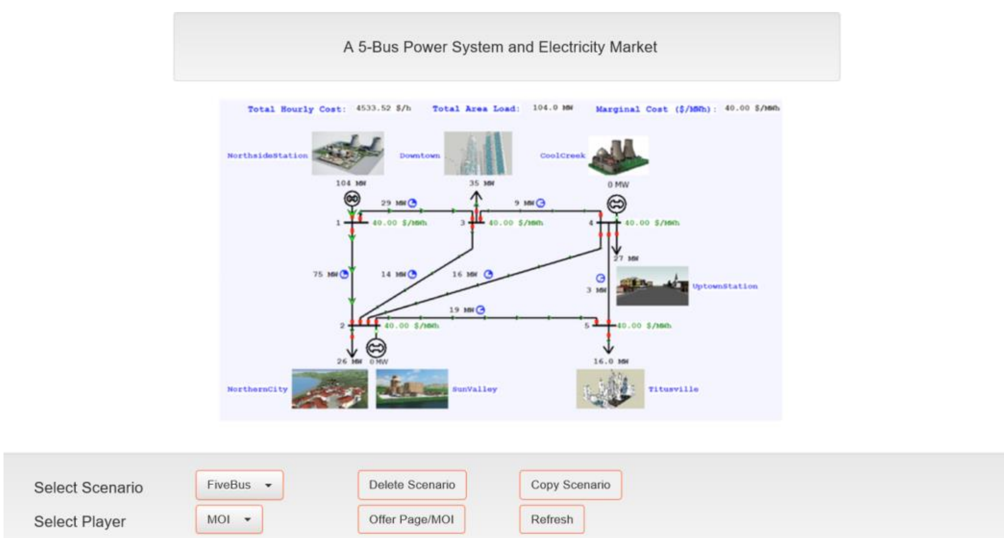


Fig. 18. Web Interface of five bus case

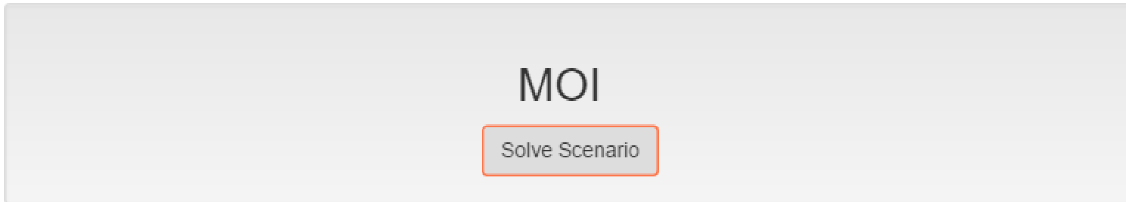
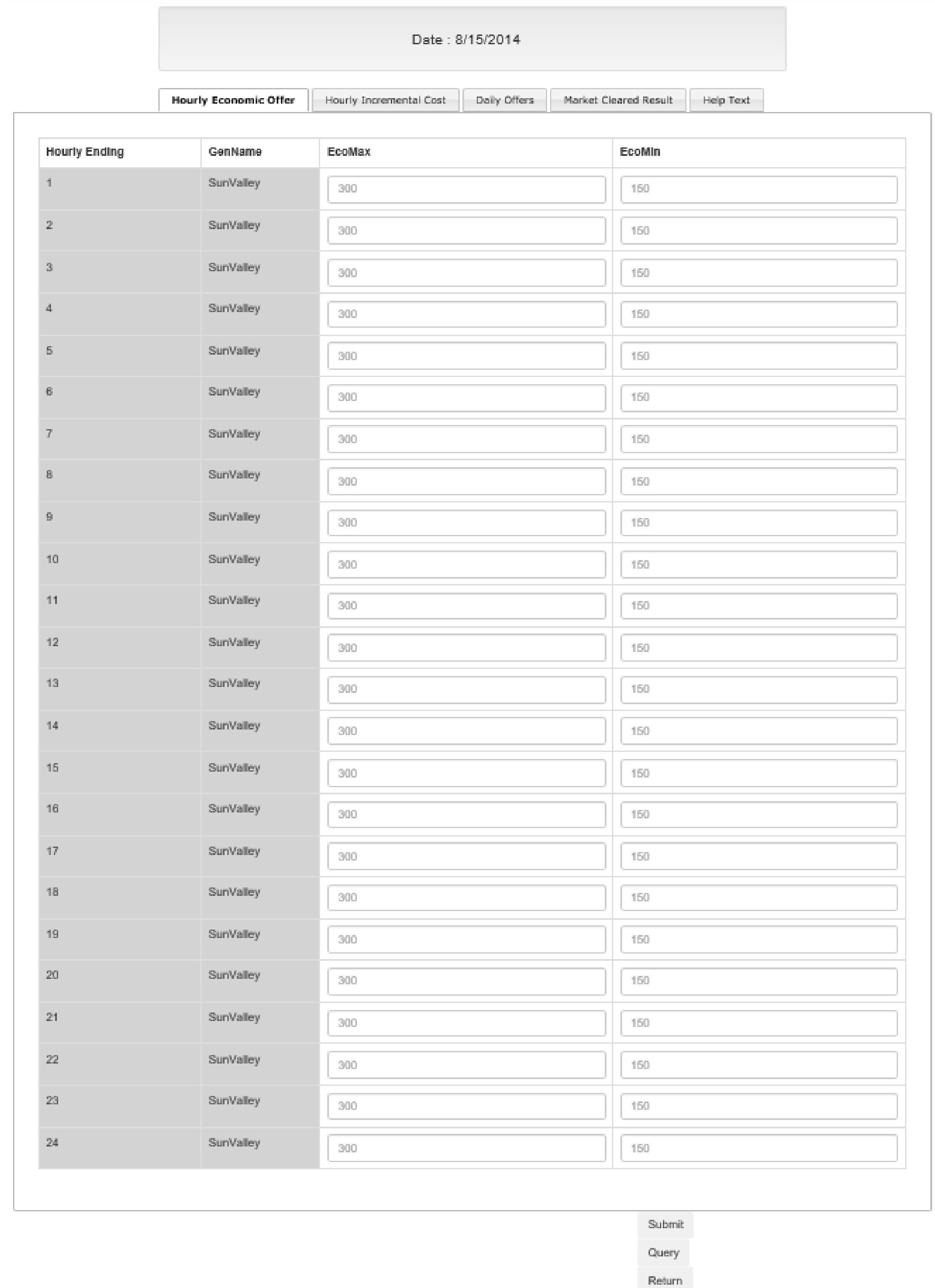


Fig. 19 Operation Interface of Market Operator

Fig. 20 is the offer page of generator "*NorthsideStation*". As show in Fig. 20, there are five tabs in this offer page including "*Hourly Economic Offer*", "*Hourly Incremental Offer*", "*Daily Offer*", "*Market Cleared Results and Help Text*". Specifically, in Fig. 20, generation supplier can input the generators' maximum output limit and minimum output limit information for 24 hours under the "*Hourly Economic Offer*" tab. Fig. 21 illustrates the "*Hourly Incremental Offer*" for generator's energy offer including the segments, MW and Price. The "*Daily Offer*" is showed in Fig. 22 where users can submit StartUpCost, NoloadCost, minimum run time limit and minimum down time limit under "*Daily Offer*" tab. In addition, after users submit these offer data and market is solved, they can check the commitment and dispatch results under the "*Market Cleared Results*" tab in Fig. 23



Date : 8/15/2014

Hourly Economic Offer | Hourly Incremental Cost | Daily Offers | Market Cleared Result | Help Text

Hourly Ending	GenName	EcoMax	EcoMin
1	SunValley	300	150
2	SunValley	300	150
3	SunValley	300	150
4	SunValley	300	150
5	SunValley	300	150
6	SunValley	300	150
7	SunValley	300	150
8	SunValley	300	150
9	SunValley	300	150
10	SunValley	300	150
11	SunValley	300	150
12	SunValley	300	150
13	SunValley	300	150
14	SunValley	300	150
15	SunValley	300	150
16	SunValley	300	150
17	SunValley	300	150
18	SunValley	300	150
19	SunValley	300	150
20	SunValley	300	150
21	SunValley	300	150
22	SunValley	300	150
23	SunValley	300	150
24	SunValley	300	150

Submit

Query

Return

Fig. 20. Hourly Economic Offer Page

Date : 8/15/2014

Hourly Economic Offer | **Hourly Incremental Cost** | Daily Offers | Market Cleared Result | Help Text

Hour	GenName	SegID	BreakpointMW	BreakpointPrice	SegID	MW	Price	SegID	MW	Price
1	SunValley	1	75	5	2	200	11	3	9999	22
2	SunValley	1	75	5	2	200	10	3	9999	20
3	SunValley	1	75	6	2	200	10	3	9999	20
4	SunValley	1	75	7	2	200	10	3	9999	20
5	SunValley	1	75	4	2	200	10	3	9999	20
6	SunValley	1	75	5	2	200	11	3	9999	20
7	SunValley	1	75	8	2	200	10	3	9999	20
8	SunValley	1	75	9	2	200	10	3	9999	20
9	SunValley	1	75	7	2	200	10	3	9999	20
10	SunValley	1	75	5	2	200	10	3	9999	20
11	SunValley	1	75	5	2	200	10	3	9999	20
12	SunValley	1	75	5	2	200	10	3	9999	20
13	SunValley	1	75	5	2	200	10	3	9999	20
14	SunValley	1	75	5	2	200	10	3	9999	20
15	SunValley	1	75	5	2	200	10	3	9999	20
16	SunValley	1	75	5	2	200	10	3	9999	20
17	SunValley	1	75	5	2	200	10	3	9999	20
18	SunValley	1	75	5	2	200	10	3	9999	20
19	SunValley	1	75	5	2	200	10	3	9999	20
20	SunValley	1	75	5	2	200	10	3	9999	20
21	SunValley	1	75	5	2	200	10	3	9999	20
22	SunValley	1	75	5	2	200	10	3	9999	20
23	SunValley	1	75	5	2	200	10	3	9999	20
24	SunValley	1	75	5	2	200	10	3	9999	20

Submit

Query

Return

Fig. 21. Hourly Incremental Offer Page

Date : 8/15/2014

Hourly Economic Offer | Hourly Incremental Cost | **Daily Offers** | Market Cleared Result | Help Text

GenName	StartupCost	NoLoad Cost	Min Runtime	Min Downtime
SunValley	1000	1000	1	1

Submit
Query
Return

Fig. 22. Daily Offer Page

Date : 8/15/2014

Hourly Economic Offer | Hourly Incremental Cost | Daily Offers | **Market Cleared Result** | Help Text

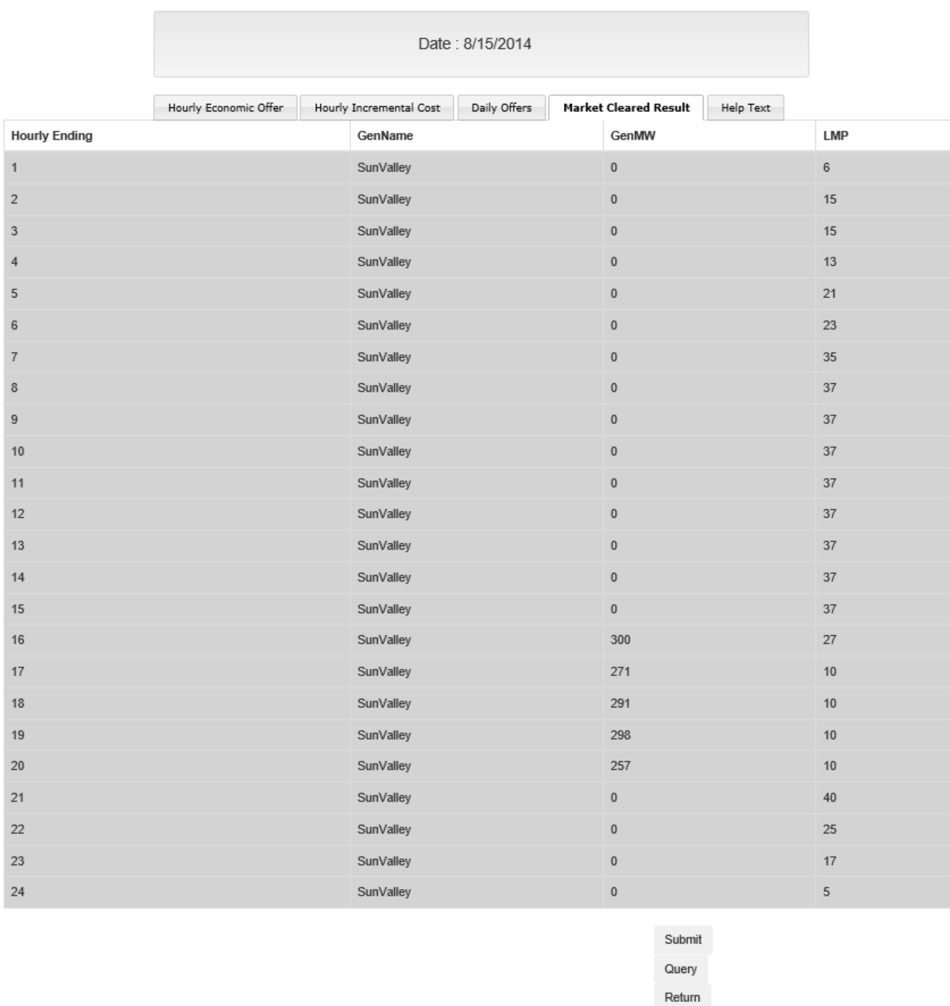
Hourly Ending	GenName	GenMW	LMP
1	SunValley	0	15
2	SunValley	0	22
3	SunValley	0	21
4	SunValley	0	20
5	SunValley	0	21
6	SunValley	0	23
7	SunValley	248	10
8	SunValley	300	20
9	SunValley	300	29
10	SunValley	300	27
11	SunValley	300	27
12	SunValley	300	27
13	SunValley	300	20
14	SunValley	300	20
15	SunValley	300	20
16	SunValley	300	27
17	SunValley	271	10
18	SunValley	291	10
19	SunValley	298	10
20	SunValley	257	10
21	SunValley	300	19
22	SunValley	189	5
23	SunValley	0	24
24	SunValley	0	22

Submit
Query
Return

Fig. 23 Market Cleared Results Page I

The detailed process of the simulation of this five bus system is illustrated as follows. (1) First, generators need to go to their offer page and submit their supply offers under the "*Hourly Economic Offer*", "*Hourly Incremental Offer*" and "*Daily Offer*". Users can replace the default data by simply inserting their own data into the corresponding cells. After the users have input all the required data, they need to click the "*Submit*" button in Fig. 20-23 to allow the system to write the data into the market database. The users can exit the system by clicking the "*Return*" button. (2) The market operator will click the "*Solve Market*" button to call the market clearing engines to clear the market and after the clearing procedure, the system will write the cleared results into the market database. (3) In order to get access to the cleared results, generation suppliers need to go to the "*Market Cleared Results*" tab in their offer page by clicking the "*Query*" button. Then they can know the commitment and dispatch results for 24 hours.

The following case study illustrates the impact of startup costs, no-load costs and minimum runtime for the commitment decisions. As shown in Fig. 22, the startup costs, no-load costs and minimum runtime can be set under the "*Daily Offer*" tab. After changing the startup costs, no-load costs of the generator "SunValley"' up to 5000 and 10000 dollars while maintaining minimum runtime as 1 hour, the newly obtained commitment results of "SunValley" have been changed as shown in Fig. 24. When startup costs and no-load costs are not high, the generation units in Fig. 23 are committed from hour 2-hour23, but in contrary, in Fig. 24 that the generator is only committed from **. The reason for the difference between the commitment results in Fig. 23 and Fig. 24 is that the no-load costs of generator "SunValley" is defined with a high value in the new case, and as a result, the final optimal commitment decisions will avoid to stat up this generator with high no-load costs for a long time since the objective of UC problem is to minimize the sum of the no-load costs, startup costs, production costs and operating reserve costs of all generators. To illustrate the commitment impact from the minimum runtime, we change the minimum runtime of “SunValley” to 19 and also set the startup costs and no-load costs with high values. The commitment result of this new case is shown in Fig.25. As illustrated in Fig. 25, “SunValley” must be committed for 19 hours to satisfy the requirement of the minimum runtime constraint. So from this simulation, we can draw a conclusion that the minimum runtime of generators will also largely impact the final decision results.



Hourly Ending	GenName	GenMW	LMP
1	SunValley	0	6
2	SunValley	0	15
3	SunValley	0	15
4	SunValley	0	13
5	SunValley	0	21
6	SunValley	0	23
7	SunValley	0	35
8	SunValley	0	37
9	SunValley	0	37
10	SunValley	0	37
11	SunValley	0	37
12	SunValley	0	37
13	SunValley	0	37
14	SunValley	0	37
15	SunValley	0	37
16	SunValley	300	27
17	SunValley	271	10
18	SunValley	291	10
19	SunValley	298	10
20	SunValley	257	10
21	SunValley	0	40
22	SunValley	0	25
23	SunValley	0	17
24	SunValley	0	5

Fig. 24 Market Cleared Results Page II

Date : 8/15/2014

Hourly Economic Offer | Hourly Incremental Cost | Daily Offers | **Market Cleared Result** | Help Text

Hourly Ending	GenName	GenMW	LMP
1	SunValley	0	40
2	SunValley	0	22
3	SunValley	0	21
4	SunValley	148	8
5	SunValley	199	9
6	SunValley	248	11
7	SunValley	198	20
8	SunValley	278	24
9	SunValley	300	29
10	SunValley	300	27
11	SunValley	300	27
12	SunValley	300	27
13	SunValley	298	24
14	SunValley	283	24
15	SunValley	283	24
16	SunValley	300	27
17	SunValley	271	30
18	SunValley	291	30
19	SunValley	298	27
20	SunValley	257	27
21	SunValley	253	24
22	SunValley	239	13
23	SunValley	0	24
24	SunValley	0	22

Submit

Query

Return

Fig. 25 Market Cleared Results Page III

3.3 Scenario C: Five Bus Model Simulation for RT Market

Currently, the web interface for the real time market simulation is still under development. Thus the detailed graphical illustration for the interface of real time market is not provided in this section. On the market side, the five bus case is solved by the second market clearing engine to solve SCED. PowerWorld can get access to the supply offer and demand bid data provided by generation supplies and the load demand directly through market database. The designed VBA functions in market operator interface are called by market operator to write data into PowerWorld and solve the SCED through PowerWorld. VBA functions that designed to read the results from PowerWorld is also designed to write the results data into market database. The real time simulation is operated for every 5 minutes. The graphical illustration of the cleared result for the five-bus system is shown in Fig. 24.

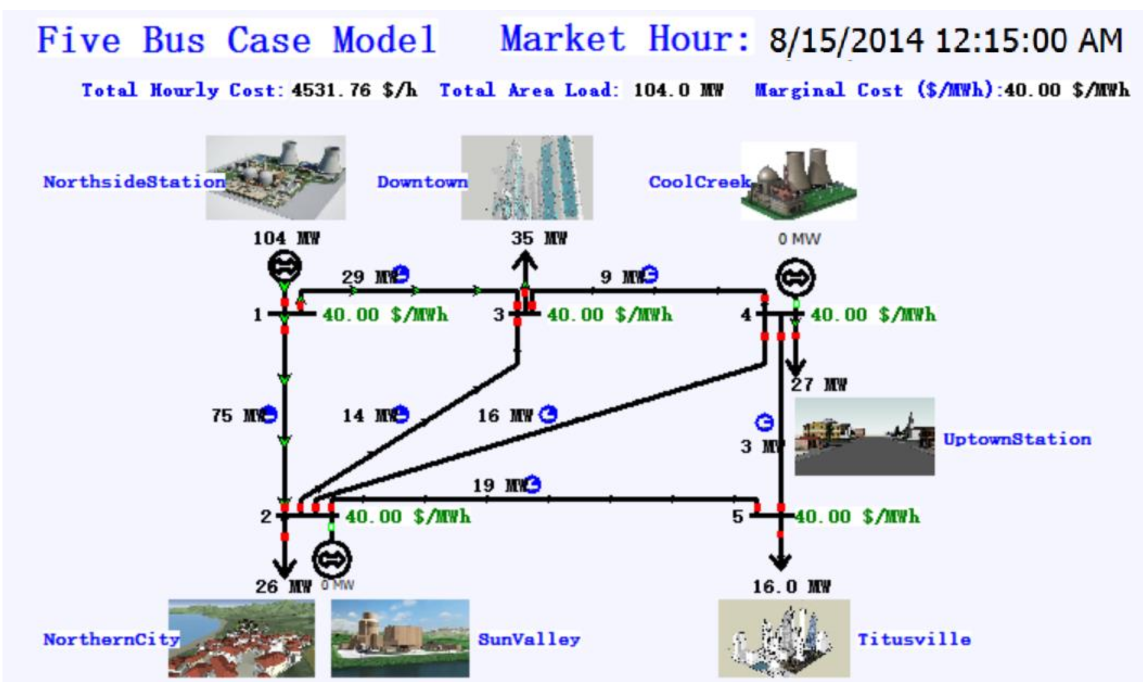


Fig. 26. Transmission Network Model of five bus case for Real Time market

3.4 Undergoing Simulator Features with Enhancements

This section will conclude some undergoing features of this market simulator. These features are developed together with an intern in our group. We made some enhancements on market operator interface and the market database.

Different with the market operator interface shown in Fig. 4, some improvements have been implemented on the market operator interface as shown in Fig. 25. The market operator interface in Fig. 25 can automatically check the database tables and have the ability to solve the market once new data are submitted to the web interface. With this new feature, instead of controlling the market clearing engine by the MS Access, the simulator can control the operation of the clearing engine directly by the web interface in Fig. 19.

In addition, in the previous version, the users can only perform the simulation of the predefined cases for instance, the predefined three-bus system in section 3.1 and the five-bus system in section 3.2 and 3.3. The improved version of market database can support the copy and delete operation of multiple scenarios and solve each scenario assigned in the web interface simultaneously. As a result, the users can design their own case study through our simulator.

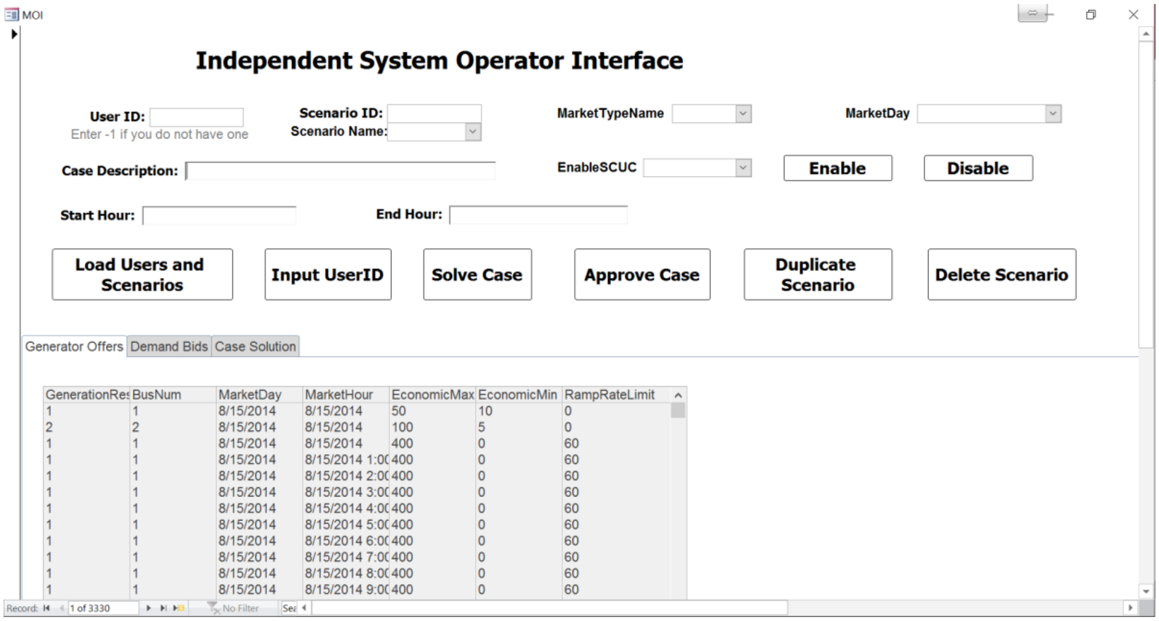


Fig. 27. Current Market Operator Interface

Chapter 4. Conclusion

This thesis is written based on the project of Electricity Market Simulator. A brief introduction of electricity market including the restructuring procedure and different components is firstly discussed. The major functions of power system operation and economics are described and current commercial and open-source market simulators are reviewed. After the introduction of the necessary background, the detailed design and implementation of the electricity market simulator is illustrated. The simulator is mainly designed for an effective classroom teaching and training tool on the electricity market. Our electricity market simulator can also be enhanced to enable electricity market researchers conduct market operation tests with human market participants and realistic transmission system operation conditions.

The future development plans include the web interface development for real time market simulation, features to simulate more complicated market scenarios, enhancement with the 10-generator, 39-bus model and renewable generator and enhancement with the ancillary services market.

Reference

[1] Market for Electricity. (n.d.), In PJM Learning Center, Retrived from http://learn.pjm.com/electricity-basics/market-for-electricity.aspx

[2] Michaels, Robert J. Vertical Integration and the Restructuring of the U.S. Electricity Industry (September 2004). Retrieved from http://ssrn.com/abstract=595565.

[3] Bhattacharya, Kankar, Math Bollen, and Jaap E. Daalder. *Operation of restructured power systems*. Springer Science & Business Media, 2012

[4] Willis, H. Lee, and Lorrin Philipson. *Understanding electric utilities and de-regulation*. Vol. 27. CRC Press, 2005.

[5] W.M. Warwick, *A Primer on Electric Utilities, Deregulation, and Restructuring of U.S. Electricity Markets*, Pacific Northwest National Laboratory, July 2000.

[6] Glover, J. Duncan, Mulukutla S. Sarma, and Thomas Overbye. *Power System Analysis and Design*. Cengage Learning, 2011.

[7] Wholesale Electricity Market Operations. (n.d.), NREL, Retrieved from http://www.nrel.gov/electricity/transmission/electricity_market.html

[8] M. Shahidehpour, W. F. Tinney, Y. Fu, "Impact of Security on Power Systems Operation", *Proceedings of the IEEE*, VOL. 93, NO.11. November 2005.

[9] Reliability Standards. (n.d.). NERC, Retrieved from http://www.nerc.com/~filesz/standards/Reliability_Standards.html.

[10] PLEXOS [Computer Software], Energy Exemplar, Retrieved from http://energyexemplar.com/software

[11] PJM market efficiency modeling practices (January 27 2015), PJM, Retrieved from https://www.pjm.com/~/media/planning/rtep-dev/market-efficiency/pjm-market-efficiency-modeling-practices.ashx

[12] GridView [Computer Software], ABB, Retrieved from http://new.abb.com/enterprise-software/energy-portfolio-management/market-analysis/gridview

[13] Jim McCalley, "Notes on Steady-state analysis", Notes for EE553, Iowa State University, Fall 2012

[14] PROBE [Computer Software], PowerGem, Retrieved from http://www.power-gem.com/PROBE.html

[15] Open Source Software (OSS) for Electricity Market Research (October 2 2016), Teaching, and Training, Iowa State University, Retrieved from http://www2.econ.iastate.edu/tesfatsi/ElectricOSS.htm

[16] "Tools for Assessment of Bidding into Electricity Auctions Final Project Report", Steven L. Puller, Ross Baldick, PSERC Publication 8-24, October 2008

[17] Energy Information Dashboard (EIDA), Case Western Reserve University, eida.case.edu

[18] AIMMS Corporation, www.aimms.com

[19] *Kallrath, Joseph (2004).* Modeling Languages in Mathematical Optimization. *Kluwer Academic Publishing.* ISBN 978-1-4020-7547-6.

[20] *Roelofs, Marcel (2010).* AIMMS Language Reference*). lulu.com.* ISBN 978-0-557-42456-6.

[21] *Roelofs, Marcel (2010).* AIMMS User's Guide . *lulu.com.* ISBN 978-0-557-06360-4.

[22] Hong, M., Sperry, K., Williams, J., & Gardner, J. (2012, July). Electricity market and operations reliability. In *2012 IEEE Power and Energy Society General Meeting* (pp. 1-7). IEEE.

[23] PowerWorld Corporation, www.powerworld.com