

An ISO Market Settlement Design Validation Approach using Probabilistic Market Simulation Study Techniques

Valentin Petrov, Jayant Kumar, Charles Richter, *Members, IEEE*

Abstract— In the Independent System Operator organizations, Settlement activity is an integral part of the market operation. One of the key components for measuring Independent System Operator market operational design correctness is to analyze how well market price signals create incentives and/or penalties for creating an efficient market to achieve the design goals under uncertainties due to unpredictable market conditions. The market settlement rule is an important vehicle to implement such price signals. This paper presents an approach for a probabilistic method to validate Settlement Systems design using probabilistic techniques for Independent System Operator market simulation. In addition, the paper presents critical comments on various differences in Settlement design approach.

Index Terms— Market System Design, Shadow Settlement, Settlement System, Market Simulation.

I. INTRODUCTION

Market competition is an important driver to achieve efficient allocation of resources. The recent emergence of de-regulated electricity markets in the energy sector directly exposes participants to the forces of supply and demand. An important characteristic of well-designed markets is that they produce transparent and believable price signals. A dispatch that is based on a price signal that does not remain consistent through the settlement process, is a dispatch price that does not incent the participants to follow the dispatch.

Proper wholesale electricity market design is critical to the success of an Independent System Operator (ISO) organization. While they may follow the development of the auction market clearing rules, the rules and procedures supporting settlement functions are an integral part of the market design. One of the key components for measuring the correctness of the market operational design is to analyze how well market price signals create incentives for maintaining an efficient market to achieve the design goals under uncertainties due to unpredictable market conditions. The market settlements rules are an important vehicle to implement such price signals.

Evolving market regions in the energy sector seem to be determined to build their market according to a new set of rules. While the specific implementation varies from market to market, market prices are generally based on offer information and the condition of the electricity network. In markets that have adopted locational marginal pricing (LMPs) the primary drivers for determining the price are the offer curve of the marginal generator(s), the amount of congestion on the grid, and the impact of each generator on the losses observed on the system. Such pricing schemes mean that the prices can and do vary in a way that is quite difficult to predict. This means there is an inherent price risk that comes with the market based pricing.

To mitigate risk, electricity markets generally support the notion of hedging using longer term markets or various financial instruments (e.g., seasonal and monthly financial transmission rights, futures contracts at points of interest, day ahead markets, hour ahead markets, etc.). At each point along the timeline, different market rules may be in place. Ultimately they should all work together to achieve an efficient allocation of resources.

Regulators and market designers have much economic theory on which they may base their rule-making. However, as with complex systems everywhere, high-level designs that seem adequate can still be subject to problems when implemented in reality if close attention is not paid to the details. If market participants can find an exploitable rule, they will. Designers can implement a market and observe the flaws in their design, but this can be an expensive discovery process. Experimental economics offers a non-financially binding method for finding the flaws prior to market go-live.

One such possible experimental-economical approach to validating Settlement Systems design is by using probabilistic techniques for ISO market simulation. In order to address the uncertainties and the impossibility to predict market conditions, this paper presents an approach for a probabilistic method to validate Settlement Systems design using probabilistic techniques for Independent System Operator market simulation. In addition, the paper presents critical comments on various differences in Settlement design approach.

The method presented here is directed towards all Central Market designers of Settlement rules. It can be used during the design phase, as well as during the test phase of the particular market settlement system for testing the correctness and

V. P. Author is with AREVA T&D Corporation, San Jose, CA 95110, USA (email: valentin.petrov@areva-td.com).

J. K. Author is with AREVA T&D Corporation, San Jose, CA 95110, USA (email: jayant.kumar@areva-td.com).

C. R. Author is with AREVA T&D Corporation, San Jose, CA 95110, USA (email: charles.richter@areva-td.com).

expected behavior of the rule set.

This paper is organized as follows. Section II contains a brief introduction to the use of probabilistic based experimental economics. Section III contains the problem formulation. Section IV includes an implementation approach, Section V presents the results and possible experiments to be setup, and finally Section VI presents a discussion of the results and finally section VII presents the conclusions of this work.

II. METHODS OF PROBABILISTIC APPROACH

One possible candidate for such a device is a probabilistic analysis of the market settlement rules. In this approach, the implemented set of rules is regarded as a black box. A large number of possible market scenarios generated at random is fed through the market. The results from these experiments are observed whether they conform to the initial market design expectations. A conclusion is then drawn to the fitness of the market design.

A. Black Box Testing

One of the commonly employed probabilistic evaluation techniques is called black-box testing. Black box testing approaches the set of rules encoded in the solution engine as fixed. Inputs are presented to the black box. The computation (within the black box) is performed which produces outputs. The outputs are captured. Outputs and the Inputs that produced the outputs become input-output pairs which can be analyzed as a set.

B. White Box Testing

A sometimes useful technique in system analysis is termed white box testing. White box testing is done by analytically proving the correctness of the settlement rule set against a set of test cases with known results. The set of test cases and results are derived from the requirement specifications and the results from the testing are compared against the results. Also, the correctness of the configuration is compared against the set of specifications by a formal review process.

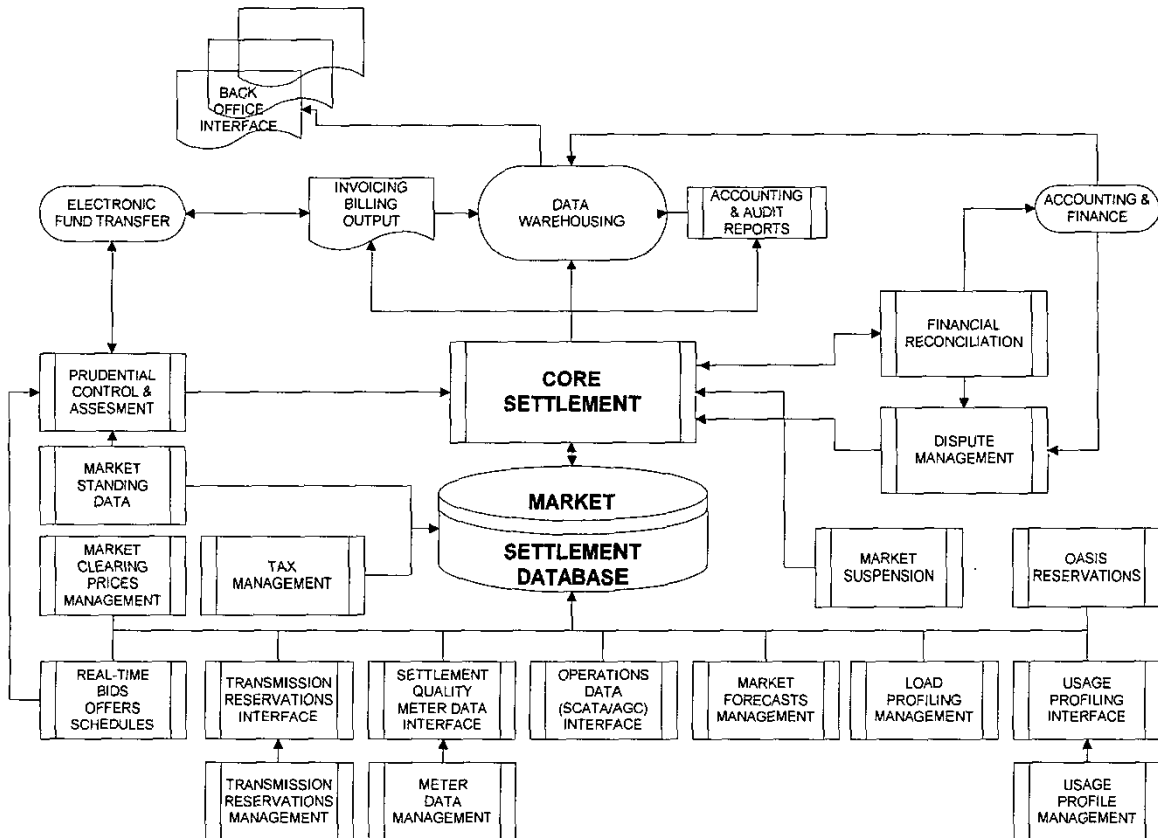


Fig. 1. Settlement Modular Setup.

C. Comparison

The approach taken is a probabilistic black box testing. Black-box testing [1, 2, 3, 4] can be easily automated and involves less processing than white box testing. The reason is that this type of testing does not use information about the program structure, that is, the settlement rule set is assumed to be a "black box". Also, large volumes of test data can be generated automatically.

The test data is generated based on high level specifications [5]. The test behavior is based on the requirement specifications for the particular settlement system to be tested.

III. PROBLEM FORMULATION

The market settlements rules are the most important vehicle to implement price signals. The price signals get communicated to the market participants via a complex system of settlements rules implementing a particular market design. In certain markets, the full set of rules is between fifty to more than a hundred. A particular implementation of the detailed rule set might lead to breaking them into elementary sub-rules which might run into the high hundreds. For example, the Electric Reliability Council of Texas Settlement Protocols describe forty-two rules. The amended California ISO marked design MD02 describes about hundred and eighty settlement business rules.

This system of rules covers the complete set of design requirements needed to financially balance the market, taking as inputs product prices, all traded product volumes, grandfathered contracts, reliability must run (RMR) contracts, various market conditions such as transmission congestion, outages, generation and transmission losses, as well as imports and exports on the boundary interfaces.

To illustrate the magnitude of the problem to be tested, above is a pictorial description of a typical ISO settlement system. Any settlement system is highly modular, with a large number of input and output interfaces and data sources. The data flowing into the Core Settlement is produced by a large number of sources, and the timing varies from near-real time to weekly, monthly or in some cases, up to yearly frequency.

The settlement rule set configuration is contained within the Core Settlement module, which takes inputs from the Market Settlement database. The results are stored in a Settlement Database within the Core Settlement module, and then reported to the internal staff and market participants as Billing and Invoicing output and accounting and audit reports.

The price signals get propagated back to the market participants after being run through the set of settlement rules and depending on the specific market events and conditions the participants are being invoiced for the corresponding debits or credits incurred by them participating on the market. Each settlement rule appears as an invoice line item of the invoice each participant receives for a particular billing period specified by the market protocols.

The correctness of the complex system of rules is crucial to

the successes of the market design. However, to analytically prove this correctness could be a daunting task given the complexity of a particular market implementation. Therefore, a way must be devised to decisively prove that the market implementation is designed to properly telegraph market signals to the participants, which is also economically easy to implement, and can be applied to a wide variety of market designs.

IV. IMPLEMENTATION APPROACH

A. Overall Design

The solution can be implemented first by defining the full set of expected results from the experiment. As such, a set of hypotheses is built, postulating the expected results from the market experiments with respect to the inputs.

The expected results against which the simulated results are compared are based on general guidelines set by the market designers and generally are presented as a written document rule set. Then the rule set is compared against the simulated results and the correctness of the settlement system can be estimated.

In order for that to be done, metrics is defined for deviations of experimental results from the expected hypothesis results. The metrics in this case is the distance of the simulated results from the expected ones. This will allow later for ranking the experiment outcomes and the eventual incremental changes in the marked designs. This will also allow for comparison of different similar runs during any sensitivity analysis conducted.

Subsequently, the inputs that can be changed are defined. These are all inputs to the system of settlement rules that will be generated by the market simulation engine and fed to the settlement system.

As part of the input definition, all input boundaries for the inputs that can be changed have to be defined. This will be set in the market simulation engine that will vary the inputs within these boundaries only.

In addition to the input bounds, all constraints on the input set of data have to be specified. As such, functional constraints must be defined to act on the inputs, since the inputs are generally functionally dependent on each other.

Once the above is completed, an input generation engine that can generate large number of market scenarios based on input boundaries and constraints have to be designed and built. This engine will take the inputs, their bounds, any initial conditions on the inputs and the functional constraints acting on the inputs, and generate multiple sets of probable and viable market scenarios, which will be fed to the settlement system.

An output test engine that calculates the fitness of the result based on the distance of the calculated results from the ones expected have to be designed. The input to the fitness engine will be the outputs of the settlements system, the expected results and a set of metrics rules acting on both inputs. The

output will be the distance of the calculated from the expected results based on certain metrics rules.

Once this is in place, the settlement system will be run on the market rule set with the generated inputs, and look at the calculated results deviation from the expected results. Conclusions to the quantity of the settlement rules for that particular market design then can be drawn. Also other experiments can be constructed, such as input sensitivity, or incremental rule sensitivity changes on the expected marked design behavior.

B. System Implementation

The above design of the implementation approach can be achieved only if the settlement system being tested is properly designed. This is possible if the separate functionalities are clearly abstracted, such as input interface modules, rule set

configuration modules, calculation modules, output interface modules, etc. Thus, the following modules have to be defined as fully abstracted and configurable:

- Business rule sets;
- System parameters;
- Audit trails and archiving;
- Standardized settlement data input;
- Standardized settlement data output;
- Validation rules;
- Settlement calculation and run control;
- Multi-market capability;
- Rule validation and error reporting;

Below is given the test setup needed to implement the settlement system testing in a reusable fashion.

As seen, a data generation configuration based on the market rules in conjunction with a test data generation engine is feeding the settlement system with data. The data output is then stored in a database, which data then is presented to another metrics calculation engine, reading the data, calculating output data metrics and producing test reports. The metrics calculation is supplied with configurable metrics rules, which can be changed flexibly from one implementation to another.

V. RESULTS

Using this method, as a result different market scenarios can be built based on the particular market protocols. By changing the rule generating engine more than one market can be simulated, thus making the approach truly general and re-usable.

We can give an example with a charge type from the ERCOT market, namely, the Out of Merit Down Energy (OOMDE). Below is the charge type rule:

$$\text{OOMDE} = \text{QUANTITY} * \text{PRICE}$$

where

$$\text{QUANTITY} = (-1) * \text{Max}(0, (\text{Min}((\text{RESOURCE_PLAN_GSITETOT}), \text{OOMDEQ}))))$$

and

$$\text{PRICE} = (\text{Max}(0, (\text{MCPER} - \text{RCGFC})))$$

Here

$$\text{OOMDEQ} = \text{Max}(0, (\text{RESOURCE_PLAN_INSTRUCTION}))$$

and

$$\text{RCGFC} = \text{FUEL_INDEX} * \text{HEAT_RATE}.$$

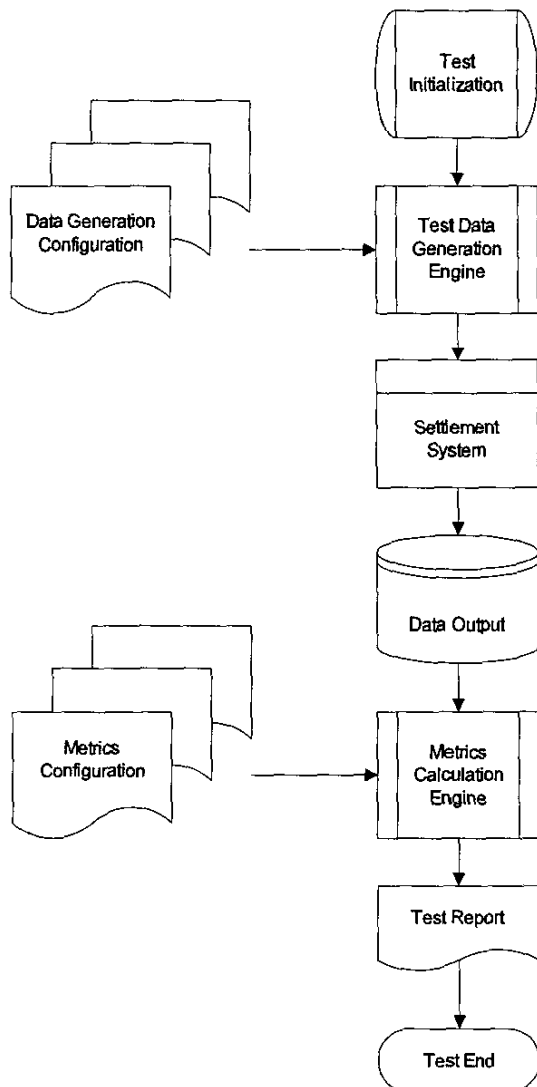


Fig. 2. Test Flow Setup.

Above, RCGFG is the resource category generic fuel cost, which is the product of the fuel index price FUEL_INDEX and the unit heat rate, HEAT_RATE. The OOMDEQ is the out of merit down energy quantity as the maximum of the unit deviation from the resource plan based on the out of merit instruction and zero, whichever is greater. The price is the maximum of the difference between the market clearing price for energy and the resource category generic fuel cost, and zero, whichever is greater. The out of merit charge type quantity QUANTITY takes into account the difference actual metered value with respect to the unit out of merit deployment instruction. The final value, AMOUNT, is simply the product of the quantity and price for that charge type.

As we can see, the following are the inputs to the OOMDE:

The Min and Max column in Table 1 define the range within which the values of the inputs can be changed. The Units column is the Unit of Measurement for a particular input. The testing is done by varying the inputs within the

TABLE I

Input	Description	Min	Max	Units
FUEL_INDEX	Fuel Index Price	9.00	10.00	\$/MMBtu
GSITETOT	Metered Generation	100.00	300.00	MWh
HEAT_RATE	Unit Heat Rate	9.50	9.70	MMBtu/MWh
MCPER	Market Clearing Price	0.00	1000.00	\$/MMBtu
INSTRUCTION	Out of Merit Instruction	10.00	100.00	MWh
RESOURCE_PLAN	Unit Schedule	100.00	300.00	MWh

boundaries and calculating the Out of Merit Down Energy amount.

The expected results against which the simulated results are compared are based on general guidelines set by the market designers and generally are presented as a written document rule set. Then the rule set is compared against the simulated results and the correctness of the settlement system can be estimated.

VI. DISCUSSION

One of the key components for measuring the correctness of the market operational design under uncertain market conditions is to analyze how well market pricing signals create incentives for maintaining an efficient market to achieve the design goals. This can be done by simulating the market uncertainties *a priori* by generating a large number of random market conditions. An advantage of this method is that it does not require a formal proof of the settlement rules and the correctness of the settlement system can be judged by estimating the impact of the monetary incentives on the market participant behavior.

Other advantage of the method can be extending it into an agent base market simulation, where each of the agents is competing based on the a certain rule engine configuration. Thus, also market gaming could be studied, similar to the studies done in Refs [6, 7].

Another benefit of this approach is that sensitivity analysis can be performed on the settlement rule set. Thus, it is possible to estimate the impact of a particular market condition on the full market response, for example, a particular congestion constraint, or a given price on the market outcome. This can be achieved by fixing all inputs but one or more and changing the inputs across the valid input range, *ceteris paribus*. Then this can be repeated for multiple randomly selected fixed set of input rules to gauge the viability of any conclusions drawn from the experiment.

VII. CONCLUSION

It this paper it was shown that one possible approach to validate Settlement Systems design under uncertain market conditions is by using probabilistic techniques for ISO market simulation. In this approach, the implemented set of rules was regarded as a black box. A large number of possible market scenarios generated at random is fed through the market. As a result, this approach does not require a formal proof of the settlement.

It this paper it was demonstrated that this approach is relatively easy to implement and can be generalized to work across multiple market designs regardless of the market rule complexity, since it does not require analytical proof of the correctness of the settlement rules.

REFERENCES

- [1] "Software Reliability Methods", Doron Peled, Bell Labs/Lucent Technologies, Murray Hill, NJ, USA, Springer-Verlag, ISBN: 0-387-95106-7
- [2] B. Beizer. *Software Testing Techniques*. Van Nostrand Reinhold, New York, NY, 2nd edition, 1990.
- [3] G. Myers. *The Art of Software Testing*. John Wiley and Sons, New York, NY, 1979.
- [4] Y. K. Malaiya. "Antirandom testing: Getting the most out of black-box testing". Technical Report 96-129, Computer Science Department, Colorado State University, Fort Collins, CO, 1996.
- [5] C. Meuded, *Automatic Generation of Software Test Cases From Formal Specifications*, PhD Thesis, The Queen's University of Belfast, 1998.
- [6] V. Petrov, C. Richter, G. Sheble, "Gaming Strategies on Electric Power Markets Driven by Agents", *International Journal for System Sciences*, Special Issue, 2000
- [7] V. Petrov, G. B. Sheblé, "Power Auctions Bid Generation With Adaptive Agents Using Genetic Programming", presented on NAPS 2000, University of Waterloo, Canada.