Simulation-Based Validation for Smart Grid Environments

Wonkyu Han Mike Mabey Gail-Joon Ahn
Laboratory of Security Engineering for Future Computing (SEFCOM)
Arizona State University
{whan7,mmabey,gahn}@asu.edu

Abstract

Large and complex systems, such as the Smart Grid, are often best understood through the use of modeling and simulation. In particular, the task of assessing a complex system's risks and testing its tolerance and recovery under various attacks has received considerable attention. However, such tedious tasks still demand a systematic approach to model and evaluate each component in complex systems. In other words, supporting a formal validation and verification without needing to implement the entire system or accessing the existing physical infrastructure is critical since many elements of the Smart Grid are still in the process of becoming standardized for widespread use. In this paper, we describe our simulation-based approach to understanding and examining the behavior of various components of the Smart Grid in the context of verification and validation. To achieve this goal, we adopt the discrete event system specification (DEVS) modeling methodology, which allows generalization and specialization of the entities in the model for a customized simulation with specific scenarios. In addition, we articulate metrics for supporting our simulation-based verification and validation and demonstrate the feasibility and effectiveness of our approach with a real-world use case.

1 Introduction

The Smart Grid is a pervasive new concept intended to provide sophisticated features to the electrical grid, including energy resource sharing, distribution, and load balancing [14, 19, 11, 20]. A wide variety of research has been conducted to determine what technological and risk aspects should be considered in the creation of the Smart Grid, such as smart metering technology [18], information system development [13], future standards, and so on.

As for the future standards, the National Institute of Standards and Technology (NIST) has released their newest framework [7]. In this version, they provide a conceptual model to describe the overall Smart Grid system, and propose eight research areas which should be standardized with high priority. The most significant difference between this release and their previous one (i.e., Release 1.0) is the emphasis on improving interoperability among various distributed systems and reducing the number of threats in the Smart Grid. In addition, there exist several functional and non-functional requirements associated with the Smart Grid. For instance, the Energy Power Research Institute (EPRI) published the Integrated Energy and Communication Systems Architecture (IECSA), which describes many functional requirements and scenarios and is helpful for understanding specific domains of the Smart Grid [6]. The Organization for the Advancement of Structured Information Standards (OASIS) and Zigbee also published additional specifications, Energy Market Information Exchange (EMIX) and Smart Energy profile (SEP) 2.0 respectively, with the goal to develop common object models which can be applied in a Smart Grid system [12, 8]. Also, EPRI releases various use cases (or scenarios) that still need to be verified and validated by scientists and engineers [1].

Even though the common interest of these research groups clearly expresses the growing risks to the Smart Grid, there exist no systematic method to leverage use cases and articulate critical flaws in a dynamic and large-scale system in the Smart Grid. Since it is more difficult to discover vulnerabilities and threats in a large system, a simulation-based verification and validation process is indispensable. Also, a simulation-based approach helps perform verification and validation without requiring considerable time and resources and needing to implement the entire system or accessing the existing physical infrastructure, which would avoid hampering its operations and causing system failures on running systems in the Smart Grid. Also, such an effective approach is critical since many elements of the Smart Grid are still in the process of becoming standardized for widespread use. In this paper, we propose a novel framework to harness the power of simulation in the

[1] As of January 2013, 213 use cases are available at http://smartgrid.epri.com/Repository/Repository.aspx.

IEEE IRI 2013, August 14-16, 2013, San Francisco, California, USA
978-1-4799-1050-2/13/$31.00 ©2013 IEEE

Table 1. Domains in the Smart Grid Conceptual Model [7]

Domain	Actors in the Domain
Customers	The end users of electricity. May also generate, store, and manage the use of energy. Traditionally, three customer types are discussed, each with its own sub-domain: home, commercial/building, and industrial.
Markets	The operators and participants in electricity markets.
Service Provider	The organizations providing services to electrical customers and utility companies.
Operations	The managers of the movement of electricity.
Bulk Generation	The generators of electricity in bulk quantities. May also store energy for later distribution.
Transmission	The carriers of bulk electricity over long distances. May also store and generate electricity.
Distribution	The distributors of electricity to and from customers. May also store and generate electricity.

verification and validation processes for Smart Grid environments. Our framework adopts the discrete event system specification (DEVS) modeling methodology [23, 26]. The DEVS modeling methodology enables to articulate states of each entity so that our framework can easily identify and trace all activities during the simulation.

The rest of this paper is organized as follows. We give an overview of the related work in Section 2 including NIST's conceptual model and validation work. Section 3 describes our framework called Simulation-Based Validation Framework along with DEVS-based model validation. In Section 4, we discuss details of our design and implementation of a specific use case to verify that our framework is capable of performing validations against system models as expected, along with details of our evaluation results. Section 5 concludes the paper.

2 Related Work

This section presents NIST conceptual model and existing validation approaches in the Smart Grid.

2.1 NIST Conceptual model

NIST conceptual model divides the Smart Grid into seven domains [2] and each of which contains various actors and applications. Actors can be physical devices, software programs or organizations which own those devices. Applications are designated tasks performed by actors. Domains are made by actors who have the same objectives and maintaining similar characteristics when they are communicating within the same domain. In the Customer domain, all customers are not just consuming electricity, but managing their energy usage and generating Distributed Energy Resources (DER). Basically Market domain consists of all operators and participants including commercial service providers, energy brokers, and end users. Actors in the Operation domain delivers electricity from generators to end users. The Service Provider domain shares information to corporate with other domains such as Market, Operation, and Customer domains. The organization in Service Provider domain provides energy installation, facility maintenance, billing service and account management. Companies in Bulk Generation domain generate electricity for customers and transfer/distribute energy via Transmission and Distribution domain respectively. During these various domain activities, each domain exchanges numerous information each other to operate their tasks in a Smart Grid system.

2.2 Validation Approaches in the Smart Grid

Various concerns in the Smart Grid have received great attention for several years. One of the leading research groups, called the Cyber Security Working Group (CSWG), made a three year plan (started on April 2011) to develop a standardized framework which consists of examining use cases, evaluating threats, and suggesting countermeasures [4]. In their plan, a use case is first selected for the threat evaluation. Risk assessment is then performed to identify what vulnerabilities the use case is associated with and how they would impact the overall Smart Grid system. From the risk assessment, high-level requirements and mitigation solutions can be specified. After the risk assessment, either a new architecture is developed to prevent the identified risks or existing standards are assessed for possible flaws. These procedures describe how early-stage validation is critical to the next generation of Smart Grid standards. Although the CSWG framework is well-organized, it lacks details on how use cases should be examined and evaluated in their framework.

Another relevant evaluation model is the Electricity Subsector Cybersecurity Capability Maturity Model (ES-C2M2) [5]. This model includes 10 domains and 4 ma-

[2] Ericsson et al. [13] suggested four domains: Generation, Transmission, Distribution and Markets, respectively, which is mostly covered in NIST model.

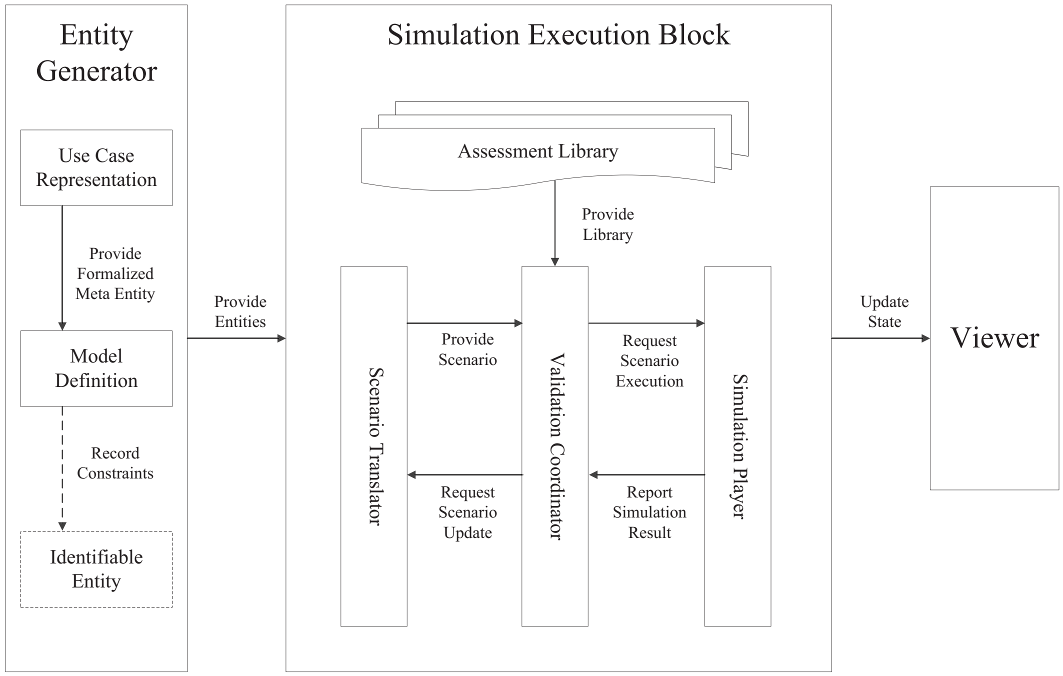

Figure 1. Simulation-Based Validation Framework

turity indicator levels that are used to measure how secure each system is. However, this approach uses its own conceptual model which makes it hard to perform the evaluation tasks in a standardized manner. Also, measuring the indicator level may be subjective. Other researchers have taken different approaches such as agent-based [17, 22], model-based [24, 21], and attack-scenario-based [16]. Even though these approaches demonstrated interesting evaluation results, their work omitted real use cases. To assess assurance of each component in the Smart Grid, it is necessary to have a comprehensive but generic framework for considering real use cases systematically.

3 Simulation-Based Validation Framework

This section describes our framework, called Simulation-Based Validation Framework, which combines the power of simulation with the validation process. As mentioned in Section 2, threats in the Smart Grid continues to gain attention; however, there still lacks a systematic, comprehensive, and repeatable framework with which to validate a wide variety of use cases. To accommodate these goals, our framework consists of three core components: (i) *Entity Generator* initiates a simulation by generating a number of entities described in an existing use case; (ii) *Simulation Execution Block* establishes relations between the entities and executes the assessment based on specified requirements and model definitions; and (iii) *Viewer* displays messages that are exchanged between entities during the state transition. Our systematic process allows for the validation to be repeated. Figure 1 shows how three components cooperate with each other.

Entity Generator The most important role of the *Entity Generator* component is to create entities that are identifiable by the *Simulation Execution Block*. To achieve this goal, the original *Use Case Representation* is modified to be a composition of entities. Each entity is then defined using either a certain expression or formal language and entered into the *Model Definition*, which allows the *Simulation Execution Block* to understand what the entity is. During this process, conditions and constraints, called a *Meta Entity*, are added to the entity in the *Model Definition*. Note that the entity made in the *Entity Generator* is not connected to any entity in this phase. Since identifying entities from the use case and making a formalized *Meta Entity* are labor intensive, it may require load balancing and optimization modules to save a substantial amount of time to complete this process.

Simulation Execution Block Once entities are generated, establishing relations between entities should be carried out first. *Scenario Translator* decides what the next entity is followed by previous one and what messages two entities exchange. As each scenario might not be unique, *Scenario Translator* considers all possible scenarios where a scenario set $S = \{S_1, S_2, ..., S_N\}$. When *Scenario Trans-*

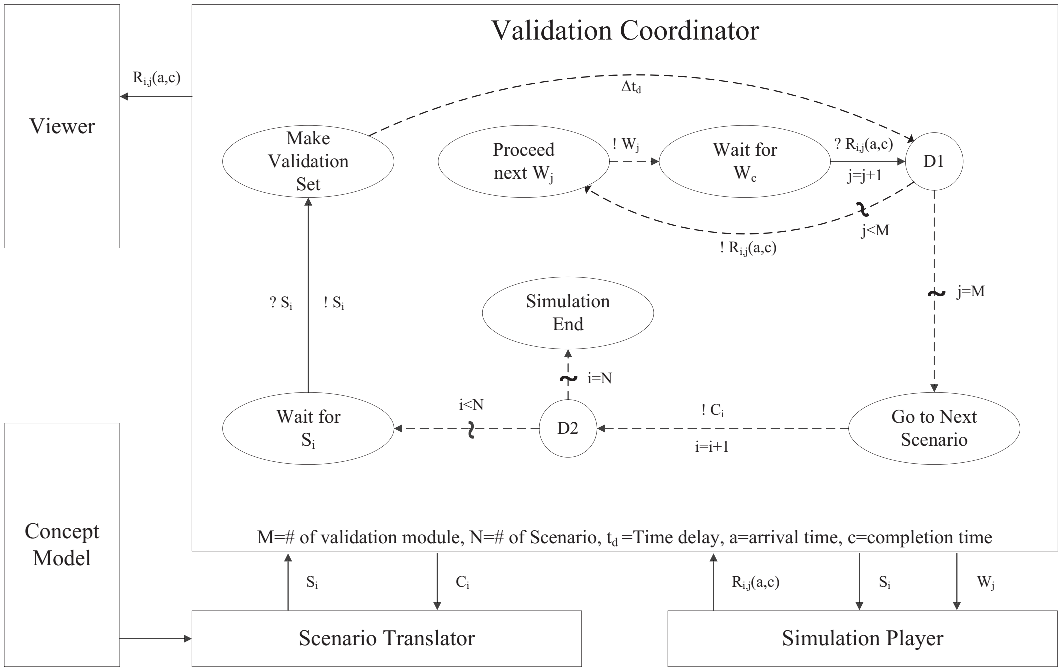

Figure 2. Validation Coordinator using DEVS formalism

lator receives *Request Scenario Update*, *Scenario Translator* provides the next scenario (S_{i+1}). *Validation Coordinator* is a core part in our framework. *Validation Coordinator* searches all possible validation methods in the *Assessment Library* which maintains requirements and testing modules. If *Validation Coordinator* completes to make a validation set which is denoted as $W = \{W_1, W_2, ..., W_M\}$, it sends *Request Scenario Execution* to *Simulation Player*. Once it receives *Report Simulation Result* from *Simulation Player*, the same action is continued until W_M is finished. This is one round of scenario validation. *Simulation Execution Block* repeats again until the last scenario S_N is completed. These automated validation procedure can easily expand by adding another library and allow to evaluate numerous use cases by changing scenarios.

Viewer The *Viewer* enables to monitor what events are occurred and what results have been generated by the simulation. Its functionality is not just displaying the result of simulation but also educating the user what risks are involved and how it can be resolved. Through *Viewer*, the effectiveness and reliability of countermeasures can be evaluated.

Since all the components in our framework cooperate interactively each other, providing an adequate description of these interactions using a single algorithm would be prohibitively difficult. Instead, we adopt the DEVS modeling methodology to describe such dynamic interactions [26]. Figure 2 shows the internal and external structure of the *Validation Coordinator* (VC). When the execution begins, VC stays in the *Wait for* S_i state until a scenario is sent from the *Scenario Translator*. Once the scenario is received, VC sends S_i to the *Simulation Player* and then transits to the *Make Validation Set* state. The *Simulation Player* then generates atomic models and establishes relations between atomic models. In the *Make Validation Set* state, VC obtains a validation set $W = \{W_1, W_2, \ldots, W_M\}$, where $|W| = M$, from the *Assessment Library* with the time delay Δt.

As an initial j value equals to zero, VC moves to the *Proceed next* W_j state. Before moving to the next state, VC sends W_j (jth validation at the ith scenario) to the *Simulation Player*. Then, VC waits until it receives $R_{i,j}(a, c)$ (result of the jth validation at the ith scenario, arrival at time a and completion at time c) at the *Wait for* W_c state. If received, VC updates $j = j+1$ and compares j value to M. If $j < M$, VC repeats the same behavior, otherwise ($j = M$) VC moves to the *Go to Next Scenario*. After comparing i value with N value, if $i < N$ then VC goes another round of simulation, otherwise ($i = N$) VC moves to the *Simulation End*. During simulation, the *Viewer* updates simulation results periodically.

Note that although we adopt the DEVS modeling methodology to describe the VC in this work, any other methodology can be leveraged in our framework.

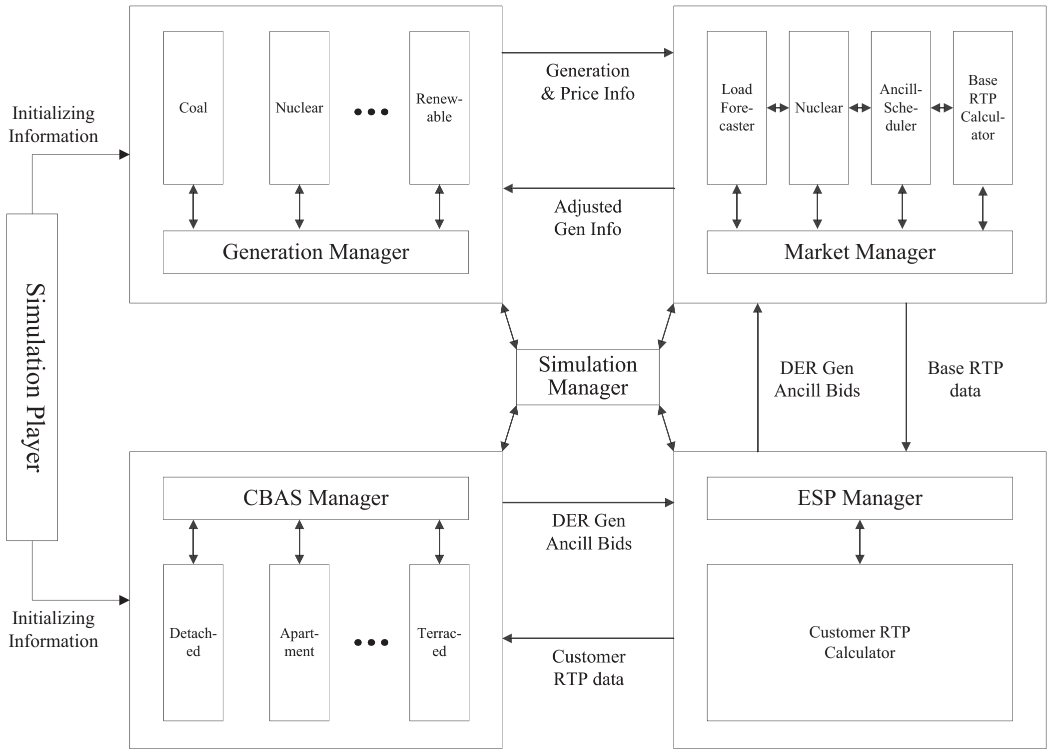

Figure 3. RTP scenario generation

4 Case Study: Implementation Details and Evaluation Results

To demonstrate the feasibility and reliability of our framework, this section starts with a use case from real-time pricing scenario and articulates critical components in this use case. Next, we describe how requirements specified in this use case can be realized in our framework. Also, we elaborate results from our evaluation.

4.1 Requirements for Real-Time Pricing

In the *Real-Time Pricing* (RTP) scenario [6], each of the domain stakeholders correspond with each other to circulate pricing information and exchange their constraints (such as power outage, ancillary services, etc.) through various course of actions. The motivation for RTP arises from the disparity between the amount of electricity generated by power plants and the amount of energy demanded by customers. Ideally, power companies would be able to accurately predict exactly how much demand there would be at any given time, but the reality is that sporadic usage spikes and ebbs may create energy surpluses and shortages all the time, resulting in either wasted energy production or customers that do not have any electric power.

Intuitively, there is a direct relationship between the demand for electricity and its price, increasing during periods of peak usage[3], and decreasing when demand is low, such as during the night. However, Service Providers typically use what is called a *fixed price list* or *fixed tariff*, which does not reflect a fine-grained view of market circumstances. Hence, the RTP approach, which updates prices hourly, provides greatly improved price data and is able to vitalize the energy market. In other words, it would significantly contribute to the fulfillment of business continuity objective that is one of important requirements for critical infrastructures including the Smart Grid. Hourly price calculation models have been proposed by many researchers [9, 25, 19]. For our case study, we selected Allcott's model [10] since this approach formulates accurate price changes according to customers' demand.

In order to identify the target requirements for our case study, we first examine the decision process: *the Bulk Generation company announces the initial raw prices at which it will sell energy in the energy market. After adding transmission and distribution costs, each company finalizes their base prices. At the same time, each energy service provider gathers customers' predicted energy demand and sends an aggregated demand amount to the energy market, where the real time price is calculated.*

[3] Peak usage times may vary for each Energy Service Provider, but are usually from 3PM to 6PM in Arizona [3, 1].

Table 2. Number of exchanged messages

Intra-domain		Inter-domain	
Bulk Generation (BG)	18	Simulation Player ↔ Domains	6
Market Operation (MO)	1,046	BG ↔ MO	19
Energy Service Provider (ESP)	409	MO ↔ ESP	418
Customer Building Automation System (CBAS)	408	ESP ↔ CBAS	800
Total	1,881	Total	1,243

Table 3. Standard deviation comparison

Scenario	Detached		SemiDetached		Apartment		Terraced	
	SRP	APS	SRP	APS	SRP	APS	SRP	APS
$\alpha = 0.1$	0.049313	0.172181	0.077588	0.107786	0.103452	0.057526	0.053144	0.1214
$\alpha = 0.3$	0.02106	0.027133	0.019465	0.02707	0.023043	0.028997	0.016388	0.02779
$\alpha = 0.5$	0.011554	0.015463	0.011423	0.016457	0.008256	0.020822	0.012499	0.016702
Total	0.031302	0.105675	0.046573	0.064329	0.061451	0.03866	0.032567	0.072146

Based on this decision process, we notice that protecting customers' privacy and maintaining price data integrity are essential in RTP. However, since the latter requirement is closely coupled with RTP model, our simulation mainly focuses on how a RTP scenario can be realized in our framework and how our simulation can detect key components involved in the RTP decision process.

4.2 Design and Implementation

To design an RTP use case in our framework, we adopt the conceptual model from NIST. We use only four of the domains by making the assumption that there is zero cost incurred by the Transmission and Distribution domains. In the Bulk Generation domain, there are five types of power plants according to their energy source: coal, natural gas, nuclear, hydro electric, and renewable. Customers' residency styles in the Customer domain (represented in our model by the Customer Building Automation System) can be one of four types: detached, semi-detached, apartment, and terraced [15].

Once electricity is generated, the next step performs load-balancing and pricing for the electricity. The energy scheduler balances total supply and expected demand by mediating between the Bulk Generation entity and the Service Provider entity (equivalent to the Energy Service Provider). The real-time pricing decision is made by the Base RTP Calculator in the Market domain, but prices may fluctuate since customers' energy usage may be affected by the set price. Once the real-time price is calculated, pricing information is delivered to the customers. Figure 3 illustrates these components and its relationships along with the real time price decision process.

Based on our framework, to realize the RTP use case with the DEVS modeling methodology, we utilize a DEVS supporting simulator called MS4 [2]. By adopting the DEVS supporting simulator, we realize the procedures illustrated in Figure 2. One of the advantages of using MS4 is that it provides a simulation viewer, eliminating the need to construct our own specialized viewer. To support our framework, four simulation entities representing the four domains were implemented. State transitions in each entity and message exchanges among entities were analyzed for each step. After the simulator completely executed the use case, we produced simulation statistics. Moreover, result graphs were generated for supporting further analysis [4].

4.3 Simulation Results

To perform a realistic simulation, we considered two Energy Service Providers in the state of Arizona (SRP [3] and APS [1]) and used production information and retail energy prices for their power plants. In addition, we took the customers' daily energy usage behavior available from [15]. By applying real data, simulating RTP use case makes more reliable and meaningful.

Table 2 shows the number of exchanged messages when the number of power plants is 9 and the number of customers is 200 (100 for each ESP). It shows that 56% of all intra-domain messages (1,046/1,881) are exchanged within the Market Operation (MO) domain, which means MO is the key infrastructure to be protected for supporting a fail-safe RTP decision process. Furthermore, 64% of all inter-domain messages (800/1,243) are generated between

[4] The simulation viewer also provides state updates, animated exchanging messages, as well as a mechanism for advancing time.

the ESP and the Customer Building Automation System (CBAS). Hence, the network between the ESP and the CBAS needs to be carefully supervised to prevent potential data leakage.

The next set of results depicts the simulation under different scenarios. Three test scenarios with different RTP ratios (α) were considered with diverse residency types as follows, where $\alpha = \frac{RTP_{users}}{All_{users}}$:

$$S = \{S_{\alpha=0.1}, S_{\alpha=0.3}, S_{\alpha=0.5}\}$$

$$W = \{W_{Detached}, W_{SemiDetached}, W_{Apartment}, W_{Terraced}\}$$

As shown in Table 3, the standard deviation of each residency type is considerably reduced when α value increases. This result shows how radical price changes can be produced when the value of α is small, which can cause severe distrust in the RTP system. In Figures 4 and 5, detached residency type shows how α value impacts overall RTP system's safety. When $\alpha = 0.1$, RTP fluctuation is very huge compared to $\alpha = 0.3$ or $\alpha = 0.5$. It means that RTP can be facing unexpected huge fluctuation when RTP is in its early stage. Moreover, maximum price of SRP is 5.63 times as large as SRP's fixed tariff in Figure 6 and maximum price of APS is 5.15 times as large as APS's fixed tariff in Figure 5. These radical price changes may cause distrust in RTP, hence any countermeasure (e.g., modifying elasticity constant or revision of high price) to mitigate these changes should be considered [5].

Through our case study of RTP, we evaluated the feasibility of our framework. First, our framework provides an easy transformation from design to implementation, which enables us to understand RTP use case without requiring significant effort. Second, our simulation-based approach using DEVS modeling methodology gives us various practical results that can answer the critical requirements of RTP that have not been previously identified and validated.

5 Conclusion

As the Smart Grid system has become more complex, its validation and verification process heavily depends upon realistic use cases, such as new requirements, energy resource relocation, and so on. Moreover, such a critical process is a tedious and difficult task without supporting of appropriate systematic approach. To resolve these problems, we have proposed a Simulation-Based Validation Framework with DEVS modeling methodology. Our framework consists of three core components: Entity Generator, Simulation Execution Block and Viewer. We demonstrated how Entity

[5] Due to the page limit, we mainly address compulsive cases from our evaluation results in this paper.

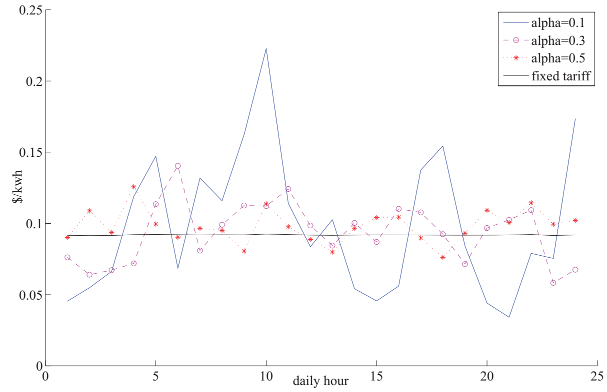

Figure 4. SRP RTP fluctuation (Detached type only)

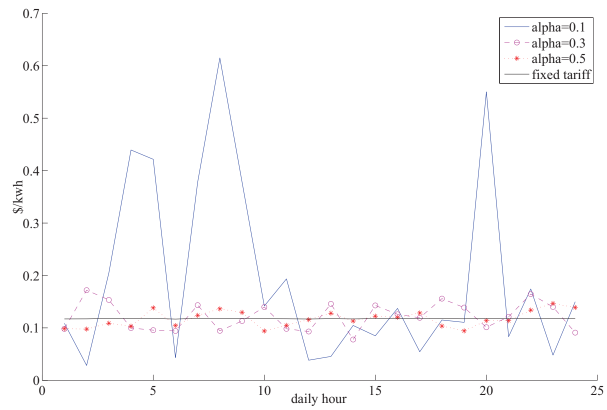

Figure 5. APS RTP fluctuation (Detached type only)

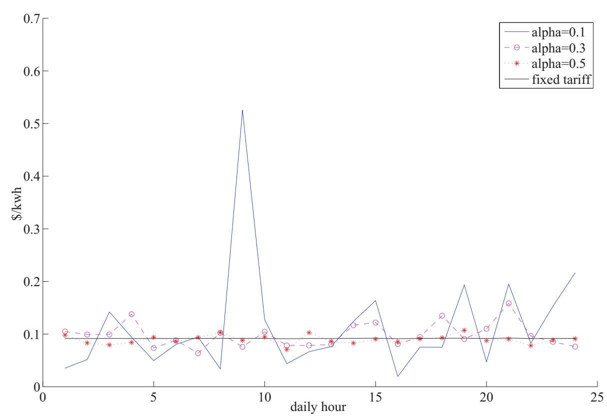

Figure 6. SRP RTP fluctuation (Apartment type only)

Generator could make individual entity as identifiable format, and Simulation Execution Block could generate a number of scenarios (with entities made by Entity Generator) and perform simulation execution while Viewer could update simulation progress.

Also, by performing various simulation experiments on a real time pricing use case, we showed how critical issues in use cases could be simulated and discovered based on the proposed framework. In the future work, we would articulate various requirements with our framework and further enhance our approach to support use case generation and validation intuitively, particularly focusing on security requirements.

Acknowledgments

This work was partially supported by the grants from National Science Foundation and Department of Energy.

References

[1] Aps home page. http://www.aps.com.

[2] Ms4me home page. http://www.ms4systems.com/pages/main.php.

[3] Salt river project (srp) home page. http://www.srpnet.com.

[4] Cybersecurity working group final three-year plan. http://collaborate.nist.gov/twiki-sggrid/bin/view/SmartGrid/CSWGRoadmap, April 2011.

[5] Electricity subsector cybersecurity capability maturity model (es-c2m2). http://energy.gov/oe/services/cybersecurity/electricity-subsector-cybersecurity-capability-maturity-model, May 2012.

[6] Energy power research institute, real-time pricing - top level. http://smartgrid.epri.com/Repository/Repository.aspx, February 2012.

[7] Nist framework and roadmap for smart grid interoperability standards. http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf, February 2012.

[8] Zigbee smart energy 2.0 draft 0.9 public application profile. http://www.zigbee.org/Standards/ZigBeeSmartEnergy/ZigBeeSmartEnergy 20PublicApplicationProfile.aspx, July 2012.

[9] H. Allcott. Real time pricing and electricity markets. http://www-prd-0.gsb.stanford.edu/ facseminars/events/applied_microecon/documents/ame_03_09_allcott.pdf, January 2009.

[10] H. Allcott. Real-time pricing and electricity market design. http://economics.stanford.edu/files/Allcott3_13.pdf, February 2012.

[11] M. Arora, S. Das, and R. Biswas. A de-centralized scheduling and load balancing algorithm for heterogeneous grid environments. In *Parallel Processing Workshops, 2002. Proceedings. International Conference on*, pages 499 – 505, 2002.

[12] W. Cox, D. Holmberg, and D. Sturek. Oasis collaborative energy standards, facilities, and zigbee smart energy. In *Grid-Interop Forum 2011*, 2011.

[13] G. Ericsson. Cyber security and power system communication — essential parts of a smart grid infrastructure. *Power Delivery, IEEE Transactions on*, 25(3):1501 –1507, july 2010.

[14] R. Garcia, J. Contreras, M. van Akkeren, and J. Garcia. A garch forecasting model to predict day-ahead electricity prices. *Power Systems, IEEE Transactions on*, 20(2):867 – 874, may 2005.

[15] S. Ghaemi and G. Brauner. User behavior and patterns of electricity use for energy saving. *Internationale Energiewirtschaftstagung an der TU Wien, IEWT*, 2009.

[16] E. Jonsson and T. Olovsson. A quantitative model of the security intrusion process based on attacker behavior. *Software Engineering, IEEE Transactions on*, 23(4):235 –245, apr 1997.

[17] J. Lin, S. Sedigh, and A. Miller. Modeling cyber-physical systems with semantic agents. In *Computer Software and Applications Conference Workshops (COMPSACW), 2010 IEEE 34th Annual*, pages 13 –18, july 2010.

[18] A. Metke and R. Ekl. Security technology for smart grid networks. *Smart Grid, IEEE Transactions on*, 1(1):99 –107, june 2010.

[19] A.-H. Mohsenian-Rad and A. Leon-Garcia. Optimal residential load control with price prediction in real-time electricity pricing environments. *Smart Grid, IEEE Transactions on*, 1(2):120 –133, sept. 2010.

[20] A. Molderink, V. Bakker, M. Bosman, J. Hurink, and G. Smit. Domestic energy management methodology for optimizing efficiency in smart grids. In *PowerTech, 2009 IEEE Bucharest*, pages 1 –7, 28 2009-july 2 2009.

[21] D. Nicol, W. Sanders, and K. Trivedi. Model-based evaluation: from dependability to security. *Dependable and Secure Computing, IEEE Transactions on*, 1(1):48 – 65, jan.-march 2004.

[22] M. Pipattanasomporn, H. Feroze, and S. Rahman. Multi-agent systems in a distributed smart grid: Design and implementation. In *Power Systems Conference and Exposition, 2009. PSCE '09. IEEE/PES*, pages 1 –8, march 2009.

[23] S. Schutte, S. Scherfke, and M. Troschel. Mosaik: A framework for modular simulation of active components in smart grids. In *Smart Grid Modeling and Simulation (SGMS), 2011 IEEE First International Workshop on*, pages 55–60. IEEE, 2011.

[24] F. Stevens, T. Courtney, S. Singh, A. Agbaria, J. Meyer, W. Sanders, and P. Pal. Model-based validation of an intrusion-tolerant information system. In *Reliable Distributed Systems, 2004. Proceedings of the 23rd IEEE International Symposium on*, pages 184 – 194, oct. 2004.

[25] T. N. Taylor, P. M. Schwarz, and J. E. Cochell. 24/7 hourly response to electricity real-time pricing with up to eight summers of experience. *Journal of Regulatory Economics*, 27:235–262, 2005.

[26] B. Zeigler, H. Praehofer, and T. Kim. *Theory of modeling and simulation: Integrating discrete event and continuous complex dynamic systems*. Academic Press, 2000.