Simulation-Based Validation for Smart Grid Environments: Framework and Experimental Results

Wonkyu Han, Mike Mabey, Gail-Joon Ahn and Tae Sung Kim

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 chapter, 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 the generalization and specialization of entities in the model and supports a customized simulation with specific variables. In addition, we articulate metrics

A preliminary version of this chapter appeared under the title "Simulation-Based Validation for Smart Grid Environments," in Proceedings of the 14th IEEE International Conference on Information Reuse and Integration, San Francisco, USA, August 14–16, 2013. All correspondences should be addressed to Dr. Gail-Joon Ahn at gahn@asu.edu.

W. Han (✉) · M. Mabey · G.-J. Ahn (✉)
Laboratory of Security Engineering for Future Computing (SEFCOM), Arizona State University, Phoenix, AZ, USA
e-mail: whan7@asu.edu

M. Mabey
e-mail: mmabey@asu.edu

G.-J. Ahn
e-mail: gahn@asu.edu

T. S. Kim
Chungbuk National University, Cheongju-si, South Korea
e-mail: kimts@chungbuk.ac.kr

T. Bouabana-Tebibel and S. H. Rubin (eds.), *Integration of Reusable Systems*,
Advances in Intelligent Systems and Computing 263, DOI: 10.1007/978-3-319-04717-1_2,

for supporting our simulation-based verification and validation and demonstrate the feasibility and effectiveness of our approach with a real-world use case.

Keywords Smart grid · Discrete event system specification · Risk assessment · Simulation · Validation

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 [1–4]. A wide variety of research has been conducted to determine what technological aspects and risks should be considered in the creation of the Smart Grid, such as smart metering technology [5], information system development [6], future standards, and so on (Table 1).

As for the future standards, the National Institute of Standards and Technology (NIST) released the final version of their Smart Grid "Framework 2.0" roadmap in February of 2012 [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 [8]. Also, the Organization for the Advancement of Structured Information Standards (OASIS) and Zigbee published the Energy Market Information Exchange (EMIX) and Smart Energy profile (SEP) 2.0, respectively, which are additional specifications with the goal to develop common object models which can be applied in a Smart Grid system [9, 10]. 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 exists 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, needing to implement the entire system, or accessing the existing physical infrastructure, which could hamper its operations

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

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

or cause failures on running systems. 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 chapter, we propose a novel framework to harness the power of simulation in the verification and validation processes for Smart Grid environments. Our framework leverages use case repositories to change its form to identifiable simulation entities and performs automatic validation tasks with corresponding assessment library. We also adopt one of the well-known formal modeling methodology, discrete event system specification (DEVS), to achieve scalability and ease of use. The DEVS modeling methodology allows modelers to articulate the states of each entity so that our framework can easily identify and trace all activities during the simulation.

The rest of this chapter is organized as follows. We give an overview of the related work in Sect. 2 including NIST's conceptual model and validation work. Section 3 describes our framework, called the Simulation-Based Validation Framework, along with DEVS-based model validation. In Sect. 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 chapter and addresses several future directions.

2 Related Work

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

2.1 NIST Conceptual Model

The NIST conceptual model divides the Smart Grid into seven domains,[2] 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 consist of actors who have the same objectives and maintain 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). The Market domain consists of all operators and participants including commercial service providers, energy brokers, and end users. Actors in the Operation domain deliver electricity from generators to end users. The Service Provider domain shares information to cooperate with other domains such as the Market, Operation, and Customer domains. Organizations in the Service Provider domain provide energy installation, facility maintenance, billing services, and account management. Companies in the Bulk Generation domain generate electricity for customers and transmit/distribute energy via the Transmission and Distribution domains, respectively. During these various domain activities, each domain exchanges information with each other to operate their tasks.

2.2 Validation Approaches in the Smart Grid

Various concerns in the Smart Grid have received attention for several years. One of the leading research groups, called the Cyber Security Working Group (CSWG), made a three year plan (beginning April 2011) to develop a standardized framework which consists of examining use cases, evaluating threats, and suggesting countermeasures [11]. 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) [12]. This model includes 10 domains and 4 maturity 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

[2] Ericsson et al. [6] suggested four domains: Generation, Transmission, Distribution and Markets, respectively, which is mostly covered in NIST model.

level may be subjective. Other researchers have taken different approaches such as agent-based [13, 14], model-based [15, 16], and attack-scenario-based [17] evaluation. Even though these approaches demonstrated interesting evaluation results, their work omitted real use cases. To assess the 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 the Simulation-Based Validation Framework, which leverages the benefits of simulation with the validation process.

3.1 Overview

As mentioned in Sect. 2, threats in the Smart Grid continue 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.

3.2 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 *Use Case Representation* module modifies the original use case into a composition of entities, actors, and activities. 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. For example, most use cases utilize UML diagrams to illustrate functional/non-functional features of actors and activities in the entity. In particular, exchanging messages between actors plays a major role to describe relations of actors and their activities.

We noticed that exchanging messages between actors can be a key criterion to make state-based diagrams, which are message-based state derivations that define the number of states with regard to the number of messages. A use case has a

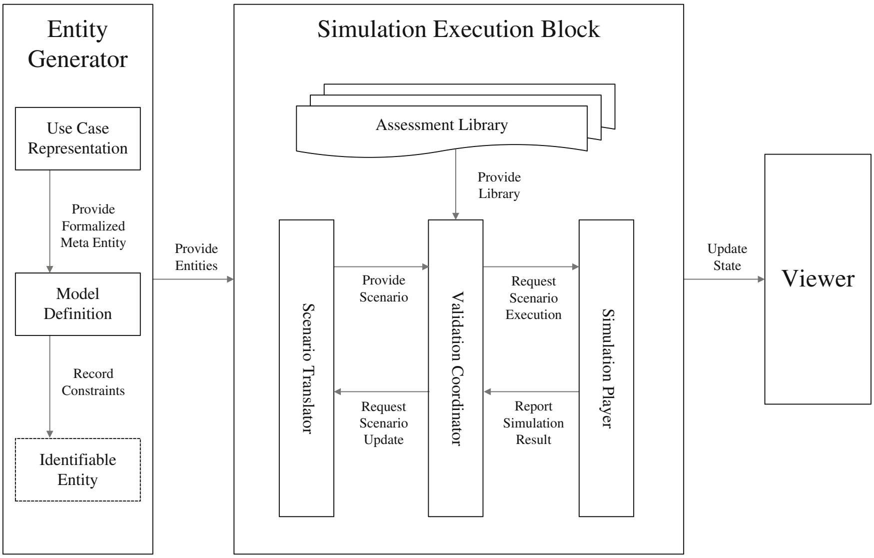

Fig. 1 Simulation-based validation framework

number of incoming messages, which are denoted by $M^I = \{M_1^I, M_2^I, \ldots, M_U^I\}$, and outgoing messages, which are denoted by $M^O = \{M_1^O, M_2^O, \ldots, M_V^O\}$ where $\mid M^I \mid = U$ and $\mid M^O \mid = V$. The entire state set of this entity is defined as $S^E = \{(S_1^E, S_2^E, \ldots, S_i^E, \ldots, S_k^E) \mid S_i^E = (x, y)$, where $x \in M^I$, $y \in M^O$ and $k = U * V\}$. The total number of states in the entire set is $\mid S^E \mid = U * V$, which has too many states because some states may not be used if, according to the use case definition, certain pairs of incoming and outgoing messages cannot be coupled. To accommodate this, we minimize $|S^E|$ by serializing messages which yields the reduced set, denoted by $S^R = M^I \cup M^O$. Note that the controlling logic of the entity is created along with the states during message-based state derivation. With message-based state derivation, most flow charts and sequence diagrams can be translated to the state diagram easily and can be recognized by the *Simulation Execution Block*.

While translating use cases into identifiable entities, conditions and constraints, called *Meta Entities*, are added to the entity in the *Model Definition*. Another item of note is that the entity made by the *Entity Generator* is not connected to any other entities in this phase. Identifying entities from the use case and making a formalized *Meta Entity* are labor intensive, and it may require load balancing and optimization modules to save any substantial amount of time while completing this process.

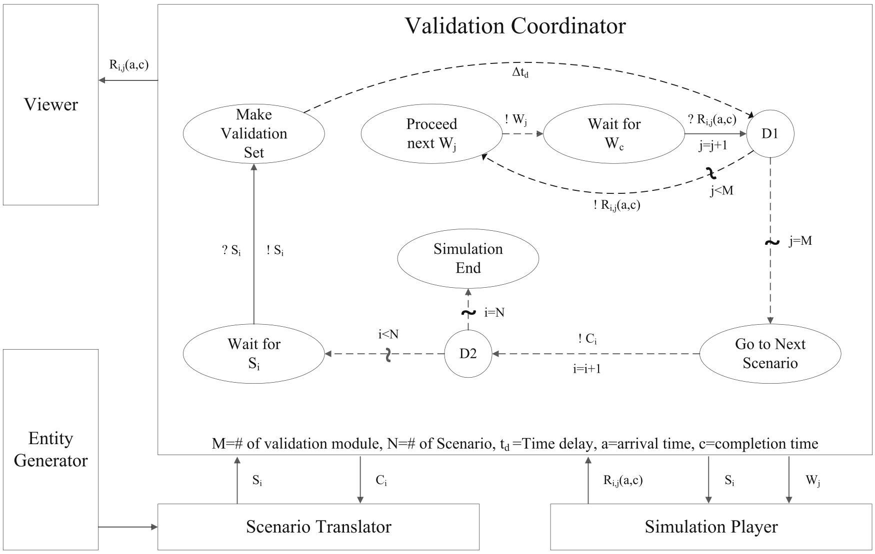

Fig. 2 Validation coordinator using DEVS formalism

3.3 Simulation Execution Block

Once entities are generated, establishing relations between entities should be carried out. The *Scenario Translator* creates these relations by deciding what messages are exchanged between all unique pairs of entities. A scenario is a distinct combination of entities, denoted by $S = \{S_1, S_2, \ldots, S_N\}$. When *Scenario Translator* receives *Request Scenario Update*, *Scenario Translator* provides the next scenario (S_{i+1}). The *Validation Coordinator* is a core element of our framework. It searches all possible validation methods in the *Assessment Library* which maintains requirements and testing modules. The *Validation Coordinator* makes selections from the *Assessment Library* and with those selections creates a validation set, denoted as $W = \{W_1, W_2, \ldots, W_M\}$, and sends a *Request Scenario Execution* message to the *Simulation Player*. Once it receives a *Report Simulation Result* message from the *Simulation Player*, the same action is continued until W_M is finished, which concludes one round of scenario validation. The *Simulation Execution Block* repeats the process again until the last scenario S_N is completed. This automated validation procedure can be easily expanded by adding another library, allowing for the evaluation of numerous use cases by simply changing scenarios.

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 [18]. Figure 2 shows the internal and external structures of the *Validation Coordinator* (VC). When the execution begins, the VC

Algorithm 1: Simulation Execution Block

```
Input: A set of entities, E.
Output: A set of results, R.
/* Simulation Translator: generate set of scenarios */
S ⟵ ScenarioTranslator(E);
foreach s_i ∈ S do
    /* Validation Coordinator: generate set of validation */
    W ⟵ AssessmentLibrary(s);
    foreach w_j ∈ W do
        a ⟵ arrival time;
        /* Simulation Player: run i^th scenario and j^th validation */
        r ⟵ SimulationPlayer(s_i, w_j);
        c ⟵ completion time;
        R.Append(r, a, c);
return R;
```

stays in the *Wait for* S_i state until a scenario is sent from the *Scenario Translator*. Once the scenario is received, the VC sends S_i to the *Simulation Player* and then transitions 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, the 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.

With an initial j value equal to zero, the VC moves to the *Proceed next* W_j state. Before moving to the next state, the VC sends W_j (jth validation at the ith scenario) to the *Simulation Player*. Then, the 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. When received, the VC updates $j \longleftarrow j + 1$ and compares the values of j and M. If $j < M$, the VC repeats the process; otherwise ($j = M$) the VC moves to the *Go to Next Scenario* state. After comparing the values of i and N, if $i < N$ then the VC goes through another round of simulation; otherwise ($i = N$) the VC moves to the *Simulation End* state. During simulation, the *Viewer* updates simulation results periodically. To sum up, the overall *Simulation Execution Block* algorithm is shown in Algorithm 1, including all the sub modules of the *Simulation Execution Block*.

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.

3.4 Viewer

The *Viewer* enables the user to monitor what events occur and what results the simulation generates. Its functionality is not only to display the results of a simulation, but also to educate the user what risks are involved and how they can be resolved. Through the *Viewer*, the effectiveness and reliability of countermeasures can be evaluated.

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 a 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 upon the results from our evaluation.

4.1 Requirements for Real-Time Pricing

In the *Real-Time Pricing* (RTP) scenario detailed in [8], each of the domain stakeholders correspond with each other to circulate pricing information and exchange their constraints, such as power outage, ancillary services, etc. 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 create energy surpluses and shortages all the time, resulting in either wasted energy production or shortages in the amount of electric power delivered to customers.

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 the business continuity objective that is one of the important requirements for critical infrastructure, including the Smart Grid. Hourly price calculation models have been proposed by many researchers [2, 19, 20]. For our case study, we selected Allcott's model [21] since this approach formulates accurate price changes according to customers' demand. The following is a slightly modified RTP calculation equation.[4] Based on Allcott's model, we additionally introduce a distributed energy resource (DER) factor, $\sum_i d_{it}$, in Eq. 1 and an ancillary service cost, P_a, in Eq. 2.

$$Q_t^s(P_t) = \sum_j k_{jt} + \sum_i d_{it} \tag{1}$$

---

[3] Peak usage times may vary for each Energy Service Provider, but are generally weekday afternoons from 2 pm to 6 pm in Arizona. The relevant reference is available at http://www.azenergy.gov/SavingTips/TimeOfUse.aspx.

[4] $\alpha = \frac{RTP_{users}}{All_{users}}$, P_t = real-time price, $\bar{P}$ = fixed tariff price, P_c = capacity market cost, P_a = ancillary service cost, η = elasticity of demand variable, ϵ_t =error fixing variable.

$$Q_t^d(P_t, \bar{P}, P_c) = \{\alpha(P_t + P_c + P_a)^\eta + (1-\alpha)(\bar{P} + P_c + P_a)^\eta\} \cdot \epsilon_t \quad (2)$$

$$Q_t^s(P_t) = (1+m)Q_t^d(P_t, \bar{P}, P_c) \quad (3)$$

Equation 1 is the total generation function that sums power plants' generation and DER generation where t is a specific time period, j is a power plant instance, and i is a customer. Equation 2 defines the expected customer's demand. Allcott used three kinds of prices which are P_t, $\bar{P}$ and P_c and we added one more price factor P_a. Equation 3 shows the equilibrium equation which determines the real-time price. For this calculation, we adopt a reserve margin index m which can be obtained at [22].

In order to identify the target requirements for our case study, we first provide a summary of 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 price. At the same time, each energy service provider gathers customers' estimated energy demand and sends an aggregated demand amount to the energy market, where the real-time price is calculated.

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 an 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 [23]. Data types of our case study are shown in Table 2.

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.

Table 2 Input data types

CBAS domain		Bulk generation domain	
Variable name	Data type	Variable name	Data type
CustomerID	int	BulkID	int
CustomerType	String	BulkType	String
Period	int	BulkCapacity	double
CorrespondingESP	int	BulkDestination	double
EnergyDemand	double[]	Period	int
DERCapacity	double	LoadElectricity	double
DERLoad	double	RawPrice	double
AncillaryService	String	Constraints	String
Constraints	String		

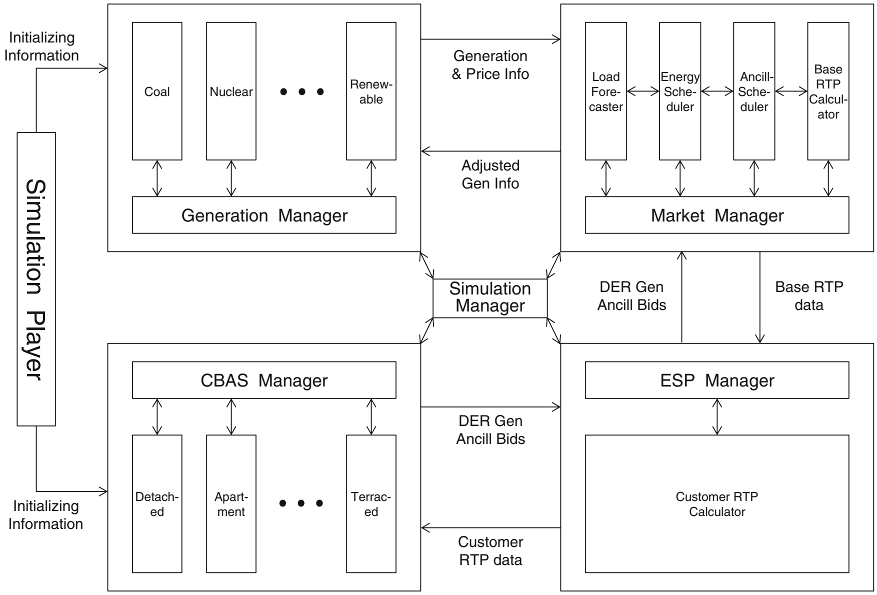

Fig. 3 RTP scenario generation

Based on our framework, to realize the RTP use case with the DEVS modeling methodology, we utilize a DEVS supporting simulator called MS4.[5] By adopting the DEVS supporting simulator, we realize the procedures illustrated in Fig. 2. One of the advantages of using MS4 is that it provides a simulation viewer, eliminating the need to construct our own specialized viewer.

As shown in Fig. 4, the overall appearance is quite similar to the RTP design. To support our framework, four simulation entities representing the four domains

[5] MS4 software is available at http://www.ms4systems.com/pages/ms4me.php.

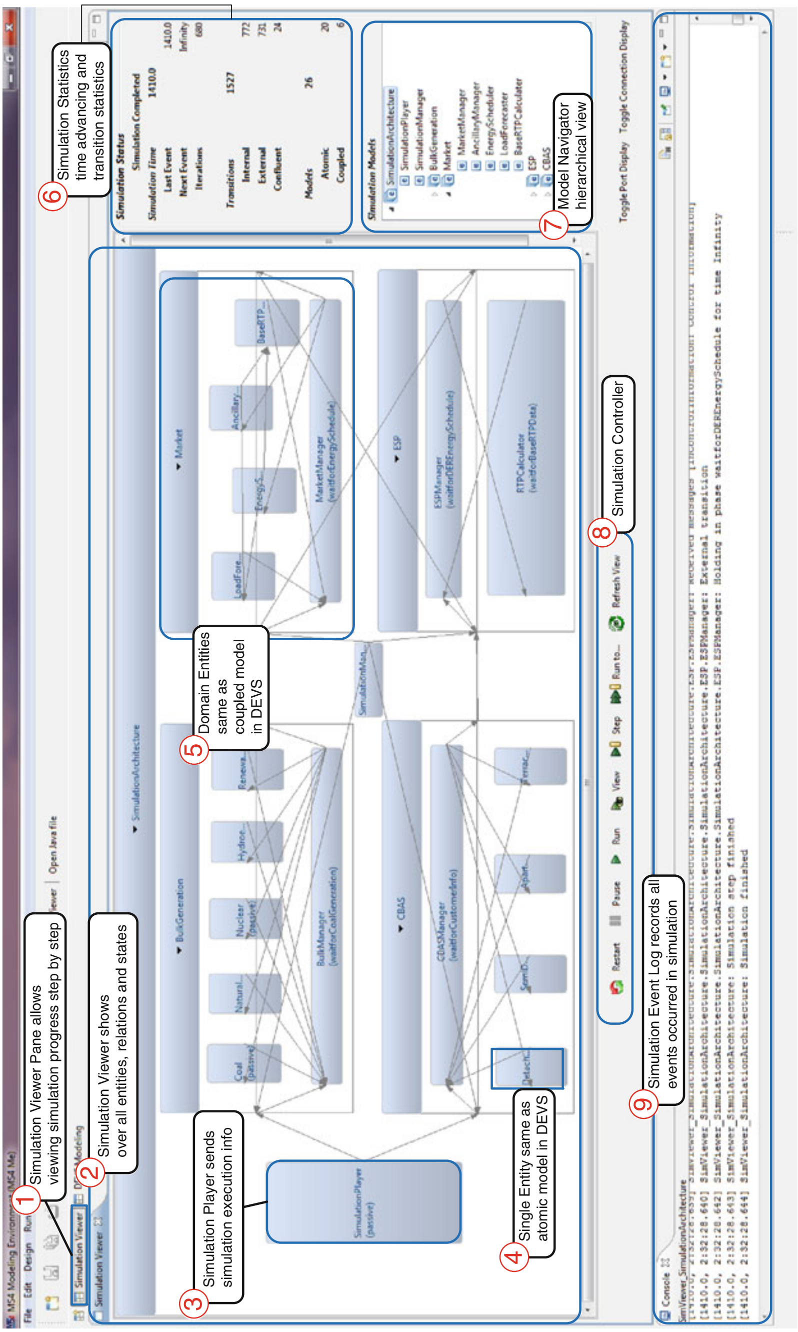

Fig. 4 Simulation entities in MS4 simulator

were implemented. State transitions in each entity and message exchanges among entities were analyzed for each step (see simulation controller). After the simulator completely executed the use case, we produced simulation statistics. Moreover, result graphs were generated for further analyzing the simulation results.[6]

4.3 Simulation Results

To perform a realistic simulation, we considered two Energy Service Providers in the state of Arizona (SRP and APS)[7] and used production information and retail energy prices for their power plants. In addition, we took the customers' daily energy usage behavior available from [23]. By applying real-world data, simulating an RTP use case is more reliable and meaningful.

Table 3 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 protect for supporting a reliable RTP decision process. Furthermore, 64 % of all inter-domain messages (800/1,243) are generated between 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. The RTP ratio, denoted by α, represents the percentage of customers that have elected to use the RTP model for their service. Three test scenarios with different values of α were considered with diverse residency types as follows.

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

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

As shown in Table 4, the standard deviation of each residency type is considerably reduced when the value of α 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 Figs. 5 and 6, detached residency type shows how α value impacts overall RTP system's safety. When $\alpha = 0.1$, RTP fluctuation is huge compared to $\alpha = 0.3$ or $\alpha = 0.5$. This means that RTP may face unexpected, huge fluctuations when RTP is in its early stages of customer adoption. Moreover, SRP's maximum real-time price is 5.63 times its fixed tariff price, as shown in Fig. 7, and APS's maximum real-time price is 5.15 times its fixed tariff price, as shown in Fig. 6.

[6] The simulation viewer also provides state updates, message exchange animations, as well as a mechanism for advancing time.

[7] The information of each energy service provider is available at https://www.srpnet.com and http://www.aps.com/en/residential/Pages/home.aspx, respectively.

Table 3 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 4 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

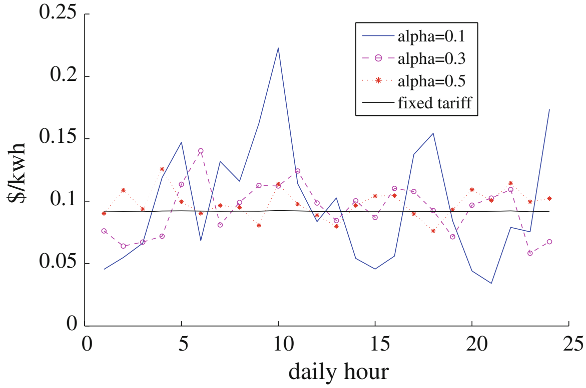

Fig. 5 SRP RTP fluctuation (detached type only)

These radical price changes may cause further distrust in RTP, hence countermeasures to mitigate these changes, such as modifying the elasticity constant or revising high prices, should be considered.[8]

We conducted another experiment to test the elasticity of price. As presented in Eq. 2, η is a crucial factor which determines the fluctuation of real-time prices. Typically η is between -0.04 and -0.15 [20, 24–26]. In order to test which η value is valid for our case study, we changed η value at intervals of 0.04 and measured the deviation of prices by taking $|\text{PRICE}_{RTP} - \text{PRICE}_{Fixed}|$. As shown in Fig. 8, the result at $\eta = -0.12$ deviates much less than $\eta = -0.04$ and $\eta = -0.08$. Due to the nature of Eq. 2 (expected customer's demand), we can reduce RTP fluctuation by

[8] In order to reduce redundancy, we mainly address compulsive cases from our evaluation results in this chapter.

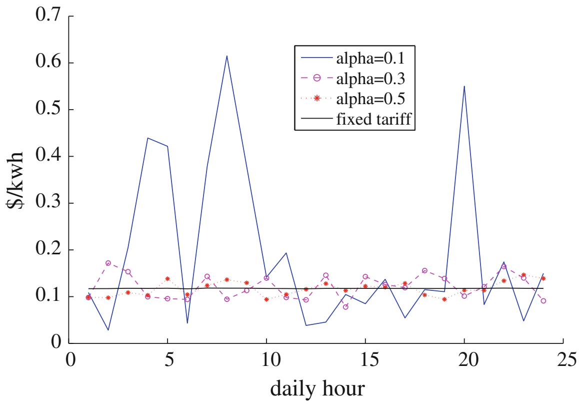

Fig. 6 APS RTP fluctuation (detached type only)

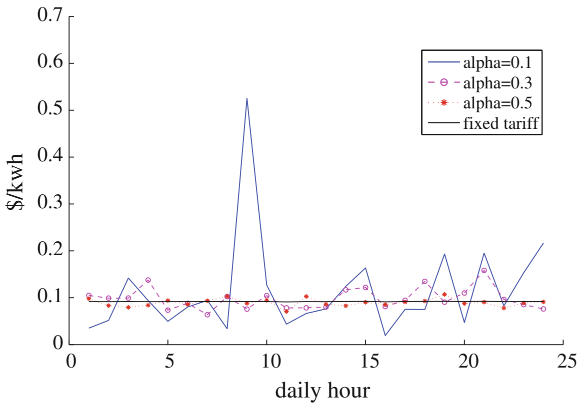

Fig. 7 SRP RTP fluctuation (apartment type only)

setting a small value for η when making a decision on a real-time price. However, finding an appropriate fluctuation level is our goal for RTP simulation, instead of mainly reducing RTP fluctuation. Hence, we measured a break-even η value that satisfies the same gross sales amount of electricity between two cases: under fixed-tariff scenario and under RTP scenario. This helps us understand what price is acceptable for both fixed tariff users and RTP users. We assumed that customers' demand has not been changed by the real-time price. We found that when $\eta = -0.083$, gross sales have the same value for all scenarios. This result leads us to determine that an appropriate η value is also important for RTP, and improper values of η may cause distrust of the system because of the large fluctuations in the simulation results.

Through our case study of RTP, we evaluated the feasibility of our framework. First, our framework provides an easy transformation from entity design to simulation execution, which enables us to understand RTP use case without requiring

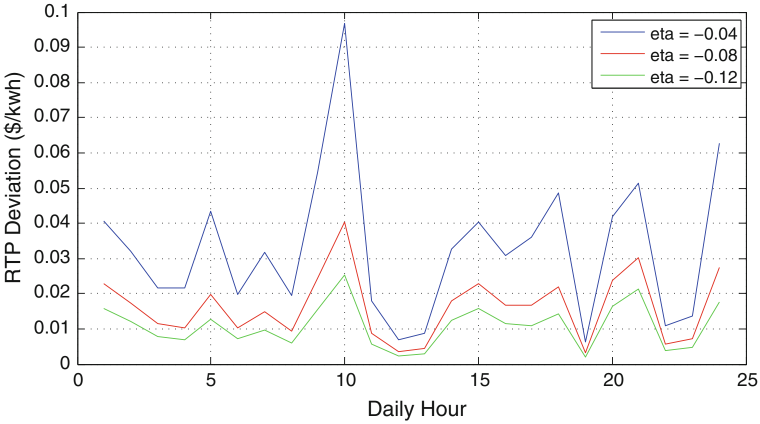

Fig. 8 Changes in elasticity of price (ESP = SRP, Type = Detached, $\alpha = 0.3$)

significant effort. Second, our simulation-based approach using the DEVS modeling methodology gives us various practical results that can answer the critical requirements of RTP that have not been previously identified or validated.

5 Conclusion and Future Directions

As the Smart Grid system has become more complex, its validation and verification process depends heavily 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 the support of an appropriate systematic approach. To resolve these problems, we have proposed the Simulation-Based Validation Framework using the DEVS modeling methodology. Our framework consists of three core components: Entity Generator, Simulation Execution Block and Viewer. We demonstrated how the Entity Generator can create individual entities in an identifiable format, and the Simulation Execution Block can generate a number of scenarios (with entities made by the Entity Generator) and execute the simulation while the Viewer provides updates on simulation's progress. Also, by performing various simulation experiments on a real-time pricing use case, we showed how critical issues in use cases can be simulated and discovered based on the proposed framework.

In our future work, we will articulate various requirements with our framework and further enhance our approach to support use case generation and validation intuitively, particularly focusing on security requirements. In particular, the following areas can be further studied:

Business intelligence Simulation-based approaches utilize various kinds of use cases to search for and find any possible risks that can impact running systems. As James [27] claimed that "acknowledging the business impact of cyber,…leveraging timely business intelligence,…[and] broaden[ing] awareness" are crucial to establishing resilient cyber systems, such characteristics of simulation-based approach can be also extended to the area of business intelligence. For instance, our framework can provide useful data for producing various scenarios that need to be investigated. Moreover, predictive analysis can be performed by our framework. Watson et al. [28] introduced the importance of a business intelligence system that fosters "the use of information and analytics." As shown in Sect. 4.3, forecasting variables of interest and testing scenarios with different conditions in our framework would be tremendously helpful to make business decisions in an effective manner.

Risk management As our RTP use case illustrates the impact of economical issues and the price of electricity on Smart Grid systems, our framework can be further extended to assess potential risks in large-scale distributed systems. Varaiya et al. [29] pointed out that the price of electricity is the biggest risk with respect to the economic challenges of Smart Grid systems. Chao [30] even claimed that fixed uniform price policies remain a considerable barrier and may prevent the success of the Smart Grid. Although RTP is an essential factor to realize the Smart Grid, our study showed that fluctuations of RTP should be restricted due to potential risks and it is necessary to mitigate such fluctuations to a manageable and acceptable risk level for preventing customers from evading the use of RTP systems. Consequently, our approach would help discover potential risks and evaluate diverse mitigation methods to minimize potential risks. In addition, our approach can be further extended to support other domains by articulating uses cases for those target domains.

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

References

1. Garcia, R.C., Contreras, J., van Akkeren, M., Garcia, J.B.C.: A garch forecasting model to predict day-ahead electricity prices. IEEE Trans. Power Syst. **20**(2), 867–874 (2005)
2. Mohsenian-Rad, A.-H., Leon-Garcia, A.: Optimal residential load control with price prediction in real-time electricity pricing environments. IEEE Trans. Smart Grid **1**(2), 120–133 (2010)
3. Arora, M., Das, S.K., Biswas, R.: A de-centralized scheduling and load balancing algorithm for heterogeneous grid environments. In: Proceedings of the International Conference on Parallel Processing Workshops (ICPPW'02), pp. 499–505 (2002)
4. Molderink, A., Bakker, V., Bosman, M.G.C., Hurink, J.L., Smit, G.J.M.: Domestic energy management methodology for optimizing efficiency in smart grids. In: Proceedings of the IEEE Bucharest PowerTech 2009, 1–7 July 2009
5. Metke, A.R., Ekl, R.L.: Security technology for smart grid networks. IEEE Trans. Smart Grid **1**(1), 99–107 (2010)
6. Ericsson, G.N.: Cyber security and power system communication—essential parts of a smart grid infrastructure. IEEE Trans. Power Delivery **25**(3), 1501–1507 (2010)
7. Nist framework and roadmap for smart grid interoperability standards. http://www.nist.gov/public_affairs/releases/upload/smartgrid_interoperability_final.pdf (2012). Accessed Feb 2012
8. Energy power research institute, real-time pricing—top level. http://smartgrid.epri.com/Repository/Repository.aspx (2012). Accessed Feb 2012

9. Cox, W., Holmberg, D., Sturek, D.: Oasis collaborative energy standards, facilities, and zigbee smart energy. In: Grid-Interop Forum 2011 (2011)
10. Zigbee smart energy 2.0 draft 0.9 public application profile. http://www.zigbee.org/Standards/ZigBeeSmartEnergy/ZigBeeSmartEnergy20PublicApplicationProfile.aspx (2012). Accessed July 2012
11. Cybersecurity working group final three-year plan. http://collaborate.nist.gov/twikisggrid/bin/view/SmartGrid/CSWGRoadmap (2011). Accessed Apr 2011
12. Electricity subsector cybersecurity capability maturity model (es-c2m2). http://energy.gov/oe/services/cybersecurity/electricity-subsector-cybersecurity-capability-maturity-model (2012). Accessed May 2012
13. Lin, J., Sedigh, S., Miller, A.: Modeling cyber-physical systems with semantic agents. In: Proceedings of the IEEE 34th Annual Computer Software and Applications Conference Workshops (COMPSACW) 2010, 13–18 July 2010
14. Pipattanasomporn, M., Feroze, H., Rahman, S.: Multi-agent systems in a distributed smart grid: design and implementation. In: IEEE/PES Power Systems Conference and Exposition (PSCE'09), pp. 1–8 Mar 2009
15. Stevens, F., Courtney, T. Singh, S., Agbaria, A., Meyer, J.R., Sanders, W.H., Pal, P.: Model-based validation of an intrusion-tolerant information system. In: Proceedings of the 23rd IEEE International Symposium on Reliable Distributed Systems (SRDS'04), pp. 184–194 Oct 2004
16. Nicol, D.M., Sanders, W.H., Trivedi, K.S.: Model-based evaluation: from dependability to security. IEEE Trans. Dependable Secure Comput. **1**(1), 48–65 (2004)
17. Jonsson, E., Olovsson, T.: A quantitative model of the security intrusion process based on attacker behavior. IEEE Trans. Softw. Eng. **23**(4), 235–245 (1997)
18. Zeigler, B.P., Praehofer, H., Kim, T.G.: Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems. Academic Press, San Diego (Feb 2000)
19. Allcott, H.: Real time pricing and electricity markets. http://www-prd-0.gsb.stanford.edu/facseminars/events/applied_microecon/documents/ame_03_09_allcott.pdf (2009). Accessed Jan 2009
20. Taylor, Thomas N., Schwarz, Peter M., Cochell, James E.: 24/7 hourly response to electricity real-time pricing with up to eight summers of experience. J. Regul. Econ. **27**, 235–262 (2005)
21. Allcott, H.: Real-time pricing and electricity market design. https://files.nyu.edu/ha32/public/research/Allcott-Real-TimePricingandElectricityMarketDesign.pdf (2013). Accessed Mar 2013
22. Levelized cost of new generation resources in the annual energy outlook 2013. http://www.eia.gov/forecasts/aeo/electricity_generation.cfm (2013). Accessed Jan 2013
23. Ghaemi, S., Brauner, G.: User behavior and patterns of electricity use for energy saving. Internationale Energiewirtschaftstagung an der TU Wien, IEWT (2009)
24. Patrick, R.H., Wolak, F.A.: Estimating the customer-level demand for electricity under real-time market prices. Technical report, National Bureau of Economic Research, Washington (Apr 2001)
25. Herriges, J.A., Baladi, S.M., Caves, D.W., Neenan, B.F.: The response of industrial customers to electric rates based upon dynamic marginal costs. Rev. Econ. Stat. **75**(3), 446–454 (1993)
26. Boisvert, R.N., Cappers, P., Goldman, C., Neenan, B., Hopper, N.: Customer response to rtp in competitive markets: a study of niagara mohawk's standard offer tariff. Energy J. **28**(1), 53–74 (2007)
27. White, J.: 12 steps toward cyber resilience. InfoSecurity Professional INSIGHTS 2(2). https://www.isc2.org/infosecurity-professional-insights-archives.aspx?terms=12-Steps-toward-Cyber-Resilience (2013)
28. Watson, H.J., Wixom, B.H.: The current state of business intelligence. Computer **40**(9), 96–99 (2007)
29. Varaiya, P.P., Wu, F.F., Bialek, J.W.: Smart operation of smart grid: risk-limiting dispatch. Proc. IEEE **99**(1), 40–57 (2011)
30. Chao, Hung-po: Price-responsive demand management for a smart grid world. Electr. J. **23**(1), 7–20 (2010)

http://www.springer.com/978-3-319-04716-4

Integration of Reusable Systems
Bouabana-Tebibel, T.; Rubin, S.H. (Eds.)
2014, X, 342 p. 121 illus., Softcover
ISBN: 978-3-319-04716-4