energies

Article

On Conceptual Structuration and Coupling Methods of Co-Simulation Frameworks in Cyber-Physical Energy System Validation

Van Hoa Nguyen [1,*], Yvon Besanger [1], Quoc Tuan Tran [2] and Tung Lam Nguyen [1,2]

[1] Univ. Grenoble Alpes, CNRS, Grenoble INP, G2Elab, F-38000 Grenoble, France; Yvon.besanger@g2elab.grenoble-inp.fr (Y.B.); tung-lam.nguyen@g2elab.grenoble-inp.fr (T.L.N.)

[2] Alternative Energies and Atomic Energy Commission (CEA), National Institute for Solar Energy (INES), 50 Avenue du Lac Léman, F-73375 Le Bourget-du-Lac, France; Quoctuan.tran@cea.fr

* Correspondence: van-hoa.nguyen@grenoble-inp.fr; Tel.: +33-4-7692-7187

Received: 31 October 2017; Accepted: 22 November 2017; Published: 29 November 2017

Abstract: Co-simulation is an emerging method for cyber-physical energy system (CPES) assessment and validation. Combining simulators of different domains into a joint experiment, co-simulation provides a holistic framework to consider the whole CPES at system level. In this paper, we present a systematic structuration of co-simulation based on a conceptual point of view. A co-simulation framework is then considered in its conceptual, semantic, syntactic, dynamic and technical layers. Coupling methods are investigated and classified according to these layers. This paper would serve as a solid theoretical base for specification of future applications of co-simulation and selection of coupling methods in CPES assessment and validation.

Keywords: cyber-physical energy system; co-simulation; conceptual structuration; coupling method

1. Introduction

Moving towards a decarbonized scenario, the power grid is expecting a high penetration of distributed and renewable energy resources and advanced Information and Communication Technologies (ICT) [1,2], which has a strong impact on the system architecture and is transforming the classical grid into a cyber-physical energy system (CPES)—Smart Grid [3]. The traditional design and validation methods, which focus in single domain, do not quite keep up with the changes [4,5]. On the other hand, emerging issues such as cyber-security requires also new tools and methods for assessment. It is, therefore, necessary to develop an integrated approach for such complex system in a holistic manner, taking into account the interaction and inter-dependencies among domains [6].

Power systems and communication networks, however, are very different in term of dynamic behavior and hierarchy. The simulation of these systems require therefore different model of computation and solvers (i.e., a power system is often simulated as continuous system with capability of generating discrete events; ICT system is in general simulated as discrete event simulation) [7]. A holistic approach requires essentially a consistent semantics for specification of the complete CPES, across multiple domains, which is rich enough to support heterogeneous design. While such a unified approach is not easy to achieve, researchers often employ the co-simulation approach: creating joint simulation of the already well-established tools and semantics; to consider the impact of ICT solutions to the power system when they are simulated with their suitable solvers [8]. Co-simulation offers a flexible solution which allows consideration of network behavior and physical energy system state at the same time. As the calculation load is shared among simulators, co-simulation also enables the possibility of large scale system assessment.

Moreover, the CPES requires the physical infrastructure and computational cyber-infrastructure to holistically and consistently coordinate to ensure its efficient and reliable functionality. It introduces a huge data influx for which big data analytic and applications are therefore of paramount importance [9]. Co-simulation framework, gathering data and analysis from multiple sources and domains, is potentially the solution to experiment such big data applications over large scale systems. On the other hand, distributed simulation with appropriate time scale, besides the benefit of computational load sharing, will avoid overloading slower simulation with unnecessary data influx from faster applications, and thus increases the performance.

Recent developments of co-simulation have led to an important portfolio of experimental and demonstration platforms. Some well-detailed, but application-oriented reviews on co-simulation frameworks [4,5,8,10], in particular, and test-bed for CPES validation in general [11] exist in the literature. In the context of this paper, we aim to providing a more systematic point of view from a higher abstraction level by offering a conceptual structuration of co-simulation framework in CPES assessment as well as the associated problems and a detailed review on coupling methods used and can potentially be used in this framework. It would serve as a solid theoretical base for specification of future applications of co-simulation and selection of coupling methods in CPES assessment and validation.

The paper is organized as follows: Section 2 presents the structuration and the different layers of a co-simulation framework in CPES context; Section 3 provides a systematic review on coupling methods according to the different layers, with a particular emphasis on usage and requirement of operational integration and formal integration, in the vision of helping users to establish their own co-simulation framework. The applications and associated abstraction levels of the coupling methods are finally considered in Section 4, along with some further perspectives.

2. Conceptual Structuration of Co-Simulation Framework

Establishing a standardizable holistic framework for CPES using co-simulation is a difficult and complex task because it requires a strong interoperability among the participating elements, especially in case of multiple partner involvement. This implies necessary efforts on harmonization, adaptation and eventually changes of actual employed standards and protocols in individual models to be able to integrate into holistic experiments.

Coupling different simulators introduces several new issues with respect to the classical modelling and simulation approaches. A generic layered structuration of co-simulation framework is therefore necessary to improve the interoperability of simulators as well as to highlight the intersection of domains and the issues that need to be solved in the process of designing a co-simulation framework. Based on existing models for generic interoperability [12] and multi-modeling [13,14], a generic five layers structuration of a co-simulation framework can be proposed (cf. Table 1).

2.1. Conceptual Layer: Architecture

As aforementioned in Table 1, the conceptual layer is highest layer of a co-simulation framework. It involves the meta-modeling process and the topology of the framework (e.g., in Figure 1 the CPES can be considered in a system of systems approach or the in layered approach: market-driven ICT network governing the power system). In this layer, the chosen scenario is analyzed around the three main points: system configuration according to the scenario, purpose of investigation (deduced from the scenario and the desired research/contribution) and use-case/test-case definition; as the scenario and the system configuration influences strongly the conceptual design. Moreover, the questions of abstraction level, interaction to environment and complex system representation manner (e.g., System of systems, coupled systems or multi-agent system) are subjects of interest in this layers.

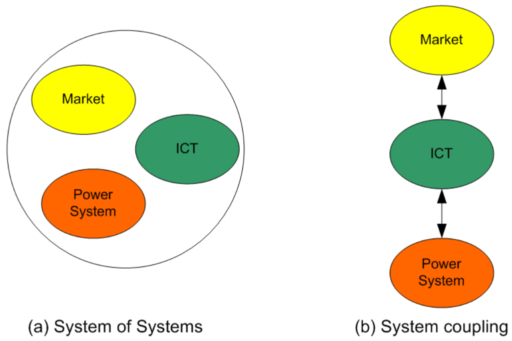

Figure 1. Illustration of different possible topologies in conceptual layer of a CPES co-simulation framework.

Table 1. Generic layers of a co-simulation framework.

Layer	Description	Associated Problems
Conceptual	Highest level where the models are considered as "black boxes" and the level concerns the co-simulation framework representation.	Generic structure of the framework; Meta-Modeling of the components.
Semantic	The level concerns the signification and the role of the co-simulation framework with respect to the open questions of the investigated CPES and studied phenomenon.	Signification of individual models; Interaction graph among the models; Signification of each interaction.
Syntactic	The level concerns the formalization of the co-simulation framework.	Formalization of individual models in the respective domains; Specification and handling the difference among the formalism.
Dynamic	The level concerns the execution of the co-simulation framework, the synchronization techniques and harmonization of different models of computation.	Order of execution and causality of models; Harmonization of different models of computation; Resolution for potential conflict in simultaneity of actions.
Technical	The level concerns the implementation details and evaluation of simulation.	Distributed or centralized implementation; Robustness of the simulation; Reliability and efficiency of the simulation.

The conceptual layer should define the structure on which the co-simulation framework is developed and the formal semantic relations/syntactic formulation are integrated.

2.2. Semantic Layer: Formal integration

The semantic layer concerns the signification and role of individual models in the general framework, as well as their in-between interactions. It is necessary to note that the models may be represented at different spatial and temporal scales, with possible intersection among their abstraction domains. In that boundary, one needs to consider and specify how information in a model can be perceived in another one (cf. Figure 2). The interaction among models has to provide a semantic coherence throughout the whole system.

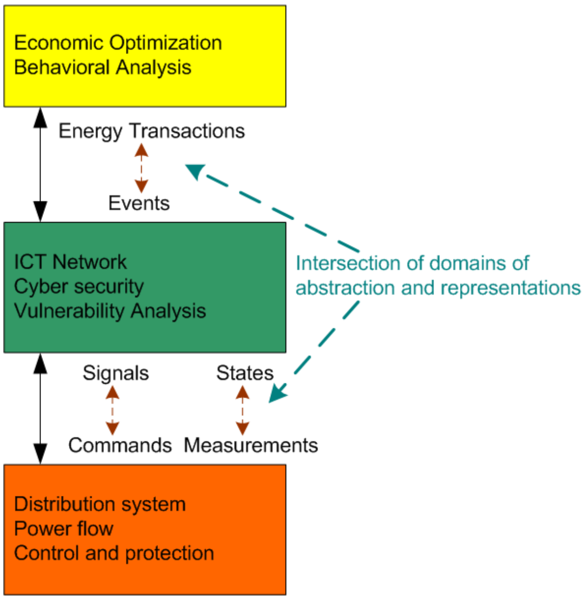

Figure 2. Illustration of semantic layer of a CPES co-simulation framework.

It is necessary to clarify that the notion of semantic layer here is addressed towards the co-simulation framework, and is not necessarily the semantic model of the energy system, although there is an overlapping area between the two concepts. More specifically, the semantic layer of a co-simulation framework explains the behavior and interaction of involved individual simulators and models, and not necessarily the semantic relations among their representing counterparts. It can be considered as an abstract layer specifying the experiments, and in some ways, similar to the concept of abstract test suite in TTCN-3 standard [15]. The dichotomies and communication among entities defined in this layer will decide how the experiment works and will be governed by the master algorithm.

The required interoperability in semantic layer and in the following syntactic layer needs to be done via applying a common information model to the framework. However, taking into account that a co-simulation framework often includes multiple domains (e.g., power system, ICT, thermal), it may be necessary to include several existing information models and thus to resolve interfacing issues. It is recommended to employ standardized information models to promote reusability and the possibility of further integration of new elements into the framework.

2.3. Syntactic Layer

This layer of a co-simulation framework influences strongly the reusability of models and interoperability of the framework. In a co-simulation framework, each model comes from different scientific domains, and it is very likely that the employed formalisms are different. A consistent co-simulation framework has to manage and harmonize these differences. Not only it has to provides a meaningful and rigorous translation of inter-formalism information, but also, the differences in term of representation, underlying model of computation and spatial-temporal scale need to be handled (cf. Figure 3).

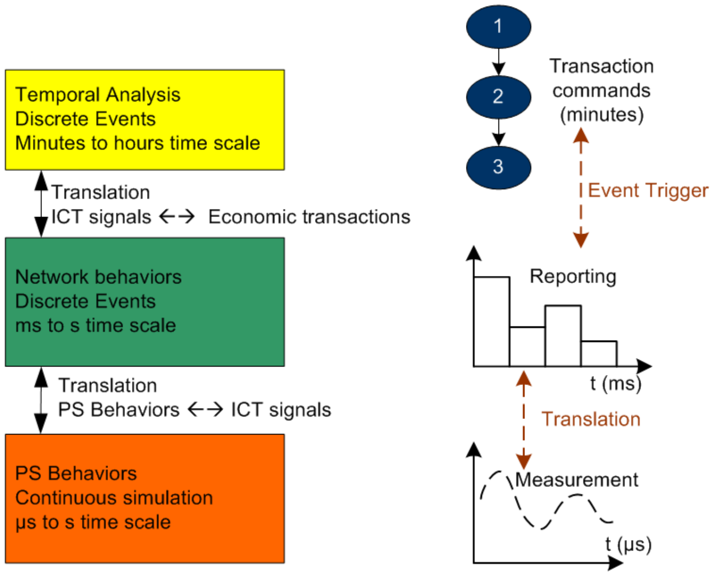

Figure 3. Illustration of syntactic layer of a CPES co-simulation framework.

In order to improve the reusability of models, it is required that the models are syntactic coherent. In general, *syntactic interoperability* of a co-simulation framework often requires the utilization of ontologies (not only for modeling, but also for data exchange). As for power system, we acknowledge several information model standards issued by the International Electrotechnical Commission (IEC) that can be considered as candidates: IEC 61850 [16], Multispeak (http://multispeak.org) and Common Information Model (CIM) [17,18]. Object Linking and Embedding for Process Control - Unified Architecture (OPC UA) [19] can also be implemented as a semi-information model when it is necessary to involve Supervisory Control and Data Acquisition (SCADA) system (i.e., hardware-in-the-loop situation). Several works are underway to harmonize the difference and to bring these standards together [20–22]. There exist a few semiautomatic converters between their models [23–25]. The mapping of ontological models to physical power system components and use-cases is classically implemented using hard-coding. One of the challenges is to enable auto-configuration of the topology of the considered CPES on-the-go and to update the database accordingly. In the case with CIM, an adaptive CIM compliant approach to resolve this challenge is proposed in [26]. The utilization of ontology in CPES is mainly for exchanging data between applications and encapsulating entire power system models in a standardized manner. As aforementioned, the concept of syntactic layer in a co-simulation framework is not always the same and is often larger than just using an ontology. Not only should it include ontological information on individual parts of the system and their interconnections, but also specify the format, syntax of the interconnections among simulators and models, as well as specifications of interfacing ontologies of different domains or transition among different time scale, which are not covered by current existing ontologies.

While being very similar to semantic layer, the syntactic layer can be simply imagined as the manner in which semantic models and interactions are represented. The notions of semantic and syntactic layers here, even though involving the utilization of energy system ontologies (e.g., IEC 61850 or CIM), are larger and require also an experiment description ontology, e.g., TTCN-3 [15]. These ontologies, if employed, can be considered as a common (or transitive) zone alongside the abstraction layers.

2.4. Dynamic Layer: Execution and Synchronization

The dynamic layer concerns executions aspects of a co-simulation framework. One needs to define the order of execution and causality of models as well as the resolution for potential conflict (i.e., cyclic dependency, deadlock) in simultaneity of actions. Synchronization techniques are necessary to ensure the consistency of the co-simulation outcome, especially when the framework consists of models with different model of computation (i.e., Continuous simulation and Discrete Event Simulation) or when the simulation is distributed (cf. Figure 4).The necessary message payloads model should also be defined.

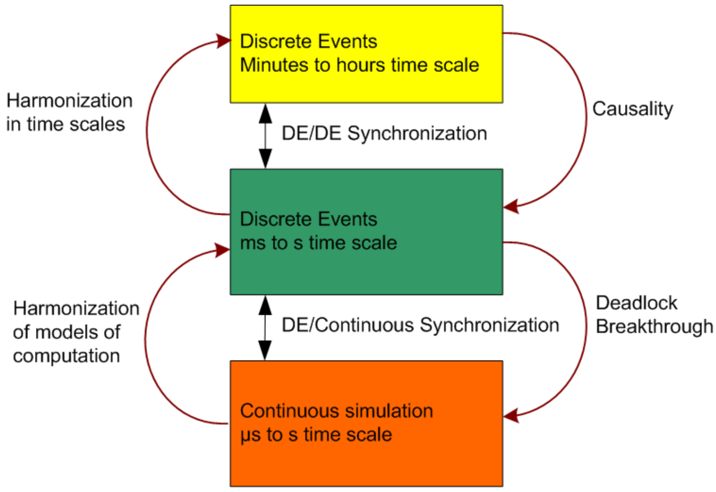

Figure 4. Illustration of dynamic layer of a CPES co-simulation framework.

Even though it is not directly related to semantic layer, it is demonstrated that different choices in dynamic layer may lead to differences in the result of simulation [27,28].

In general, for the dynamic layer, the two main issues influencing the correctness and the reliability of a co-simulation framework are harmonization of models of computation and synchronization of simulators. Harmonization of different models of computation is also a critical problem because CPES often involves a juxtaposition of various domains, where their dynamic behavior requires different solvers and consequently various models of computation. On the other hand, one can ensure, via synchronization, that operations occur in the logically correct order, whether they proceed concurrently or they must obey causality. As the size of system increases (physically or by increasing the number of simulators) or as the speed of operation increases, synchronization plays an increasingly dominant role in the stability of the framework.

Models of computation can be classified as of [29] : Imperative (e.g., Emulators), Finite State Machine (e.g., a set of states, rule-based control), Dataflow (e.g., ODEs, DEAs), Discrete Event (e.g., communication, zero-crossing), etc. Discrete Event simulation can be further broken down into: event scheduling, process interaction and activity scanning [30].

Power system models are, in general, represented by continuous models which is also capable to produce events (e.g., zero crossing, switching, etc.). The continuous models often use the *imperative* or *dataflow* model of computation, in which the model react to the availability of data at their inputs by performing some computation, via mean of according solvers, and producing data on their outputs. Dataflow is concurrent with no notion of time. In general case of power system, the model is based on a set of differential equations defining the peculiarity of the state variables and the environment factors of a system (e.g., steady-state simulations, electromagnetic transients and circuit simulations, or electromechanical phenomena). ICT and market models are often represented in discrete events chaining over a discrete set of points in time (thus associated to discrete events model of computation). More specifically, *discrete event* models are discrete, dynamic and stochastic in nature and are "run" whereas continuous model can be "run" or "solved" according to their models of computation [31]. General speaking, the two most encountered MoC for CPES assessment are Dataflow and Discrete Event (Figure 5), and they are most of the time, the subjects of harmonization.

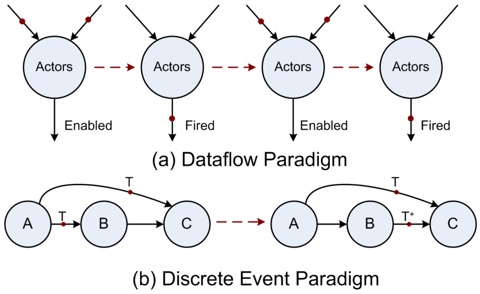

Figure 5. Examples of (**a**) Dataflow process and (**b**) Discrete Event process.

As for synchronization techniques, two well established approaches are called conservative processing and optimistic processing. In general, the conservative approach requires all the participating simulators to wait for each other to finish their step before advancing to the next step. The synchronization is checked with time stamps, via Null Message algorithm [32] or Global Synchronization approaches, such as: Bounded Lag algorithm [33], Time Buckets [34] or Composite Synchronization [35]. On the other hand, the optimistic approach allows the individual simulators to advance on their own events. When a conflict is detected, then the simulators must perform a leap backwards (time warp [36]) and discard the all the results from the moment in question.

2.5. Technical Layer: Implementation and Evaluation

This layer involves the choices of techniques for implementation and the evaluation of results. Various practical issues need to be considered: the choice of distributed or centralized simulation, global model of computation, technical implementation of interface or latency assessment. As co-simulation requires many coupling of simulators, it is necessary to consider in the end, the reliability, efficiency of the coupling, the stability of co-simulation framework (especially in cases involving hardware-in-the-loop), as well as to evaluate the robustness and accuracy of the results (cf. Figure 6). An in-depth discussion on the technical aspects of co-simulation is however further than the scope of this paper (i.e., conceptual structuration).

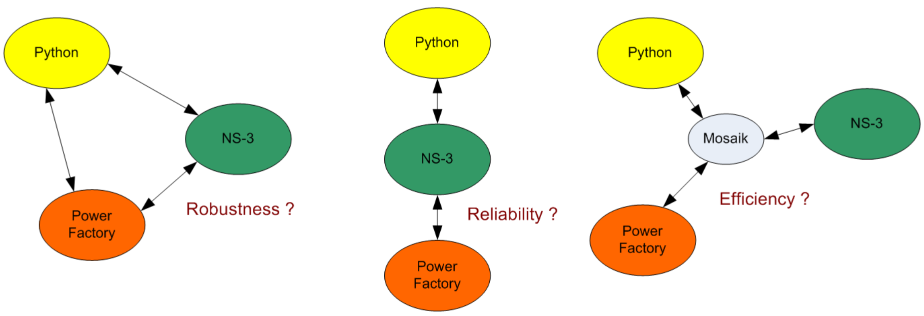

Figure 6. Illustration of technical layer of a CPES co-simulation framework.

The structuration proposed in this section is important to classify different models in a co-simulation framework into their according level of abstraction. It provides insights to identify the potential issues as well as to search for solution for coupling across different scale and abstraction level. As a consequence, the modularity, scalability and interoperability of a co-simulation framework might be improved.

In the context of this paper, we are interested also, in next sections, different existing coupling methods and synchronization techniques for co-simulation as well as their position with respect to the proposed structuration. It would provide a methodological and structural review on the state of art in co-simulation for CPES assessment.

3. Review on Coupling Methods

In general, a co-simulation framework requires the joint and simultaneous executions of models in different tools, via mean of information exchange during the execution of the simulations. Information is exchanged through either ad-hoc interfaces (Figure 7a) or via intermediate buffer governed by a master algorithm (Figure 7b). Master algorithm (where exists) is responsible for instantiating the simulators and for orchestrating the information exchange (simulator-simulator or simulator-orchestrator).

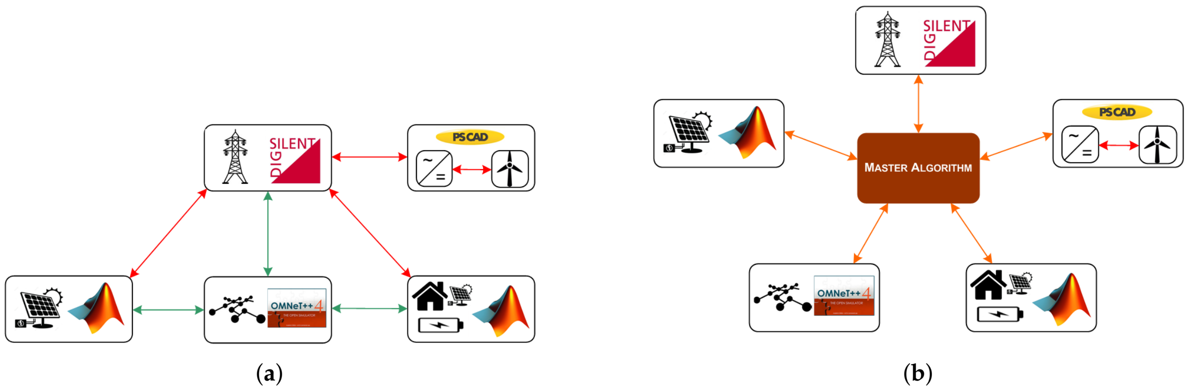

Figure 7. (**a**) Ad-hoc Co-Simulation; (**b**) Co-Simulation with an orchestrator.

The existing methods for simulator coupling in co-simulation can be classified into operational integration and formal integration. The operational coupling technique is used for specific problematic and is often not generic. In general, the operational integration aims for co-simulation at dynamic and technical layers while semantic and syntactic interrelation among the simulator are not properly addressed, thus does not provide reusability and modularity. On the other hand, formal integration aims at solving the problem at semantic and syntactic level (via either model coupling or simulator coupling). The semantic coherency among entities is ensured by encapsulating the models in a more generic layer or by orchestrating the simulators via mean of a master algorithm. The semantic and syntactic of the interaction is then governed by the said master federate.

3.1. Operational Integration

Operational integration approach most of the time case-specific and provides a good flexibility in practice. In this approach, the questions in semantic layer, which are directly related to the objectives of the test, as well as experiment behaviors are considered and interpreted outside of the simulation framework. The co-simulation focuses on engineering solutions at syntactic, dynamic and technical layers. The advantage of this approach relies in the simplicity of set-up which correctly answers to the user's need. However, the particular specifications make it hard for others to reproduce and compare the experiment result. In a scientific and industrial context, it may sometimes limit the valorization of the research.

Two most popular approaches of operational integration for co-simulation are: Model exchange (i.e., coupling the models together into one simulator) and ad-hoc coupling.

3.1.1. Model Exchange—Functional Mock-up Interface

In this approach, each model is built in its own formalism, then implanted to the same tool. The syntactic level in this case is completely hidden as the differences among the formalism are handled vie the implanting tool, i.e., technical layer. The models are forced to be simulated in the same rate, governed by the simulator. Therefore, this approach present a lot of limit in term of coupling models with different time scale.

In term of application, the approach is widely used in ICT domain simulation. The platforms such as NS-2 and NS-3, OMNet++ [37] or GTNetS [38] provides the possibility of using various models representing different aspects of the network at the same time. In CPES system assessment, it is not uncommon either to encapsulate a model and integrate it to another one; as an alternative when online co-simulation is hard to achieve.

Functional Mock-up Interface (FMI) is a standard for co-simulation issued by the consortium of the European ITEA2 Modelisar project in 2010 and in revised version in 2014. The basic component of FMI is the Functional Mock-up Unit (FMU), which gives access to the equations of the model (mode Model Exchange) or implements a solver manipulating the equations of the model (mode Co-simulation). The main purpose of FMI is providing a common interface by defining abstract functions to be implemented by every simulation component and defining how to export that said component as a shared library. FMU for model exchange can be considered as an operational integration while FMU for co-simulation can be applied to either operational integration or formal integration, depending on the level of semantic coherency among the models. It allows the integration of simulation models in a distributed and parallel way, thus possibility to apply to large-scale system [39,40].

FMI has been widely applied to the domain of mecatronics and automobile. Until recently, researchers have investigated application of FMI to CPES assessment in an effort to improve interoperability and reusability of models [41,42]. In [43], a new method of exploiting FMU via web-service and Service Oriented Architectures (SOA) is proposed. The architecture is promising and provides the framework for a potential "co-simulation as a service".

3.1.2. Ad-Hoc Coupling of Simulator

The most common architecture for simulator coupling in an operational integration is linking and integrating existing independent domain specific tools. In the case of CPES, most of the time, a power system simulator is coupled with the communication system simulator. This configuration is intuitive, concise and efficient and is often realized via sockets or direct integration to ICT simulator, as of survey result in [44].

Some co-simulation frameworks using ad-hoc connections are presented in Table 2.

Table 2. Some co-simulation frameworks using ad-hoc connections in the domain of CPES.

Co-Simulation Framework	Power System Simulator	Network Simulator	Synchronization Strategy	Applications	Time Scale
GECO [45,46]	PSLF	NS-2	Global Event-Driven	PMU-based WAMPAC	ms to s
ADEVS [47]	ADEVS	NS-2	DEVS	WAMPAC	ms to s
VPNET [48]	VTB	OPNET	Master-slave	WAMPAC	ms to s
GridSim [49]	PowerTech TSAT	GridStat	Time stepped	WAMPAC	ms to s
PowerNet [50]	Modelica	NS-2	Master - Slave	Control	Unknown
Bergmann et al. [51]	NETOMAC	NS-2	Time stepped	Evaluation of DERs	µs to ms
Babazadeh et al. [52,53]	RTlab	OPNET SITL	Asynchronous	WAMC, HVDC	µs to ms
Godfrey et al. [54]	OpenDSS	NS-2	Unknown	DER Integration	ms to s
TASSCS [55]	PowerWorld	OPNET	Unknown	Cyber-security of SCADA	µs to ms
Greenbench [56]	PSCAD	OMNet++	Global Event-Driven	Cyber-security LV Grid	ms to s

In the ad-hoc approach, the interfaces are specifically configured for to be implemented only within the simulators, with possibly high requirements of adaptations. In this set-up, the inter-dependency of involved simulators is not observed and assessed by an orchestrator, therefore, time synchronization is difficult when more than three simulators are involved in the experiment. If not properly configured, the inter-dependency of simulators may lead to deadlock and inaccurate results, especially in case of presence of communication delay among them. Scalability and stability of the experiments are therefore questionable and most of the time, ad-hoc co-simulation approach is not suitable for large scale system.

Moreover, most of the proprietary simulation tools do not provide direct interfaces to advances simulation tools and co-simulation. The interfaces developed for ad-hoc co-simulation depend on the simulators it connects; therefore, the re-usability of models and interfaces is not always evident.

Most of the time, re-utilization of the simulator in another co-simulation framework requires heavy adaptation efforts. As aforementioned, one can resort to the Functional Mock-up Interface (FMI) standard to remedy the issue. As an interface, FMI can be used in both ad-hoc and with master algorithm mode.

In general, operational integration responds quickly to the needs of a specific study, however, these approaches do not satisfy the requirements in conceptual, semantic and syntactic levels. As a consequence of this lack of generality and information, the operational models are strongly related to the technical implementation and are very limited in term of interoperability and reusability. On the other hand, operational integration demonstrates also difficulty in handling time scale and abstraction level transition among models.

3.2. Formal Integration

Formal integration is a more generic approach than operational integration as the co-simulation framework needs to ensure correct inter-relation among models in semantic and conceptual layers. In order to provide and handle, it is generally required to be orchestrated by a master algorithm. Analogy to the operational integration approach, the aforementioned master algorithm can be implemented in a single simulator where all the models are encapsulated into, or can be implemented as an orchestrator of different simulators. The principle difference between the two methods is that the models in this case are required to be semantically coherent and the exchanged signals are syntactically ready to be processed, without any further interpretation at higher layers.

This approach can be implemented in several forms:

- Implantation the elementary model into a more generic formalism. This requires however a lot of efforts and collaborations, especially in case of multi-domain experiments, where the behaviors of elements in one domain are not always evident to the experts of the other domain.
- Building and integrating an interface for translating from a particular formalism of a model to a more generic one.
- Simulator coupling handled and governed by an orchestrator.

The first two approaches requires the existence of a formalism generic enough to correctly and completely encapsulate the other involved models, without losing their semantic and conceptual meaning. This requirement appears to be a challenge. In ICT network simulation, Discrete Event System Specification (DEVS) and its derivations are generally accepted to satisfy such conditions.

3.2.1. Encapsulation of Formalism—Discrete Event System Specification (DEVS)

In general, DEVS is a generic formalism that allows encapsulating other Discrete-event models and discrete-time continuous models [57]. In DEVS, a model is represented by a 7-tuple $(I, O, S, \delta_{ext}, \delta_{int}, \lambda, t_a)$, where I and O represent the inputs and outputs of the model, S is the vector of state variables, t_a represents time advancement in simulation, δ_{int} is the evolution of internal states of model while δ_{ext} represent the influence of external events to model's internal states. Model coupling in DEVS consists of encapsulating the individual models into a big global DEVS multi-model [58–60].

The approach of encapsulation of formalism provides a strong syntactic coherence for the individual models as well as the link among them. Moreover, there are various algorithms to simulate in a distributed manner the DEVS-based model [57,60], leading to a good flexibility in term of management of simulation-related constraints.

On the other hand, the approach requires a considerable effort to encapsulate the models into DEVS, thus limits the reusability of existing models. Moreover, it is arguable that encapsulation of formalism could provide semantic and conceptual coherence and consistency [14].

3.2.2. Waveform Relaxation Method

Waveform relaxation method (WRM) [61] is a family of techniques popularly used in solving large systems of nonlinear ordinary differential equations (ODE) [62] or Differential Algebraic Equations (DAE) [63]. The basic idea of WRM is replacing the problem of solving a differential in multiple variables by one of solving a sequence of differential equations in one variable, in which the waveform of other variables are predefined. The solutions obtained from these equations are then substituted into the others one dimensional differential equations, which are then re-solved using the new waveform. The procedure is repeated until convergence condition is reached. Consider a first order two dimensional differential equations in $x(t) \in \mathbb{R}^2$ on $t \in [0, T]$:

$$\begin{cases} (\mathbf{A}) & \dot{x}_1 = f_1(x_1, x_2, t), \quad x_1(0) = x_{10} \\ (\mathbf{B}) & \dot{x}_2 = f_2(x_1, x_2, t), \quad x_2(0) = x_{20} \end{cases}$$

Then the iterative algorithm of WRM can be applied to solve this system of ODEs as illustrated on Figure 8. Similar to the case of nonlinear algebraic equations, Gauss-Seidel (or Serial Sequence) [62] and Jacobi (or Parallel Sequence) [64] techniques can be applied.

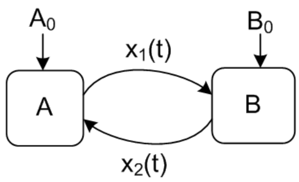

Figure 8. Illustration of the waveform relaxation method.

Originally aimed to solve complex ODEs, the WRM techniques recently draws attentions of the researchers for its applications to co-simulation, in particular, the coupling of continuous simulation [43]. In this approach, an individual simulator is represented by complete I/O waveform at the coupling points, to the other simulators. The WRM is executed until the convergence conditions are reached (Figure 9). Therefore, WRM is a method of co-simulation of semantic, syntactic and dynamic levels.

The WRM technique is of great interest in case of strong coupling of simulators, especially in the presence of communication latency. In general, WRM requires multiple executions of simulators until reaching the convergence conditions, which is the main source of execution time of the co-simulation framework. On the other hand, the other methods of coupling require one single execution of simulators, with multiple exchanges in one run instance. In the case without presence of latency or with very small latency (e.g., co-simulation of different simulators in one computer), WRM has in general longer execution time. However, when communication latency is not negligible, WRM may reach convergent point way much faster than the other methods [43]. In theory, the performance of WRM for differential systems can be improved with parallel WRM [65] and the accuracy/convergence rate can be improved with two-stage WRM [66]. However, application of these methods to co-simulation is not yet investigated.

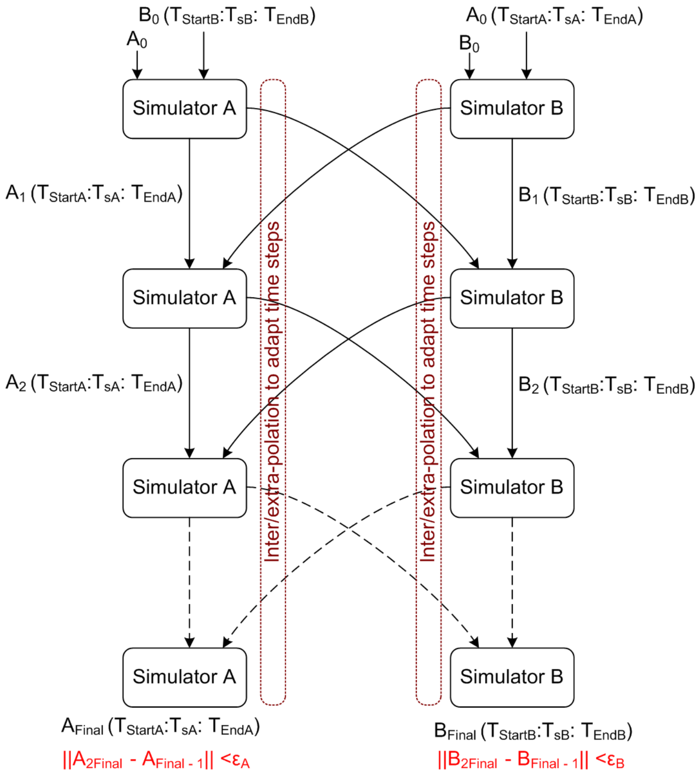

Figure 9. WRM method applied to Co-simulation.

Three important issues need to be considered in the implementation of a co-simulation framework based on WRM techniques:

- Until now, the method is exclusively applied for coupling of continuous simulation, particularly of dataflow MoC type. Discretized WRM and applications of WRM to discrete systems [67,68] is still a new research subject.
- In the implementation of WRM, the convergence condition needs to be properly addressed. In literature, there exist works on convergence conditions of WRM for ODE [69] and DAE [70,71]. A convergence analysis for Discrete-time WRM is also proposed in [72].
- in case of coupling models with asynchronous time steps, it is necessary to use an inter/extra-polation algorithm to harmonize the difference.

To conclude, WRM is an interesting approach to co-simulation, with a well-established mathematical foundation. However, the applicability of the method to discrete event simulator needs further investigation.

3.2.3. Federation of Simulation—Simulator Coupling with Orchestrator

The first two approaches of formal integration ensure semantic coherence by converting the models into a more generic formalism (i.e., DEVS and waveform), in this section, we consider formal integration by simulator coupling. While ad-hoc co-simulation is concise and efficient, for semantic and conceptual levels, it is necessary to implement a coordinator/master algorithm to orchestrate the framework. In this approach, the master algorithm is responsible for instantiating, interfacing and coordinating the involved simulators. The simulators can communicate via a common buffer/message bus governed by the master algorithm or directly among each other's (in which the master algorithm define the routing). Beside the benefit of semantic coupling, co-simulation with a master algorithm also helps to simplify the framework architecture (i.e., N interfaces for a N domains system, rather than N^2 interfaces that would be required if each domain had to explicitly interface to each other domain).

As of today, the master algorithm is most of the time implemented manually via self-developed (e.g., Java, Python, Mathlab) or third party software such as Ptolemy (http://ptolemy.eecs.berkeley.edu/) (used in [73–75], Simantics (https://www.simantics.org/) or Mosaik (http://mosaik.offis.de/) (e.g., [45,76,77]. The IEEE 1516 High level Architecture (HLA) is one of a few standards for co-simulation coordinator [78]. HLA is developed by the US M&S Co, mainly driven by military research in enabling joint simulation training. The first civil version is published by IEEE in 2000 and the evolved HLA is published in 2010. A HLA-based simulation consists of federates (participating simulators) and Run-Time Infrastructure (RTI), which provides an API for bidirectional communication from and to federates. The advantage of HLA is the possibility of highly paralleled simulations for large-scale systems; whereas its disadvantage is the introduction of additional time-synchronization issues [44]. Thus, we can register various researches focusing on the optimization aspect of the synchronization techniques in HLA [79,80].

Table 3 provides a non-exhaustive list of examples of co-simulation frameworks orchestrated by a master algorithm.

Table 3. Some co-simulation frameworks with master orchestrator in the domain of CPES.

Co-Simulation Framework	Power System Simulator	Network Simulator	Master Algorithm	Applications	Time Scale
EPOCHS [81]	PSCAD/EMTDC,PSLF	NS-2	HLA	Protection and Control	ms to minutes
INSPIRE [82]	PowerFactory	OPNET	HLA	WAMPAC	ms to minutes
SINARI [83]	PSCAD	NS-2	JAVA	vulnerability analysis	ms to s
VIRGIL [84]	PowerFactory	OMNET++	Ptolemy 2	Control and optimization	ms to s

DEVS and WRM are specific for discrete event and continuous simulation with possible adaptations to cover the other types of simulations. On the other hand, simulator coupling with an orchestrator provides more flexibility in term of coherence level. In general, depending on the specific set up and the master algorithm, co-simulation in this approach requires consistency in from dynamic to conceptual level.

3.2.4. Multi-Agent Approach for Co-Simulation

Beside the aforementioned approaches, we can also mention the multi-agent simulation approach for co-simulation at semantic and conceptual layers, which is interaction-oriented and can be considered as an overlapping of operational integration and formal integration. While offering conceptual and semantic interoperability, multi-agent simulation approach provides an operational integration of models and the simulators are executed via a well specified master algorithm. As a consequence, this approach does not always offer reusability and modularity for existing simulators.

The principle idea of the approach is considering the co-simulation framework as a multi-agent system. The most notable framework using this approach is *Agents and Artifact for multi-modeling* (AA4MM). The meta-model AA4MM considers the framework as interactive separated layers [85,86]. In semantic and syntactic levels, it defines a set of concepts representing the framework as a junction of interacting autonomous and heterogeneous models. In dynamic layer, the execution of the framework is considered to be a set of interacting autonomous and heterogeneous simulations. In addition, at last, in technical layer, the meta-model specifies an environment of interacting autonomous and heterogeneous simulators. An agent is associated to its models, simulations and simulators. The co-simulation framework is built upon the interaction among agents and the management of shared resources and environment (artifacts) (Figure 10). In general, AA4MM offers a homogeneous view on aspects of a co-simulation framework at semantic, syntactic, dynamic and technical level [14].

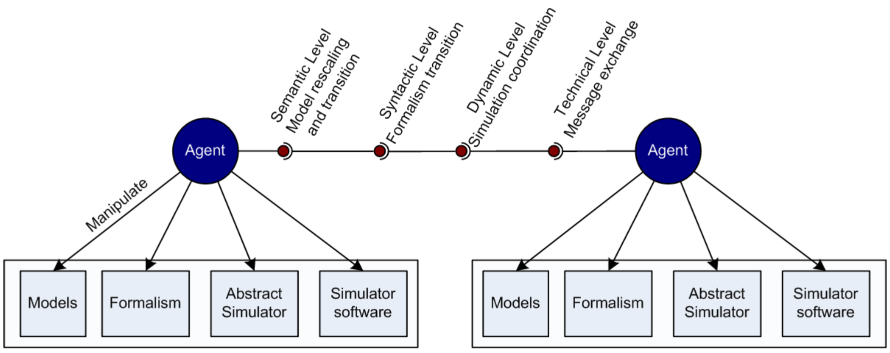

Figure 10. Interaction among Agents in the AA4MM meta-model (adapted from [14]).

Providing a great synergy and strong coupling capability, the implementation of multi-agent approach to CPES assessment is however still in the very first stage.

4. Conclusions

As the smart grid evolves into a cyber-physical energy system with inter-dependent domains, it is necessary to consider the design and validation problems at system level. That leads to the necessity of a holistic approach combining multi-domain validation, including hardware testing and simulation. While it requires a lot of effort to establish a general framework to cover all the domains in a CPES system, researchers have adopted the co-simulation approaches to consider the inter-dependency among domains as well as to validate the system as a whole. Flexible and adaptive, co-simulation is considered suitable for CPES assessment. However, in order to build a co-simulation framework, it is imperative to take into account various problems in different abstract levels, which is confusing and not always visible if one does not have a clear structuration of the framework.

In this paper, a conceptual structuration of co-simulation framework in CPES assessment is presented. A detailed review on coupling methods used and can potentially be used in co-simulation framework, along with their abstraction levels in the structuration, is then followed. It is supposed to provide the readers with a systematic view of the state of the art. This classification can be represented in a simple illustrative diagram (Figure 11).

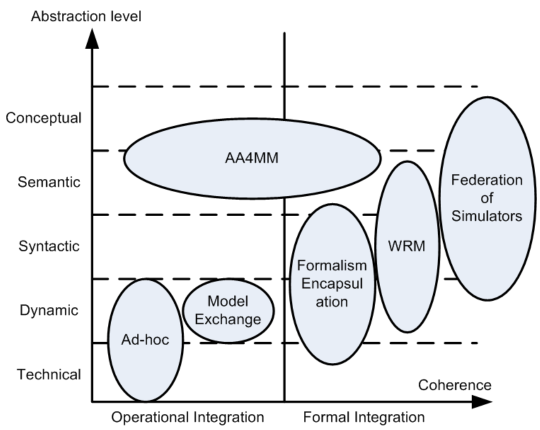

Figure 11. Reviewed coupling methods and their associated abstraction levels.

The paper highlighted multiple problems in different abstraction levels of a co-simulation framework. It is an important contribution to establish a solid theoretical base for specification of future applications of co-simulation and selection of coupling methods in CPES assessment and validation.

It is, however, necessary to take a closer look at the dynamic level where the synchronization among simulators happens. Several technical challenges persist at this level, such as the difficulty in implementing simultaneously continuous and discrete system, the lack of formalism to efficiently cover continuous simulation and discrete event simulation, harmonization of different time scales, cyclic dependency in conservative approach or time warp mechanism in optimistic approach, performance evaluation and stability analysis of different synchronization techniques, etc. These issues are interesting perspectives for further investigation.

Acknowledgments: This work is supported by the European Community's Horizon 2020 Program (H2020/2014-2020) under project "ERIGrid" (Grant Agreement No. 654113, www.erigrid.eu), which provides also funding for its open access. The participation of G2Elab and CEA-INES is also partially supported by the Carnot Institute "Energies du Futur" under the PPInterop II project (www.energiesdufutur.eu). The authors would like to express their gratitude towards the anonymous reviewers for helping to improve the paper with their valuable comments.

Author Contributions: V.H. Nguyen initiated the study on structuration and abstraction layers of co-simulation framework in CPES assessment. Y. Besanger and Q.T. Tran proposed the classification of coupling methods. T.L. Nguyen contributed to the review of existing works in operational and formal integration. V.H. Nguyen wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the study, in the writing of the manuscript and in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

CHIL	Controller-Hardware-in-the-loop
CPES	Cyber-Physical Energy System
DAE	Differential Algebraic Equation
DEVS	Discrete Event System Specification
DRES	Distributed Renewable Energy Resources
DRTS	Digital Real-Time Simulation
ICT	Information and Communication Technologies
IEC	International Electrotechnical Commission
FMI	Functional Mock-up Interface
FMU	Functional Mock-up Unit
HIL	Hardware-in-the-loop
HLA	High Level Architecture
HUT	Hardware Under Test
ODE	Ordinary Differential Equation
OLE	Object Linking and Embedding
OPC	OLE for Process Control
OPC UA	OPC Unified Architecture
PDES	Parallel Discrete-Event Simulation
PHIL	Power-Hardware-in-the-loop
SCADA	Supervisory Control And Data Acquisition
WAMPAC	Wide Area Monitoring Protection And Control
WRM	Waveform Relaxation Method

References

1. International Energy Agency. *Smart Grids in Distribution Network-Roadmap Development and Implementation*; IEA: Paris, France, 2015.
2. European Commission. *Energy Roadmap 2050*; Communication from the Commission to the European Parliament, the Council, the European Economic And Social Committee and the Committee of the Regions, Smart Grid Task Force; European Commission: Brussels, Belgium, 2011.
3. Farhangi, H. The path of the smart grid. *IEEE Power Energy Mag.* **2010**, *8*, 18–28.

4. Mets, K.; Ojea, J.A.; Develder, C. Combining Power and Communication Network Simulation for Cost-Effective Smart Grid Analysis. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1771–1796.
5. Palensky, P.; van der Meer, A.A.; López, C.D.; Jozeph, A.; Pan, K. Co-Simulation of Intelligent Power Systems—Fundamentals, software architecture, numerics, and coupling. *IEEE Ind. Electron. Mag.* **2017**, *11*, 34–50.
6. Strasser, T.; Andrén, F.P.; Lauss, G.; Bründlinger, R.; Brunner, H.; Moyo, C.; Seitl, C.; Rohjans, S.; Lehnhoff, S.; Palensky, P.; et al. Towards holistic power distribution system validation and testing—An overview and discussion of different possibilities. *Elektrotech. Informationstech.* **2017**, *134*, 71–77.
7. Nguyen, V.H.; Besanger, Y.; Tran, Q.T.; Nguyen, T.L.; Boudinet, C.; Brandl, R.; Strasser, T. Using Power-Hardware-in-the-loop Experiments together with Co-simulation in a holistic approach for cyber physical energy system validation. In Proceedings of the IEEE PES International Conference on Innovative Smart Grid Technologies IEEE ISGT Europe 2017, Turin, Italy, 26–29 September 2017.
8. Palensky, P.; van der Meer, A.A.; López, C.D.; Jozeph, A.; Pan, K. Applied co-simulation of intelligent power systems: Implementation, usage, examples. *IEEE Ind. Electron. Mag.* **2017**, *11*, 6–21.
9. Dang-Ha, T.H.; Olsson, R.; Wang, H. The Role of Big Data on Smart Grid Transition. In Proceedings of the IEEE International Conference on Smart City (SmartCity), Chengdu, China, 19–21 December 2015; pp. 33–39.
10. Li, W.; Ferdowsi, M.; Stevic, M.; Monti, A.; Ponci, F. Co-simulation for smart grid communications. *IEEE Trans. Ind. Inform.* **2014**, *10*, 2374–2384.
11. Cintuglu, M.; Mohamed, O.; Akkaya, K.; Uluagac, A. A Survey on Smart Grid Cyber-Physical System Testbeds. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 446–464.
12. Tolk, A.; Muguira, J. The levels of conceptual interoperability model. In Proceedings of the 2003 Fall Simulation Interoperability Workshop, Orlando, FL, USA, 14–19 September 2003.
13. Cantot, P.; Igarza, J.L.; Luzeaux, D.; Rabeau, R. *Simulation et ModéLisation des SystèMes de SystèMes-IC2 Traité Informatique et Systèmes d'Information*; Hermes, Lavoisier: Paris, France, 2009.
14. Siebert, J. Approche Multi-Agent pour la Multi-ModéLisation et le Couplage de Simulations. Application à L'éTude des Influences Entre le Fonctionnement des RéSeaux Ambiants et le Comportement de Leurs Utilisateurs. Ph.D. Thesis, Université Henri Poincaré—Nancy I, Nancy, France, 2011.
15. Centre for Testing and Interoperability. *TTCN-3 Tutorial*; Technical Report; ESTI: Valbonne, France, 2013.
16. International Electrotechnical Commission. *TC IEC61850—Power Utility Automation*; Technical Report; 57—Power System Management and Associated Information Exchange; IEC: Geneva, Switzerland, 2003.
17. International Electrotechnical Commission. *Energy Management System Application Program Interface (EMS-API)—Part 301: Common Information Model (CIM) Base*; Technical Report; TC 57—Power System Management and Associated Information Exchange; IEC: Geneva, Switzerland, 2014.
18. International Electrotechnical Commission. *Application Integration at Electric Utilities—System Interfaces for Distribution Management—Part 11: Common Information Model (CIM) Extensions for Distribution*; Technical Report; TC 57—Power System Management and Associated Information Exchange; IEC: Geneva, Switzerland, 2013.
19. Electrical Power Research Institute. *OPC Unified Architecture—Part 1: Overview and Concepts*; Technical Report; TC 65/SC 65E–TR 62541-1:2010; Electrical Power Research Institute: Palo Alto, CA, USA, 2010.
20. International Electrotechnical Commission. *Application Integration at Electric Utilities—System Interfaces for Distribution Management—Part 14: Multispeak-CIM Harmonization*; Technical Report; TC 57—Power System Management and Associated Information Exchange; IEC: Geneva, Switzerland, 2013.
21. Gary, A.; Greg, R.; Gerald, R. MultiSpeak and IEC 61968 CIM: Moving Toward Interoperability. In Proceedings of the Grid-Interop Forum, Atlanta, GA, USA, 11–13 November 2008.
22. Nguyen, V.H.; Besanger, Y.; Tran, Q.T. CIM and OPC UA for interoperability of micro-grid platforms. In Proceedings of the IEEE PES Innovative Smart Grid Technologies Conference ISGT North America 2016, Minneapolis, MN, USA, 6–9 September 2016.
23. Kostic, T.; Preiss, O.; Frei, C. Towards the formal integration of two upcoming standards: IEC 61970 and IEC 61850. In Proceedings of the Large Engineering Systems Conference on Power Engineering, Montreal, QC, Canada, 7–9 May 2003.
24. Electrical Power Research Institute. *Harmonizing the International Electrotechnical Commission Common Information Model (CIM) and 61850—Key to Achieve Smart Grid Interoperability Objectives*; Technical Report; EPRI1020098; EPRI: Palo Alto, CA, USA, 2010.

25. Rohjans, S.; Uslar, M.; Appekath, J. OPC UA and CIM: Semantics for the smart grid. In Proceedings of the IEEE PES Transmission and Distribution Conference and Exposition, New Orleans, LA, USA, 19–22 April 2010.
26. Le, M.T.; Nguyen, V.H.; Besanger, Y.; Tran, Q.T. CIM compliant multiplatform approach for cyber-physical energy system assessment. In Proceedings of the IEEE PES Innovative Smart Grid Technologies Conference ISGT Asia, Auckland, New Zealand, 4–7 December 2017.
27. Michel, F. Formalisme, Outils et éLéMents MéThodologiques pour la Modélisation et la Simulation Multi-Agents. Ph.D. Thesis, Université des Sciences et Techniques du Languedoc, Montpellier, France, 2004.
28. Chevrier, V.; Fates, N. How important are updating schemes in multi-agent systems? An illustration on a multi-turmite model. In Proceedings of the 9th International Conference on Autonomous Agents and MultiAgent System AAMAS 2010, Toronto, ON, Canada, 9–14 May 2010.
29. Chang, W.T.; Ha, S.; Lee, E.A. Heterogeneous Simulation–Mixing Discrete-Event Models with Dataflow. *J. VLSI Signal Process.* **1997**, *15*, 127–144.
30. Kesaraju, V.; Ciarallo, F. Integrated simulation combining process-driven and event-driven models. *J. Simul.* **2012**, *6*, 9–20.
31. Banks, J.; Carson, I.I.; Nelson, B.L.; Nicol, D. *Discrete Event System Simulation*, 4th ed.; Prentice-Hall International: New York, NY, USA, 2005.
32. Chandy, K.; Misra, J. Distributed Simulation: A case study in design and verification of distributed program. *IEEE Trans. Softw. Eng.* **1979**, *24*, 440–452.
33. Lubachevsky, B.; Shwartz, D.; Weiss, A. An analysis of rollback-based simulation. *ACM Trans. Model. Comput. Simul.* **1991**, *1*, 154–193.
34. Steinman, J. Breathing Time Warp. In Proceedings of the Seventh Workshop on Parallel and Distributed Simulation, San Diego, CA, USA, 16–19 May 1993.
35. Nicol, D.; Liu, J. Composite synchronization in parallel discrete-event simulation. *IEEE Trans. Parallel Distrib. Syst.* **2002**, *13*, 433–446.
36. Jefferson, D. Virtual time. *ACM Trans. Program. Lang. Syst.* **1985**, *7*, 404–425.
37. Pongor, G. OMNET: Objective modular network testbed. In Proceedings of the MASCOTS93 International Workshop on Modeling, Analysis and Simulation on Computer and Telecommunication Systems, San Diego, CA, USA, 17–20 January 1993.
38. Riley, G. The Georgie Tech Network Simulator. In Proceedings of the ACM SIGCOMM Workshop on Models, Methods and Tools for Reproductible Network Research, Bolton Landing, NY, USA, 19–22 October 2003.
39. Awais, M.; Gawlik, W.; DeCillia, G.; Palensky, P. Hybrid Simulation using SAHISim Framework. In Proceedings of the 8th EAI International Conference on Simulation Tools and Techniques, Athens, Greece, 24–26 August 2015.
40. Galtier, V.; Vialle, S.; Dad, C.; Tavella, J.P.; Lam, Y.M.; Plessis, G. FMI-based Distributed Multi-Simulation with DACCOSIM. In Proceedings of the Spring Simulation Multi-Conference, Alexandria, VA, USA, 12–15 April 2015.
41. Widl, E.; Delinchant, B.; Kubler, S.; Li, D.; Muller, W.; Norrefeldt, V.; Nouidui, T.S.; Stratbucker, S.; Wetter, M.; Wurtz, F.; et al. Novel simulation concepts for buildings and community energy systems based on the Functional Mock-up Interface specification. In Proceedings of the Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES), Berlin, Germany, 14 April 2014.
42. Nguyen, T.L.; Tran, Q.T.; Caire, R.; Besanger, Y.; Hoang, T.T.; Nguyen, V.H. FMI Compliant Approach to Investigate the Impact of Communication to Islanded Microgrid Secondary Control. In Proceedings of the IEEE PES International Conference on Innovative Smart Grid Technologies (ISGT Asia 2017), Auckland, New Zealand, 4–7 December 2017.
43. Raad, A.; Reinbold, V.; Delinchant, B.; Wurtz, F. Energy building co-simulation based on the WRM algorithm for efficient simulation over FMU components of Web Service. In Proceedings of the BS'15 Building Simulation Conference, Hyderabad, India, 7–9 December 2015.
44. Müller, S.C.; Georg, H.; Nutaro, J.J.; Widl, E.; Deng, Y.; Palensky, P.; Awais, M.U.; Chenine, M.; Kuch, M.; Stifter, M.; et al. Interfacing Power System and ICT Simulators: Challenges, State-of-the-Art and Case Studies. *IEEE Trans. Smart Grid* **2016**, *PP*, doi:10.1109/TSG.2016.2542824.
45. Lin, H.; Sambamoorthy, S.; Shukl, S.; Thorp, J.; Mili, L. A study of communication and power system infrastructure interdependence on PMU-based wide area monitoring and protection. In Proceedings of the Power and Energy Society General Meeting, San Diego, CA, USA, 22–26 July 2012.

46. Lin, H.; Veda, S.; Shukla, S.; Mili, L.; Throp, J. GECO: Global Event-Driven Co-Simulation Framework for Interconnected Power System and Communication Network. *IEEE Trans. Smart Grid* **2012**, *3*, 1444–1459.
47. Nutaro, J.; Kuruganti, P.; Miller, L.; Mullen, S.; Shankar, M. Integrated hybrid-simulation of electric power and communication systems. In Proceedings of the IEEE PES General Meeting 2007, Tampa, FL, USA, 24–28 June 2007.
48. Li, W.; Monti, A.; Luo, M.; Dougal, R. VPNET: A co-simulation framework for analyzing communication channel effects on power systems. In Proceedings of the IEEE Electric Ship Technologies Symposium, Alexandria, VA, USA, 10–13 April 2011; pp. 143–149.
49. Anderson, D.; Zhao, C.; Hauser, C.; Venkatasubramanian, V.; Bakken, D.; Bose, A. A virtual smart grid–real-time simulation for smart grid control and communication design. *IEEE Power Energy Mag.* **2012**, *10*, 49–57.
50. Liberatoire, V.; Al-Hammouri, A. Smart Grid Communication and Co-Simulation. In Proceedings of the IEEE Energytech, Cleveland, OH, USA, 25–26 May 2011; pp. 1–5.
51. Bergmann, J.; Glomb, C.; Götz, J.; Heuer, J. Scalability of Smart Grid Protocols: Protocols and Their Simulative Evaluation for Massively Distributed DERs. In Proceedings of the 1st IEEE International Conference on Smart Grid Communication, Gaithersburg, MD, USA, 4–6 October 2010; pp. 131–136.
52. Babazadeh, D.; Chemine, M.; Kun, M.; Al-Hammouri, A.; Nordstrom, L. A platform for Wide Area Monitoring and Control System ICT analysis and Development. In Proceedings of the IEEE PowerTech, Grenoble, France, 16–20 June 2013; pp. 1–7.
53. Babazadeh, D.; Nordstrom, L. Angent-based control of VSC-HVDC Transmission Grid—A Cyber Physical System Perspective. In Proceedings of the IEEE MSCPES 2014, Berlin, Germany, 14 April 2014.
54. Godfrey, T.; Mullen, S.; Dugan, R.; Rodine, C.; Griffith, D.; Golmie, N. Modeling smart grid applications with co-simulation. In Proceedings of the 1st IEEE International Conference on Smart Grid Communication, Gaithersburg, MD, USA, 4–6 October 2010; pp. 291–296.
55. Mallouhi, M.; Al-Nashif, Y.; Cox, D.; Chadaga, T.; Hariri, S. A testbed for analyzing security of SCADA control systems (TASSCS). In Proceedings of the 2011 IEEE PES Innovative Smart Grid Technologies Conference, Anaheim, CA, USA, 17–19 January 2011.
56. Wei, M.; Wang, W. Greenbench: A benchmark for observing power grid vulnerability under data-centric threats. In Proceedings of the IEEE INFOCOM 2014—IEEE Conference on Computer Communications, Toronto, ON, Canada, 27 April–2 May 2014; pp. 2625–2633.
57. Zeigler, B.; Praehofer, H.; Kim, T. *Theory of Modeling and Simulation*; Academic Press: Cambridge, MA, USA, 2000.
58. Himmelspach, J.; Uhrmacher, A. Plug'n Simulate. In Proceedings of the ANSS07 40th Annual Simulation Symposium, Norfolk, VA, USA, 26–28 March 2007.
59. Himmelspach, J.; Rohl, M.; Uhrmacher, A. Component based models and simulation experiments for multi-agent systems in JAMES II. In Proceedings of the International Workshop on Agent Theory to Agent Implementation, Estoril, Portugal, 12–13 May 2008.
60. Quesnel, G.; Duboz, R.; Ramat, E. The virtual laboratory environment—An operational framework for multi-modeling, simulation and analysis of complex dynamical systems. *Simul. Model. Pract. Theory* **2008**, *17*, 641–653.
61. Lelarasmee, E.; Ruehli, A.; Sangiovanni-Vincentelli, A. *The Wave form Relaxation Method for Time-Domain Analysis of Large Scale Integrated Circuits*; University of California: Oakland, CA, USA, 1982; Volume 1.
62. White, J.; Vincentelli, A.; Odeh, F.; Ruehli, A. Waveform Relaxation: Theory and Practice. *Trans. Soc. Comput. Simul.* **1985**, *2*, 95–133.
63. Crow, M.; Ilic, M. The waveform relaxation method for systems of differential/Algebraic Equations. *Math. Comput. Model.* **1994**, *19*, 67–84.
64. Sand, J.; Burrage, K. A Jacobi Waveform relaxation method for ODE. *SIAM J. Sci. Comput.* **1998**, *20*, 534–552.
65. Burrage, K.; Dyke, C.; Pohl, B. On the performance of parallel waveform relaxations for differential systems. *Appl. Numer. Math.* **1996**, *20*, 39–55.
66. Garrappa, R. An analysis of convergence for two-stage waveform relaxation methods. *J. Comput. Appl. Math.* **2004**, *169*, 377–392.
67. Sun, W.; Fan, W.G.; Wang, T. A discretized waveform relaxation method in flexible holonomic systems. In Proceedings of the 2014 International conference on Information Science, Electronics and Electrical Engineering, Sapporo, Japan, 26–28 April 2014.

68. Liu, H.; Fu, Y.; Li, B. Discrete waveform relaxation method for linear fractional delay differential-algebraic equations. *Discrete Dyn. Nat. Soc.* **2017**, *2017*, 6306570.
69. Hout, K. On the convergence of waveform relaxation methods for stiff nonlinear ordinary differential equations. *Appl. Numer. Math.* **1995**, *18*, 175–190.
70. Bartoszewski, Z.; Kwapisz, M. On the convergence of waveform relaxation methods for differential-Functional System of Equations. *J. Math. Anal. Appl.* **1999**, *235*, 478–496.
71. Jiang, Y.L.; Wing, O. A note on convergence conditions of waveform relaxation algorithms for nonlinear differential-algebraic equations. *Appl. Numer. Math.* **2001**, *36*, 281–297.
72. Crisci, M.; Russo, E.; Vecchio, A. Discrete-time waveform relaxation Voltera-Runge-Kutta methods: Convergence analysis. *J. Comput. Appl. Math.* **1997**, *86*, 359–374.
73. Lee, E.; Zheng, H. Operational Semantics of Hybrid Systems. In *Hybrid Systems: Computation and Control, Proceedings of the 8th International Workshop, HSCC 2005, Zurich, Switzerland, 9–11 March 2005*; Springer Science Business Media: Berlin, Germany, 2005; pp. 25–53.
74. Wetter, M. Co-simulation of building energy and control systems with the building controls virtual test bed. *J. Build. Perform. Simul.* **2011**, *3*, 185–203.
75. Chatzivasileiadis, S.; Bonvini, M.; Matanza, J.; Yin, R.; Liu, Z.; Nouidui, T.; Kara, E.; Parmar, R.; Lorenzetti, D.; Wetter, M.; et al. Cyber physical modeling of distributed resources for distribution system operations. *Proc. IEEE* **2015**, *104*, 789–806.
76. Georg, H.; Wietfeld, C.; Muller, S.; Rehtanz, C. A HLA based simulator architecture for co-simulating ICT based power system control and protection systems. In Proceedings of the 3rd Smart Grid communications (SmartGridComm), Tainan City, Taiwan, 5–8 November 2012.
77. Galus, M.D.; Waraich, R.A.; Noembrini, F.; Steurs, K.; Georges, G.; Boulouchos, K.; Axhausen, K.W.; Andersson, G. Integrating Power systems, transport systems and Vehicle Technology for Electric Mobility Impact assessment and efficient control. *IEEE Trans. Smart Grid* **2012**, *3*, 934–949.
78. IEEE. *IEEE Standard for Modeling and Simulation (M & S) High Level Architecture (HLA)—Framework and Rules*; Technical Report; IEEE SA-1516-2010; IEEE: Piscataway, NJ, USA, 2010.
79. Hemingway, G.; Neema, H.; Nine, H.; Sztipanovits, J.; Karsai, G. Rapid synthesis of high-level architecture-based heterogeneous simulation: A model-based integration approach. *J. Simul.* **2012**, *88*, 217–232.
80. Wang, X.; Turner, S.J.; Yoke Hean Low, M.; Gan, P.B. Optimistic Synchronization in HLA-Based Distributed Simulation. *J. Simul.* **2005**, *81*, 279–291.
81. Hopkinson, K.; Wang, X.; Giovanini, R.; Thorp, J.; Birman, K.; Coury, D. EPOCHS: A platform for Agent-Based Electric Power and Communication Simulation Built from Commercial Off-The-Shelf Components. *IEEE Trans. Power Syst.* **2006**, *21*, 548–558.
82. Georg, H.; Muller, S.; Rehtanz, C.; Wiefeld, C. Analyzing Cyber-Physical Energy Systems: The INSPIRE Co-Simulation of Power and ICT system using HLA. *IEEE Trans. Ind. Inform.* **2014**, *10*, 2364–2373.
83. Caire, R.; Sanchez, J.; Hadjsaid, N. Vulnerability analysis of coupled heterogeneous critical infrastructures: A Co-simulation approach with a testbed validation. In Proceedings of the IEEE ISGT Europe 2013, Lyngby, Denmark, 6–9 October 2013.
84. Chatzivasileiadis, S.; Marco, B.; Javier, M.; Rongxin, Y.; Thierry, S.N.; Emre, C.K.; Rajiv, P. Cyber–Physical Modeling of Distributed Resources for Distribution System Operations. *Proc. IEEE* **2016**, *104*, 789–806.
85. Ricci, A.; Viroli, M.; Omicini, A. Give Agents their artifacts: The A & A approach for engineering working environments in MAS. In Proceedings of the AAMAS07 6th International Joint Conference on Autonomous Agents and Multi-Agent Systems, Honolulu, HI, USA, 14–18 May 2007.
86. Omicini, A.; Ricci, A.; Viroli, M. Artifacts in the A & A meta-model for multi-agent systems. *Auton. Agent Multi-Agent Syst.* **2008**, *17*, 432–456.