A Co-Simulation Framework for Power Systems and Communication Networks

Welin Zhong, Muyang Liu, *Student Member, IEEE*, and Federico Milano, *Fellow, IEEE*
AMPSAS, School of Electrical & Electronic Engineering, University College Dublin, Ireland
{weilin.zhong, muyang.liu}@ucdconnect.ie, federico.milano@ucd.ie

***Abstract*—This paper presents a co-simulation framework for power systems and communication networks, using DOME, a Python-based power system analysis tool, and NS-3, an open-source discrete-event network simulator. The objects of the paper are twofold. First, the paper outlines DOME and NS-3, and describes the design of a co-simulation framework based on these software tools. Then a real-world dynamic model of the all-island Irish transmission system is used for testing the performance of this framework.**

***Index Terms*—Co-simulation, communication networks, communication delays, delay differential algebraic equations, discrete events.**

I. INTRODUCTION

A. Motivation

Power systems are becoming intertwined with communication networks increasingly. Notably, power system dynamics can be influenced by the transient behaviors, i.e., latency, congestion, package dropouts, etc, of the communication network. A framework that can adequately capture the dynamics of both power systems and communication networks is a relevant analysis tool. A monolithic software tool that includes both power systems and communications networks, however, is a sort of chimera. Power system transient stability models are based on continuous differential-algebraic equations with a time step of a few tens or hundreds of milliseconds, whereas communication networks are generally simulated through discrete event models with the time step of the order of nanoseconds.

Co-simulation appears as a promising approach to merge these two substantially different simulation approaches. The co-simulation framework discussed in this paper considers two Python-interfaced software tools: DOME for power system dynamic analysis [1]; and NS-3 for the simulation of communication networks [2].

B. Literature Review

Relevant researches on power system analysis with considering the impact of communication networks include [3]–[6]. However, the parameters of the communication networks in these works are theoretical, and communication networks have some optimistic assumptions which are not realistic. To fully integrate power system and communication network, the co-simulation of these two systems has become an exciting research area.

EPOCHS (electric power and communication synchronizing simulator) is one of the first simulators to integrate realistic communication networks with power systems [7]. EPOCHS is a platform for agent-based electric power and communication simulation and is based on three off-the-shelf simulators: PSCAD/EMTDC, PSLF, and Network Simulator 2 (NS-2). PSCAD/EMTDC is an electromagnetic transient simulator; PSLF is a transient electromechanical simulator; and NS-2 is a communication network simulator. The primary limitation of EPOCHS is that, due to the size of the synchronization step, the accuracy and efficiency of the simulation cannot be met at the same time.

A global event-driven co-simulation framework named GECO was proposed in [8]. This framework is implemented using GE's Positive Sequence Load Flow (PSLF) and NS-2. Compared to EPOCHS, GECO has the advantage that it can achieve better synchronization accuracy by using the global event-driven method. However, PSLF is not open-source software. Therefore, designing some new devices or control loops to optimize the simulation is sharply limited.

The co-simulation framework called FNCS utilizes a federated approach to integrate three open-source simulators: the distribution simulator GridLAB-D, the transmission simulator PowerFlow, and the communication network simulator NS-3 [9]. The main contribution of FNCS is the utilization of two synchronization strategies to improve the performance of co-simulation by 20% on average. However, this project was discontinued in 2015, and no longer maintained. Instead, the team that developed FNCS has started to develop a joint national lab tool called HELICS (Hierarchical Engine for Large-scale Infrastructure Co-Simulation) [10]. HELICS is an open-source co-simulation framework designed to support the co-simulation with power systems, communication networks, and markets.

Two other co-simulation frameworks were reported in 2007 and 2011 respectively. First, A Discrete Event system Simulator (ADEVS) [11], based on the Discrete Event System Specification (DEVS), is designed to integrate DEVS and NS-2. Then, ORNL [12] power system simulator is designed for combining controls, communications and electro-mechanical dynamics in the smart grid, using ADVES and OMNeT++. However, DEVS is designed for discrete event system modeling. Since power systems are effectively continuous systems, ADEVS and ORNL do not appear particularly suited to for

Weilin Zhong, Muyang Liu, and Federico Milano are supported by Science Foundation Ireland (SFI) under Investigator Programme Grant No. SFI/15/IA/3074.

power system modeling.

There are many other co-simulation frameworks reported in recent years. Reference [13] describes a co-simulation framework that merges of OpenDSS and NS-2. Reference [14] reports an integration of MATLAB/Simulink and OPNET to co-simulate the power systems and communication networks. References [15] and [16] provide a similar SCADA testbed, respectively integrated PowerWorld and RINSE, PowerWorld and OPNET.

Table I summarizes the co-simulators mentioned above.

TABLE I: Existing power system/communication network co-simulators

	Simulation Tool		
Reference	Power System	Comm. Network	Year
EPOCHS [7]	PSCAD, PSLF	NS-2	2006
[15]	PowerWorld	RINSE	2006
ADEVS [11]	ADEVS	NS-2	2007
[13]	OpenDSS	NS-2	2010
[17]	PowerWorld	Anylogic	2010
ORNL [12]	ADVES	OMNeT++	2011
[14]	MATLAB/Simulink	OPNET	2011
VPNET [18]	Virtual Test Bed	OPNET	2011
PowerNet [19]	Modelica	NS-2	2011
[16]	PowerWorld	OPNET	2011
GECO [8]	PSLF	NS-2	2012
GridSim [20]	TSAT	GridStat	2012
[21]	DigSilent	OPNET	2012
FNCS [9]	GridLAB-D, PowerFlow	NS-3	2014
HELICS [10]	GridLAB-D, GridDyn	NS-3	2017

C. Contributions

Most of the simulation software in the power system (e.g., PowerWorld or PSCAD) and the communication network (e.g., OPENT) are proprietary software tools, which significantly limits their integration in a co-simulation framework. In this paper, we consider only tools based on open-source compilers and libraries. In particular, we used the Python language as the glue between DOME and NS-3. Based on its mature ecosystem of scientific libraries, the Python programming language is now one of the most famous scientific computing languages [22].

The specific contributions of the paper are the following.

- The description of a new co-simulation framework for integrated power system and communication network. This framework integrates two open-source software tools DOME and NS-3, which allows users to customize devices and is particularly suited to education and research.
- A discussion on the impact of realistic communication delays on power system dynamic response and stability. Thanks to the modeling capability of NS-3, the considered wide-area communication delay model appears more precise than the models previously discussed in the literature [23].

D. Organization

The paper is organized as follows. Section II-A provides a description of the Python-based power system analysis tool DOME. Section II-B introduces the discrete-event network simulator NS-3. Section II-C presents the co-simulation framework in this paper and the implementation of using DOME and NS-3. Section III presents a case study testing the framework in this paper in a 1479-bus all-island Irish Transmission system with inclusions of wide-area communication networks. Conclusions and future work are summarized in Section IV.

II. Overview of the Co-simulation Framework

A. Power system analysis tool DOME

DOME, a Python-based power system analysis tool, is developed based on the experience maturated with the Power System Analysis Toolbox (PSAT), which is one of the first open-source software for power system analysis [24]. Compared to PSAT, the architecture of DOME has been upgraded to provide a more advanced and stable performance.

The first major difference between PSAT and Dome is the choice of Python as the primary development language. Python is a dynamically typed and high-level programming language, which is easy to use. Some relevant characteristics of Python are the following:

- Compared with the traditional statically typed programming language (e.g., C/C++, Java, etc.), the syntax of Python is easier to understand, and it is friendly to beginners.
- Python has a huge variety of standard and third-party libraries that allows easily extending new applications on the original project.
- Libraries such as NumPy and CVXOPT provide an efficient method to analysis multidimensional arrays, linear algebra, eigenvalue computation, etc., which are the critical function in power system analysis (e.g., time domain simulation, eigenvalue analysis).

Based on these features of Python, DOME can execute various power system analysis by invoking several independent Python modules. DOME also provides the application programming interface (API) to extend some efficient open-source libraries such as ARPACK [25] and SLEPc [26], which are based on other programming languages (C/C++).

Currently, DOME provides about 950 devices, such as synchronous machines, automatic generation controller (AGC), automatic voltage regulator (AVR), power system stabilizer (PSS). It also provides a large variety of wind and marine current turbine models and controllers, energy storage devices and models based on stochastic processes [27] and delay differential equations [28]. DOME can be used to solve the power flow analysis such as continuation power flow and three-phase unbalanced power flow, and used to address other analysis, e.g., time domain simulation, eigenvalue analysis, electromagnetic transients. The detailed design principles of power system modeling and scripting is provided in [29].

Compared with other power system analysis tools, DOME has the advantage to use a semi-implicit formulation of

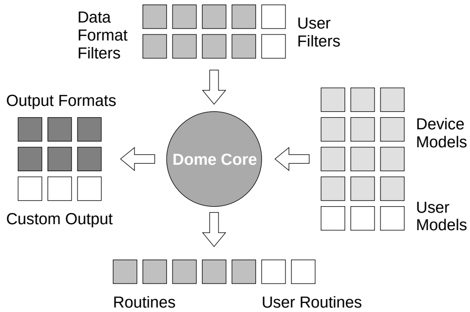

Fig. 1: Modular structure of DOME [1].

differential-algebraic equations [30]. Moreover, it allows users to develop their new device models. Therefore, DOME can also be used as an education tool. Undergraduates can use DOME for their final projects, while Master and Ph.D. students can develop some new routines. Fig. 1 illustrates the modular structure of DOME. A description of the educational usages of DOME can be found in [31].

B. *Communication network simulator* NS-3

NS-3 is an open-source discrete-event network simulator for internet systems, designed for networking education and research [32], and on a replacement for the network simulator version 2 (NS-2). Similar to the NS-2, NS-3 is distributed under the GNU GPLv2 license that allows the third-party developers to use the library and contribute code freely. Compared with NS-2, the advantages of NS-3 are the following.

- NS-3 supports Python as a scripting interface, instead of OTcl in NS-2, to improve scalability and integration.
- Owing to the components of NS-2 are written in different languages (e.g., some are written in C++ and others in OTcl), it is impossible to start simulation in NS-2 without OTcl. By contrast, the core of NS-3 is solely written in C++ but with several optional Python bindings. Therefore, simulation scripts can be written in either C++ or Python.
- NS-3 has detailed models spanning several popular research areas, such as LTE, WiFi, and 5G models.
- NS-3 has a distributed architecture to achieve scalability for large-scale simulations (e.g., one billion nodes network) [2], [33], [34].
- NS-3 provides an easy way to analyze the results generated by the simulation. The tracing system provides the user with detailed statistics of the output data. The generated ASCII or PCAP trace files are valuable for data analysis.

NS-3 is based on several key abstractions. Relevant concepts and definitions are as follows.

- Network *nodes*, the basic computing device abstraction, which represents the end systems such as computers, routers, and switches.
- Network *devices*, cover both software driver and simulated hardware equipment in NS-3, are used to connect the nodes to the communication channels.

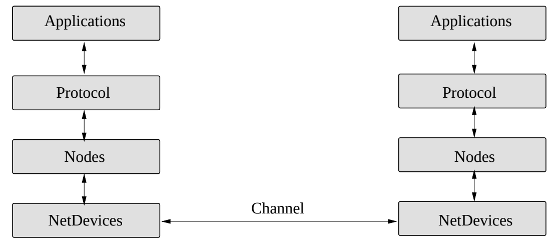

Fig. 2: The basic architecture of NS-3.

- Communication *channels*, the medium used to connect network devices. NS-3 provides three kinds of channels, Point-to-Point channel, Carrier Sense Multiple Access (CSMA) channel, and WiFi channel.
- *Topology Helpers*, which help to easily set up the network topology such as arrange several connections between nodes, and assign the IP addresses.
- *Applications*, the basic abstraction of simulating users activities in the network, as software applications run on computers.

Fig. 2 illustrates the basic architecture of NS-3. The detailed tutorial, manual, and model introduction can be found in the official website of NS-3 [32].

C. DOME/NS-3 *Co-Simulation Framework*

The design principle of the proposed framework is as follows. DOME is the "master" and NS-3 is the "slave." All input data are passed to DOME, which takes care of initializing both the power system and the communication network. The latter is sets up in NS-3 and consists a set of the point-to-point communication channel effectively. Then DOME runs the time domain simulations and defines the time steps (fixed or adaptive). At every time step, say t, DOME solves the integration of the differential-algebraic equations that define the power systems and, at the same time passes to NS-3 the current simulation time. NS-3 is run to simulate the each Point-to-Point/CSMA communications and the delays with which the transmitted signals arrive at the destination are passed back to DOME. The signals are then modeled in DOME as delayed variables and properly accounted for in the integration scheme. Fig. 3 illustrates the proposed co-simulation framework.

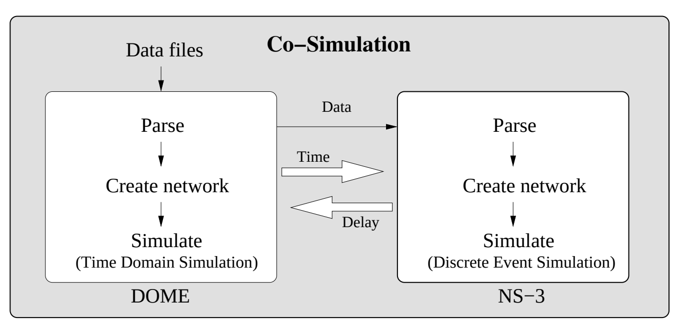

Fig. 3: The architecture of the co-simulation framework.

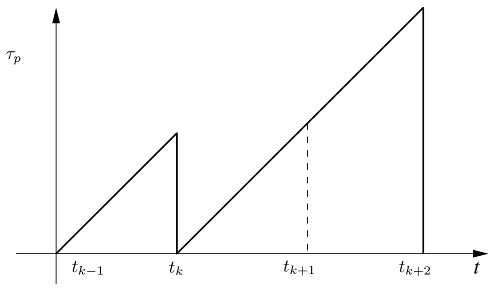

Fig. 4: Stochastic WAC delay model including packet loss.

III. Case Study

This section tests the co-simulation framework discussed in Section II and studies the impact of Wide-Area Communication (WAC) delays on a real-world power system. The co-simulation results are compared with the results obtained with the software tool DOME.

A. Wide-Area Communication Delay Model

The realistic WAC delay [35] can be formulated as:

$$\tau(t) = \tau_f + \tau_p(t) + \theta(t), \tag{1}$$

where τ is the total delay, τ_f is the fixed delay associated with transducers used, data processing, τ_p is the transmission delay, and θ is the associated random jitter resulting from network-induced issues.

In an ideal WAC network, the transmission delay τ_p for each data packet is an identical constant period:

$$T = t_{k+1} - t_k \ , \tag{2}$$

where $t(k)$ is the time that k-th data packet arrives. The transmission delay at a specific time t can be derived as:

$$\tau_p(t) = t - t_k \ . \tag{3}$$

In a real-world WAC network the $k+1$-th packet can be lost. If the packet drop-out occurs, the Zero-Order Holder (ZOH) will hold the latest state as the feedback signal to the controllers until the next packet has been received, which means that the delay of the last lost packet is automatically added to the next packet. Fig. 4 shows the case when the packet $k+1$ is lost.

The time-varying WAC delays can be obtained through the proposed co-simulation framework or the mathematical model developed by the second and the third authors in [23].

1) Delay generated by co-simulation: In the co-simulation framework, the fixed delay τ_f is directly set as a parameter in DOME, as this is a feature of the PMU not a part of the communication network. The other terms of (1) depends on the communication network and are determined with NS-3.

In NS-3, the transmission delay is considered as:

$$\tau_p = \tau_{po} + \frac{L}{R}, \tag{4}$$

where τ_{po} is the propagation delay decided by the transmission medium, L is the size of each packet, and R is the data rate in the transmission channel.

The jitter θ in (1) is decided according to the background traffic, network topology and routing protocol considered in NS-3.

2) Delay model generated with the stochastic WAC delay model: The stochastic WAC delay model proposed in [23] depends on several manually-set parameters. Similar to the co-simulation framework, τ_f is set *a priori*. The transmission delay τ_p is represented with a sawtooth function is shown in Fig. 4, and is defined by the transmission period T and the data packet loss rate p. The jitter θ is assumed to be Gamma distributed and changes for each data packet. The Gamma distribution is defined by a scale factor a and a shape factor b.

B. Comparison of Delay Models

Consider the following settings of the WAC delay in NS-3:

- The fixed delay $\tau_f = 50$ ms, considering the PMU reporting rate at 25 frames per second PMU time, namely 40 ms for each packet extra 10 ms for data processing [36].
- The PMU-sent data packet size in this simulation is set to 100 Bytes.
- A Point-to-Point link is utilized to connect PMUs to Phasor Data Concentrators (PDCs). The data rate is set as 5 Mbps, and the propagation delay of the channel is 5 ms.
- A Carrier Sense Multiple Access (CSMA) link is utilized to connect PDCs to the control center. The CSMA link simulates the high-speed Ethernet network; the data rate is set as 34 Mbps, and the propagation delay of the channel is 2 ms.
- As the CSMA channel is established to simulate the high-speed Ethernet channel, other data are simultaneously transferred over this network. The RTU data and the video surveillance data streams are considered as the background traffic. The destination of these background traffic is the same as PMU data.
- Assume the communication network is weak for a high packet dropout rate.

Note that, in the remainder of the paper, the delay model obtained with the co-simulation is called "Ethernet delay", as it is based on a model of a high-speed Ethernet network, whereas the model defined in [23] is called "stochastic WAC delay."

With above settings, NS-3 generates a Ethernet delay with packet loss rate 19.04%, magnitude of transmission delay of each packet $\tau_{p,\max} = 23.8$ ms, mean jitter $\bar{\theta} = 3.05$ ms. The corresponding settings for the stochastic WAC delay model are the following: $\tau_f = 50$ ms, $T = 23.8$ ms, $p = 19.04\%$, $a = 3.05/2$ ms, and $b = 2$. Sample trajectories of the Ethernet delay and the stochastic WAC delay are shown in Fig. 5.

According to Fig. 5, the Ethernet delay model and the mathematical delay model proposed in [23] show small but not negligible differences. The major reason for these differences is the modeling of the jitter θ. In the co-simulation framework,

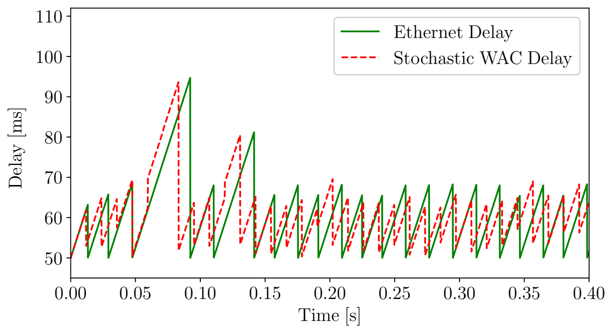

Fig. 5: Time-varing wide-area communication delay models.

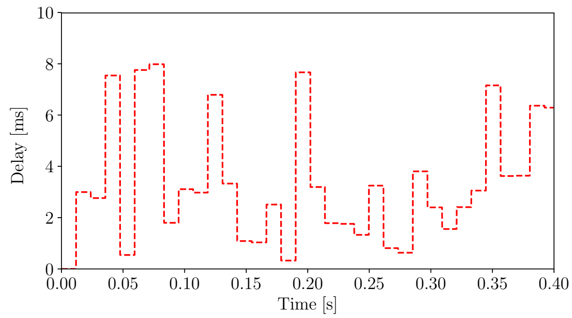

Fig. 6: Gamma distributed delay jitter.

the network-induced delay is a consequence of the background traffic, network topology and routing protocol. While in the stochastic WAC delay model, the jitter is simplified with a gamma-distributed stochastic value for each data packet. To clarify the simplified jitter in the mathematical model, Fig. 6 depicts the corresponding gamma-distributed jitter of the stochastic WAC delay shown in Fig. 5. In the Ethernet delay model, the jitter is regarded as a part of the sawtooth delay as explained in Fig. 4, which better mimics actual WAC delays.

C. Power System Time-domain Simulations

This subsection compares the impact of the two delay models discussed above on a real-world power system, i.e., the all-island Irish transmission system that consists of 1479 buses, 1851 transmission lines, 176 wind power plants, 22 conventional synchronous power plants, and 6 power system stabilizers (PSSs).

The feeding signals of the PSSs included in the all-island Irish system model are assumed to be obtained from the wide-area networks with the WAC delays discussed in Section III-B. The contingency is the outage of the synchronous power plant connected to bus 1378. The time step of time domain simulation is 1 ms. The other settings of DOME are the same as [23].

The all-island Irish power system has a very good stability margin. Therefore, to study the effect of the difference of communication delays generated by the co-simulation and stochastic model proposed in [23], the gains of the PSSs are artificially increased 70 times, thus leading to a high sensitivity of the dynamic response of the system to the delays.

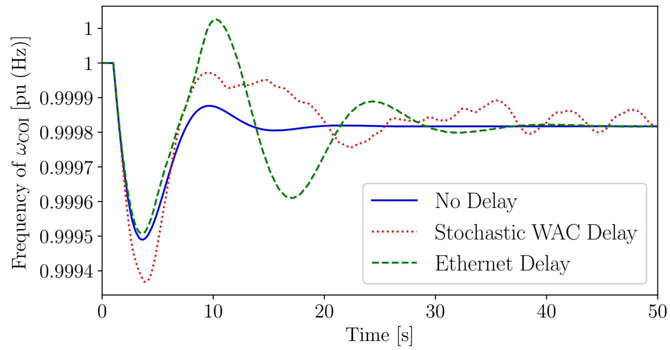

Fig. 7: Transient behavior of the frequency of the COI for the all-island Irish power system following a power plant outage, with high PSS gains.

Figure 7 shows the transient behavior of the frequency of the center of inertia for the Irish system for various scenarios without and with inclusions of the delays. The running time for the delayed scenario under the co-simulation framework is 593 s and 451 s for DOME implemented with the stochastic WAC delay model.

Compared with the no-delay scenario, both delay models impact on the stability of the power system. However, their impact is significantly different. The scenario tested under the co-simulation framework damps the dynamic oscillation within 50 s, while the scenario considering the model proposed in [23] shows an irregular behavior due to the stochastic jitter included in the model.

The co-simulation framework appears to be a promising tool to study the impact of WAC delays on power system dynamics but clearly has a higher computational burden with respect to delay models that are directly embedded into the power system equations. This co-simulation framework can be thus utilized as a guideline to develop mathematical models that better resemble real-world communication delays, since the related references and measurement data are very limited.

IV. CONCLUSIONS

This paper proposes a framework which integrates the simulations of power systems and communication networks. The co-simulation framework is implemented using DOME and NS-3. Compared to other co-simulators, the proposed framework allows implementing customized devices to model specific devices. Such a framework is able to properly simulate a large-scale model of the all-island Irish power system with inclusion of wide-area communication delays.

We believe that the co-simulation framework presented in this paper has great potential. Future work will focus on further developing this framework, especially to design more complex communication network topologies and technologies (e.g., 5G) for power system analysis, as well as optimizing the performance of the overall co-simulation framework.

REFERENCES

[1] F. Milano, "A Python-based software tool for power system analysis," in *Power and Energy Society General Meeting (PES), 2013 IEEE*. IEEE, 2013, pp. 1–5.

[2] G. F. Riley and T. R. Henderson, "The ns-3 network simulator," in *Modeling and tools for network simulation*. Springer, 2010, pp. 15–34.

[3] J. W. Stahlhut, T. J. Browne, G. T. Heydt, and V. Vittal, "Latency viewed as a stochastic process and its impact on wide area power system control signals," *IEEE Transactions on Power Systems*, vol. 23, no. 1, pp. 84–91, 2008.

[4] A. Phadke and J. Thorp, "Communication needs for wide area measurement applications," in *Critical Infrastructure (CRIS), 2010 5th International Conference on*. IEEE, 2010, pp. 1–7.

[5] M. Chenine and L. Nordstrom, "Modeling and simulation of wide-area communication for centralized pmu-based applications," *IEEE Transactions on Power Delivery*, vol. 26, no. 3, pp. 1372–1380, 2011.

[6] S. Wang, X. Meng, and T. Chen, "Wide-area control of power systems through delayed network communication," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 2, pp. 495–503, 2012.

[7] K. Hopkinson, X. Wang, R. Giovanini, J. Thorp, K. Birman, and D. Coury, "Epochs: a platform for agent-based electric power and communication simulation built from commercial off-the-shelf components," *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 548–558, 2006.

[8] H. Lin, S. S. Veda, S. S. Shukla, L. Mili, and J. Thorp, "Geco: Global event-driven co-simulation framework for interconnected power system and communication network," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1444–1456, 2012.

[9] S. Ciraci, J. Daily, J. Fuller, A. Fisher, L. Marinovici, and K. Agarwal, "Fncs: a framework for power system and communication networks co-simulation," in *Proceedings of the Symposium on Theory of Modeling & Simulation-DEVS Integrative*. Society for Computer Simulation International, 2014, p. 36.

[10] B. Palmintier, D. Krishnamurthy, P. Top, S. Smith, J. Daily, and J. Fuller, "Design of the helics high-performance transmission-distribution-communication-market co-simulation framework," in *Proc. 2017 Workshop on Modeling and Simulation of Cyber-Physical Energy Systems, Pittsburgh, PA*, 2017.

[11] J. Nutaro, P. T. Kuruganti, L. Miller, S. Mullen, and M. Shankar, "Integrated hybrid-simulation of electric power and communications systems," in *IEEE Power Engineering Society General Meeting*, 2007, pp. 1–8.

[12] J. Nutaro, "Designing power system simulators for the smart grid: Combining controls, communications, and electro-mechanical dynamics," in *Power and Energy Society General Meeting, 2011 IEEE*. IEEE, 2011, pp. 1–5.

[13] T. Godfrey, S. Mullen, D. W. Griffith, N. Golmie, R. C. Dugan, and C. Rodine, "Modeling smart grid applications with co-simulation," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*. IEEE, 2010, pp. 291–296.

[14] K. Zhu, M. Chenine, and L. Nordstrom, "Ict architecture impact on wide area monitoring and control systems' reliability," *IEEE transactions on power delivery*, vol. 26, no. 4, pp. 2801–2808, 2011.

[15] C. Davis, J. Tate, H. Okhravi, C. Grier, T. Overbye, and D. Nicol, "Scada cyber security testbed development," in *Power Symposium, 2006. NAPS 2006. 38th North American*. IEEE, 2006, pp. 483–488.

[16] M. Mallouhi, Y. Al-Nashif, D. Cox, T. Chadaga, and S. Hariri, "A testbed for analyzing security of scada control systems (tasscs)," in *Innovative Smart Grid Technologies (ISGT), 2011 IEEE PES*. IEEE, 2011, pp. 1–7.

[17] L. Lugaric, S. Krajcar, and Z. Simic, "Smart cityplatform for emergent phenomena power system testbed simulator," in *Innovative Smart Grid Technologies Conference Europe (ISGT Europe), 2010 IEEE PES*. Citeseer, 2010, pp. 1–7.

[18] W. Li, A. Monti, M. Luo, and R. A. Dougal, "Vpnet: A co-simulation framework for analyzing communication channel effects on power systems," in *Electric Ship Technologies Symposium (ESTS), 2011 IEEE*. IEEE, 2011, pp. 143–149.

[19] V. Liberatore and A. Al-Hammouri, "Smart grid communication and co-simulation," in *Energytech, 2011 IEEE*. IEEE, 2011, pp. 1–5.

[20] D. Anderson, C. Zhao, C. Hauser, V. Venkatasubramanian, D. Bakken, and A. Bose, "Intelligent design" real-time simulation for smart grid control and communications design," *IEEE Power and Energy Magazine*, vol. 10, no. 1, pp. 49–57, 2012.

[21] H. Georg, C. Wietfeld, S. C. Müller, and C. Rehtanz, "A hla based simulator architecture for co-simulating ict based power system control and protection systems," in *Smart Grid Communications (SmartGridComm), 2012 IEEE Third International Conference on*. IEEE, 2012, pp. 264–269.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[23] M. Liu, I. Dassios, G. Tzounas, and F. Milano, "Stability analysis of power systems with inclusion of realistic-modeling of wams delays," *IEEE Transactions on Power Systems*, 2018.

[24] F. Milano, "An open source power system analysis toolbox," *IEEE Transactions on Power systems*, vol. 20, no. 3, pp. 1199–1206, 2005.

[25] "ARPACK – Arnoldi Package," available at www.caam.rice.edu/software/ARPACK.

[26] "SLEPc – Scalable Library for Eigenvalue Problem Computations," available at www.grycap.upv.es/slepc.

[27] F. Milano and R. Zárate-Miñano, "A systematic method to model power systems as stochastic differential algebraic equations," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4537–4544, Nov 2013.

[28] F. Milano, "Small-signal stability analysis of large power systems with inclusion of multiple delays," *IEEE Transactions on Power Systems*, vol. 31, no. 4, pp. 3257–3266, July 2016.

[29] ——, *Power System Modelling and Scripting*. London: Springer, 2010.

[30] ——, "Semi-implicit formulation of differential-algebraic equations for transient stability analysis," *IEEE Transactions on Power Systems*, vol. 31, no. 6, pp. 4534–4543, Nov 2016.

[31] L. Vanfretti and F. Milano, "Facilitating Constructive Alignment in Power Systems Engineering Education using Free and Open Source Software," *IEEE Trans. on Education*, vol. 55, no. 3, pp. 309–318, 2012.

[32] "The ns-3 network simulator,," available at https://www.nsnam.org/.

[33] S. Nikolaev, E. Banks, P. D. Barnes Jr, D. R. Jefferson, and S. Smith, "Pushing the envelope in distributed ns-3 simulations: one billion nodes," in *Proceedings of the 2015 Workshop on ns-3*. ACM, 2015, pp. 67–74.

[34] S. Smith, D. R. Jefferson, P. D. Barnes Jr, and S. Nikolaev, "Improving per processor memory use of ns-3 to enable large scale simulations," in *Proceedings of the 2015 Workshop on ns-3*. ACM, 2015, pp. 60–66.

[35] B. Naduvathuparambil, M. C. Valenti, and A. Feliachi, "Communication delays in wide area measurement systems," in *System Theory, 2002. Proceedings of the Thirty-Fourth Southeastern Symposium on*. IEEE, 2002, pp. 118–122.

[36] I. S. Association, "Ieee standard for synchrophasor data transfer for power systems," *IEEE Std C*, vol. 37.