Development of HELICS-based High-Performance Cyber-Physical Co-simulation Framework for Distributed Energy Resources Applications

Jianhua Zhang[1,2]*, Jeff Daily[3], Ryan A. Mast[4], Bryan Palmintier[2], Dheepak Krishnamurthy[2], Tarek Elgindy[2], Anthony Florita[2] and Bri–Mathias Hodge[2,5]
[1]Department of Electrical and Computer Engineering, Clarkson University, Potsdam, NY 13699, USA
[2]National Renewable Energy Laboratory (NREL) Golden, CO 80401, USA
[3]Pacific Northwest National Laboratory (PNNL), Richland, WA 99352, USA
[4]Lawrence Livermore National Laboratory (LLNL), Livermore, CA 94550-9234, USA
[5]Department of Electrical and Computer Engineering, The University of Colorado, Boulder, Colorado 80309, USA
Email: jzhang@clarkson.edu, mast9@llnl.gov,{Bryan.Palmintie, Dheepak.Krishnamurthy, Tarek.Elgindy, Anthony.Florita, Bri.Mathias.Hodge}@nrel.gov

***Abstract*—The rapid growth of distributed energy resources (DERs) has prompted increasing interest in the monitoring and control of DERs through hybrid smart grid communications resulting in the typical smart grid cyber-physical system. To fully understand the interdependency between them, we propose to integrate the Network Simulator 3 (NS3) into the High Engine for Large-Scale Infrastructure Co-Simulation (HELICS), a new open-source, cyber-physical-energy co-simulation platform. This paper aims to the development and case study of the HELICS-based high performance distribution-communication co-simulation framework for the DER coordination. The novel co-simulation framework for the NS3 integrating into the HELICS is developed. The DER monitoring application about hybrid smart grid communication network design is simulated and validated on this proposed HELICS-based cyber-physical co-simulation platform.**

***Index Terms*—co-simulation, cyber-physical system, distributed energy resources (DER).**

I. Introduction

The increasing penetration of distributed energy resources (DERs) into the distribution grid has prompted great interest and multiple development efforts in the area of monitoring and control of millions of DERs at the levels of the distribution system operator (DSO) and transmission system operator (TSO) [1]–[3]. DERs refer to distributed renewable generation, e.g. rooftop solar photovoltaic (PV) panels, small wind turbines, residential battery energy storage systems, and controllable loads. The emerging concept of smart distribution networks indicates that DERs need to been effectively monitored and controlled to achieve the autonomous distribution grid with two features: 1. optimizing energy performance of DERs to address stochastic and dynamic challenges, 2. supporting grid services of frequency and voltage regulation [4]. Thus, new control, protection and communication systems are required. As a groundbreaking cyber-physical energy system, the smart distribution grid is relying more heavily on hierarchical and distributed monitoring and control systems with greater dependency on a variety of communication systems [5]. However, we lack modeling and simulation capabilities for both industries and academia to understand the inter-dependency among Transmission, Distribution and Communication (TDC) in the planning and operation of such smart distribution grid.

To this purpose, the power system and communication network co-simulation for smart grid applications has been proposed and studied as a promising solution to investigate the interactions among different infrastructures because of the balanced trade-off between the investment of developing the brand-new simulator from the scratch and benefits of taking advantage of existing stand-alone, well-trusted simulators. Specifically, dedicated power system and communication network co-simulations, such as PSLF transmission grid simulator-Network Simulator 2 (NS2), PSCAD-NS2, OpenDSS-NS2, GridLAB-D-NS2, OpenDSS-NS3, were developed with dedicated co-simulation approaches, and they face a number of challenges, particularly, such as full convergence among simulators and scalability issue due to the size and complexity of the practical power grid [6]–[10]. Co-simulation framework across three infrastructures of power system, Artificial Intelligence (AI), and communication networks were proposed by implementing the dedicated Python-based coordination script [11]. To overcome the limitations of convergency and scalability, we have proposed the generic, open-source, cyber-physical energy co-simulation framework named High Engine for Large-scale Infrastructure

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding is provided by U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Solar Energy Technologies Office. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government.

Co-Simulation (HELICS), which is to support large-scale co-simulation with off-the-shelf power system, communication, market, and AI-based end use simulators [12].

As the part of the HELICS development, this paper proposes an innovative co-simulation framework to integrate the communication simulator, NS3, into the HELICS platform to enable the large-scale cyber-physical co-simulation. The main contribution of this paper is threefold: 1) the co-simulation framework between the NS3 simulator and the HELICS platform is developed and implemented; 2) a co-simulation application of the DER monitoring through hybrid communication networks at the utility-scale distribution grid is developed; 3) an assessment of impact of a suite of hybrid communication network designs on the DER monitoring in terms of latency and packet loss rate (PLR) is conducted. Note that this paper focuses on the DER monitoring application of rooftop solar PV panels because it is pervasive in the distribution grid. The proposed co-simulation platform can be extended into other smart grid cyber-physical applications.

II. HELICS Framework Design

To better understand transmission distribution and communication (TDC) inter-dependency and have confidence in deploying systems that will meet, or improve upon, current reliability, efficiency, and cost-effective benchmarks, the flexible and scalable open-source co-simulation HELICS platform have been designed and developed through the efforts of multiple national laboratories [12]. The HELICS architecture overview, data exchange and time management, and three interface functions are briefly summarized in what follows. Note that one simulator refers to a federate and a group of simulators in the co-simulation is called the federation.

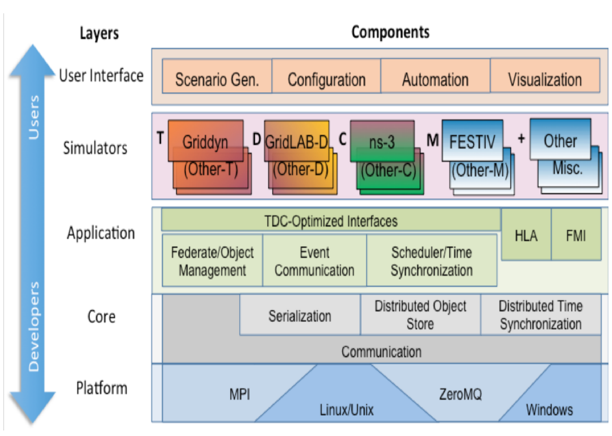

Fig. 1. Layered Design Architecture of the HELICS Platform [12]

A. Architecture of the HELICS Platform

The HELICS architecture, as shown in Fig. 1, is designed through a layered approach. It consists of five layers from the bottom up: 1) Platform layer, it enables HELICS working across multiple operating systems and multiple computational scales through Message Passing Interfaces (MPI) and the fast messaging library, ZeroMQ; 2) Core layer, where two essential mechanisms: data exchange and time management/synchronization, are implemented for the combined discrete event (e.g. communication network) and time series (e.g. power system) simulation; 3) Application layer, it supports two types of low-level interfaces: a) general-purpose interfaces, including High Level Architecture (HLA) and Functional Mockup Interface (FMI) co-simulation interfaces, for arbitrary user-provided federates, such as customized optimization/AI controllers; and b) optimized interfaces for the common TDC+M simulator type. These interfaces enable application federates interacting with the co-simulation framework; 4) Simulator layer, it supports a diversity of off-of-shelf simulators, such as transmission simulators (e.g., GridDyn, PSS/E), distribution simulators (e.g., GridLAB-D, OpenDSS, CYMDIST), communication simulators (e.g., NS3), market simulators (e.g., FESTIV), and customized controllers, through two key data exchange extensions; and 5) User interface layer, it provides tools and standardized approaches to assemble all required input data, organize and run the co-simulation, and parse the results.

B. Data Exchange and Time Management

The design of two critical features of the cyber-physical co-simulation: data exchange and time synchronization/management, plays an important role in the run-time mechanics. In the HELICS framework, the inter-federate data exchange design consists of two mechanisms: 1) standardized data exchange pattern, including variable naming, types, and timing/synchronization. It enables flexible data exchange among three types of federates in terms of data types: value federate, message federate, and message filter federate; and 2) end-point construct, it models value based and message based interaction resulting in the end-point communication. Each federate needs to register publish/subscribe end-points and explicit pairwise communication among end-points.

Time management is a key capability to allow federates/simulators to operate with different timescales and co-iterate at any time step to achieve convergence in the physical simulation. In the HELICS platform, the core coordinates among different federates to determine which federate can proceed in an ordered manner through the global synchronization mechanism. At each federate, two global synchronization calls are employed to enter initialization and execution stages of the co-simulation, then time advancement mechanism is used to indicate the core that the federate is ready to proceed to the next time step or that it requires co-iteration exchange.

C. TDC-optimized Interface Functions

The Application layer has three interface functions: federate/object management, event communication, and scheduler/time synchronization to enable TDC+M simulators co-simulating with the HELICS platform. The federate/object management is used to manage the registration information of each simulator when it joins the co-simulation. The event communication function is responsible for the data exchange among federates. And the scheduler/time synchronization function enables time synchronization among different simulators with different time steps and scheduling all events in an ordered manner during the co-simulation.

The purpose of this paper is to develop the co-simulation framework for NS3 integrating to the HELICS platform. The HELICS framework design rational and TDC-optimized interfaces enable the conduction of this study.

III. Co-Simulation Framework of HELICS-NS3

Based on the design and development of the large-scale HELICS platform, the co-simulation framework between the HELICS platform and the Network Simulator 3 is proposed to enable the general-purpose cyber-physical energy system simulation. It is designed in terms of three TDC-optimized interface functions provided by the HELICS platform and described in the above Subsection C, because the NS3 is a common style of simulator. The proposed co-simulation framework, shown in Fig.2, consists of three modules: HELICS-NS3 helper, HELICS-customized simulator, and HELICS-customized applications, whose design rational and implementation are discussed in detail as below.

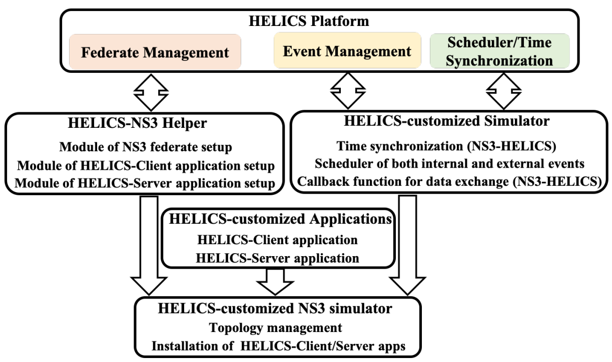

Fig. 2. Co-Simulation Framework Design of HELICS-NS3

A. HELICS-NS3 Helper Module

The HELICS-NS3 helper module is designed to achieve the first interface function of federate management. It is to create the NS3 federate and join the federation. It consists of three sub-modules: 1) HELICS-helper, it creates the NS3 message federate, which connects to the HELICS core, and sets the NS3 time-step; 2) HELICS-client-helper, it helps to declare the new client application container which allows the installation of the end-point object, and 3) HELICS-server-helper, it is similar with HELICS-client-helper.

B. HELICS-customized Simulator Module

To enable the co-simulation between the NS3 and HELICS platform, the default standalone NS3 simulator has to be redesigned to host three vital functions: time synchronization, scheduler for both internal and external events, and data exchange. In the default single process simulator, the NS3 simulation runs its own timer. The time synchronization function makes sure that the NS3 simulation timer is synchronized with the global timer of the HELICS platform. In the HELICS-NS3 co-simulation, NS3 internal events refer to regular communication events, such as packet acceptance, packet read, sending/receiving packet, and schedule of next packet transmission, and NS3 external events mainly refer to send/receive HELICS messages. It requires that the HELICS scheduler should be designed to handle both internal and external events properly. In the HELICS co-simulation, each federate exchanges the data with the HELICS platform, instead directly with the destination federate. From the NS3 point of view, the way of sending/receiving HELICS messages is a design challenge, because the end-point mechanism is different from the external interface provided by the NS3 simulator. The callback function of the end-point object is designed to invoke the sending/receiving of HELICS messages, and it is required to install at the communication node.

As the vital coordinator in the NS3, the HELICS-customized simulator module is specifically developed to implement the above three functions by combining the HELICS-NS3 helper module. At the first step, the NS3 simulator runs at the co-simulation mode instead of the standalone simulation mode by setting HELICS-customized simulator. Secondly, at the initialization stage, the NS3 message-type federate is created to set up the time synchronization with the HELICS, and to install the end-points. Finally, the NS3 federate enters the execution stage and executes the loop, within which, when the internal event list is empty, the simulator executes the external event of exchanging the HELICS message by conducting the time advancement scheme. This process enables the proper way of scheduling internal and external events.

C. HELICS-customized Application Module

Two HELICS-customized applications are developed for three separate purposes. To accommodate the DER coordination application, a HELICS-customized client module is developed to collect the real DER data packet from the power system federate through the HELICS platform, and send them out immediately to the destination server node. The server module is to receive the packet, and responsible for the autonomous online tracing and data post-processing, statistically collecting network performance metrics. The third purpose is to register the end-point object in the client/server application for the HELICS message exchange.

IV. Case Study of Hybrid Communications Effects on DER Monitoring

The Distributed Energy Resources (DERs) monitoring critically depends on the performance of hybrid communications networks for visualizing power systems. The distribution and communication co-simulation (HELICS + GridLAB-D + NS3) enables the design and evaluation of hybrid communications systems. Thus, we implement the open-loop case study about the hybrid communications design effects on DER monitoring. In this section, the Reference Test Case A (RTC-A), implementation of D+C HELICS co-simulation, and evaluation results are presented and discussed in what follows.

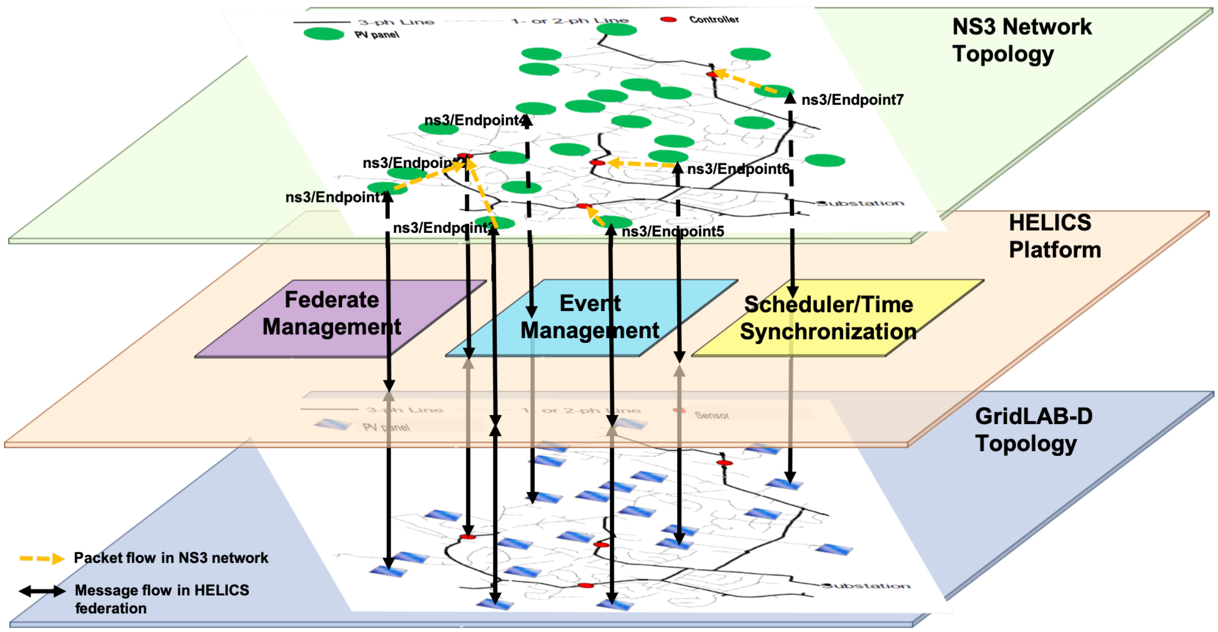

Fig. 3. Implementation Architecture of DER Monitoring Co-Simulation

A. Reference Test Case A

The taxonomy feeder titled R2-25.00-1 containing 1,080 nodes, referred to as Reference Test Case A (RTC-A), has been selected for the hybrid communications architecture testing. Penetration of 10% solar uptake is modeled on this feeder. The location and availability of the existing communications infrastructure is modeled using data from the specific municipal utility district, which has rolled out smart meters across their utility network and uses the dedicated networks for data communications infrastructure. Typical smart meter installation rates and placement of data concentrators that have been built for current smart meter communications requirements are scaled to the R2-25.00-1 feeder. Therefore, modified communications infrastructure of the RTC-A consists of 51 PV nodes in yellow dots, 275 smart meters in blue dots, 10 data concentrators in big-size green dots, one edge router, as shown in Fig. 4. All PV nodes and smart meter nodes are divided into 10 neighborhood areas.

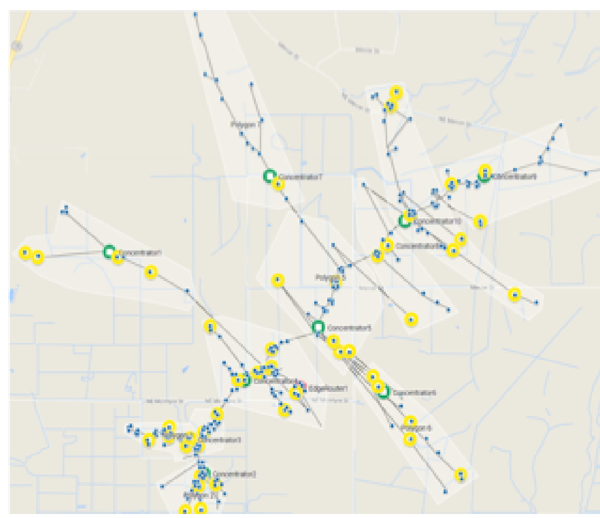

Fig. 4. Power Grid and Communication Infrastructures of RTC-A

B. Implementation of DER Monitoring

This co-simulation consists of GridLAB-D and NS3 simulators on the HELICS platform. The implementation architecture is shown in Fig. 3. The RTC-A taxonomy feeder with 10% PV penetration is modeled in the GridLAD-D simulator, as shown in the bottom level of Fig. 3. The GridLAB-D federate is enabled by adding the connection module and the *helics_msg* object. This object is configured with the JSON input file where the physically installed location of publish end-points and the corresponding NS3 end-points are defined. For the hybrid communications network, we consider the low-power wireless personal area networks (LoWPAN) within the home-area network consisting of the smart meter and the attached PV panel and both WiFi and Ethernet cable communication options for the neighborhood area which consists of a bundle of smart meters and a data concentrator. Thus, there are LoWPAN-WiFi and LoWPAN-Ethernet hybrid designs. In the top level, the corresponding network topologies are simulated in the NS3 simulator. In the NS3 simulation script, the network topology is created, the HELICS-customized client application is installed into each PV node, and the HELICS-customized server application is installed in each data concentrator node. It is worthy to note that the NS3 network topology exactly overlays on the power system topology in the GridLAB-D model. The time-step for both simulators is set to 1.0 second. In the middle-level, the HELICS broker is set to ZeroMQ.

In the co-simulation, each PV meter data at the GridLAB-D federate is recorded and sent to the HELICS platform through three interface functions, as shown in the middle level of Fig.3. At the NS3 federate, each PV node equipped with the global end-point object, receives the corresponding PV meter data and then transmits them to the destination data concentrator through the hybrid communications network. The HELICS message flow is shown as the black solid double-arrow line and the packet flow in the NS3 communication network is shown as the yellow solid double-arrow line in Fig. 3. The server application at each data concentrator receives the PV meter data and collects the statistical information about the network performance in terms of the packet loss rate and latency. The co-simulation runs for the 5 minutes with 300 packets and 20 minutes with 1200 packets, respectively.

C. Result Analysis

In this subsection, the impact of hybrid communications design on the DER monitoring in terms of the network performance metrics of latency and PLR are examined in detail. Fig. 5 and 6 show the average latency and packet loss rate performance at each neighborhood area for two hybrid designs and two simulation periods. In Fig. 5, each bar indicates the average latency value in microsecond. In Fig.6, the empty bar for some neighborhood area indicates the PLR is zero. There are two interesting observations: 1) the average latency performance are satisfactory, namely less than the critical latency of 300 ms, in all ten neighborhood areas, regardless the simulation period and the hybrid designs; 2) However, for the packet loss rate performance, the Lowpan-WiFi design has the worse performance.

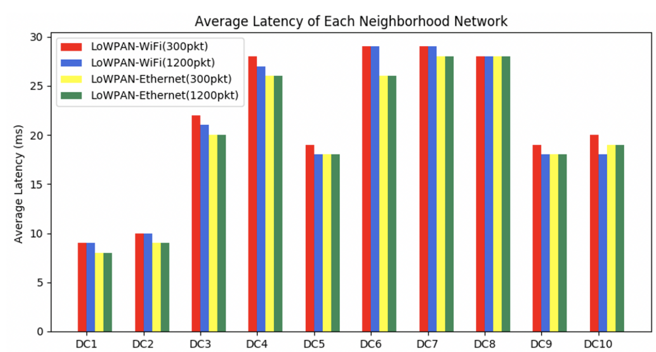

Fig. 5. Latency Performance at Each Data Concentrator Area

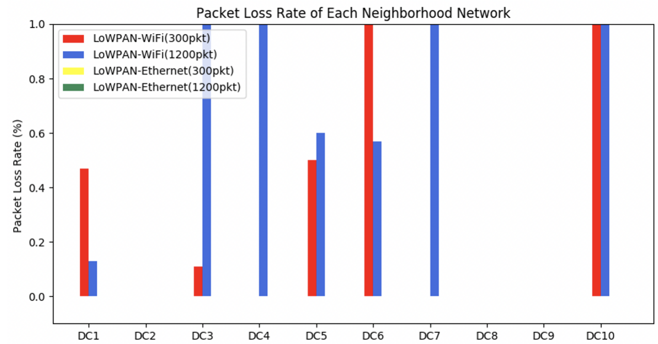

Fig. 6. Packet Loss Rate at Each Data Concentrator Area

V. Conclusion

In this paper, we propose a new co-simulation framework for the NS3 integrating to the HELICS platform to enable the real cyber-physical energy simulation for the evaluation of emerging DER applications. The proposed co-simulation framework features with the scalability and high-performance through the design rational of the HELICS platform. The case study about hybrid communications design for the DER monitoring is conducted on this platform. Both the hybrid communications design and the co-simulation framework are validated this way. Future research includes the implementation of DER control algorithms as the closed-loop case study to further validate the HELICS-based cyber-physical co-simulation platform.

References

[1] "Wind and solar data projections from the u.s energy information administration: Past performance and planned enhancements," July 2017. [Online]. Available: https://www.eia.gov/outlooks/aeo/supplement/renewable/

[2] "The global annual market for the department of distributed solar pv plus energy storage is expected to exceed $49 billion by 2026," May 2017. [Online]. Available: https://www.navigantresearch.com/newsroom/the-global-annual-market-for-the-deployment-of-distributed-solar-pv-plus-energy-storage-is-expected-to-exceed-49-billion-by-2026

[3] M. G. Molina, "Distributed energy storage systems for applications in future smart grids," in *2012 Sixth IEEE/PES Transmission and Distribution: Latin America Conference and Exposition (T D-LA)*, Sept 2012, pp. 1–7.

[4] D.Giustina and S. Rinaldi, "Hybrid communication network for the smart grid: Validation of a field test experience," *IEEE Transcation on Power Delivery*, vol. 10, no. 1, pp. 251–261, March 2015.

[5] M. Zajc, M. Kolenc, and N. Suljanovic, "Virtual power plant communication system architecture," *Smart Power Distribution Systems*, pp. 231–250, 2018.

[6] K. Hopkinson, Xiaoru 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, May 2006.

[7] Hua Lin, S. Sambamoorthy, S. Shukla, J. Thorp, and L. Mili, "Power system and communication network co-simulation for smart grid applications," in *ISGT 2011*, Jan 2011, pp. 1–6.

[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, Sep. 2012.

[9] A. Razaq, B. Pranggono, H. Tianfield, and Hong Yue, "Simulating smart grid: Co-simulation of power and communication network," in *2015 50th International Universities Power Engineering Conference (UPEC)*, Sep. 2015, pp. 1–6.

[10] M. Garau, G. Celli, E. Ghiani, F. Pilo, and S. Corti, "Evaluation of smart grid communication technologies with a co-simulation platform," *IEEE Wireless Communications*, vol. 24, no. 2, pp. 42–49, April 2017.

[11] I. Ahmad, J. H. Kazmi, M. Shahzad, P. Palensky, and W. Gawlik, "Co-simulation framework based on power system, ai and communication tools for evaluating smart grid applications," *2015 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)*, pp. 1–6, 2015.

[12] 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 *2017 Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, April 2017, pp. 1–6.