Co-Simulation Framework based on Power System, AI and Communication Tools for Evaluating Smart Grid Applications

Ishtiaq Ahmad, Jawad Haider Kazmi, Mohsin Shahzad
Energy Department
AIT Austrian Institute of Technology
Vienna, Austria
ishtiaq.ahmad.fl@ait.ac.at

Peter Palensky
Electrical Engineering, Mathematics and Computer Science
TU Delft, Netherland

Wolfgang Gawlik
Institute of Energy Systems and Electrical Drives
TU Wien, Austria

***Abstract*—Numerical simulation has become essential tool of the design process and is widely used for the validation and evaluation of smart grid solutions and applications. However, domain specific tools are able to model a system only for a particular domain i.e., electric power system tool can model only power systems and for a intelligent control it has no provision to design AI based decision making algorithms, which can be developed in Multi-agent System tool. Thus for in-depth realistic analysis a co-simulation approach is required for simulating multi-domain systems. This paper presents a framework for co-simulation with the tools from three different domains covering power system, artificial intelligence and communication to simulate and analyse the smart grid applications. DigSILENT PowerFactory, a dedicated power system simulator, is interfaced with JADE (Java Agent Based Environment), an AI tool, using Application Programming Interface (API) through Python script. OMNeT++ was used as a communication simulator. Problem of voltage stabilization in power distribution network was addressed using Multi-Agent System (MAS) based approach for the validation of the proposed framework. Results show that co-simulation approach is effective for the realistic analysis of smartgrid applications.**

***Index Terms*—Co-simulation, Multi-Agent System, ICT, Distributed Control, Power Distribution Network.**

I. Introduction

De-regulation, security of supply, increased used of renewables, environmental considerations and rapid growth in ICT are the main motivation behind smart grid. Major smart grid functionalities include; advanced metering infrastructure, demand response, electric vehicles, wide-area situational awareness, distributed energy resources/storage and distribution grid management. Many smart grid applications are developed to fulfill the objective of intelligently integrating the behaviour and actions of all users to ensure sustainable, economic, and secure electricity supply. Before deployment of these intelligent solutions extensive simulations are required for analyzing the expected results. This highlights the importance of tools and methods used for simulation because application of these solutions heavily depend upon the results obtained after simulation. Currently many computer tools are available for power system simulations and one can model power system in detail for the analysis purpose. However, these specialized software tools lack modeling capability of other features of smart grid applications like intelligent control and communication which put the question mark on validation of results, as in case of smart grid applications there is tight coupling between power system, communication and intelligent control [1]. Figure 1 shows intelligent control as merger of the three areas while smart grid is combination of application from three domains i,e., power system, communication and intelligent control. As smart grid is considered to be a complex system with growing interconnection and heterogeneous nature of generation sources as well as operation paradigms, Multi-Agent Systems (MAS) application has attracted various researchers. MAS is a combination of intelligent control and distributed systems [2].

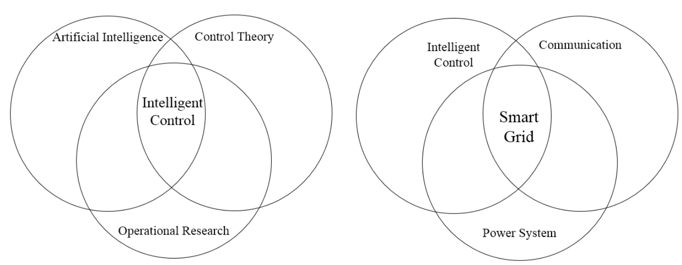

Fig. 1. Intelligent control and smart grid as combination of application from various domains

Above mentioned facts demand for development of a framework where applications from different domains can be modeled in detail and simulate in a realistic manner. An alternate is to develop a simulation tool from scratch with capabilities of modeling and simulating all the desired components of smart grid i.e., power system, Intelligent control and communication. However, this approach seems not a realistic due to the time and cost involved. Moreover, complex requirements like different timestep, and nature of they system (continuous or

AIT, Austrian Institue of Technology, Austria discrete event) make it further complicated and difficult task to build a single tool capable of simulating systems from different domains. A counter approach to avoid these difficulties is to extend the features of existing individual simulation tools to enable the simulation of all involved domains. Such an approach can be useful for simple scenario but might not produce correct results for complex scenario [3]. Based on these arguments, co-simulation has become a promising technique where different didicated tools are made to work together for simulating different parts of overall complex system simultaneously [4].

The rest of the paper is organized as follows: A brief literature review of co-simulation of power system, MAS and communication tools is presented in Section II. Section III provides detail of the used tools and methods for the proposed framework. Section IV describes a simple smart grid application example simulation and results while conclusions are made in section V.

II. Related Work

A lot of efforts are being done to develop co-simulation framework for smart grid applications. There are standard co-simulation interfaces like High Level Architechture (HLA), Functional Mockup Interface (FMI) and Distributed Interactive Simulation (DIS) developed for purpose of large scale system model integration. There main application is in automotive industry, space and defense projects. However learning curve for the HLA, FMI and DIS is steep and considerably long [5].

Ref. [6] developed a simulation engine EPOCHS that combines PSCADE/EMTDC, PSLF and NS2 together with an agent component. This simulator was claimed to be the first to combine realistic network communications with electric power components by the authors. Run-Time Infrastructure (RTI) was used to routes all messages between simulation components and manages simulation times while distributed wide area control and protection schemes were implemented through agents. Ref. [7] developed interface between PowerWorld (power system simulator) and JADE (agent environment) through TCP/IP and used MATLAB as simulation controller. No communication tools was used in the framework to model communication network. Ref. [8] proposes GridIQ, a simulation framework for co-simulation of JADE and PSAT (Power System Analysis Tool). The GridIQ acts as interface between the two tools and execution of the agent network and power system analysis is interleaved. However, communication between agents was considered to be ideal i.e., no communication delay between agents, which makes it unrealistic. Ref. [9] presented a software framwork MACSim which co-simulate MAS application and communication network. Here, JADE was used as MAS environment and OPNET as network simulator but no power system simulation tool was considered. Similar co-simulation framework MACSimJX was developed in [10] which provides modeling and development of multiagent driven control systems. Simulink, a tool used for control systems development and JADE, an environment for developing agents were used.

VPNET is proposed in [1], which is a co-simulation of all three area i.e., power sytem, communication networks and MAS. It consists of three parts: Virtual Test Bed for simulation of power systems, OPNET for the simulation of communication networks, and the Co-simulation Coordinator. VPNET is first known attempt for co-simulation of all the three components of smart grid. However, data exchange and time synchronization between the other two simulators were carried out using co-simulation coordinator. The co-simulation coordinator was used for the MAS integration.

Literature review reveals that in almost all the efforts for developing smart grid co-simulation framework either MAS tool or Communication tool is missing. This paper presents co-simulation of all the three tools from intelligent agents, power system and communication domains to validate smart grid solutions.

III. Tools and Methods

This section provides the details of power system tool, MAS tool and communication network tool used for the framework like capabilities, possible integration techniques and advantages. Methods used for interfacing of tools is also presented in this section.

A. Tool Selection

While selecting tool various aspects needs to be analyzed and evaluated on the basis of its intended use. These aspects include tool capabilities, availability, ease of use, efficiency etc [11]. Table I describes time, size and complexity scales in power system, MAS and communication tools.

B. Power System Tool

1) Description: DigSILENT PowerFactory was used as a power system simulator. It is a powerful simulator capable of modeling analysis and simulation of complex power system models and scenarios. Some of the features are Network Models, Data Management, Results and Reporting, External Data Format Support, Scripting Languages. It has many functionalities from basic Power Flow Analysis, Fault Analysis, Voltage Stability Analysis, Load Flow Sensitivities, Contingency Analysis to advanced functionalities like Protection Functions, Distribution Network Optimization, Harmonic and Power Quality Analysis, Optimal Power Flow (OPF), Techno-Economical Calculation, Reliability Analysis, State Estimation, Quasi Dynamic Simulations [12].

2) Interface Methods: Power factory also provides many options for coupling with external tools. Figure 2 describes all possible techniques that can be used to couple PowerFactory with external tools [13]. In this work, Python API was used to integrate PowerFactory with other tools as it gives full access and control of all the objects, features and functionalities of PowerFactory.

TABLE I
SCALES FOR THE THREE TOOLS

Scale	Tools		
	Power System	MAS System	Communication
Time	Transient static	Discrete (packets) Continuous	MAS Protection reconfiguration
Size	single node multi-node	LAN WAN	Single Agent Multi-Agent
Complexity	Balanced/unbalanced Symmetrical/Asymmetrical	Link layer Application layer	Monitoring Intelligent decision

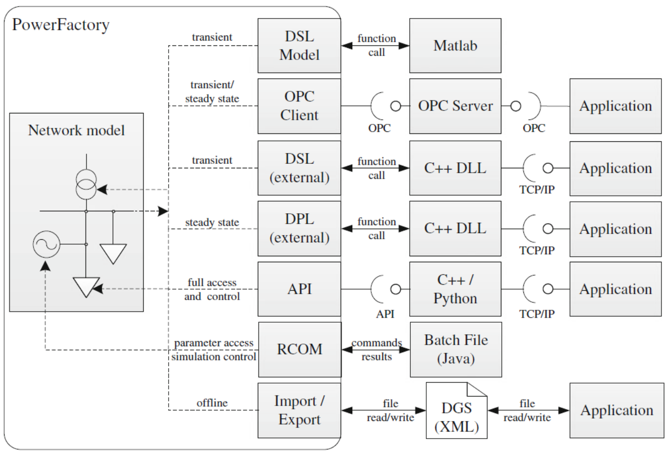

Fig. 2. Integration methods of DigSILENT PowerFactory with external tools

C. MAS Environment Tool

1) Description: Java Agent DEvelopment Framework (JADE) was used as Multi-Agent System modeling and simulation which is free open source JAVA based platform independent software and is distributed by Telecom Italia [14]. It provides a run-time environment to execute agent programs as well as a development framework which aids developers to create agents. JADE has a nice Graphical User Interface (GUI) that support the debugging and deployment phases. It provides agent abstraction along with agent task execution and composition model, peer to peer agent communication based on the asynchronous message passing paradigm. It also has directory services which support publishing subscribe discovery mechanism and many advanced features of distributed system [15] . JADE comply with the FIPA (Foundation for Intelligent Physical Agents) specifications for interoperable intelligent multi-agent systems. FIPA is an IEEE computer society standard organization for the agent based technology. Figure 3 shows architecture of JADE.

2) Interface Methods: The JADE (API) is a part of JADE implementation architecture and allows to use the JADE engine in any application that can execute JAVA .jar flies. JADE can be interfaced with other application running in a JAVA Virtual Machine through this API. It provides access to classes and methods of JADE. The other option is to use sockets for data transfer and sharing. JADE contained `SocketProxyAgent` which can be used to send/receive data/message on multiple sockets in FIPA standard Agent Communication Language (ACL) format [16]. However, it can connect to 50 clients at a time. Further, an agent in JADE can be implemented to send/receive data on socket through JAVA sockets and authors in [7] used this method.

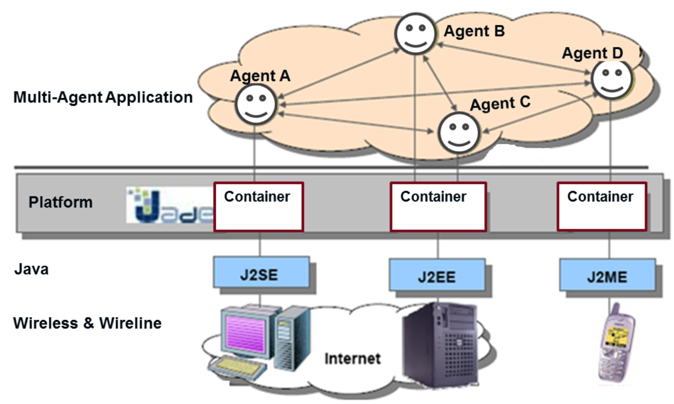

Fig. 3. JADE Architecture [15]

D. Communication Modeling Tool

1) Description: OMNeT++ [17] (Objective Modular Network Testbed in C++) is an open-source, moduler, extendable and component-based discrete event simulation environment for modeling the communication systems. In the proposed co-simulation, the modeling of the communication system is carried out using the tool. Its a widely used simulation tool for the modeling of the communication network and the protocols. There are many communication simulation models available and among the popular ones are INET, INETMANT, MiXiM etc. A *module* is the basic building block in an OMNeT++ model which communicate with other modules by passing `messages` through *gates*. Each model can contains one or more of such models. The modules supports a hierarchically structure. There can be two types of models; a *simple module* or the *active module* is programmed in C++ and described with the NED (Network Description) language while a compound module is created by *connecting* more than one simple module. An OMNeT++ *network* is a special type of compound module having no gates [18].

2) Interface Methods: OMNeT++ can be extended to include custom functionality for example, to enable real-time and hardware-in-the-loop (co-)simulations. One way to do it is by replacing the simulation scheduler with a custom one, according to the application needs. Another possibility is to embed OMNeT++ simulation kernel in a simulation environment by using the API (Application Programming Interface).

E. Co-Simulation Methods

1) Co-Simulation setup: Figure 4 shows the co-simulation set up for the three tools. DigSILENT Power Factory was used for electric grid simulation while JADE as a Agents simulation. OMNeT++ was used for communication modeling and simulation.

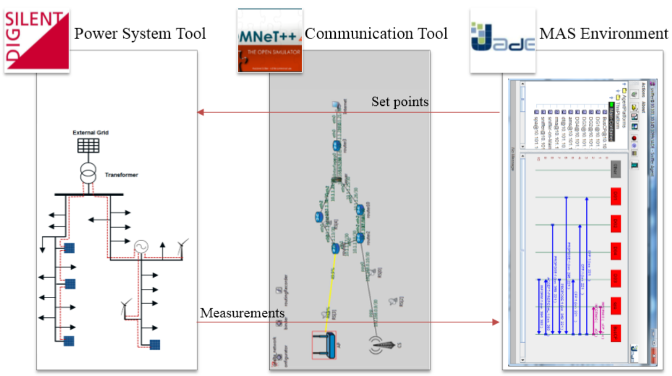

Fig. 4. Co-simulation set up.

2) Interfacing Methods used in Co-simulation: Co-simulation framework was developed in a way that it can also be used for hardware-in-loop (HIL) simulations. Thus it was developed to run real time simulations in which three simulators run in parallel in real-time and exchange data as and when required. A script was written in Python to interface PowerFactory with JADE and OMNeT++. PowerFactory instance was controlled by a Python script which uses sockets to comunicate with other tools. A custom OMNeT++ `scheduler` developed in [19] was used for interfacing OMNeT++ and running it in real-time. `SocketProxyAgent` in JADE was used as interface over sockets.

IV. Framework Application Example and Results

A. Voltage Regulation Problem

Problem of voltage regulation in distribution network with distributed generation was considered. When voltage at a certain point goes out of limit, voltage support can be provided by the distributed generators by injection(absorption) of reactive power output. Problem can be formulated as:

Min $f(Q) = \sum_{m}^{j=1} Q_j$

st. $S_i = f_i$ power flow at each node $i = 1...n$

$0.95 \leq V_i \leq 1.05$ voltage constraint at each node

$0 \leq Q_j \leq Q_j^{max}$ var limits for each DG

The cost of dispatch in the above presented objective function is the total amount of reactive power required from all Distributed Generators(DGs).It is assumed that voltage regulator setting remains unchanged as this emergency support by DGs is much faster than the voltage regulator response.

1) Power System Model: In order to test the proposed framework, Modified IEEE 13 node test feeder system with distributed generation was considered [20]. Other data like load, generations and line parameters were adopted from [21] with some modifications. There are six generation buses and only three DGs are able to provide voltage support services (inject/observe reactive power). Distributed generators are supposed to be connected through a communication link and able to change their reactive power (absorption or injection) while measurements are taken from some critical points. Distribution generators are considered to be intelligent agents which can negotiate/communicate with each other to fulfill their objective (voltage stabilization in this case). Power network was build in DigSILENT PowerFactory and agents are modelled in JADE agent environment.

2) Communication Network: A wide area network was considered keeping in view the distance between the Distributed Generators. Communication model has been implemented in OMNeT++ and is shown in Figure 5.

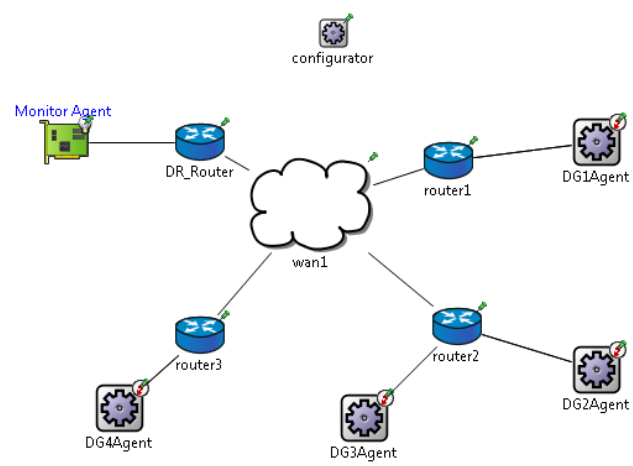

Fig. 5. Communication Model in OMNeT++

3) Agent Modeling: Two types of agents were considered. Monitoring agent and Distributed Generator agent (DG agent). Monitoring agents are considered to be deployed at the critical point and monitor the bus voltage. In case of violation in voltage limits, Monitor agent is responsible for initiating the process of voltage stabilization and communicate with DG agents through sending/receiving messages using algorithm explained in the next section.

4) Control Algorithm: In MAS application challenging part is devising a policy for agent interaction. In this test case

Contract-Net Protocol (CNP) based algorithm was used for agent interaction. In case of voltage violation, Monitor Agent sends RFP (Request For Proposal) message to all the DG agents. DG agents reply with their current reactive power, limits of reactive power and voltage sensitivity w.r.t reactive power change. Monitor agent decides set point for the DG Agents on the basis of sensitivity and again sends messages to the selected DG agents to update their reactive power output. Algorithm is explained in detail in [22].

B. Results

Following three cases were simulated using the proposed framework.

1) Base Case: Base case was simulated with out considering any voltage support by the DGs. One of the load bus was taken as a target bus (critical point). Figure 6 shows that after about fifty seconds voltage at target bus drop from the permissible limit and thus voltage support is required at this point.

2) Voltage Support by DGs with ideal communication: In this scenario it was assumed that there is ideal communication between agents and result of voltage support process is shown in Figure 6, which reflects that voltage is in permissible range after voltage support process. Figure 7 shows agent communication in JADE. CNP protocol was used for interaction between agents while DG for voltage regulation was selected on the basis of sensitivity of voltage w.r.t reactive power. Detail of algorithm is given in [22] .

3) Voltage Support by DGs considering communication network: In this scenario a communication network was used for communication between agents as described in section communication network. Figure 8 shows the results after considering communication. It is clear that communication has effects on performance of algorithm. This example shows that co-simulation approach is effective in studying effects of communication on power networks. Further , communication design can be changed as per requirement of the application/algorithm and its effects can be analysed before deployment in the field.

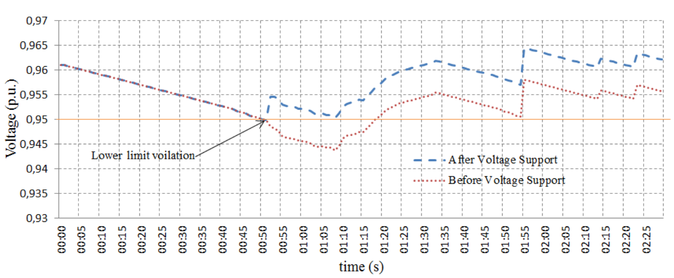

Fig. 6. Voltage at target bus before and after voltage support by DGs

V. Conclusions

A co-simulation framework comprising of power system, MAS and communication tool is introduced in this paper. Possible interface options and capabilities of the individual

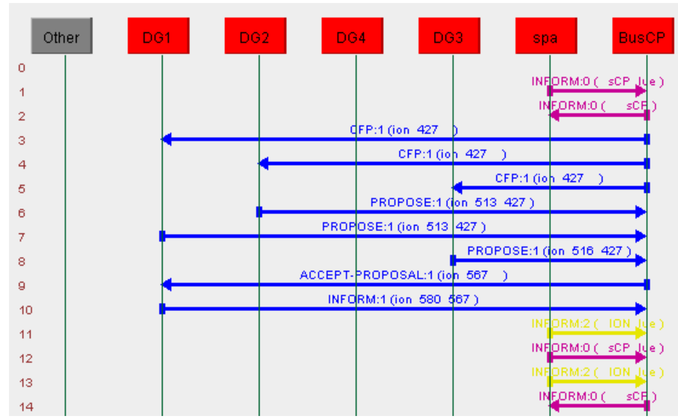

Fig. 7. Interaction between agents in JADE

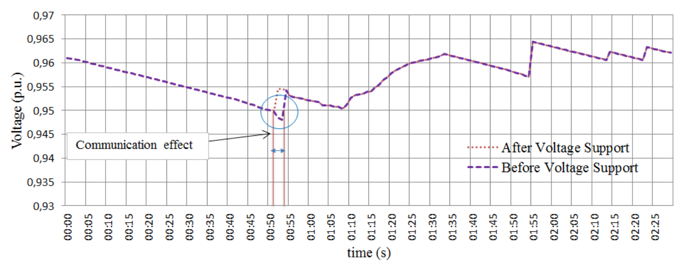

Fig. 8. Voltage support with and without communication

tools were discussed. Problem of voltage regulation in power distribution network was simulated as an example application to test the proposed co-simulation approach. Comparison of results were made to analyse co-simulation approach. Results indicate that, this approach is very effective in analysing the coupled effects between power system and communication. Although , it is quite intuitive that communication may effects on results , however effects need to be studied and will depend upon nature of application and communication network parameters. Co-simulation approach is quite useful in this case. Moreover it is very convenient to co-simulate different existing tools instead of building a single tool with all the capabilities (power system modelling, intelligent control and communication) which is time consuming and expensive. Future work will be to test this framework in laboratory with Hardware-In-Loop simulation.

References

[1] W. Li, M. Luo, L. Zhu, A. Monti, and F. Ponci, "A co-simulation method as an enabler for jointly analysis and design of mas-based electrical power protection and communication," *Simulation*, p. 0037549712472939, 2013.

[2] A. Saleem, "Agent based control of electric power systems with distributed generation," Ph.D. dissertation, Department of Electrical Engineering, Centre for Electric Technology (CET) , Technical University of Denmark, 2010.

[3] F. Ponci, A. Monti, and A. Benigni, "Simulation for the design of smart grid controls," in *Smart Grid Modeling and Simulation (SGMS), 2011 IEEE First International Workshop on*, Oct 2011, pp. 73–78.

[4] W. Li, X. Zhang, and H. Li, "Co-simulation platforms for co-design of networked control systems: An overview," *Control Engineering Practice*, vol. 23, pp. 44 – 56, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0967066113001925
[5] C.-H. Yang, G. Zhabelova, C.-W. Yang, and V. Vyatkin, "Cosimulation environment for event-driven distributed controls of smart grid," *Industrial Informatics, IEEE Transactions on*, vol. 9, no. 3, pp. 1423–1435, Aug 2013.
[6] 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," *Power Systems, IEEE Transactions on*, vol. 21, no. 2, pp. 548–558, May 2006.
[7] R. Roche, S. Natarajan, A. Bhattacharyya, and S. Suryanarayanan, "A framework for co-simulation of ai tools with power systems analysis software," in *Database and Expert Systems Applications (DEXA), 2012 23rd International Workshop on*, Sept 2012, pp. 350–354.
[8] C. J. Bankier, "Gridiq - a test bed for smart grid agents," 2010.
[9] F. Perkonigg, D. Brujic, and M. Ristic, "Mac-sim: A multi-agent and communication network simulation platform for smart grid applications based on established technologies," in *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, Oct 2013, pp. 570–575.
[10] C. R. Robinson, P. Mendham, and T. Clarke, "Macsimjx: A tool for enabling agent modelling with simulink using jade," *Journal of Physical Agents*, vol. 4, no. 3, pp. 1–7, September 2010.
[11] R. Verma, A. Gupta, and K. Singh, "Simulation software evaluation and selection: a comprehensive framework," *Journal of Automation and System Engineering*, 2008.
[12] D. GmbH. Digsilent powerfactory. [Online]. Available: http://www.digsilent.de/index.php/products-powerfactory.html
[13] M. Stifter, F. Andrén, R. Schwalbe, and W. Tremmel, "Interfacing powerfactory: Co-simulation, real-time simulation and controller hardware-in-the-loop applications," in *PowerFactory Applications for Power System Analysis*. Springer, 2014, pp. 343–366.
[14] T. Italia. Java agnet development framework (jade). [Online]. Available: http://jade.tilab.com/
[15] F. L. Bellifemine, G. Caire, and D. Greenwood, *Developing multi-agent systems with JADE*. John Wiley & Sons, 2007, vol. 7.
[16] *FIPA ACL Message Structure Specification*, FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS Std.
[17] http://www.omnetpp.org/, OMNeT++ Homepage.
[18] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in *Simutools '08: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, 2008, pp. 1–10.
[19] M. Stifter, J. Kazmi, F. Andre?n, and T. Strasser, "Co-simulation of power systems, communication and controls," in *Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES), 2014 Workshop on*, April 2014, pp. 1–6.
[20] D. T. Feeders, "Ieee pes distribution system analysis subcommittee," *OnlineAvailable: http://www. ewh. ieee. org/soc/pes/dsacom/testfeeders/index. html*, 2011.
[21] M. Brenna, E. De Berardinis, L. Delli Carpini, F. Foiadelli, P. Paulon, P. Petroni, G. Sapienza, G. Scrosati, and D. Zaninelli, "Automatic distributed voltage control algorithm in smart grids applications," *Smart Grid, IEEE Transactions on*, vol. 4, no. 2, pp. 877–885, June 2013.
[22] I. Ahmad, P. Palensky, and W. Gawlik, "Multi-agent system based voltage support by distributed generation in smart distribution network," in *2015 International Symposium on Smart Electric Distribution Systems and Technologies (EDST) CIGRE SC C6 Colloquium, Vienna, Austria - September 8-11, 2015 (accepted)*, 2015.