Proceedings of the 2017 $4^{th}$ International Conference on Advances in Electrical Engineering
28-30 September, Dhaka, Bangladesh

# Enhancing Engineering Studies in Developing Countries using OpenModelica

Mohammed Ahsan Adib Murad*, Luigi Vanfretti†, Mohammad Rokonuzzaman‡, and Rashedul Amin Tuhin§
*School of Electrical & Electronic Engineering
University College Dublin, Dublin, Ireland
†Rensselaer Polytechnic Institute, Troy, NY, USA
‡University of Asia Pacific, Dhaka, Bangladesh
§East West University, Dhaka, Bangladesh

***Abstract*—Laboratory experiences are crucial for the students learning in higher educational institutions, and are specially required to ensure a high standard of education for engineering programs. However, for universities in developing countries, such as Bangladesh, it is difficult to provide state of the art laboratory facilities and associated support functions for specialized areas such as electrical power systems, mechanical engineering, etc., due to high capital and operational costs related to complex/dedicated equipment. In this paper, we propose the use of an open-standard-based modeling language and an open-source simulation software tool. The language used is Modelica, an equation-based object-oriented modeling language, and the open source simulation software used is OpenModelica. To illustrate the potential use of Modelica and OpenModelica when physical laboratories are unavailable and there is no access to proprietary software, different power system dynamic analysis studies are illustrated using test models from the OpenIPSL library.**

***Index Terms*—Modeling, Simulation, Modelica, Open Source Software, OpenModelica.**

## I. Introduction

### A. Motivation & Literature Review

Bangladesh, with a population of over 160 million people, is one of the most densely populated countries in the world. Effective education of this vast human resource is one of the keys to the development of the country. Higher Education Institutes (HEIs) carry this crucial responsibility. The number HEIs has skyrocketed since 1992 from 14 to 131 [1], as shown in Fig. 1. Because of the globalization of the economy, almost all the universities in Bangladesh provide engineering education to promote innovation, productivity and overall socio-economic progress, a comprehensive overview of engineering education in Bangladesh are presented in [2]. The objectives of concurrent engineering programs in Bangladesh are to develop the country's knowledge and skill base to apply methods and resources in addressing Bangladesh's own development challenges [3].

Ensuring a high-quality education while the number of engineering students increases is challenging because of several factors, insufficient funding being the most relevant one at both private and public universities, even after being highly subsidized by the Board of Trustees (BoTs) and the government. Due to this limitation, only a few number of universities have up-to-date laboratory facilities. On the other hand, computer-based laboratories are easier to setup, with access to relevant software packages, or even through the internet [4]. Computer-based simulations are one viable alternative to conventional laboratory experiences that can help, in theory, reduce costs while at the same time enhancing engineering students' learning. However, due to the specialization of the different engineering disciplines, the majority of the computer-based simulation tools are proprietary, imposing costly licenses that are well out of the economic reach for most of the universities in Bangladesh. An overview of commercial and open-source simulation tools given in [5] w.r.t domain specific energy systems, while Open Source Software (OSS) power system tools are discussed in [6].

While some proprietary tools offer "student versions" or "educational versions" free-of-cost, there are important drawbacks of their use for educational purposes. First, the tools come with technological locks that limit their applicability to "real-life" problems. Second, they subjugate the user by creating a tool-specific dependency, through a technological lock-in to their own models/data or know-how. Lastly, but more importantly, there are ethical, philosophic and socio-political implications in the use of proprietary software that students are not aware of; imposing the use a proprietary tool without choice is a questionable practice, especially when done at universities [7]. Realizing such issues, to enhance education quality in developing countries, various free and open source software for education are adopted, one great initiative taken by India is Free and Open Source Software for Education (FOSSEE) project [8]. This paper aims to illustrate how the Modelica language [9] and the OpenModelica tool can be used as free/libre and OSS alternative for educational purposes.

Modelica is a standardized equation-based open source object-oriented modeling language, used for Hybrid Differential Algebraic Equations (HDAE) based complex mathematical models, supports multi-domain descriptions i.e. systems that mix different kinds of subsystems, such as electrical, mechanical, chemical, etc. Its development started from the late

978-1-5386-0869-2/17/$31.00 

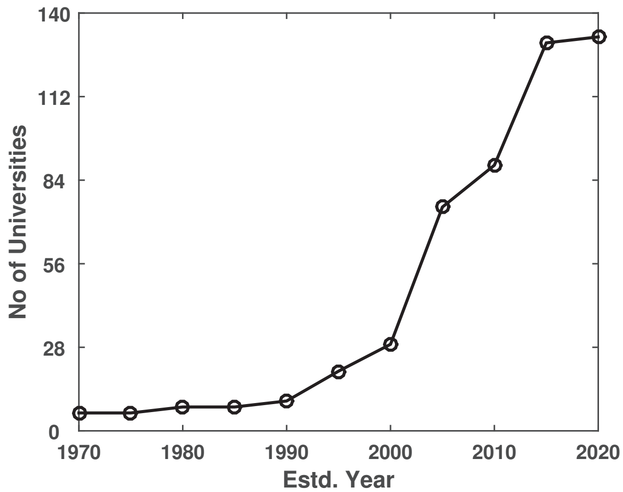


Fig. 1. University growth in Bangladesh.

Nineties and by this time this language is already well accepted by several engineering communities. The OpenModelica software tool offers an environment to model and simulate cyber physical systems using the Modelica language [11].

### *B. Paper Contribution*

The contributions of this paper are:

- To present the prospect of Modelica as open standard modeling language for universities in developing countries like Bangladesh for computer based laboratory experiments, undergraduate and graduate thesis work and research.
- To demonstrate examples of power system analysis networks implemented using OpenModelica as a proof of concept, to demonstrate the capabilities of this simulation tool.

### *C. Paper Organization*

The remainder of this paper is organized as follows. Section II provides a brief overview of Modelica language, standard libraries and OpenModelica modeling and simulation environment. Section III presents aspects of Modelica for modeling energy systems, while Section IV includes examples of power system networks modeled and simulated using OpenModelica. Finally, in Section V, conclusions are drawn.

## II. The Modelica Modeling Language

### *A. Overview of Modelica*

The Modelica language is used for mathematical modeling of complex cyber physical systems and their simulation. It differs in many ways from traditional simulation tools:

- Modelica allows modeling by declaring equations rather than using assignment statements.
- Its modeling approach is acausal which differs from block diagram representations in other simulation tools, is that there is no need to describe the input output relations between variables of different components.
- It allows multi-domain modeling and simulations, so physical objects from different domains can be connected together.

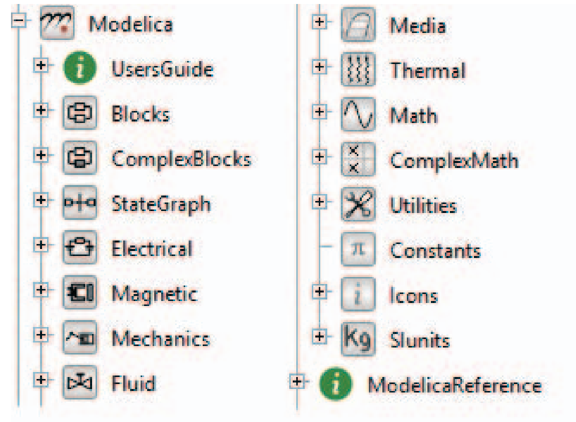


Fig. 2. Modelica Standard Library (MSL) packages.

Due to this unique approach modeling, Modelica offers several advantages: it allows multidisciplinary modeling, offers large standard free library known as the Modelica Standard Library and the models are solver independent. Apart from these, the Modelica language syntax is equation based, and thus it is very easy to learn and program for engineering students. Two seminal books [11] and [12] includes all the details about Modelica language, its specifications, syntax, examples etc which can be a very useful introduction to this language.

*1) Modelica Standard Libraries:* The standardized library package `Modelica` is provided together with the Modelica language, which are developed and maintained by Modelica Association and is called the Modelica Standard Library (MSL). In the MSL constants, types, connector classes, model classes of various application areas are available. Some example of these application areas are: Electrical, Magnetic, Mechanics, Thermal and others shown in Fig. 2. The open source MSL contains about 1600 model components and 1350 functions from different domains [9].

*2) Modelica Simulation Environments:* There are commercial and open-source simulation environments available to simulate the models developed using Modelica. A few of the commercial tools are: Dymola, MapleSim, SimulationX, Wolfram SystemModeler; while the main free-of-cost and OSS tools are: OpenModelica [13], JModelica [14].

*3) Modelica Free Libraries:* Modelica user community grew very rapidly in last decade and it became popular to different engineering and science community. Around 85 cost-free libraries are available to download from *https://www.modelica.org/libraries* and cost-free to use.

### *B. Open-Modelica*

OpenModelica is a free/libre and open-source Modelica-based modeling and simulation environment intended for industrial and academic usage, developed by the Open Source Modelica Consortium (OSMC) [13]. OpenModelica uses the Advanced Interactive Modelica compiler (OMC) to compile the Modelica code to C for simulation. OpenModelica provides several basic environments for creating models and advanced uses:

- **OMEdit:** OpenModelica Connection Editor (OMEdit) is a graphical user interface for model creation, connection, editing, simulation, and plotting of results. It supports user defined extensions/models both in textual and graphical form.
- **OMNotebook:** OpenModelica Notebook (OMNotebook) is a platform independent, interactive electronic book primarily used for teaching.
- **OMShell:** Interactive OpenModelica Shell (OMShell) is an interactive session handler that parses and interprets commands and Modelica expressions for evaluation, simulation, plotting, etc.

OpenModelica also provides OMDebugger for debugging the implemented models, an optimization tool (OMOptim), a python scripting tool with OpenModelica (OMPython) among several other tools [13].

### *C. Modelica Limitations*

Modelica satisfies all the basic requirements for scientific mathematical studies but yet it has its own limitations. It is not a general purpose programming language e.g. Python rather it is built only for mathematical modeling. For example, Python can be used to web development whereas Modelica is designed only for system modeling.

## III. Modeling and Simulation of Power and Energy System

### *A. Modeling Power Systems using Modelica*

There are several software packages available for power system studies, giving a description of all is clearly out of the scope of this paper. However, the relevance of open source software for power system analysis, research and education is given in [6]. The object of this section, is to show how Modelica is useful to model power systems for the case of electro-mechanical dynamics where the power system model is described by a set of nonlinear DAEs:

$$\begin{aligned} \dot{x} &= f(x, y, t) \\ 0 &= g(x, y, t) \end{aligned} \tag{1}$$

where $x$ and $y$ are state and algebraic variables, $t$ is time, $f$ and $g$ are differential and algebraic equations respectively. Using an implicit numerical integration method to run time domain simulation, for a generic time $t$ assuming a step length $\Delta t$ one has to solve the following problem [16]:

$$\begin{aligned} 0 &= \hat{q}(x(t+\Delta t), y(t+\Delta t), f(t)) \\ 0 &= g(x(t+\Delta t), y(t+\Delta t)) \end{aligned} \tag{2}$$

where $\hat{q}$ is a function that depends on the implicit numerical method. In conventional power system tools due the non linearity of the equations in (2) their solution is typically obtained by using Newton's method which in turn consists of computing iteratively the increments $\Delta x^{(i)}$ and $\Delta y^{(i)}$ of the state and algebraic variables and updating the actual variables given by,

$$\begin{aligned} \begin{bmatrix} \Delta x^{(i)} \\ \Delta y^{(i)} \end{bmatrix} &= -\left[\Delta A_c^{(i)}\right]^{-1} \begin{bmatrix} \hat{q}^{(i)} \\ g^{(i)} \end{bmatrix} \\ \begin{bmatrix} \Delta x^{(i+1)}(t+\Delta t) \\ \Delta y^{(i+1)}(t+\Delta t) \end{bmatrix} &= \begin{bmatrix} \Delta x^{(i)}(t+\Delta t) \\ \Delta y^{(i)}(t+\Delta t) \end{bmatrix} + \begin{bmatrix} \Delta x^{(i)} \\ \Delta y^{(i)} \end{bmatrix} \end{aligned} \tag{3}$$

where $\Delta A_c^{(i)}$ is a matrix, depends on the state and algebraic Jacobian matrices of the system. So, in conventional power system tools, whenever a user wants to implement a custom or user defined model, the user needs to provide all differential equations reduced to index-0 [15] and the elements of the Jacobian matrices. In additions the users need proficient programming skills to develop their own solver too. For example, open source power system simulation tool PSAT received a lot of attention in power system research community [17] but adding a new component in this tool can be difficult and time consuming [18]. With respect to this problem, in Modelica the user only need to provide the $f$ and $g$ equations, after that the Modelica tool automatically performs index-reductions along other tasks and creates the Jacobian matrices for the solver to run time-domain simulations. In addition, the user can linearize the equation based model [19] using a Modelica tool of their choice, and use this result into their environments for their own control system design.

### *B. Scope of use of Modelica in Energy Systems*

Modelica can be used for modeling and simulating a variety of domains that comprise energy systems [20]. It provides a much simpler approach towards modeling any system than conventional techniques, which can be easier for the students to adapt. Modelica can especially be useful for the computer based laboratory experiments related to analyzing power systems which are essential for energy system studies.

Another important aspect needs to be considered is the prospect of research in the field of power and energy systems using Modelica. Currently, the research scope is very narrow in Bangladesh, mainly due to the unavailability of proper laboratories and/or domain specific modeling tools. In this regard, Modelica can be an invaluable resource to researchers in the field of power and energy system in Bangladesh.

## IV. Example

To demonstrate the modeling capabilities of the Modelica language two example of power systems are implemented in OpenModelica and compared the simulation results of the same models implemented in PSAT. The test systems are implemented in Modelica using the OpenIPSL [21] library which is available in online as open source. The OpenIPSL library can be used to perform dynamic simulation of any power system network, all the details about this library is available in [22].

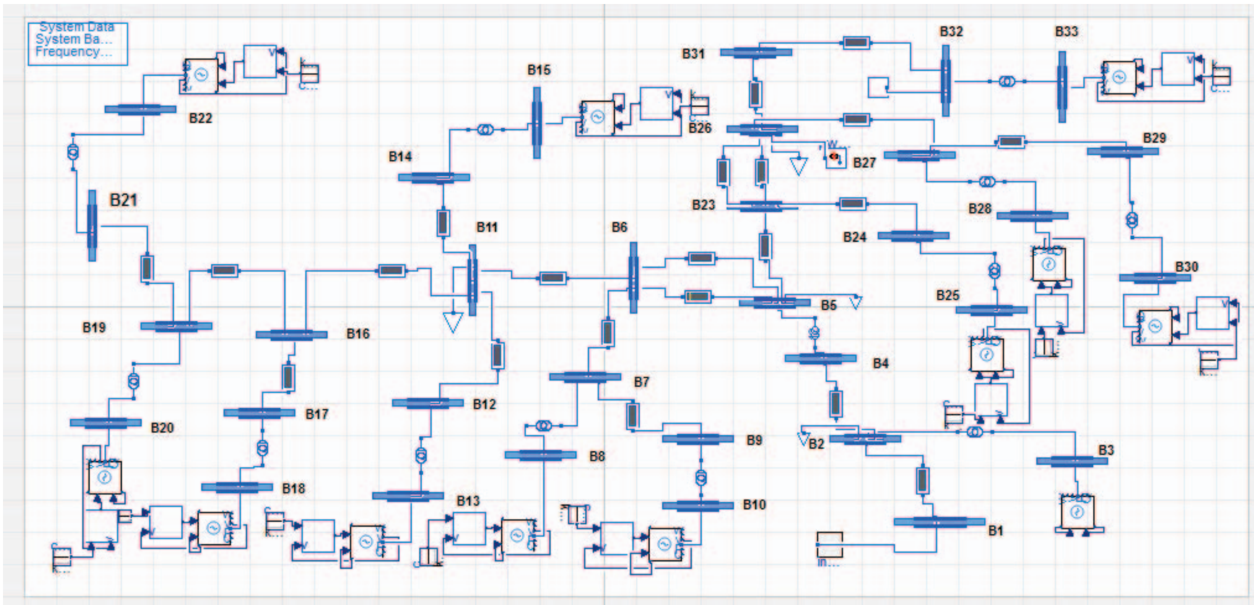


Fig. 3. Namsskogan distribution grid in Modelica using OpenIPSL.

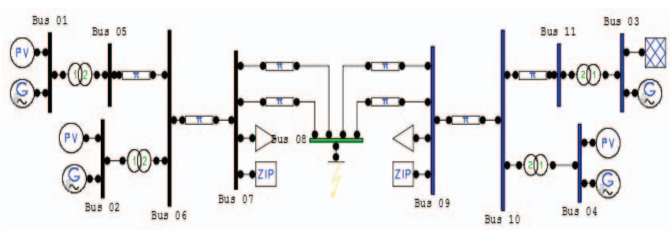


Fig. 4. Two Area system in PSAT.

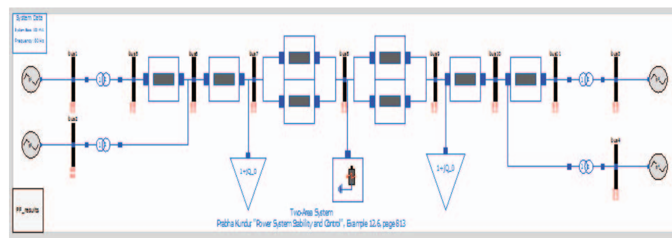

Fig. 5. Two Area system in Modelica using OpenIPSL.

## A. Two-Area Klein-Rogers-Kundur Test System

The Two-Area system test case is taken from Prabha Kundurs textbook [23], originally presented in [24]. The two areas are connected with weak tie lines, and thus this test system is used for different kinds of studies: transient stability, stability enhancement, power exchange, inter-area oscillation analysis and damping. The system contains eleven buses, four generators, four transformers, transmission lines and loads. The two-area test systems in OpenIPSL and PSAT are shown in Figs. 4-5.

The simulation was carried out for the Two-Area test system by applying a three phase fault at Bus 8 in OpenModelica and PSAT. The fault applied at 1s with the duration 50 ms. The fault resistance and reactance used were 0 and 1e-5 respectively. The simulation results are shown in Fig. 7. In Fig. 7 voltage, active and reactive power at Bus 1 are shown. All the variables are in per unit (p.u.).

## B. Distribution Grid

A model of a distribution grid in Norway (Namsskogan) was implemented using both PSAT and OpenIPSL, the system data is taken from [25]. The system has total 33 buses,12 generators, 21 transmission lines, 3 loads and 2 shunt elements and shown in Fig. 3. The distribution grid in PSAT and OpenIPSL were simulated by applying a three phase fault at Bus 26. The simulation results are shown in Fig. 8. The Eigenvalues of the model provide useful information for small signal stability analysis and control design of power systems [16]. Using OMshell and OMNotebook the eigenvalues of this test network were obtained and compared against the eigenvalues of the PSAT. The eigenvalue loci is shown in Fig. 6, the simulation tools provide similar results.

# V. Conclusion

This paper shows the scope and opportunity to use Modelica as a modeling language and the OpenModelica as a simulation tool for analyzing power and energy systems. The Modelica community offers a range of cost-free and open source libraries and tools for different engineering fields, which is why it is highly attractive for universities in Bangladesh and any other developing countries.

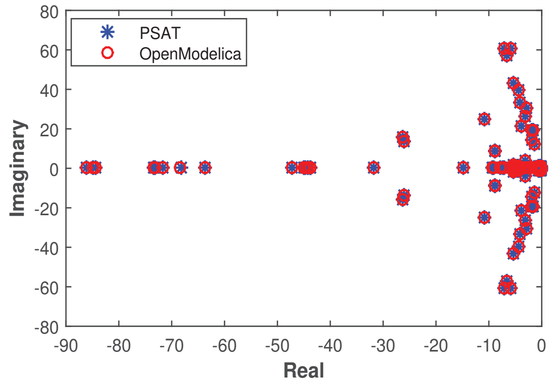


Fig. 6. Eigenvalue loci of Namsskogan Distribution Grid both in Open Modelica using OpenIPSL and PSAT.

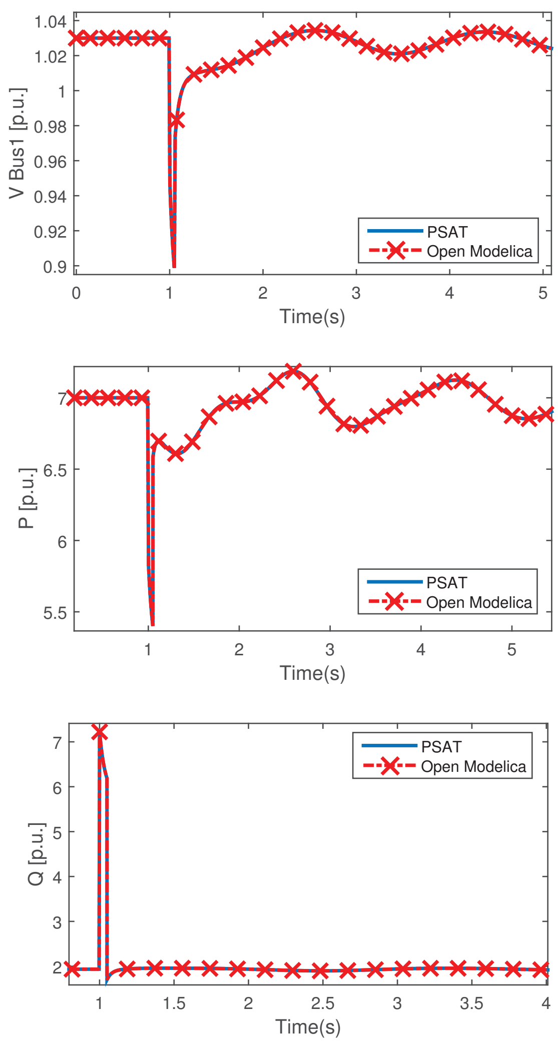


Fig. 7. Simulation result of Two Area system both in Open Modelica and PSAT.

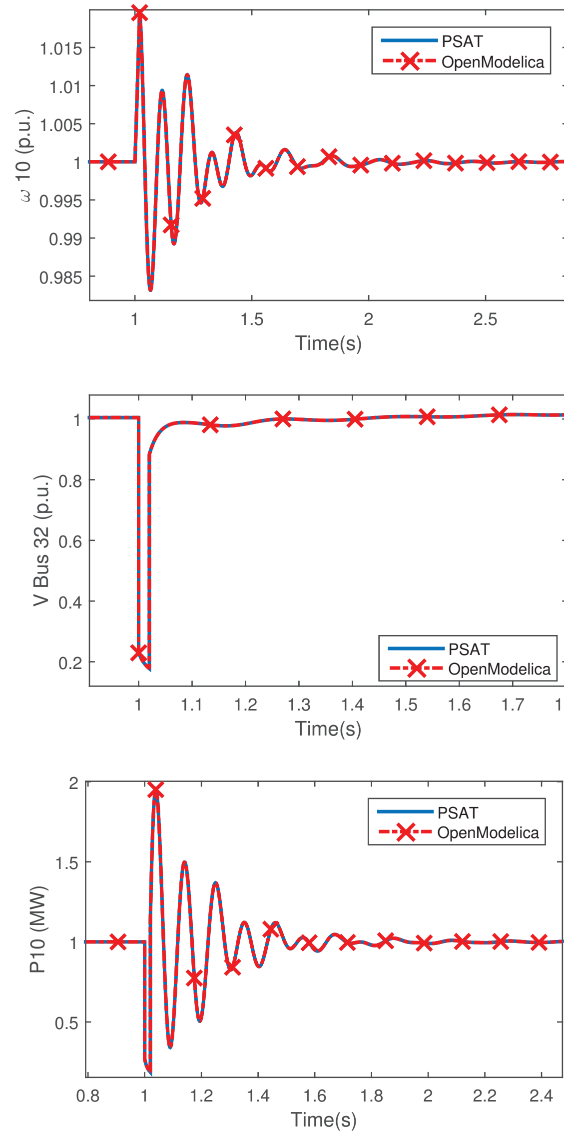


Fig. 8. Simulation result of Namsskogan Distribution Grid both in Open Modelica and PSAT.

## REFERENCES


[1] List of Public Universities, Ugc.gov.bd, 2016. [Available Online] http://www.ugc.gov.bd/en/home/university/public/120

[2] H. Chowdhury and F. Alam, "Engineering education in Bangladesh an indicator of economic development," *European Journal of Engineering Education*, vol. 37, no. 2, pp. 217-228, 2012.

[3] H. Chowdhury, F. Alam, S. Biswas, M. Islam and A. Islam, "Quality Assurance and Accreditation of Engineering Education in Bangladesh," *Procedia Engineering*, vol. 56, pp. 864-869, 2013.

[4] J. Henry, "Running Laboratory Experiments via the World Wide Web,"*In ASEE Annual Conference & Exposition*, Seattle, Washington, 1998.

[5] P. Palensky, E. Widl and A. Elsheikh, "Simulating Cyber-Physical Energy Systems: Challenges, Tools and Methods," *in IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 3, pp. 318-326, March 2014.

[6] F. Milano and L. Vanfretti, "State of the art and future of OSS for power systems," 2009 IEEE Power Energy Society General Meeting, Calgary, AB, 2009, pp. 1-7.

[7] Chopra, S. and S.D. Dexter, "Decoding Liberation: The Promise of Free and Open Source Software," New York: Routledge New York, 2009.

[8] Free and Open Source Software for Education (FOSSEE). [Online] http://fossee.in/

[9] Modelica and the Modelica Association. [Online]

http://www.modelica.org/
[10] Functional Mock-up Interface (FMI). [Available Online] http://fmi-standard.org/
[11] P. Fritzon, “Principles of object-oriented modeling and simulation with Modelcia 3.3: A cyber-physical approach”, Wiley-IEEE Press, 2nd Edition, April, 2015
[12] M. Tiller, “Modelica by Example” [On-line] http://book.xogeny.com
[13] OpenModelica. [Available Online] https://www.openmodelica.org/
[14] JModelica. [Available Online] http://www.jmodelica.org/
[15] F. E. Cellier and H. Elmqvist, “Automated formula manipulation supports object-oriented continuous-system modeling,” *in IEEE Control Systems,* vol. 13, no. 2, pp. 28-38, April 1993.
[16] F. Milano, “Power System Modelling and Scripting”, London: Springer, 2010
[17] L. Vanfretti and F. Milano, “Application of the PSAT, an open-source Software, for Educational and Research Purposes,” *Power Engineering Society General Meeting*,FL, 2007, pp. 1-7.
[18] W. Li, L. Vanfretti and Y. Chompoobutrgool,“Development and implementation of hydro turbine and governor models in a free and open-source software package,” *Simulation Modelling Practice and Theory*, Volume 24, May 2012, Pages 84-102.
[19] M. Baur, M. Otter and B. Thiele, “Modelica Libraries for Linear Control Systems,” *in: Proceedings 7th Modelica Conference,* pp. 593-602. 7th Modelica Conference, 20.-22. Sept. 2009, Italy.
[20] F. Casella, A. Bartolini, S. Pasquini and L. Bonuglia, “Object-Oriented Modelling and Simulation of Large-Scale Electrical Power Systems using Modelica: a First Feasibility Study,” *In Proceedings of the 42nd Annual Conference of the IEEE Industrial Electronics Society IECON,* Firenze, Italy, Oct. 24-27, 2016, pp. 0-6.
[21] L. Vanfretti, T. Rabuzin, M. Baudette and M. Murad, “iTesla Power Systems Library (iPSL): A Modelica library for phasor time-domain simulations,” *SoftwareX*, Volume 5, 2016, Pages 84-88, ISSN 2352-7110.
[22] Open iTesla Power System Library (OpenIPSL). [Available Online] https://github.com/OpenIPSL
[23] P. Kundur, “Power System Stability and Control”, McGraw-Hill, New York, 1994
[24] M. Klein, G. J. Rogers, S. Moorty and P. Kundur, “Analytical investigation of factors influencing power system stabilizers performance,” *in IEEE Transactions on Energy Conversion,* vol. 7, no. 3, pp. 382-390, Sep 1992.
[25] T. Nicolae Preda, “Modelling of Active Distribution Grids for Stability Analysis”, PhD Thesis, NTNU, 2016.