Multi-Domain Power and Thermo-Fluid System Stability Modeling using Modelica and OpenIPSL

Fernando Fachini [1], Marcelo de Castro [2], Mingzhe Liu [3], Tetiana Bogodorova [4], Luigi Vanfretti [5], Wangda Zuo [6]
ECSE - Rensselaer Polytechnic Institute [1,2,4,5] & CEAE - University of Colorado Boulder [3,6]
{fachif [1], decasm3 [2], bogodt2 [4], vanfrl [5]}@rpi.edu, {Mingzhe.Liu [3], Wangda.Zuo [6]}@colorado.edu

***Abstract*—This paper presents the a set of multi-domain load models that allow simulating the dynamics of coalesced electrical power and the thermo-fluid system by exploiting the Modelica language based on the Modelica OpenIPSL power system library. This allows for phasor domain representation of the electrical grid, such as that used in *de facto* power system stability software, to be combined with the electro-mechanical (e.g. motor-drive) and thermo-fluid representation of the load (e.g. heat pumps and pipes). The added dynamics of the thermo-fluid and mechanical interfaces allow for simulating the transient effects of disturbances of the load explicitly by following its own constitutive physics, thereby enabling dynamic interaction between electrical and hydraulic contingencies. The modeled components are described with emphasis on how they are modeled in Modelica and were tested for different electrical and fluid-flow contingencies, demonstrating their usability and their viability in representing higher fidelity multi-domain load systems.**

***Index Terms*—Multi-domain, power and thermo-fluid system, Dynamics, OpenIPSL, Phasor domain representation**

I. INTRODUCTION

The power system with extensive integration of renewable sources evolves not only at the transmission level, but at the distribution level. The latter includes feeders with loads, power generation and storage units forming microgrids that are able to sustain a short time disconnection from the grid. Microgrids are tightly connected with thermo-fluid systems that are usually not covered in power systems research. However, influence of the thermo-fluid on the electrical system, and vice versa, can not be neglected. In [1], the study on the influence of fluidic faults on the resiliency of the electrical grid is presented. Another application is on optimal coordination of power and heat generation with the objective of increasing system operation reliability [2], which can be better addressed through multi-domain modeling using the Modelica language. Therefore, the representation of detailed models becomes crucial, especially three-phase induction motors, which are ubiquitously used to drive district heating and water pumps.

Electrical load models, including the motors, introduce stochastic behavior into the power grid. Thus, in the past decade, Western Electric Coordinating Council (WECC) has continuously studied the effects of different load models for time domain simulation, proposing the effective aggregate load model [3], [4]. This composite load model can be found in commonly used power system tools such as Siemens PTI PSS®E, GE PSLF, and others, granting end-users with a single model that has static and dynamic behavior [5]. Although load models have slowly become more realistic in their representations for bulk power systems, the mechanical interface between the motor and the actual load is still simplified, utilizing a polynomial function to emulate the mechanical torque profile at steady-state operation.

With the advancements in modeling technologies and computational power, there is an opportunity to use more detailed models even early in design or analysis phases. The Modelica language, an object-oriented multi-domain modelling language used to simulate complex cyber-physical systems [6], allows for model representation in the form of equations, and also for seamlessly creating and connecting components of multiple domains. For example, a successful implementation of a multi-domain system interaction is proposed in [7]. In this work the Modelica-based models allow for the interaction between power system dynamics and building dynamics, in which a three-phase phasor induction motor model is implemented instead of efficiency based models, typically found in building modeling tools.

With the goal of providing industry grade representation of the power grid combined with multi-domain modeling of a typical thermo-fluid load, this paper is focused on the implementation of a multi-domain load model in the Modelica-language and integrating it into OpenIPSL [8], an open-source Modelica-based library used for power system dynamic modeling. Hence, the contributions of this work are:

1) Modeling of a multi-domain power grid, motor-drive, and thermo-fluid system;
2) Implementation of a three-phase induction motor model controlled by a Volts/Hertz regulator, and its coupling to a load, represented by a typical thermo-fluid system;
3) Implementation of a modular motor-drive (i.e. a controllable induction motor) model using object-oriented computer-based modeling techniques and tools (i.e. Dymola) and using Modelica language;

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Advanced Manufacturing Office, Award Number DE-EE0009139. The views expressed herein do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

The work of M. de Castro and L. Vanfretti was supported in part by Dominion Energy, in part by the New York State Energy Research and Development Authority (NYSERDA) under agreement number 137948, and in part by the Center of Excellence for NEOM Research at King Abdullah University of Science and Technology.

4) Development of models that are adequate for stability analysis of both electrical and thermo-fluid contingencies, which can be used to model multi-domain Micro-grids with renewable energy sources [9].

The following sections are divided as such: section II describes the standard three-phase induction motor model; section III describes the new multi-domain induction motor model, including object-oriented modeling aspects, as well as implementation and modeling details of the Volts/Hertz controller and the thermal-fluid system; section IV shows application examples, and lastly, section V lays down conclusions for this work.

II. STANDARD TYPE I INDUCTION MOTOR MODEL

The model chosen for this work is the standard three-phase induction motor type I [10], which is the simplest representation of a motor model, modeled by the swing equation. This model is well suited for demonstrating, in the subsequent section, how to create a "mechanical interface", where the motor can be coupled with a more faithful representation of the load [5]. Moreover, it helps explaining the modeling requirements that allow for coupling the induction motor with a Variable Speed Drive (VSD). In the sequel, names and variables in the figures are referred to in the text using **bold**, while the Modelica implementation of the equations appear in `courier` font for a clear distinction from the conceptual equations.

The swing equation is given by (1),

$$\frac{ds}{dt} = \frac{T_m(s) - T_{\text{elec}}}{2H_m} \tag{1}$$

where s is the slip of the motor, $T_m(s)$ is the mechanical torque polynomial function, T_{elec} is the electromagnetic torque, and H_m is the constant of inertia of the load in seconds. The solution of this equation allows determining the per unit (p.u.) rotor speed of the motor, defined as $w_r = 1 - s$, where w_r is the rotor speed in (p.u.). The electrical torque, T_{elec} in (p.u.), is roughly the same value as the consumed power in the induction motor, allowing the simplification $T_{elec} \approx P_{elec}$. The P_{elec} function is derived from the power injected into the steady-state circuit of the induction motor, given by equation (2). The reader is referred to [10] for more information.

$$P_{elec} = \frac{s r_{R1} V_{motor}^2}{r_{R1}^2 + s^2\left(x_S + x_{R1}\right)^2} \tag{2}$$

III. MULTI-DOMAIN INDUCTION MOTOR MODEL

As mentioned previously, induction motor models in power system tools have limitations when the impact of mechanical behavior that governs the load's dynamics is studied. This is the result of the simplified representation of the mechanical torque. However, by making use of multi-domain modeling in Modelica, it is possible to improve the motor model by including a VSD controller and also adding a mechanical interface for mechanical system coupling. The new motor model and the VSD model are presented in the following subsections.

A. Overview of the New Induction Motor Model

The multi-domain Modelica-based induction motor is shown in Fig. 1. The novelty of the model is in the addition of four new connection ports (**Flange**, **mech_torque**, **we**, and **wr**). These ports were added to the original Type I motor model already found in the OpenIPSL library [8]. The original motor model only has one connector, which is **pwpin**, an OpenIPSL specific connector that is ubiquitous to all of the models from the library. The **Flange** connector allows the motor model to be coupled to all components with a matching interface from the Mechanics Rotational [11] sub-library within the Modelica Standard Library (MSL) [12]. The **mech_torque** and **we** connectors are input pins for the mechanical torque and the magnetic field synchronous speed, respectively. The mechanical torque originates from a mechanical component (e.g. a pump system), while the synchronous speed is controlled by the frequency controller of the VSD. Finally, the **wr** output connector is the motor's rotor speed serving as input to the VSD controller.

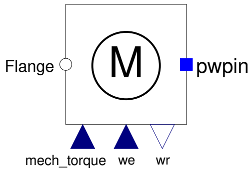


Fig. 1. Icon view of the multi-domain Type I motor model in OpenIPSL.

1) The Model's Equations: The original set of equations, which was described briefly in the previous section, handles rotor speed in (p.u.), adopting a feeding voltage signal with constant frequency. For the purpose of this paper, the frequency of the voltage signal is controllable, and is expressed in terms of the synchronous speed signal **we**. Therefore, the updated slip equation becomes:

$$\frac{ds}{dt} = \frac{\left(\frac{\text{mech\_torque}\cdot wr}{M_b}\right) - P_{motor}}{2H_m} \tag{3}$$

where mech_torque is the mechanical torque of the rotor's inertia, wr is the rotor speed in rad/s, and P_{motor} is the consumed active power of the motor in the motor's base power M_b. With the adoption of the synchronous speed input signal, the rotor speed equation is now expressed as (4), where both w_r, and w_e are expressed in rad/s.

$$w_r = (1 - s)w_e \tag{4}$$

When comparing both equations (1) and (3), it is possible to realize that the once used mechanical torque polynomial function $T_m(s)$ is now modeled as a variable that originates from the torque applied to the inertia shown in Fig. 2. This concept will be explained in more detail in the example section.

2) Mechanical Interface Modeling: The Modelica mechanical interface of the updated motor model from Fig. 1 is shown in Fig. 2. The **Rotor_Speed** block contains the rotor speed function that is used together with a **speed** component to create a mechanical rotational interface. The flange of the

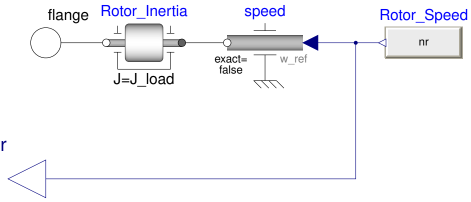


Fig. 2. Mechanical interface of the updated Type I induction motor model

speed block in Fig. [2] is then connected to the **Rotor_Inertia** inertia component which is then propagated to a flange output connector named **flange**. The inertia of the **Rotor_Inertia** block is determined through equation ([5]),

$$J = K \left(\frac{P_{rated}}{(N/1000)^3} \right)^{1.48} \tag{5}$$

where $K = 0.0043$ is a coefficient, P_{rated} is the rated power of the motor in kW, N is the pump speed in rev/min [13]. The signal from the **Rotor_Speed** block is propagated to the output connector **wr**, so that it can be used as input to the VSD controller. The **speed** component takes in the **Rotor_Speed** value that defines the reference speed in rad/s. The flange output of the **speed** component has two physical variables, the potential variable (absolute rotation angle of the flange) and the flow variable (cut torque in the flange) that allows coupling with the **Rotor_Inertia** component. As a result, the equation defined in **Rotor_Speed** block is coupled to a rotating mechanical interface, i.e. the **flange**.

B. *Modelica-based Variable Speed Drive Model*

Conventional induction motors in power system stability tools are limited to a single speed operation, as they are modeled without any form of rotor speed control. However, in practice, it is common to use a VSD [14] to operate the motor at different speeds/torques. The VSD is a power electronics based actuator that is typically built with a rectifier, a DC link circuit, an inverter, and its controller. The following subsections briefly describe the Modelica implementation of the power electronics interface and the Volts/Hertz controller.

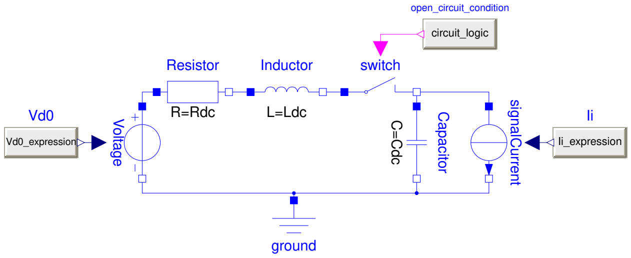


Fig. 3. Power electronics interface of the VSD

1) Power Electronics Interface: The power electronics interface model, that is shown in Fig. [3], is based on the results presented in the papers [3], [15]. The model does not account for the switching components, rather modeling both the rectifier and inverter in their averaged responses. The rectifier is a three phase diode bridge and is mathematically represented in the Modelica model as:

$$\texttt{Vd0} = 3 * \texttt{sqrt}(6) * \texttt{Vs} * (\texttt{V\_b})/\texttt{pi} \tag{6}$$

$$\texttt{Resistor.i} = (\texttt{P} * \texttt{S\_b})/\texttt{Vd0} \tag{7}$$

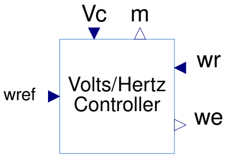


Fig. 4. Icon view of the Volts/Hertz controller model in OpenIPSL

where `Vd0` is the average rectified DC voltage, `Vs` is the phase to ground RMS voltage from the grid in p.u., `V_b` is the system voltage base, `Resistor.i` is the DC current flowing through the resistor and inductor, `P` is the active power in the AC side of the rectifier, and `S_b` is the base power. The DC link was assembled with components from the electrical sub-package of the MSL. In order to model the unidirectional flow of the DC current in the **Resistor** and the **Inductor** blocks, a logical **switch** is used in conjunction with the Boolean logic in equation ([8]):

$$\texttt{if Resistor.i} <= 0 \texttt{ then true else false} \tag{8}$$

where the logical expression is allocated in the **open_circuit_condition** block. The inverter is a voltage source converter and as such, allows control over the amplitude and frequency of the voltage signal at the AC side of the inverter. The mathematical equations that characterize the inverter in the Modelica implementation are:

$$\texttt{Ii} = \texttt{Pmotor} * \texttt{S\_b}/\texttt{Capacitor.v} \tag{9}$$

$$\texttt{Vmotor} = \texttt{Capacitor.v} * \texttt{m}/(2 * \texttt{sqrt}(2) * \texttt{V\_b}) \tag{10}$$

where `Ii` is the DC load current, `Pmotor` is the RMS active power consumed by the motor, `Capacitor.v` is the capacitor voltage, `Vmotor` is the RMS phase to ground voltage magnitude feeding the motor, and `m` is the PWM modulation index. Voltage regulation is performed using the PWM modulation index directly in the power electronics interface, while the frequency regulation is done through the **we** synchronous speed connection depicted in Fig. [1].

2) Volts/Hertz Controller: The Volts/Hertz controller implemented in Modelica is shown in Fig. [4]. This component has five different connectors: **Vc** is an input connector that takes the DC-link capacitor voltage measurement, **wr** is an input connector and takes the rotor speed from the induction motor model from Fig. [1], **wref** is the rotor speed reference value, **we** an output connector that provides the synchronous speed signal to the motor model of Fig. [1], and **m** is the PWM modulation index used in equation ([10]). The remainder of the model is built in Modelica following [14].

IV. MULTI-DOMAIN TEST SYSTEM MODEL AND SIMULATION RESULTS

The aforementioned models are integrated into a multi-domain model to illustrate its use in stability analysis. To this end, simulations are carried out introducing both electrical and fluid contingencies to a multi-domain test system.

A. *Multi-Domain Test System*

The multi-domain power and thermo-fluid model, comprised of a motor coupled to a water pump and its water piping system, is shown in Fig. [5]. The electrical grid side

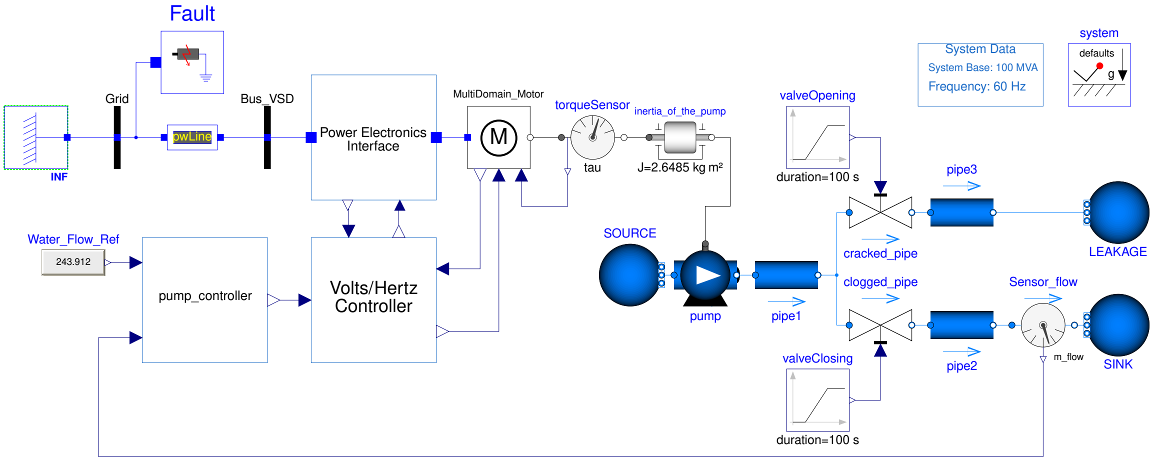


Fig. 5. Multi-domain test system

of the multi-domain test system is comprise of: an infinite bus **INF**, two buses named **Grid** and **Bus_VSD**, an electrical fault component **Fault**, and a distribution line **pwLine**. The VSD is modeled by two components, the **Power Electronics Interface** and the **Volts/Hertz Controller**. The rotor speed reference **wref** signal of the **Volt/Hertz Controller** model is the output of the **pump_controller** block, modelled as a proportional–integral (PI) controller. The PI controller is used to adjust the motor speed reference in the **Volts/Hertz Controller** in order to maintain a constant mass flow rate, even after a fluid system contingency occurred.

On the mechanical domain, the **torqueSensor** block measures the torque in the **inertia_of_the_pump** component, representing the torque in the pump shaft. On the thermo-fluid side, all model components used are from the MSL's Fluid library [16]. The **SOURCE** component represents an infinite water source, defining the starting pressure of the piping system as well as the liquid's temperature. The **SINK** drain represents the output of the normal pipe system while **LEAKAGE** represents a crack, modeled by **pipe3**. Different fluidic contingencies are created by opening and closing the **clogged_pipe**, and **cracked_pipe** valves. In order to emulate a clogged pipe, the **valveClosing** component sends a 100 s. ramp signal to the **clogged_pipe** component to close the valve to 50% of its initial position. In contrast, a crack is emulated by opening the **cracked_pipe** valve. **valveOpening** sends a 100 s duration opening signal to **cracked_pipe** where at the end of **pipe3** the sink has an ambient pressure of 1 bar. Additional model parameters are ommited due to space limitations.

B. Simulation Results

Two tests were performed to validate the presented multi-domain power and thermo-fluid models. First test includes a contingency at electrical side of the system, while the second test introduces a contingency at a thermo-fluid side.

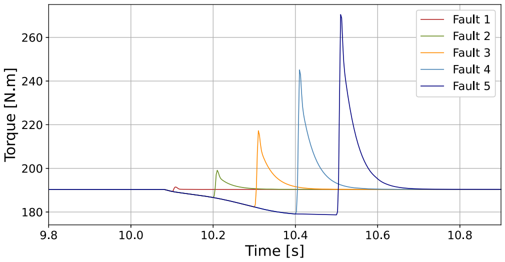


Fig. 6. Mechanical torque spike in the rotor shaft

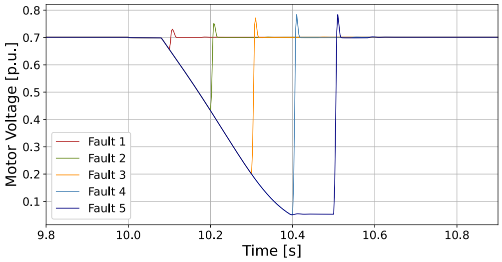


Fig. 7. Voltage magnitude at the motor terminal

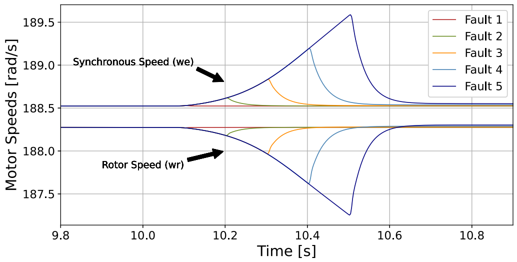


Fig. 8. Motor Rotor Speed versus Synchronous Speed

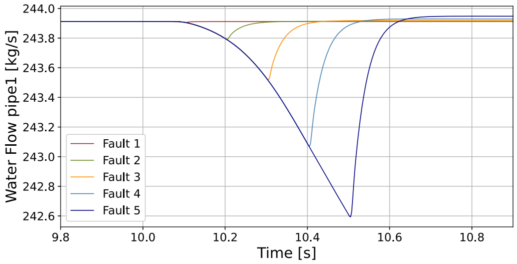


Fig. 9. Water flow in **pipe1** in different contingency scenarios

1) Test 1 — Electrical contingencies: These simulation tests aim to showcase the impact that increasing a fault's duration has in the overall system. The five different contingencies start at 10 s, however for each case the fault is cleared at increments of 0.1 s, i.e. Fault 1 clears at 10.1 s, Fault 2 at 10.2 s, etc. Figure 6 depicts different torque values in the rotor shaft for the increasing fault duration scenarios. Torque spikes appear when faults are cleared, as the feeding voltage at the motor terminal decreases as shown in Fig 7. Concurrently, the rotor speed **wr** also decreases, and consequently, the VSD controller increases frequency to compensate the rapid voltage drop, shown by the synchronous speed **we** in Fig. 8. Once the fault is cleared, both rotor and synchronous speeds return to their previous states. The effects of the electrical fault propagate to the thermo-fluid system, shown in Fig. 9 by the water mass flow in **pipe1**, displaying changes in the water mass flow rate due to the decrease of the rotor speed during the fault.

2) Test 2 — Fluid system contingencies: These contingencies consider two different scenarios: (a) leakage in a pipe, and (b) clog in a pipe. For the leakage scenario, the **cracked_pipe** valve is opened at a constant rate for 100 s. until the maximum stress in the piping system is reached. This allows to test the system's ability of regaining its previous fluid-flow state through the combined responses of the **pump_controller** and **Volts_Hertz Controller**. In Fig. 10 the PWM modulation index control output increases as more water from the main pipe leaks. As the PWM modulation index increases, so does the voltage applied to the motor. The synchronous and rotor speeds also increase through the interaction with the VSD controller, displayed in Fig. 11, which consequently also increases the motor's power consumption. The fluid mass flow throughout the entire simulation is shown in Fig. 12, where it is possible to check that **pipe2** regains its previous flow state through the influence of the **pump_controller** control signal.

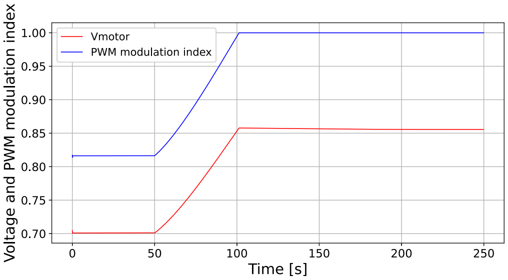


Fig. 10. Motor Voltage and PWM modulation index in time

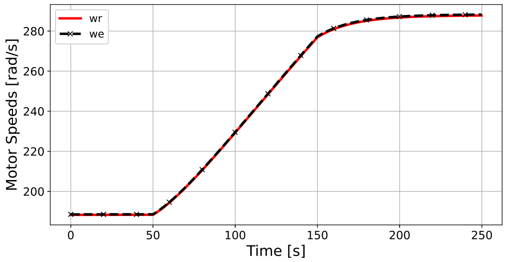


Fig. 11. Rotor and Synchronous speed in the motor

V. CONCLUSION

This work presented the implementation and validation of a multi-domain power and thermo-fluid system based on the Modelica OpenIPSL library. The models are emulating a VSD controlled motor that is coupled to the load, such as a thermo-fluid pump system. The case studies showed that the developed components are crucial for the detailed study of multi-domain systems, such as the electrical and thermo-fluid heating-cooling system in a microgrid. In smaller sized systems, having detailed load representation allows for more fidelity of the electrical system design to model contingencies across domains. For future work, we plan to develop a multi-domain microgrid system that incorporates renewable energy sources and the aforementioned models. This model is necessary to perform stability study and control design using model predictive control and reinforcement learning.

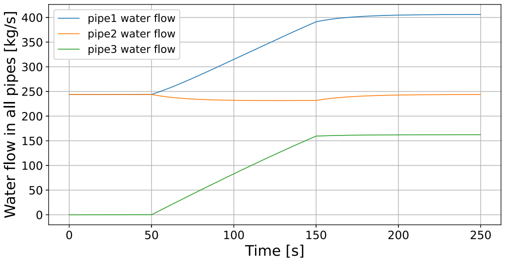


Fig. 12. Water flow in all the pipes of the system

REFERENCES

[1] R. Song, T. Hamacher, and V. S. Perić, "Impact of hydraulic faults on the electric system in an integrated multi-energy microgrid," in *Proceedings of the 9th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems*, 2021, pp. 1–6.

[2] J. Chen, B. Qi, Z. Rong, K. Peng, Y. Zhao, and X. Zhang, "Multi-energy coordinated microgrid scheduling with integrated demand response for flexibility improvement," *Energy*, vol. 217, p. 119387, 2021.

[3] D. Kosterev. *et al*, "Load modeling in power system studies: WECC progress update," in *2008 IEEE PES GM*, 2008, pp. 1–8.

[4] Z. Ma et al, "Mathematical representation of WECC composite load model," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 5, pp. 1015–1023, 2020.

[5] N. L. M. T. Force, "Technical reference document dynamic load modelling," *NERC, Dec*, 2016.

[6] P. Fritzson and V. Engelson, "Modelica—a unified object-oriented language for system modeling and simulation," in *European Conference on Object-Oriented Programming*. Springer, 1998, pp. 67–90.

[7] Y. Fu *et al*, "Coupling power system dynamics and building dynamics to enabling building-to-grid integration," in *Proceedings of the 13th International Modelica Conference*, 2019.

[8] M.Baudette, *et al*, "OpenIPSL: Open-instance power system library—update 1.5 to "iTesla power systems library (iPSL): A modelica library for phasor time-domain simulations"," *SoftwareX*, vol. 7, pp. 34–36, 2018.

[9] F. Fachini, L. Vanfretti, M. de Castro, T. Bogodorova, and G. Laere, "Modeling and validation of renewable energy sources in the openipsl modelica library," in *IECON 2021–47th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2021, pp. 1–6.

[10] F. Milano, *Power system modelling and scripting*. Springer Science & Business Media, 2010.

[11] C. Pelchen *et al*, "Modeling and simulating the efficiency of gearboxes and of planetary gearboxes," in *Proceedings 2nd International Modelica Conference*, Germany, 2002, pp. 257–266.

[12] P. Fritzson, *Principles of object-oriented modeling and simulation with Modelica 3.3: a cyber-physical approach*. John Wiley & Sons, 2014.

[13] A. D. Thorley, *Fluid transients in pipeline systems: a guide to the control and suppression of fluid transients in liquids in closed conduits*. Amer Society of Mechanical, 2004.

[14] B. K. Bose, "Power electronics and ac drives," *Englewood Cliffs*, 1986.

[15] AD Del Rosso *et at*, "Stability assessment of isolated power systems with large induction motor drives," in *2006 IEEE/PES T&D Conf. & Expo.: Latin America*, 2006, pp. 1–6.

[16] H. Elmqvist, H. Tummescheit, and M. Otter, "Object-Oriented Modeling of Thermo-Fluid Systems," in *Proceedings 3rd International Modelica Conference*, 2003, pp. 269–286.

Rensselaer

To: Reviewers 1, 2, 3, and 4

From: Fernando Fachini et al.

Subject: Response to Reviewers - Paper ID 22PESGM1706

Date: February 22, 2022

Dear Program Committee Members,

Below we have addressed each of the reviewers comments. Where possible, we have made modifications to the paper to clarify or expand the results. Due to time and space limitations, we could not address all the comments in detail. However, we have made our best effort to incorporate them all.

Color Code:

- *Black Italic:* reviewer comments taken verbatim from the submission feedback.
- Black: author's response to a comment where a detailed explanation was provided in this document and no correction in the paper was possible due to space and time limitations.
- Blue: author's response to a comment where action was taken to address reviewer's feedback.
- Magenta: labels to refer reviews to a previous response to a similar question raised by another reviewer.
- Red: author's text modification to the original paper.

Reviewer 01

Comments: The reviewer would like to thank the authors for this interesting topic. The paper is well presented and written. It discusses a very important topic of co-simulation and integration between different domains. However, a few comments are provided here to consider.

The authors would like to start by clarifying that the paper does not treat co-simulation, in fact, the main goal of the paper is to develop multi-domain models so to avoid the drawbacks from co-simulation as described in [Sch+19]. There are several limitations to running specific domain tools in a co-simulation setup, for instance: a) Numerical stability issues of co-simulation, b) Difficulties in choosing the right co-simulation orchestration algorithm (master), c) Difficulties in how to define the macro step size for a specific co-simulation, just to name a few. Based on the current limitations of co-simulation, the authors propose a multi-domain modeling of cyber-physical systems approach, where models are described by a set of algebraic differential equations and can be coupled and simulated in a single software environment only.

In addition, another benefit of the author's approach is that, while the author's used Dymola, a commercial modeling and simulation environment, the model can be used in any tool compliant with the the open-access Modelica modeling language standard specification [Brü+02]. A list of Modelica-compliant tools can be found in `https://modelica.org/tools.html`. Finally, these Modelica tools support the open-access and open-source Functional Mock-up Interface (FMI) standard for model-exchange, hence, the models developed by the user can be used by more than 100 tools, as listed here: `https://fmi-standard.org/`

To address this reviewer's comment, we have emphasized this important difference in the introduction.

Revise English language.

The paper was revised and the modifications can be found in red at the end of this document.

Does the model apply for load motors only?

The models created were intended for simulating a variable speed drive controlled three-phase induction motor in the positive sequence phasor domain. The chosen load being driven by the motor is merely an example of a continuously varying load that utilizes variable speed drives, although any other load example that utilizes induction motors and variable speed drives can be simulated with these models. Observe that the key difference between this approach and conventional power system simulation is that the load is defined by the cascading process, i.e. the pump and associated thermofluidic system, whereas in power system simulation the torque is defined by a constant or a simple approximating function. Hence, our approach allows to capture the dynamic behavior of the load beyond the induction machine and it's speed drive. In other words, we are able to model and simulate the dynamics of coupled electrical/mechanical/thermal-fluid multi-domain models have during transient studies.

To further clarify from the power system point of view, the OpenIPSL library, contains many other load models, such as the traditional static and dynamic loads that are currently used in state of the art power system tools. Therefore, the user is now capable of simulating, in their own grid examples, the dynamic effects that traditional load models present. Our work in this paper expands those capabilities as explained above.

"This paper presents the multi-domain models that .." Does this paper cover all the models or a few of them?

The paper discusses the implementation of the new multi-domain models in the OpenIPSL library. The OpenIPSL library itself is a Modelica based library capable of phasor time-domain simulations [Bau+18], and it contains models that are based on and validated against models from commercially available power system software, such as Siemens PTI PSSE, and thus are focused on the electrical domain. However, as described in our paper, the electrical grid of the future is becoming more complex, meaning that different energy domains are starting to interact and impact each other's performance. That is why it is important to develop models that allow to capture the interaction between these different energy domains. With that, the paper discusses the implementation and modeling of the variable speed drive model and the multi-domain motor model, which are the key to coupling it and interface the mechanical torque, to the pump and the cascading thermofluidic network. The power electronics interface of the variable speed drive is unique, with a fixed frequency AC input signal rectifier, has a DC link component which is then controlled via switching components in the inverter section, allowing for both voltage magnitude and frequency control of the motor. It is important to notice that the modeling is done in the positive sequence phasor domain, therefore the rectifying and inverting portions of the VSD are averaged, which means that the switching components are not explicitly modeled, which is sufficient for integrated system analysis studies. The control strategy however can be different, and in our case we chose to implement the Volts/Hertz

Controller because it is the most used controller architecture in practical applications. The mechanical interface described in the paper for the Type 1 multi-domain motor can be implemented in any motor model that is available or developed in the literature, we just opted to show the least complex model to facilitate clarity of the case studies in this paper.

Paraphrase the abstract to include; why solving this problem is important, and what has been done.
The comment was taken into consideration and the modification can be found in the abstract.

What kind of thermo-fluid systems are connected to distribution levels? Give a few examples. Are they only three phase motors?
There are numerous examples of thermofluidic systems that are connected to distribution level grids. For instance, to fill a water tower with water, there must be a motor that drives a pump. Similarly, heat pumps are used for heating by pumping hot water via a motor that drives a pump. Another example could be irrigation systems to transport water to farm lands. However, the theme under which this work is conducted is the multi-domain modeling of combined heat and power networks that couple electricity production with thermofluidic systems that are present in district energy systems, such as those of campus microgrids. To address the reviewer's comment, in the introduction this example is mentioned as the thermofluidic system example: "For instance, combined heat and power networks are energy efficient systems that combine energy production with heating and cooling of fluids, and that have an intrinsic relationship with the electrical grid due to the pumping system."

Elaborate on what kind of interactions between thermo-fluid systems and power systems. is it dynamic, reliability, resilience?
All the aforementioned interactions are of interest in the literature. For example, interactions between the different controls such as the pump controller and the VSD controller are possible. The interaction between both systems can be impacted due to misoperation or lack of coordination between the two domains, such as increase in the torque of the motor while not opening the valve for water distribution, leading to cracks in the pipes and consequently increasing the motor's consumed power. Paper [Wan+19] is a literature review on the topic of modeling and simulation of energy infrastructures from a resilience perspective. As pointed out in the paper, electric power is characterized as being the most crucial energy infrastructures due to its enabling function of interacting with other energy infrastructures. Telecommunication, transportation, water supply, sewer systems, etc are all examples of systems that are dependent of electricity, and their dependence can lead to their vulnerability. Based on the aforementioned importance of the subject, the study of coupled energy systems has become of significant importance. Paper [SHP21] discusses the possibility of cascading effects in a microgrid due to a contingency in the thermal system. Another paper addresses the multi-energy coordination of microgrid scheduling, how to effectively coordinate the operation of Combined Heat and Power Microgrids for operational flexibility and resiliency [SHP21]. The comment was taken into consideration and the modification can be found in the introduction.

Although aggregated load models have been widely used and are still being widely used. Describe the trade off between including more realistic model representations, scalability, and computational limitations?
Our multi-domain models allow to couple models from different domains, in fact, our paper and current research is funded as a part of DOE project that studies the interaction between buildings and the electrical grid, in particular, investigates the interactions in district energy systems which use of Combined Heat and Power systems as part of a microgrid for electricity and steam for heating. The project also includes a study of the resiliency of these grids and

reliability proposing state of the art control methodologies. The models are developed and integrated into the Modelica based OpenIPSL library and Modelica-based Buildings library [Wet+14]. Since these models are employed, the limitations of modeling electrical components in phasor domain is imposed. However, this modeling approach is of common practice in power system community, allowing for fast dynamic simulations. In other words, this modeling approach allows to speed up computations while ignoring electro-magnetic dynamics of power system components.

In addition, the performance of the Modelica modeling language in terms of the performance of numerical solvers that run Modelica based models are constantly improving in different IDEs such as Dymola. For example, in [HOV19] the authors discuss the Differential Algebraic Equation Solvers for large scale hybrid models that are used for power systems simulations. The paper discusses efficiency experiments featuring OpenIPSL power grid models, concluding that the run times for these models are competitive with domain-specific, state-of-the-art simulation tools, and thus also affirming that Modelica based models are scalable. On a numerical aspect, paper [Ago+19] discusses a Low Level Virtual Machine (LLVM) state-of-the-art compiler framework that can lead to significantly improved performance as well as lower overall simulation costs of large-scale models. The LLVM compiler is still in the early stages of development, although there are current solution proposals that have already shown to increase computational efficiency, such as the one presented in paper [BCB+17], describing the recently implemented sparse solver in OpenModelica in order to efficiently compile and simulate large-scale Modelica models.

Add more review on what has been done in the scope of the paper. Reference [6] is not enough. The comment on reference [1] is expanded. To expand the review, a new reference [SHP21] has been added, which addresses the multi-energy coordination of microgrid scheduling, how to effectively coordinate the operation of Combined Heat and Power Microgrids for operational flexibility and resiliency. Due to limitation in the number of pages, the literature survey was not expanded further. The comment was taken into consideration and the modification can be found in the introduction.

It seems that you focused on three -phase induction motors models but you mentioned that you will present all models at the beginning of the paper. Please clarify.

As explained earlier, the main goal of this paper is to develop a model that allows us to couple power grid models with other domains. For the case we present, the VSD and multi-machine models are the "link" between the power grid and the the thermofluidic system because they couple at the mechanical shaft. Based on these two new components, we build a multi-domain system model to illustrate their application. The models of the components of the power grid are from the OpenIPSL library and the models for the thermofluidic system are from the Modelica Standard Library's Fluid subpackage. More details of the components from those libraries can be found in [Bau+18] and [Cas+06], respectively. The comment was taken into consideration and the modification can be found in the introduction.

Provide a sentence on what is Modelica and what is OpenIPSL.

The Modelica language is an object-oriented multi-domain modelling language used to simulate complex cyber-physical systems, while OpenIPSL [Bau+18] is an open- source Modelica-based library used for power system dynamic modeling. The comment was taken into consideration and the modification can be found in the introduction.

What about other types of motors? Why did you choose that model specifically? The Type I induction motor is the simplest model available in power system dynamic studies, because it includes only the mechanical swing equation of the motor. It is the well known *eletromechan-*

ical model that is ubiquitously found in the literature [Mil10]. As mentioned previously, the mechanical interface described in the paper can be implemented in any motor model, including more detailed models used in power system dynamic studies, such as those from PSSE's catalog. The reason why we chose the Type I motor model in this paper is due to its simplicity when reduced number of equations facilitate the presentation of the approach.

The author's have already developed a Type III motor model using the same approach detailed in the paper. The Type III model represent a detailed single-cage induction machine in dq reference frame. Compared to the Type I model, it includes two additional differential equations that represent the d-and-q voltages states, i.e. $\dot{e}'_d$ and $\dot{e}'_q$. To illustrate how the approach proposed by the author's can be applied to more complex motor models, Fig. 1 shows a simple test-case system of a motor driving a pump that is filling a reservoir. For this example, the authors chose not to add any controller in order to facilitate the analysis of the results.

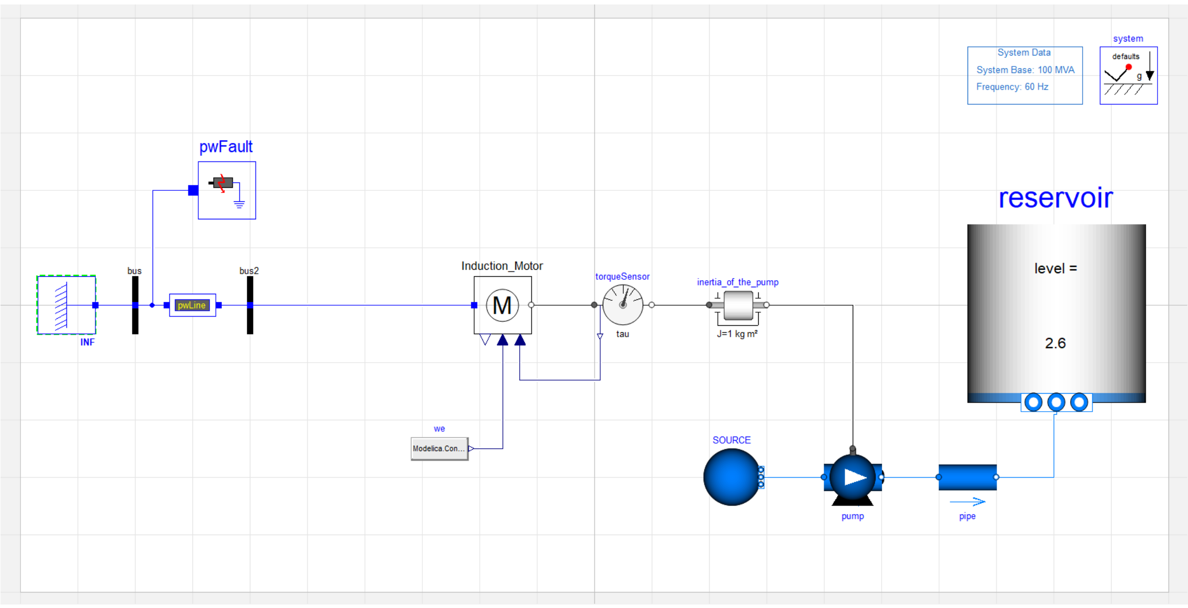


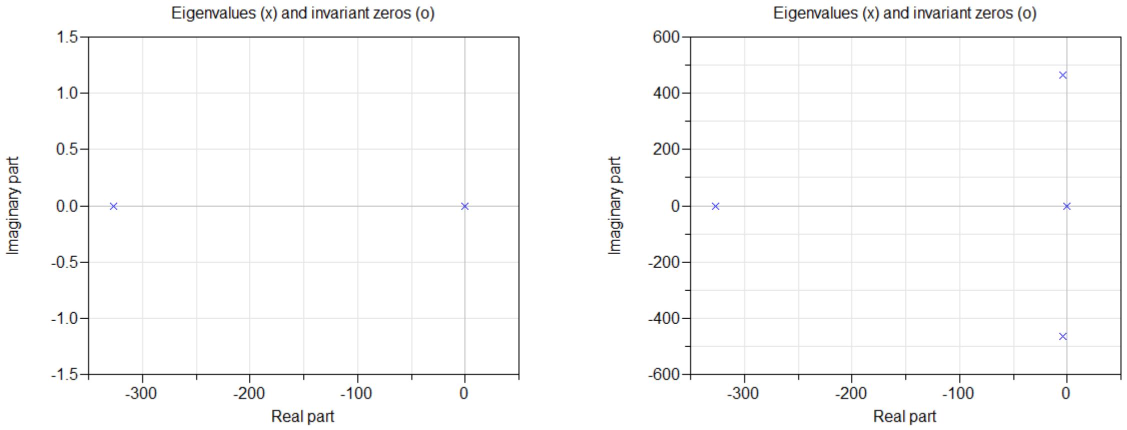
Figure 1: Simple test-case example of a motor driving a pump that is filling a reservoir

Swapping multi-domain induction motor models in this test-case is as simple as changing the used model, allowing the user to verify the simulation result differences with ease. The motor Type l model has only one differential equation, while the Type lll model has three differential equations, therefore we compute the eigenvalues of both motor models for the same simulation test-case to demonstrate differences and show that other more detailed models can be included. As expected, the linearization of the simulated test-case model with the Type l induction motor model presents one negative eigenvalue that is related to the state angular speed ω, of the motor which is a real-valued eigenvalue. Meanwhile, the linearization of the simulated test-case with the Type lll induction motor model presents one negative real-valued eigenvalue (similar to the motor model Type l) and a complex conjugate pair that is related to the d-and-q voltages states, $\dot{e}'_d$ and $\dot{e}'_q$. It is important to note that the eigenvalue displayed at the origin of both plots from Fig. 2 is related to the mathematical formulation used in power system dynamics (see [Mil10]) which is used in the OpenIPSL library, and is not related to the motor model. The distinction between models is clear and the differences in the eigenvalues are displayed in Table 1 and in Fig. 2, thus validating our point that it is possible to add the mechanical interface to any motor model.

Finally, this example serves to illustrate that because Modelica models are equation-based, the linearization of the models is performed on the same models used for time-domain simulation, which is not the case of power system tools, which require a separate linear model to be independently implemented [Mil10]. This is thanks to the symbolic analysis engine of

Dymola [Brü+02], which is capable of obtaining symbolic Jacobians without the need of user intervation. Note that other software environments, such as OpenModelica, are also capable of performing linearization of the same models.

Due to space limitations, it is not possible to include the details above in the updated paper.



(a) Eigenvalue after linearization for Type l

(b) Eigenvalues after linearization for Type lll

Figure 2: Comparison of different eigenvalues for the Type l and Type lll motor model

Test-case with Induction Motor Type l		**Test-case with Induction Motor Type lll**	
States	**Eigenvalues**	**States**	**Eigenvalues**
ω	-323.32	ω	-323.32
		e_d, e_q	-3.66 $\pm j463.63$

Table 1: Eigenvalue comparison for type l and type lll motor models

Add a figure for the corresponding motor model in OpenIPSL so that the reader can see the difference between the two models.

The previous motor model doesn't have the mechanical interface built into it, and due to the limited space of the paper the authors explained this difference in the text.

For the model described in section II.A. do you have to write down a code-script to represent its behavior. if yes, what language and which platform have you used.

R1.A.: Modelica is an acausal programming language for modeling cyber physical systems, enabling the user to create models utilizing differential and algebraic equations, and as such, the swing equation from the motor is used to create the model. The equations used are presented in [Mil10] and the Modelica implementation of the key equations can be found in the OpenIPSL's source code repository and given below in abbreviated form. Due to limited space, we refer the reader to get a comprehensive description of the models in our open source Modelica library. All models are open-source and readily available in GitHub through OpenIPSL's repository: `https://github.com/OpenIPSL/OpenIPSL`.

Because we use object orientation, the "base" class, is defined by the electromechanical equation (see full implementation in `https://tinyurl.com/BaseMotor`):

```
equation
  v = sqrt(p.vr^2 + p.vi^2);
  anglev = atan2(p.vi, p.vr);
  der(delta) = w_b*(w - 1);
```

where `delta` defines the machine's angle and `der(delta)` its derivative.

This base case is inherited by the motor model, which compliments the behavior with (see full implementation in `https://tinyurl.com/TypeIMotor`):

```
equation
  P = p.vr*p.ir + p.vi*p.ii;
  anglev = atan2(p.vi, p.vr);
  v = sqrt(p.vr^2 + p.vi^2);
  Re = Rs + Rr1/s;
  der(s) = (Tm - P)/(2*Hm);
```

where `s` is the slip and `der(s)` it's derivative. Now, while in the OpenIPSL implementation `Tm` is the machine's torque, which is approximated with a simple second order polynomial `Tm = A + B*s + C*s*s`, in our manuscript this approximation is removed and replaced with the multi-domain interface which imposes the torque demanded by the cascading thermofluidic model (i.e. imposed by the pump's inertia and demand of the pump).

As it can be observed above, using the Modelica language, it is not necessary to implement a specific "script" to solve for the model's behavior. The behavior of the model is defined by it's differential and algebraic equations. To solve those equations, a simulation environment compliant with the Modelica language is used, in this work, that is Dymola. Nevertheless, the newly created models and the OpenIPSL are open-source and can be run in any free or commercial Modelica-compliant based environment, such as OpenModelica or the tools that can be found in `https://modelica.org/tools.html`.

Due to space limitations, it is obviously impossible to include the details above in the updated paper.

Why K in equation (5) is constant?

Through a linear regression analysis of 284 pump inertia data points from different suppliers [Tho04], Thorley developed an equation that approximately models the inertia of a pump. The coefficient K in the equation came from the previous analysis. The comment was taken into consideration and this information has been added to the paper.

Section lll: Revise punctuation and English language. The comment was taken into consideration and the necessary punctuation corrections were made.

Although it is okay to use courier font to represent equations in Modelica, it is confusing. I would recommend using normal text for equations (6), (7), (8), (9), and (10) instead of courier font.

R1.B.: We appreciate the reviewer's comment, however, we have preferred to use the courier font, a way of distinguishing the equations that describe the model and the equations that are implemented using the Modelica model. Equations (1) - (5) represent the dynamics of the model, while equations (6) - (10) are their Modelica implementation. Being an acausal object-oriented programming language, Modelica allows the user to create models and their components as classes with specific attributes. For instance, in equation (6) the object **Resistor** has an electrical current attribute named **i**, therefore, calling the attribute in an equation requires typing **Resistor.i**. That is why it is important to distinguish the set of equations that represent the physics of the component and the Modelica implementation. While this might not ideal for the casual reader, it would be invaluable for those readers that are interested in the means for implementation.

Have you simulated the proposed model on a standard or realistic power system?

The proposed models expand the capabilities of the OpenIPSL library, so they can be used with any of the *de facto* standard IEEE networks (e.g. IEEE 9 Bus, 14 Bus, etc.) or other power system models provided with the library [Bau+18]. To illustrate, how the multi-domain models presented in our paper can be easily added into any modeled power grid, we provide an illustration below. The OpenIPSL library contains a representation of the Nordic Electric Grid, known as the Nordic 44. The Nordic 44 is an aggregated dynamic power system simulation model designed for analysis of dynamic phenomena in the Nordic power grid, for details see [Van+17]. Figure 3 shows a screenshot of the Nordic 44 grid with the added multi-domain model from our paper. The scaling of the image is not ideal due to the large number of components in the example, but it serves to illustrate how the proposed models can be readily used in a realistic power system model. Due to space limitations, it is obviously impossible to include the details above in the updated paper.

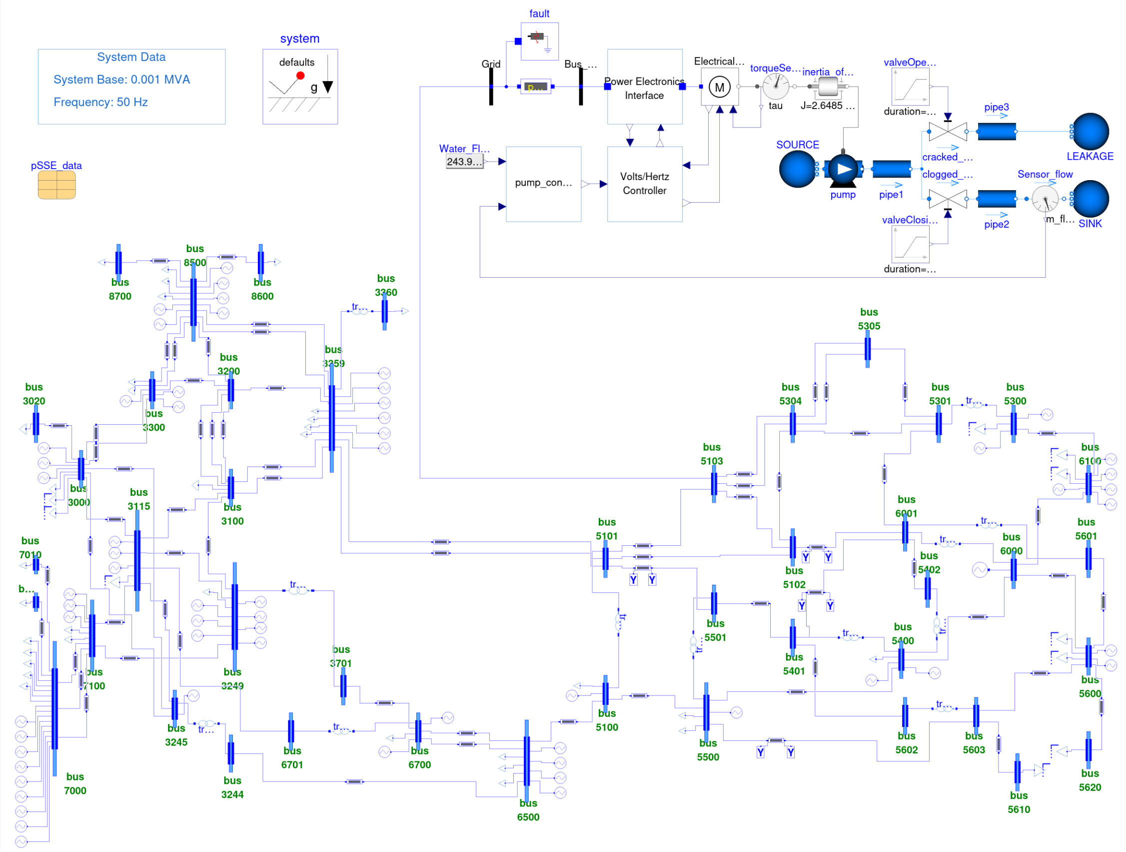


Figure 3: Modified Nordic 44 system with the multi-domain thermo-fluid load

Remove "due to the space limitations, only the leakage results are displayed". You can just mention the results of leakage scenario.

The comment was taken into consideration and the text was updated.

Section IV: Revise grammar and writing typos.

The comment was taken into consideration and the typos were corrected.

What about contingencies taking place at coupling points, i.e., VSD controller.

It is unclear what type a contingency the reviewer is describing. Assuming the reviewer means failures related to the VSD controller, the answer is yes, they can be simulated. To illustrate, to model a potential problem that a VSD controller can experience, we can model the loss of a sensor's signal. In the example below, we model and simulate a periodic loss in the signal coming from the **Sensor_flow** component starting at $t = 150$ s. Figure 4 shows the system's response due to signal loss, for the mass flow rate in all three pipes for the fluid system contingency leakage scenario. Observe that the periodic signal loss results on the VSD controller results in large deviations of the mass-flow rate for two of the pipes, similar to bang-bang response.

Due to space limitations, it is obviously impossible to include the details above in the updated paper.

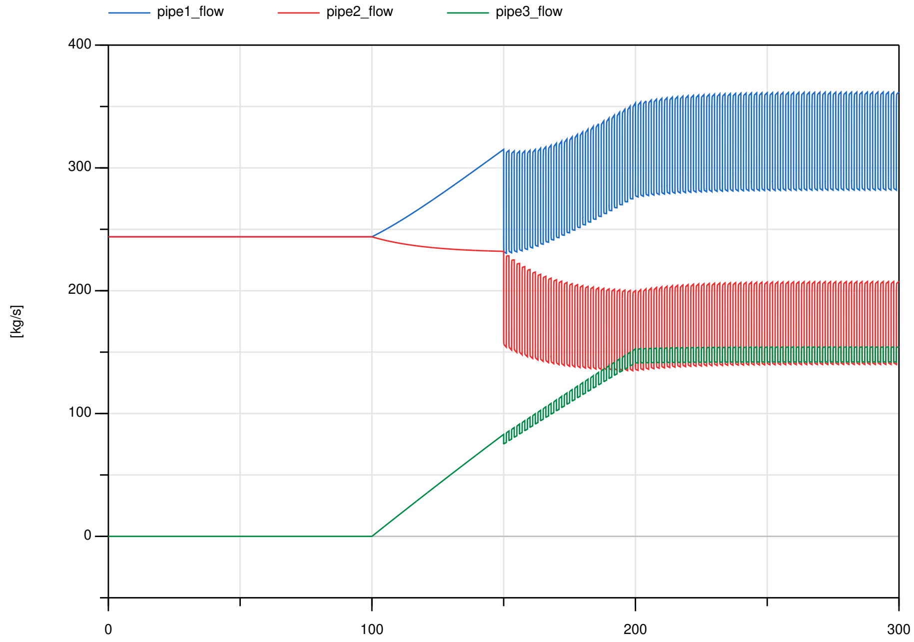


Figure 4: Mass flow rate in the three distinct pipes of the paper

Remove references from conclusion.

The comment was taken into consideration and the reference was allocated to the introduction.

Describe challenges for implementing your models on large scale systems.

Among challenges of implementing the proposed models within large scale systems, the most important one is the time required to create the system models manually. However, recent work in [Góm+19] shows that the process can be automated by consuming model information from other standards using in specific domains of engineering. As examples, for power systems using the Common Information Model (CIM) standard, while [] shows how this can be done for Building Information Models (BIM) for buildings and their energy systems. We expect that such approaches will continued to be improved as the Modelica technology is further adopted, as for example, in DOE's SPAWN of Energy Plus effort, see `https://tinyurl.com/DOESpawn`.

Due to space limitations, it is not possible to include the details above in the updated paper.

Add more references.

The comment was taken into consideration and new references were added.

Revise the formats of some references.

The comment was taken into consideration and the references were updated.

Reviewer 02

Further renewable-based models could be included.

Thank you for the suggestion. The author's have recently implemented renewable and energy storage models, which can be found in a separate publication [Fac+21].

The purpose of this paper is to describe and demonstrate the potential of the new multi-domain motor model and its VSD controller. A comprehensive microgrid system model that includes the multi-domain load models presented in this paper and renewable energy sources presented in [Fac+21] is under development, and planed for next publication.

Reviewer 03

Equation 6 and 7 needs to be defined in a standard mathematical format. Equation 6 – 10 can be combined in a form of algorithm because they seems to be conditional equation.

We thank the reviewer for his observation. Regarding the formatting, please see the rationale of the approach used in the response to Reviewer 1 above, after **R1.B.**. We kindly ask the reviewer to refer to that answer.

An algorithm is not necessary because of how the Modelica language has been used to pose the models and how the simulation environment solves the models' equations. In a nutshell, the author's do not have to solve for the models' equations in algorithmic form. We have a detailed explanation in the response to Reviewer 1 above, after **R1.A.**. We kindly ask the reviewer to refer to that answer.

It is recommended to include a short algorithm of the approach proposed in the paper.

As explained in the previous point, an algorithm is not used, models are coupled via their constitutive equations and no algorithm solution is proposed by the authors. We have a detailed explanation in the response to Reviewer 1 above, after **R1.A.**. We kindly ask the reviewer to refer to that answer.

The results are not described in a well structured format? What is the significance of the simulation results as shown in Fig 6 to Fig 12?

Due to page limitations, the authors tried to explain the differences between simulations of the two contingencies shown in Fig. 6 and Fig. 12. For instance, in Fig. 6 and Fig. 7, the electrical fault reduces the input voltage to the motor, consequently impacting the motor's torque. Meanwhile, Fig. 8 shows the synchronous and rotor speeds of the motor during electrical faults, which is a unique feature of the model, considering that VSDs are not modeled in typical power system models, such as IEEE standard models. Lastly, in Fig. 9, the water flow in the pipes was also displayed to show the impact of the electrical fault on the operation of these thermofluidic systems. The same pertains the fault in the thermofluidic system, but now focusing on the peculiarities of a contingency in the thermofluidic system.

Thus, the authors consider the presented cases as most significant to share with the reader in terms of what is possible to generate and simulate with the new multi-domain models with OpenIPSL and Modelica.

References are not properly referred in the text of the paper.

The authors used the LaTeX template for the IEEE General Meeting Conference, therefore references follow the requirements of the conference. Otherwise, the additional references are added to improve the manuscript.

What is the significance of the proposed research? Whats the novelty of the proposed research?

This work is being conducted under a project funded by the Department Of Energy's (DOE) Building Technologies Office of the United States. The project aims to study the interactions between the buildings, district energy systems, and the electrical grid. The Modelica language is already part of the modeling technologies portfolio's used by the DOE's Building Technologies Office, see for the SPAWN of Energy Plus effort at `https://tinyurl.com/DOESpawn`. Consequently, we were asked to participate in the project and expand the OpenIPSL library so that we we could perform integrated studies of building/district energy systems that interact with the grid. This paper presents our initial work to integrate, under a single model, using a single modeling language, and simulating within a single tool, both the power grid and a detailed thermofluidic system similar to that which can be found in a typical building application (i.e. a heat pump).

We are currently working on creating Microgrid test case scenarios, utilizing Modelica based libraries such as OpenIPSL [Bau+18] and Buildings [Wet+14], to study state of the art control strategies, grid resiliency, etc.

More specifically, the contribution of this work is the multi-domain motor model the VSD, and the presented multi-domain model with the case studies. In addition, the open-access Modelica language, being an open-source acausal and object oriented modeling language, enables the user to model the microgrid being able to couple different energy domains and simulate within a singular software environment. This avoids the usage of co-simulation which has several drawbacks [Sch+19].

The language of the paper, throughout is not according to the professional IEEE standards.

The comment was taken into consideration and the modification were done accordingly.

Standard variable names should be used.

We thank the reviewer for this observation. The naming convention of the components follows the variable definitions implemented in the OpenIPSL library, all names and variables are described in detail in the paper and properly referenced when needed.

We have a detailed explanation in the response to Reviewer 1 above, after **R1.A.** and **R1.B.**. We kindly as the reviewer to refer to the answer provided that justifies our rationale.

Reviewer 04

Authors have presented a relevant practical modelling of electro-mechanical system. However, it was just for the usability and viability of using the modelica OpenIPSL.

The authors' would like to clarify to the reviewer that in addition to modeling the electromechanical power system dynamics, the paper deals with integrated and coalesced modeling of mechanical (motor control), and thermofluidic systems.

The OpenIPSL library is an open-source phasor domain modeling power system library, built using the Modelica programming language, an object oriented acausal modeling language used to model large cyber-physical systems, that enables users to run dynamic simulations, similar to Siemens PTI PSSE. The models of the library are all based and validated against PSSE models. The advantage of using Modelica and the newly presented OpenIPSL models is that the users can simulate the models in different energy domains using one software only, plus the added benefit of Modelica being an open-source modeling language. The purpose of multi-domain models is to avoid using co-simulation, which has several drawbacks [Sch+19].

Author should add a discussion on limitation and impact on accuracy if modelling with other modelling software or technique, i.e., heat pumps, motor coupled load, electrical grid, etc.

The Modelica Standard Library (MSL) contains several sub-packages with models and examples for different domains. For instance, MSL has a Mechanics sub package used to model the physics of rotational bodies, translational bodies and multi-body systems. There are also the Electrical, Magnetic, Fluid, Math, and so forth. The Electrical sub-package of the MSL is used for simulating electronic circuit and, thus, is not intended for power system dynamic simulation, e.g. such as done with Siemens PTI PSSE. The OpenIPSL library, on the other hand, is well suited for studying time-domain simulation of large systems, matching the simulation results of state of the art power system tools in speed and accuracy [HOV19], in particular Siemens PTI PSSE as shown in [Bau+18]. The Fluid sub-package of the MSL, as described in [Cas+06], states that the Fluid library contains "components describing zero and one-dimensional thermo-fluid components, which can be connected in arbitrary networks. The purpose of the library is to provide standard interfaces for thermo-fluid components, demonstrate how to build such models, and include a growing set of models of common use."

In our work the thermal-fluid system was created utilizing components from the MSL's Fluid sub-package [Cas+06], the mechanical interface utilizing the rotational components under the MSL's Mechanical sub-package [PSO02], while the electrical components of the grid came from the OpenIPSL library [Bau+18].

As shown in [Bau+18], there is no impact in accuracy when simulating power systems using the OpenIPSL library and the Modelica language, as it can match precisely the simulation results of the de facto standard for simulation, Siemens PTI PSSE. Meanwhile, when it comes to the representation of other domains, the MSL is a very robust library currently in version 4, it has been under continous development since 1999. Over more than 20 years, it has gained quality via peer-review through open-source development which can be found at: `https://github.com/modelica/ModelicaStandardLibrary`. Moreover, owing to the high quality of its models it has been adopted as the basis for modeling in multiple simulation environments, see `https://modelica.org/tools.html`. For the work in this paper, it is important to emphasize that these models require a clear understanding of the physical equations that describe the specific components, and understanding how to couple them in a meaningful way in order to create these multi-domain models. Hence, accuracy issues can arise only if the other domains are not represented correctly.

If possible, represent the difference in the form of figure or result.

The purpose of the paper was to describe the multi-domain models and share the results that showcase the potential of such models. If the reviewer meant the difference between OpenIPSL and other power system software, then the authors reference paper [Van+13].

References

[Brü+02] Dag Brück et al. "Dymola for multi-engineering modeling and simulation". In: *Proceedings of modelica*. Vol. 2002. Citeseer. 2002.

[PSO02] Christopher Pelchen, Christian Schweiger, and Martin Otter. "Modeling and simulating the efficiency of gearboxes and of planetary gearboxes". In: *Proceedings of 2nd International Modelica Conference*. 2002, pp. 257–266.

[Tho04] AR David Thorley. *Fluid transients in pipeline systems: a guide to the control and suppression of fluid transients in liquids in closed conduits*. Amer Society of Mechanical, 2004.

[Cas+06] Francesco Casella et al. "The Modelica Fluid and Media library for modeling of incompressible and compressible thermo-fluid pipe networks". In: *Proceedings of the 5th international modelica conference*. 2006, pp. 631–640.

[Mil10] Federico Milano. *Power system modelling and scripting*. Springer Science & Business Media, 2010.

[Van+13] Luigi Vanfretti et al. "Unambiguous power system dynamic modeling and simulation using modelica tools". In: *2013 IEEE Power & Energy Society General Meeting*. IEEE. 2013, pp. 1–5.

[Wet+14] Michael Wetter et al. "Modelica buildings library". In: *Journal of Building Performance Simulation* 7.4 (2014), pp. 253–270.

[BCB+17] Willi Braun, Francesco Casella, Bernhard Bachmann, et al. "Solving large-scale Modelica models: new approaches and experimental results using OpenModelica". In: *12 International Modelica Conference*. Linkoping University Electronic Press. 2017, pp. 557–563.

[Van+17] Luigi Vanfretti et al. "An open data repository and a data processing software toolset of an equivalent Nordic grid model matched to historical electricity market data". In: *Data in brief* 11 (2017), pp. 349–357.

[Bau+18] Maxime Baudette et al. "OpenIPSL: Open-instance power system library—update 1.5 to "iTesla power systems library (iPSL): A modelica library for phasor time-domain simulations"". In: *SoftwareX* 7 (2018), pp. 34–36.

[Ago+19] Giovanni Agosta et al. "Towards a high-performance modelica compiler". In: *13th International Modelica Conference*. 2019, pp. 313–320.

[Góm+19] Francisco J Gómez et al. "CIM-2-mod: A CIM to modelica mapping and model-2-model transformation engine". In: *SoftwareX* 9 (2019), pp. 161–167.

[HOV19] Erik Henningsson, Hans Olsson, and Luigi Vanfretti. "DAE Solvers for Large-Scale Hybrid Models." In: *Modelica*. 2019, pp. 157–050.

[Sch+19] Gerald Schweiger et al. "An empirical survey on co-simulation: Promising standards, challenges and research needs". In: *Simulation modelling practice and theory* 95 (2019), pp. 148–163.

[Wan+19] Jing Wang et al. "Literature review on modeling and simulation of energy infrastructures from a resilience perspective". In: *Reliability Engineering & System Safety* 183 (2019), pp. 360–373.

[Fac+21] Fernando Fachini et al. “Modeling and Validation of Renewable Energy Sources in the OpenIPSL Modelica Library”. In: *IECON 2021–47th Annual Conference of the IEEE Industrial Electronics Society*. IEEE. 2021, pp. 1–6.

[SHP21] Ruihao Song, Thomas Hamacher, and Vedran S Perić. “Impact of hydraulic faults on the electric system in an integrated multi-energy microgrid”. In: *Proceedings of the 9th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems*. 2021, pp. 1–6.