Modelling and validation of control systems through the excitation of the hydroelectric power station Sopladora using OpenModelica

Melissa Ontano[1] Carla Parra[1] Edison Pogo[2] Javier Urquizo[1]

[1]Facultad de Ingeniería en Electricidad y Computación, ESPOL, Ecuador, {montano,cparra, jurquizo}@espol.edu.ec
[2]Operador Nacional de Electricidad CENACE, Ecuador, epogo@cenace.org.ec

Abstract

This paper begins with the excitation control system in a machine which is variable according to the needs of each system, and which has particular behaviour in each implementation. At the time of this simulation, there were not enough traditional models to provide us with the libraries of different software for power systems. However, free licensed software is available with object-oriented programming and the capacity to model complex heterogeneous systems; therefore, we built a mathematical model of the AVR (automatic voltage regulator) and PSS (power system stabilizer) of the Sopladora hydroelectric power station. These models were validated in a simple generator system – a bus – load. We compared the response behaviour of the voltage in the generator terminals and the field voltage with a simulation performed in software recognized and certified by its trajectory and application in engineering, and the same software was used as the reference. Once we achieved the right operation of the controllers, we proceeded to simulate the equivalent system of the 230 kV Sopladora bus. For this, we followed a construction process by adding and validating components until the equivalent system was obtained, and was ready to perform the validation qualitatively in both interfaces under the "software to software" modality, by observing the answers in the same flat and quantitatively through the mean squared error. This system was previously developed in the program referenced by a specialist of the National Electricity Operator CENACE.

Keywords: Control through excitation, Interface, Mean squared error, Oscillatory modes, Simulation, Software to software, Transient stability, Voltage control.

1 Introduction

The National Electricity Operator has the function of ensuring that the national interconnected system remains reliable and operates without interruption; for this, the power system must be within the limits of dynamic safety. One technically and economically feasible method of achieving the desired level of stability is by tuning the Power System Stabilizers (PSS). CENACE carried out a study on the development of research on this method, and later its implementation, based on previous analyses it had proposed. One proposal was to only have well-tuned power stabilizers in the machines with the highest participation in oscillatory modes, as this was considered better than the intervention of all the machines connected to the system (1). It was also suggested that the plant which would have the greatest contribution to electromechanical transients for a disturbance is the Coca Coda Sinclair hydroelectric plant. For this reason, in this plant the first field tuning of the PSS was carried out in February 2017 (2). The second most viable option was tuning in the PSS field of the Sopladora hydroelectric plant, and so, in April 2018, after the validation of the parameters in the three units of the plant, it was declared that the tuning had been successful. In Ecuador, research has been conducted on advances in the interface of a simulator in real time, and the verification of the tuning of the power stabilizers in the Coca Coda Sinclair and Sopladora hydroelectric power stations, during the development and research stage, was carried out by the eMEGASim module belonging to OPAL-RT Technologies. It was monitored by the WAMS (Wide Area Monitoring System) (3), and the simulations were performed with an equivalent machine system connected to an infinite bus (SMIB: infinite bus of a single machine). This simulator module is available in simulation bars, and is only compatible with the SimPowerSystemsTM library of Simulink.

Based on these limitations and our research needs, we transferred the development interest to the ePHASORSim module, which is part of one of the four modules of the real-time simulator, but its main function is electromechanical analysis. It also has the capacity to support a network of up to 30,000 nodes and is compatible with environments such as Simulink, Excel, PSS®E, CYME, Power Factory and OpenModelica (4). This latest software can also export models created by the user to the .FMU (functional mockup unit) format, which is a compressed document that contains XML files and compiled C code (6). It allows the development of a model in the OpenModelica environment and then uploads it to the RT-Lab or Simulink space (the latter is not included in this work, but it is necessary to mention it to explain our motivation in the use of OpenModelica). This research seeks approximation to exact models using OpenModelica free software, on the understanding that the importance of the development of this software lies in the possibility of creating any particular element not found in conventional libraries. These elements can then be used in the space of any power software that allows the adaptation of FMI blocks. To verify the approximation between the models developed in Simulink and those built in OpenModelica, we use the quantitative method of mean square error, under the criterion of proximity to the tolerance of simulation. This means that the mean square error between both answers must be less than the tolerance of the software. Simulink is a causal block-oriented language. It is a simulation network consisting of hierarchically connected blocks in a graphical representation of blocks connecting the inputs to output values. As a consequence the blocks reflects the calculation procedure rather than the structure of the modelled reality itself; whereas in complex models, this problem is addressed by modern equation-based, or acausal modelling language like Modelica. Modelica reflects the structure of the real system (5). In our paper the Modelica package extends the commonly used Simulink environment using the acausal approach, besides the fact that is not a proprietary system owned by a commercial firm.

1.1 OpenModelica

OpenModelica is free standardized software for modelling and simulation, based on object-oriented programming with open source. The OMEdit-OpenModelica Conection Editor tool is an interface that allows graphic interaction implemented in C^{++}, and provides the user with different features, among them:

- Creation of models through the Diagram View window and interaction with the code through the Text View window.
- Interface of each component with different types of connectors in its terminals.
- Ability to graph state variables, internal and external system signals, with graphs in the time, parametric, arrays and parametric arrangements (7).

1.2 OpenIPSL library

OpenIPSL is a library compatible with OpenModelica, developed in the iTesla project by SmaTS Lab, which is work led by Prof. Luigi Vanfretti. This work is formed mainly by the element of power systems previously validated under the software to software methodology (8),and can be used for dynamic analysis of power systems and simulations in the time domain.

1.3 Generator electromechanical model

This designation is assigned to the three-phase synchronous generator of the outgoing poles of hydroelectric plants (9), and is a classic electromechanical model with transfer functions to model direct and quadrature inductances. Assuming an additional circuit for the direct axis, the state variables can be described as in the following equations (10):

$$\frac{\partial \delta}{\partial t} = \Omega_b \left(\omega - 1\right) \tag{1}$$

$$\frac{\partial \omega}{\partial t} = \frac{Pm - Pe - D(\omega - 1)}{M} \tag{2}$$

$$\frac{\partial e'_q}{\partial t} = \frac{(f_s(e'_q) - (x_d - x'_d) - \frac{T''_{d0}}{T'_{d0}} \frac{x''_d}{x'_d}(x_d - x'_d) i_d + (1 - \frac{T_{AA}}{T'_{d0}}) v^*_f)}{T'_{d0}} \tag{3}$$

$$\frac{\partial e''_q}{\partial t} = \frac{(-e''_q + e'_q - (x'_d - x''_d + \frac{T''_{d0}}{T'_{d0}} \frac{x''_d}{x'_d}(x_d - x'_d)) i_d + \frac{T_{AA}}{T'_{d0}} v^*_f)}{T''_{d0}} \tag{4}$$

$$\frac{\partial e''_d}{\partial t} = \frac{\left(-e''_d + \left(x_q - x''_q\right) i_q\right)}{T''_{q0}} \tag{5}$$

The power is:

$$P_e = (v_q + r_a\, i_q)\, i_q + (v_d + r_a\, i_d)\, i_d \tag{6}$$

The voltages and currents correspond to:

$$0 = v_q + r_a\, i_q - e''_q + (x''_d - x_l) i_d \tag{7}$$

$$0 = v_d + r_a\, i_d - e''_d - (x''_q - x_l) i_q \tag{8}$$

The generator variables and its values are shown in Table 1 below

Table 1. Mean square error for each case

Parameters of the generator - Sopladora equivalent	
Synchronous reactance – d axis (pu) x_d	0.95
Synchronous reactance – q axis (pu) x_q	0.617
Sub-transient reactance – d axis (pu) x_{2d}	0.25
Sub-transient reactance – q axis (pu) x_{2q}	0.227
Open circuit transient time constant – d axis (pu) T_{1d0}	8.06997
Open circuit subtransient time constant – d axis (pu) T_{2d0}	0.1731231
Open circuit subtransient time constant – q axis (pu) T_{2q0}	0.3026119
Nominal Power (MVA) Sn	541.11
Nominal Voltaje (kV) V_n	13.8
Armadure Resistance (pu) r_a	0.002
Transient Reactance – d axis (pu) x_{1d}	0.283
Mechanical Time (kWs/kVA) M	8.28
Damping D	0

1.4 Pwpin connector

This connector can be found in the OpenIPSL Interfaces section, in which real and imaginary current and voltage information is saved. It allows the connection of the different power elements and exchanges the system phasor information on voltage and current through the Pwpin connector, in compliance with the laws of Kirchhoff.

1.5 Mean square error

The quantitative method is used in this work to ensure that the response of OpenModelica is the same model as the one created in Simulink. For this reason, we calculate the mean square error between both answers, to discretize the answers according to small intervals of time. Both values are compared through Equation 9.

$$ECM = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i\,)^2 \qquad (9)$$

1.6 OpenModelica integration methods

The integration method is used to define the duration of the simulation and the proximity of results. It is selected depending on: the order of the differential equation; the type of model (dynamic or static); the type of integration (implicit or explicit); and the control of each step, or multiple steps if used. The default integration method used OpenModelica is the DASSL (Differential / Algebraic System Solver), because it is compatible with most models (11).

1.7 Stability and generation control by excitation

Stability is the ability of a system to overcome a disturbance and return to a stable operating state. Control via excitation is achieved by affecting the field voltage of the machine, which is controlled by the voltage regulator and the power system stabilizer, although both perform different functions within the system. The AVR adds synchronizing torque to accelerate or decelerate the rotor so that it recovers its point of operation after small variations, while the PSS adds damping torque in the system, to avoid oscillatory instability (12). The signal received by the AVR is the voltage in the field terminals, but the PSS can also have as inputs rotor speed, electrical power, or the frequency of the system. It is necessary to consider that the output of the PSS will be another entry for the AVR.

1.8 Sopladora hydroelectric power station

The Sopladora hydroelectric plant is the third largest in the country (13), with a capacity of 487 MW divided into three generators with Francis-type turbines, located between the provinces of Azuay and Morona Santiago. It is part of the Paute Integral Hydroelectric Complex and receives the turbinated water from the Paute river, after they pass through the Molino hydroelectric plant. Two transmission lines come from its substation; one goes to Milagro and the other to Locks. In addition, the private hydroelectric generator San Bartolo is connected to the same bar of Sopladora 230 kV, which belongs to the La Favorita group, with a capacity of approximately 57 MW.

2 Methodology

In this section we will explain step by step the process carried out in the modelling and validation of the controllers via excitation of the Sopladora hydroelectric power station. It begins with the modelling of the voltage regulator block, for calculating the reference voltage and stabilizer of power systems. The next step is the validation of these controllers through a simple power system consisting of generator, bus and load. In this phase is where validations are made to make sure that the modelling of AVR and PSS has been the correct one. In the following phases we developed the equivalent 230 kV Sopladora system, adding each power element to form the entire system composed of generators, transformers, bars, lines and infinite buses.

2.1 Modelling the AVR and Reference signal

The inputs of the Sopladora AVR are: input voltage (Vin) and initial field voltage (Vf0). The input voltage is defined by Equation 10, while the initial field voltage assigned to the output of the AVR, in order to initialize it, is a value from the machine. Figure 1 shows in series that the output is a block limiter and, therefore, if vf0 is within the proposed limit range, the state variable X4 will be assigned the value of vf0. Next, a stable state is assumed for the initialization of the mathematical calculation in each block of transfer function. Thus, the variations in time will take the value of zero and the state variables defined as X1, X2 and X3 will also be assigned the initial value of vf0. In the block diagram of Figure 2, each transfer function or lead-lag blocks is assigned their respective parameter, and initialized under the "initial output" mode. This is because the initial input variable of the AVR is the field voltage, which represents an output in the three transfer function blocks that comprise the model.

$$Vin = Vref - Vt + Vpss \qquad (10)$$

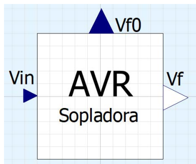

Figure 1. Interface of the Sopladora AVR

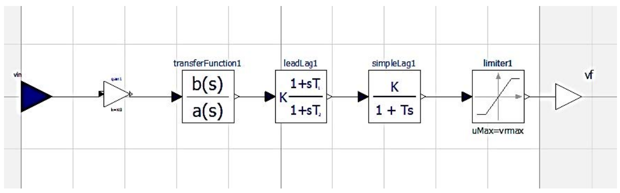

Figure 2. Block diagram of the Sopladora AVR

Continuing with the mathematical analysis of the system, we also deduce that state variable x1 is directly proportional to the input signal (Vin) through the AVR gain, thus defining Equation 11.

$$Vin = \frac{vf0}{Kavr} \qquad (11)$$

Combining Equations 10 and 11, the reference voltage of the AVR control system is defined by Equation 12, considering that the PSS voltage is zero and that the value of the voltage at the generator terminals (Vt) is known from the data preloaded from the power flow.

$$Vref = \frac{vf0}{Kavr} + Vt \tag{12}$$

In our model, this value is calculated by the block shown in Figure 3, and has a determined value; therefore, its internal programming contains only the initialization of the voltage in the generator terminals, and the AVR gain, and it has as input the voltage initial field. Thus, its output (Vref) is a constant value that will be affected only if the power flow is modified.

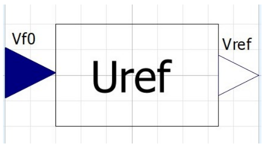

Figure 3. Reference voltage interface for AVR Sopladora

2.2 Modeling of the PSS

The model of the power system stabilizer of Sopladora is a dual type, which receives as inputs the variation of speed of the rotor (dw), and the active power (pe) in values per unit. The PSS must perform control when there is variation of speed, but in stable state the variations are zero and so the output (Vpss) is initialized with the value of zero. In addition to assigning the initial power value, it was necessary to obtain it directly from the machine through a constant, because it is one of the conventional outputs in the model of the generator. This also applies to the variation of speed, so it was necessary to apply Equation 13 by a constant and an adder block to assign the desired value to the PSS, taking the value of rotor speed, which is a conventional output of the machine. Figure 4 shows the representation of the block diagram and Figure 5 the interface created for modelling the PSS.

$$dw = w - 1 \tag{13}$$

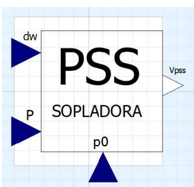

Figure 4. Interface of the Sopladora PSS

2.3 Sopladora generator modelling

The machine of order 5 type 2 includes transient and subtransient information and the general parameters equivalent to the three generating units of the Sopladora unit.

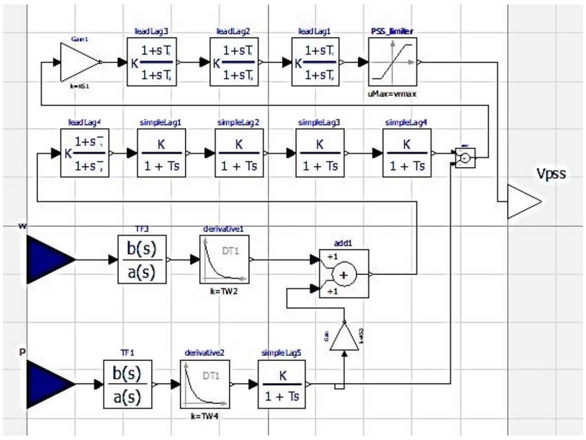

Figure 5. Block diagram of the Sopladora PSS

The output of the AVR is connected to the field voltage of the machine and the internal connection of the PSS is made by adding a gain between the output of the PSS and the input of the AVR. This switches the function by taking a value of zero or one, allowing us to activate or deactivate the PSS. A pulse of small amplitude is also added by means of an adder to the reference voltage, to produce the excitation of the oscillatory modes. We then verify the behaviour of the AVR and the PSS. This system is shown in Figure 6 and is developed within the “Generator" interface, which is the same as that in the OpenIPSL library, and is simply a previously created mask where we connect the whole system internally to the Pwpin.

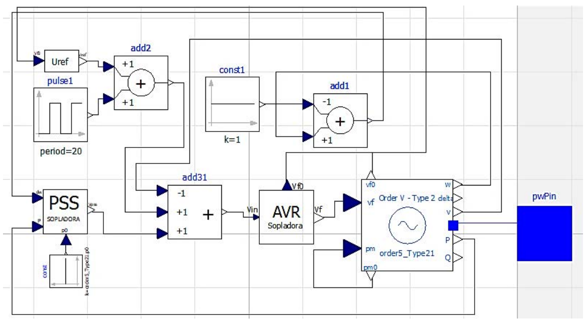

Figure 6. Internal block diagram to the Generator interface

2.4 System modelling for validation

The system for the initial validation of the PSS and the AVR was a generator, bus and load. Pulses of three different amplitudes are sent, but having the same period, pulse width and start time. Two loads are also used, and in both cases the loads are obtained from the previously validated OpenIPSL section PSAT section (8). One takes a purely resistive value of 463 MW and a mixed one of 463 + j298 MVA. In addition, the validation of the AVR and performance of both controllers are considered at the same time, resulting in a total of 12 previous cases. These are used to define whether our proposed models are consistent with the same system and respective controllers in Simulink.

2.5 System modelling Sopladora 230kV

The controllers were validated correctly via excitation modelled in OpenModelica, through the process described above. The procedure for the construction of the system equivalent to the 230 kV Sopladora bus is started, and this involves including the Sopladora step-up transformer, the San Bartolo hydroelectric power station with its elevator transformer, and the two transmission lines, Milagro and Esclusas, modelled as infinite buses. Modellica has a declarative language equations and mathematical functions that allows acausal modeling, therefore increased correctness and high level specification. The acausality makes Modelica library classes more reusable than traditional classes containing assignment statements where the input-output causality is fixed. To include the equivalent system behind each of them, a short circuit power was considered, represented by equivalent impedance, since in the OpenIPSL library the modelling of the infinite bus is ideal. Therefore, to approximate the proposed model in Simulink, where short circuit power is part of the model, it was necessary to include impedance in a series to the ideal bus, and a bus in the other terminal to define the angle and reference voltage determined by the power flow. The system is validated after including each mentioned element, in order to avoid a bad response in the final simulation and the need to correct errors in each phase to obtain the final equivalent system closest to the one proposed in Simulink. In addition, since the program does not have the capacity to calculate power flows, it was previously calculated in Simulink and the values of active power, reactive power, voltage per unit, and angle were entered in each electrical component. Figure 7 shows the process carried out to obtain the equivalent system. In each phase, an amplitude pulse generator -0.01, internal to the Sopladora generator mask, was added to the reference voltage and its response in OpenModelica compared with that of Simulink to verify the correct operation of each model. In Figure 8, only the system equivalent to the 230 kV Sopladora is shown, with mutual inductance between both transmission lines simulated as infinite buses.

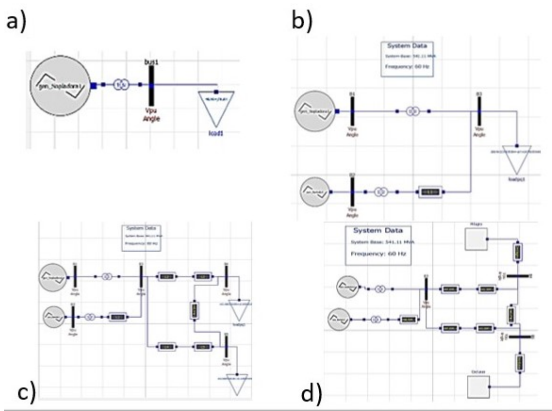

Figure 7. Phases before to the construction of the equivalent system

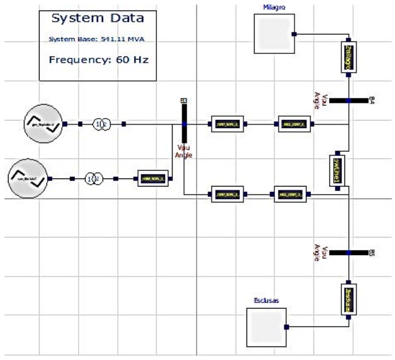

Figure 8. Diagram of the equivalent system of Sopladora

3 Results

In order to validate the AVR and PSS created in OpenModelica, twelve cases in total were created within the generator system. Bus, load, and "pulse train" signals with amplitudes of -0.01, -0.02 and -0.03 were applied, and all started in the second 20; the pulse width had a duration of 10 seconds and the period of the wave was 20 seconds, with a simulation of 50 seconds in total. Negative pulses were applied because, when there is an increase in load, the voltage at the generator terminals tends to decrease. This happens with a variation of the rotor speed of the generator, and so it is necessary to exert the synchronizing torque of the AVR. However, in addition there is also an oscillation of the point of operation that will cause instability if too little damping torque is applied, and so when applying the pulse train within the reference voltage, we activated the oscillatory modes and verified the performance of the controllers via excitation. The detail of each case can be found in Table 3, where the results of the calculation of the mean square error for each case are shown, discretizing the voltage response in the terminals, both Simulink and OpenModelica. The 12 cases can be divided into two large blocks, one with only resistive load, and another with a mixed load. Each type of load was validated under the action of three pulses with different amplitudes, and exerting control over only the AVR, versus AVR control and PSS. In Table 2, we can see the integration methods implemented in each simulator. Considering that the mathematical conditions within the model are an implicit integration of higher order and continuous type, both responses were discretized in steps of 0.002 seconds to validate a total of 25,001 points during 50 seconds of simulation. In addition, in both programs the tolerance was configured at a maximum value of x10^{-6}, which means that the values of the mean square error must be less than, or equal to, the preset tolerance to be considered valid and correct modelling. The ODE 23t method was selected by the response approach versus that provided by OpenModelica using the DASSL method; both were selected through the trial and error method until approximate answers were obtained.

Next, the characteristics of each case used during the

Table 2. Comparison between the integration of Simulink and Modelica

	Simulink	*OpenModelica*
Integration Method	ODE 23t	DASSL
Step Time	0.002	0.002
Tolerance	$x10^{-6}$	$x10^{-6}$
Simulation Time (s)	50	50
Total Number of Steps	25001	25001

validation are as detailed below;

- The first case was calculated by the action of a pulse train added to the reference voltage of the AVR, whose amplitude is -0.01. The system to which it was applied has resistive load and the voltage regulator is the only control exercised in the machine. This produced a mean squared error (MSE) value of 0.3 x 10^{-7} between the Simulink and OpenModelica approximation case, which we can guarantee is within tolerance limits

- The second validation case also involved a system of resistive load, and a train of pulses added to the reference voltage of the AVR of -0.01, but this time both controls were exerted via excitation, that is the voltage regulator and power stabilizer. The result of the MSE test between Simulink and OpenModelica was 1.6 x 10^{-7}, which is lower than the established tolerance and therefore a validated case.

- The third case applied in the validation process involved the train of pulses in the reference voltage, and an amplitude of -0.01 was added to the reference voltage. This is considered a power system with a resistive load and inductive load, and the control exercised only the voltage regulator. We obtained an MSE error of 0.62 x 10^{-7}, which is less than the established tolerance and is thus a validated case.

- The fourth case was a system of inductive and resistive load power. In the voltage regulator of the generating machine, an amplitude pulse train of -0.01 was added to the reference voltage and stabilizer and damper control was executed. We obtained an average quadratic error result of 1.9 x 10^{-7}, which again is a valid within the tolerance range.

- The fifth scenario was calculated by the action of a pulse train added to the reference voltage of the AVR whose amplitude was -0.02. The system to which it was applied had resistive load and the voltage regulator was the only control exercised in the machine. This produced an MSE of 1.6 x 10^{-7} between Simulink and OpenModelica, which is within tolerance limits.

- The sixth case of validation also involves a system of resistive load, and a train of pulses was added to the reference voltage of the AVR of -0.0. This time both controls were exerted via excitation, i.e. voltage regulator and power stabilizer. The result of the MSE test between Simulink and OpenModelica was 6.6 x 10^{-7}, a value lower than the established tolerance and therefore this is a validated case.

- The seventh case applied in the validation process involved the train of pulses in the reference voltage with an amplitude of -0.02 added to the reference voltage. This scenario involved a power system with resistive and inductive loads, and control was exercised only over the voltage regulator. We obtained a MSE of 2.4 x 10^{-7}, and as this is less than the established tolerance it is a validated case.

- The eighth case was an inductive and resistive load power system, in the voltage regulator of the generating machine. An amplitude pulse train -0.02 was added to the reference voltage and the stabilizer and damper control were executed, to obtain an average MSE result of 7.7x 10^{-7}. Again, this was a valid amount within the tolerance range.

- The ninth was calculated by the action of a pulse train added to the reference voltage of the AVR, whose amplitude was -0.03. The system to which it was applied had resistive load and the voltage regulator was the only control exercised in the machine. This produced an MSE of 3.6 x 10^{-7} between Simulink and OpenModelica approximation case, so we can guarantee it is within tolerance limits.

- The tenth case of validation also involved a system of resistive load, with a train of pulses added to the reference voltage of the AVR of -0.03, but this time both controls were exerted via excitation over the voltage regulator and power stabilizer. The result of the MSE test between Simulink and OpenModelica was 7.9 x 10^{-7}, a value lower than the established tolerance and therefore this is a validated case.

- The eleventh case applied in the validation process involved the train of pulses in the reference voltage, with an amplitude of -0.03 added. This is considered a power system with a resistive load and inductive load, and the control was only exercised over the voltage regulator. An MSE of 5.5 x 10^{-7} was produced, which is less than the established tolerance and so is a validated case.

- The twelfth case was an inductive and resistive load power system, and to the voltage regulator of the generating machine we added a train of pulses whose amplitude was -0.03. Stabilizing control was executed and buffered. We obtained an average quadratic error result of 8.4 x 10^{-7}, which is a valid amount, as established in the tolerance range.

After analysing the responses of all proposed cases, we observed that, under the action of any pulse, the system responded better to a resistive load than a mixed load. However, we know that this is far from reality because, considering that the transmission system and load must be included as the inductive part of the equivalent load, as a characteristic in validated cases, we also see that the error increased in all cases when the amplitude of the pulses was greater. This is consistent since each system (Simulink and OpenModelica) used different integration methods, and slightly different models. For example, the loading model in Simulink type Z constant could involve variables that OpenIPSL-PSAT load type PQ does not consider, or vice versa. As a last observation, we have to state that for the PSS the MSE increased. Only including the AVR in the power system remains valid, since, when including more elements, the possibility already described above is introduced that the modelling in the power system is not the same. This leads to the results in Simulink and OpenModelica being a little more spread. In Figure 9, we present in graph form the voltage at the generator terminals when the system only has a resistive load, a pulse of -0.01 has been applied, and it is only under the action of the AVR; in contrast, in Figure 10 we can observe the voltage at the generator terminals, but here the system has a mixed load, a pulse of -0.03 amplitude, and the performance of the AVR is applied next to the activated PSS.

We considered contrasting the data of these two scenarios in graph form as extreme cases; that is, the first has the lowest MSE and the last represents the greatest. With this, we can qualitatively discern that the follow-up of both signals (referring to the comparison between the response of Simulink and OpenModelica) is correct.

Table 3. Mean square error for each case

	Step 1 (amplitude = -0.01)			
	Resistive Load AVR	Resistive Load AVR+PSS	Mixed Load AVR	Mixed Load AVR+PSS
	0.300721	1.666431	0.621053	1.932726
	Step 2 (amplitude = -0.02)			
ECM ($x10^7$)	Resistive Load AVR	Resistive Load AVR+PSS	Mixed Load AVR	Mixed Load AVR+PSS
	1.645078	6.664367	2.438686	7.736053
	Step 3 (amplitude = -0.03)			
	Resistive Load AVR	Resistive Load AVR+PSS	Mixed Load AVR	Mixed Load AVR+PSS
	3.598513	7.991177	5.504289	8.424005

Given that the MSE in all cases had values lower than the allowed simulation tolerance, we believe that valid

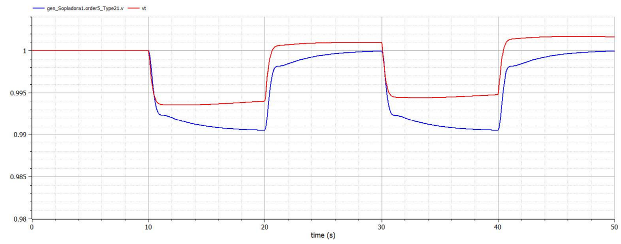

Figure 9. ((Red signal belongs to OpenModelica, blue signal belongs to Simulink.) Voltaje when the system only has a resistive load.

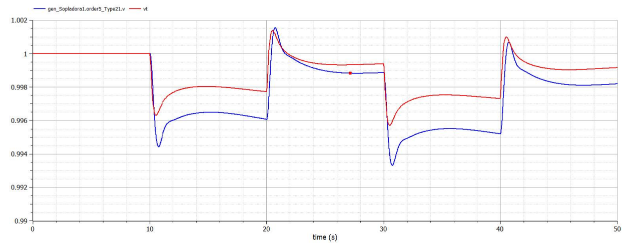

Figure 10. ((Red signal belongs to OpenModelica, blue signal belongs to Simulink.) Voltaje when the system has a mixed load and PSS.

modelling and initialization of the controllers via excitation was performed, and with this we proceed to validate the equivalent data system of the 230 kV Sopladora bus of Figure 8.

However, we do not elaborate the complete system, because if it were to find an error between both models this would be very complicated. Therefore, we added elements to the power system step by step, loading power flow, and validating each one, as explained in Figure 7, until we achieved the desired equivalent system. The first element we included was the transformer; we performed the validation of the generator, transformer, bus, and load system using the Trafo approach in OpenIPSL. This has only the star connection in both windings, and led us to make the change in Simulink, where the real connection is given as a triangle on the windings of the primary, and a star on the terminals of the secondary. This enables a better comparison of equivalent models, and in addition we applied in the machine of this system a pulse of -0.01 to the reference voltage. The voltage response at the generator terminals was validated and a MSE of 7.9 x 10^{-7} was obtained by including only AVR, and 9.2 x 10^{-7} when adding PSS. Both errors were calculated for 25,001 points, for 50 seconds, discretizing the graph in steps of 0.002, as initially proposed. Again, the system had errors within the tolerance range and therefore we continued to increase the components.

In the next validation step, the San Bartolo generator was included with its respective transient information, subtransient, and general parameterization. This unit is without any type of generation control, and it is also connected to a transformer which also has a star connection on both windings. The impedance of the equivalent transmission line, whose length is approximately 19 km, and the conductance had a value of zero, and the susceptance was a fairly small value because it is a short line (see Figure 7b). The calculation of the MSE was made under the same conditions as the previous case and 5.6 x 10^{-7} was obtained with only AVR, and 7.3 x 10^{-7} when including the PSS. Both these values were lower than the tolerance and therefore we continued with the construction of the equivalent system. The next element included was the transmission lines that leave the Sopladora bus; one arrives at Milagro and the other at Esclusas. As both travel for a long stretch together in the same transmission tower, a mutual inductance is included between both buses. In addition, both sections are long lines and therefore the value of the susceptance will be considerable. The MSE was calculated under the conditions already described, only under the action of the AVR it was 5.2 x 10^{-6}, and with the performance of the PSS it increased to 8.1 x 10^{-6}. These are unacceptable values within the allowed tolerance, but to include a greater number of power elements and because this is not our final equivalent system, these error values were allowed. Finally, we proceeded to substitute the charges for infinite buses, modelled with an impedance in series for the short circuit power and a conventional bus, to establish the reference voltage and angle. This model was described in Figure 8, and arrived at the desired values, under the established conditions of perturbations. When the MSE of Table 4 was calculated only with the AVR included, we found an error of 5.2 x 10^{-8} was produced, and when the PSS was included, the error was 3.1 x 10^{-8}; this caused an opposite effect to that which had occurred in all the previous validations, that is to say that the error in including the PSS decreased, while in all the previous cases of validation, the inclusion of the power stabilizer caused an increase in the error. This is due to the effect of the infinite bus, since they are not modelled as charges but as an equivalent system. The PSS will produce the desired action of dampening the oscillations in the said system, and this can be seen in Figure 11, where only the AVR controls the voltage. However, there is the presence of oscillatory waves, which are eliminated in Figure 12, where the PSS acts. In this model, very good approximations were obtained compared to the response of OpenModelica and Simulink, since the tracking between one response and another is almost exact and was checked with the error values shown in the Table 4; these are much smaller than the tolerance allowed.

Table 4. Mean square error of the controllers applied to the equivalent 230 kV Sopladora system

EQUIVALENT SOPLADORA 230kV		
ECM $(x10^8)$	AVR + PSS	Only AVR
	3.152280	5.241840

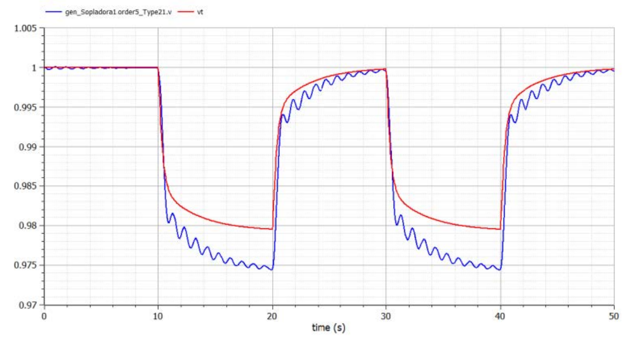

Figure 11. (Red line OpenModelica, blue line Simulink) Voltage with AVR

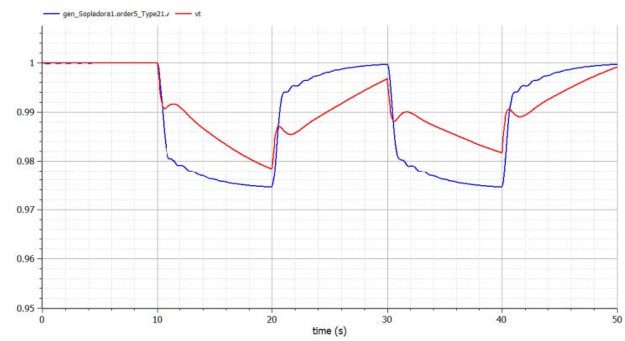

Figure 12. (Red line OpenModelica, blue line Simulink) Voltage with AVR and PSS

4 Discussion

In order to validate the control systems modelled in OpenModelica, two power systems were specifically developed. The first was used to test the method against several proposed scenarios; a simple system consisting of a generator, bus, load and different external signals was used directly with a reference voltage that affected the oscillatory mode. The system managed to respond as expected, but during the validation of both the controllers and the system, there were cases where the MSE was very close to the tolerance. When the validation of the equivalent system had an error a little greater than the allowed tolerance, we explain this by noting that the elements used to model the power system, such as loads and generators, have previously been validated with PSAT and not with Simulink, which is the reference software in this research. OpenIPSL has elements validated in the vast majority of PSAT and PSSE, and so to carry out real implementation of "software to software", as proposed by Qi Le in his research (9), we would need to perform modelling and pre-validation of the control elements, the power elements specifically of Simulink. However, the mathematical modelling of the power elements is mostly very standardized, and thus there are simply software programs which involve more variables in the modelling than others. For example, in the approximation of a salient pole machine proposed in Simulink, the number of poles and initial information of the magnetic fluxes is entered, while PSAT is limited to acquiring the transient, subtransient and general parameterization information of the machine. When we look at Table 4, we realize that despite using standard power elements in the construction of the total equivalent Sopladora system, the behaviour of the controllers via excitation demonstrates a precise approach. Observers will note that the same model of machine and bus were used in the simple system and in the Sopladora equivalent system. The difference between one model and the other was the replacing of the load with infinite buses, and therefore we attribute a bad approximation between the constant load of Simulink and the one validated in OpenIPSL using PSAT.

Our main interest in this research focuses on the development of control models and, when looking at Figure 8, we realize that the mathematical modelling managed to replicate Simulink to Modelica almost exactly. We also see that the integration method in both is quite similar to the idea in about mathematical integration, as Modelica responds to the level of Simulink, which is software recognized for its usefulness and implementation in engineering. Finally, although OpenModelica is not specialized software for power systems, the preloaded power flow from Simulink was able to exert very satisfactory behaviour. Nevertheless, although this could be seen as a disadvantage due to its dependence on another program to perform the power flow calculation, it actually decouples the model from using a specific calculation method, thus allowing an exchange between different simulation spaces according to the research needs. It could be developed as a module that performs power calculation under an algorithm, and automatically added to the elements of the system, but we consider that this would mean that Modelica becomes a simulator of power systems; we prefer not to establish its application in power systems, as the modality of independent and open software allows us to approach any mathematical model, static or dynamic necessary in the development of any field investigation. In fact, it really implies a disadvantage because it would transform any dependent model to a solver type of power flow, and therefore the created models could only be exported in compatible interfaces, not only with FMI, but also with the current power method stablished.

Matlab is one of the most recognized programs at the engineering level, mainly due to its capacity for numerical analysis. In the field of power systems, its Simulink tool, specifically the "powergui" block, has the ability to perform continuous, discreet phasor-type calculations. OpenModelica, despite its versatility, depends on an initialization given by a program that calculates the power flow. This means that we are not comparing two programs of equal implementation; on the one hand, Matlab/ Simulink, Power Factory/ DigSilent or PSSE, to name a few, are software programs with a well-recognized trajectory and recognition in the field of power systems. However, OpenModelica has become a novelty by giving open source code in each of its pre-established models, object-oriented programming based on mathematical equations, and other characteristics mentioned in the development of this work. What is remarkable is that, while OpenModelica will not replace any of the power flow programs mentioned above, since each one has a justified and developed implementation, it can complement and extend the libraries with more precise elements created in the space of OpenModelica, according to research needs.

5 Conclusions

It is easy to find information about Simulink or Matlab, because it is a program that has been involved in a great deal of development and application, but much less research has been conducted with free software and open development such as OpenModelica; therefore, it is necessary to be a little more empirical and intuitive. Fortunately, the object-oriented language implemented by OpenModelica is quite user-friendly and the programming logic code is similar to C^{++}. However, it is essential to read the basic manuals provided during the download, as well as the information that you obtain from its own web page. The OpenIPSL library represents an important development in the area of power systems and can have many applications since there are conventional control elements developed within it, but it is important to continue expanding and validating different elements of other power simulators. These could be converted into a multivariate library offering the possibility of mixing approximations of models, for example a PSAT machine controlled by an existing governor in Simulink, but within the modelling interface of OpenModelica. This idea is interesting because it persists in the creation of hybrid models, which represents a great advance in the field of research. According to the results obtained in Figure 8 and Table 4, which represent the final result of this work, OpenModelica is software enabled for the modelling of generation control systems based on complex mathematical equations, using a simplified language. It also has advanced integration methods of continuous and discrete types like Simulink, giving us equivalent answers. The ability to export models created in OpenModelica to other work interfaces removes the limitations of not having an accurate model according to research needs, which means that it is becoming an ideal tool for any engineering area.

OpenModelica, through OpenIPSL, has become a pseudo-simulator of power systems, because it does not have a specific method of performing the flow calculation. If it can allow response analysis, perform disturbances, produce faults, introduce noise signals, among other features, it becomes an interesting option for use and development. The PSAT load model proposed in OpenIPSL versus the Simulink model show notable differences, which can be seen when analyzing the table of results between the simple system (Table 3) and the Sopladora equivalent system (Table 4). In the first, only three power elements were implemented, the generator, bus and load. In the second model, in spite of including elements such as transmission lines and transformers, these behaved simply as series impedances, but when substituting the charges with infinite buses, the MSE decreased from 10^{-7} to 10^{-8}.

The development of power elements modelled and validated with Simulink is necessary to perform an accurate "software to software" validation of the control models. Moreover, in general, before carrying out any validation of control elements designed by the user, it is necessary to have the same elements of the power systems modelled in OpenModelica. That is, if Power Factory is used as a reference software, for example, to develop the modelling of power elements in OpenModelica prior to validation of the control system (in this case, the software provided with the simulated systems was Simulink), a better option instead of using this same software as a reference could have been to migrate the whole system to the PSAT interface, as many elements of this software are already validated through the OpenIPSL library. The continuation of this work would be to simulate the rest of the controllers via excitation and primary frequency controls in the machines planned to intervene in the tuning project of PSSs directed by CENACE. This will combine these models in the ePHASORSim workspace, and later develop all the SNI, including the machines with their respective controllers, and verify optimal operation. Such verification will not only be local but also consider the interaction of the control exerted in other machines to take corrective actions if necessary, looking for the damping of the whole system being within the ideal limits. Because OpenModelica is open, it could be considered that its implementation is in the field of education, since for the creation of any model in this interface it is necessary not only to understand the modelling process, but also the operation and mathematical approach and the ideal resolution tool. This is the criterion used to select the type of solver during the integration.

To perform validations between two responses from different environments, the MSE method is effective, but it is necessary to consider that for the tolerance in both simulators must be the same and be close to the real results, as many points as possible must be included. This means discretizing the outputs in small steps, considering whether the environment being plotted has the capacity to support the data numbers resulting from the discretization; in this work, we used the data import from Simulink towards the OpenModelica workspace to compare the graphs qualitatively, mainly because of the ease offered by OpenModelica in uploading result files in a .CSV format in its environment of "plotting". It is easy to export from Excel but the .CSV file must contain time in the first column (if that is the domain used), and the second or third column must take the values of the variables to be plotted on the vertical axis.

References

[1] P. Fritzson, Principles of object-oriented modeling and simulation with Modelica, WILEY-IEEE Press, 2014.

[2] Ministerio de Electricidad y Energía Renovable, Ministerio de Electricidad y Energía Renovable, 2016. [Online]. Available: https://www.energia.gob.ec/sopladora/. [last access: 05 09 2018].

[3] F.Armendariz, D. Panchi y H. Arcos, Sintonización de los Estabilizadores de Sistemas de Potencia del Sistema Nacional Interconectado Ecuatoriano Aplicación y Comparación de Métodos Heurísticos en Ambiente Multimáquina. Revista Técnica Energía, vol. Ene 2018, pp. 26-32, 2018.

[4] P. Verdugo and J. Cepeda, Sintonización en campo de los estabilizadores de sistemas de potencia de la Central Coca Codo Sinclair, V Congreso CIER de la Energía, 2017.

[5] Filip Jezek, Tomas Kulhanek, Karel Kalecky and Jiri Kofranek. Lumped models of the cardiovascular system of various complexity. Biocybernetics and Biomedical Engineering, Volume 37, Issue 4, Pages 666-678, 2017.

[6] P. X. Verdugo and J. C. Cepeda, Testbed for PSS Tuning Using Synchrophasor Measurements and a Real-Time Digital Simulator, IEEE PES.

[7] RT TECHNOLOGIES, opal.rt, 2018. [Online]. Available at: www.opal-rt.com/systems-ephasorsim/. [last access: September 2018].

[8] L. Vanfretti,T. Bogodorova, and M. Baudette. Power System Model Identification Exploiting the Modelica Language and FMI Technologies, IEEE International Conference on Intelligent Energy and Power Systems (IEPS), 2004.

[9] OpenModelica, Modelica, 2018. [Online]. Available at: `https://doc.modelica.org/help/ModelicaBlocksContinuous.html`.

[10] M. Baudette, M. Castro, T. Rabuzin, J. Lavenius, T. Bogodorova and L. Vanfretti, 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, no. ISSN 2352-7110, pp. 34-36, 2018.

[11] L. Qi, Modelica Driven Power System Modeling, Simulation and Validation, Thesis, KTH Institute of Technology, Stockholm, 2014.

[12] F. Milano, Power system analysis toolbox documentation for PSAT, 2013.

[13] OpenModelica, OpenModelica Navigation, [Online]. Available at: `https://openmodelica.org/doc/OpenModelicaUsersGuide.html`. [Last access: 2018].

[14] IEEE Power & Energy Society, IEEE Tutorial Course Power System Stabilization Via Excitation Control, Tampa, Florida, 2007.