

# Parameter Estimation of User-Defined Control System Models for Itaipú Power Plant using Modelica and OpenIPSL

Meaghan Podlaski<sup>1</sup> Luigi Vanfretti<sup>1</sup> Marcelo de Castro Fernandes<sup>1</sup> Jonas Pesente<sup>2</sup>

<sup>1</sup>Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, USA, {podlam, vanfrl, decasm3}@rpi.edu

<sup>2</sup>Itaipú Binacional, Brazil, pesente@itaipu.gov.br

## Abstract

The power industry heavily relies on power system modeling to understand system operations, perform system planning studies, and identify and correct problems that arise within the system. By minimizing the error between the models and actual physical system, it can be ensured that the models provide representation of both the existing and future the power system. Many of these models in the system are user-defined, i.e. they are specialized representations of a specific system component in the system. It is important that these customized models produce an accurate response. However, maintaining such models is costly, so it is of value to determine if those models can be replaced with a generic model. Phasor measurement data can be used to calibrate model parameters and reduce error. In this paper, this process is automated for a generator in the Itaipu power plant using RaPID, a MATLAB toolbox that integrates measurements, models using Modelica/FMI standards, and optimization routines. This is achieved by using a combination of particle swarm optimization (PSO) algorithm and classical gradient optimization routines to calibrate the model parameters. In this paper, the calibration of a generic model of a synchronous generator, automatic voltage regulator, and power system stabilizer are estimated and compared to the user-defined models for an automatic voltage regulator, power system stabilizer, and turbine governor.

*Keywords: Modelica, FMI, synchronous generator parameter estimation, PSO, system identification*

## Glossary

<b>GENSAE</b>	Salient pole generator with exponential saturation
<b>AVR</b>	Automatic voltage regulator
<b>PSS</b>	Power system stabilizer
<b>TG</b>	Turbine governor
<b>SMIB</b>	Single machine infinite bus system
<b>Model description</b>	Source of the definition of the model via a standard or another document

<b>OpenIPSL model</b>	Model in Modelica available in the OpenIPSL library adapted from the model description
<b>Efd</b>	Field exciter voltage
<b>PMU</b>	Phasor measurement unit
<b>P</b>	Active power
<b>Q</b>	Reactive power
<b>Pmech</b>	Mechanical power

## 1 Introduction

### 1.1 Motivation

New advancements in renewable power generation and control systems creates a resilient power grid and limits our negative impact on the environment. Simulation-based studies are helpful in determining which technologies have the highest benefit for the grid and the potential impacts of integrating a resource with the bulk electric grid.

Highly accurate dynamic power system models are necessary, especially in cases where user-defined models are used to simulate system conditions. Many renewable resources and control systems utilize user-defined models in their simulations, creating the burden of maintaining multiple models for system operators. It is necessary to determine the accuracy of these models and investigate if these models can be replaced by a generic, standardized model. Low confidence in the parameters of these models leads to more conservative and possible erroneous assessments of their responses to an event. There is also inherent uncertainty due to changes in the system parameters due to wear and aging of system components. Limited opportunities exist to test the physical power system because the existing system cannot be compromised for experimentation; building a new system for testing is not a viable option, as it would be too costly (L. Vanfretti, W. Li, T. Bogodorova and P. Panciatici, 2013). The results of this paper expand upon (M. Podlaski, L. Vanfretti, J. Pesente and P. H. Galassi, 2019), to show alternate ways to model power systems and derive user-defined model parameters with accuracy using Modelica and FMI. Previously, the IEEE standard models were used to represent the system. These models were calibrated using the same algorithms and methods used to calibrate the user-defined models. These results also compare the performance of

the user-defined power system models to IEEE standard power system models.

Real-world measurement data from phasor measurement units (PMU) can be used to improve power system models. By implementing these models with Modelica and interfacing with other softwares using FMI, the parameters can be calibrated for various power system components. Using a set of measurements obtained from PMUs attached at the terminal bus of Itaipu Binacional, the world's second-largest renewable hydro-electric dam, the parameters of different components of the power generation can be calibrated. This particular generator studied in this work produces 700 megawatts (Itaipu Binacional), an amount of power capable of supplying a city of 1.5 million people.

## 1.2 Related Works

Previous studies for power system model calibration using PMU measurements focus on using different solver methods and standardized models. PMU measurements have been used for dynamic model validation and calibration using various methods such as extended Kalman filter techniques (Z. Huang, P. Du, D. Kosterev, and S. Yang, 2013). Existing conventional and renewable plants are calibrated in (J. Chen, P. Shrestha, S. Huang, N. D. R. Sarma, J. Adams, D. Obadina, and J. Ballance, 2012) using PMU data to help determine the cause of faults within the system. The dynamic parameter identification uses a combination of particle swarm optimization (PSO) and sensitivity analysis for a system consisting of a wind turbine, its reactive power support, and step up and step down transformers. The parameter identification produces good results for an undamped oscillation under weak grid conditions. The calibrated models helped operators find problems with the AVR of the plant for the fault studied, allowing for improved operations under weak grid conditions in the future. This is especially important for a plant like Itaipú, which provides such a large amount of power.

Modelica and the FMI standard have also been used extensively in power system model calibration. In (Andersson and Strömner, 2013), a multi-domain model for a wind turbine is calibrated using synthetic and real-life measurement data. The model calibration follows a sequential approach in calibrating system components similar to the one outlined in this paper. The calibration process outlined in (Andersson and Strömner, 2013) utilizes the optimization features in the Modelica Design Library in Dymola rather than exporting the models as FMUs to utilize optimization routines in other software.

## 1.3 Paper Contribution

This paper contributes a study focused on the calibration of a standardized generator model and user-defined models for its control system. Both a classical gradient optimization method and a particle swarm optimization method are utilized in model calibration. FMI Toolbox and RaPid in MATLAB were used to calibrate the models discussed in this study.

## 1.4 Paper Organization

This paper is organized as follows. Section 2 outlines the setup of the models using Modelica/FMI as well as the software-to-software validation. Section 3 outlines the optimization problem defined in MATLAB that will be solved using RaPid. Section 4 shows the results of the optimization using the methods outlined in the previous sections, comparing the fitness of each calibration method.

# 2 Creating Power System Models Using Modelica and OpenIPSL

The power system model was implemented in Modelica using Dymola, the OpenIPS library, as well as the Modelica standard library to create the user-defined models from Itaipu. The Itaipu generators are salient pole generators with exponential saturation, which is modeled using the GENSAE model, as derived in (Kundur). The turbine-governor (TG), automatic voltage regulator (AVR), and power system stabilizer (PSS) models are user-defined models that were originally implemented in Anatem, the simulation software Itaipu uses for their plant. The voltage measurements from the PMU are injected into the system at the terminal bus to calibrate the generator, TG, AVR, and PSS.

## 2.1 Modelica model

### 2.1.1 Modelica model overview

The model was implemented in Dymola using in the Modelica language using the OpenIPSL (M. Baudette, L. Vanfretti, J. Rabuzin, M. Murad) and Modelica Standard (MSL) libraries. The Itaipu generators are salient pole generators with exponential saturation, so the IEEE standard GENSAE model is used to create the system model. The equations used to model the generator are included in the Appendix. User-defined models from Itaipu's modeling software, Anatem, were implemented using Modelica for the AVR, PSS, and TG. These user defined models feature functions from both the MSL and OpenIPSL used to model the components. The components are modeled using transfer functions and behaviors specific to the Itaipu plant. In the cases where the IEEE controller models are studied, the models' behaviors and transfer functions are derived from the IEEE standard for excitation system models (IEEE, 2016). These models are configured in a manner such that we can export them as FMUs to be used in MATLAB for model calibration.

The power system model was developed using the Modelica language and Dassault's Dymola software as shown in Figure 1. The components are labeled as follows:

- A. Tables containing the PMU data for the active and reactive power measurements.
- B. System data contains frequency and base power for the system. The machineData block contains parameter data stored in a record, which is propagated to all system components. A record exists with the results of every parameter calibration test run.

- C. User-defined turbine governor model from Itaipu.
- D. GENSAE generator model <sup>1</sup>, a salient pole generator with exponential saturation.
- E. User-defined AVR model from Itaipu.
- F. User-defined PSS model from Itaipu.
- G. Active and reactive power outputs to be used in the FMU.
- H. Controllable voltage source component.
- I. Tables containing real and imaginary voltage components from the PMU measurements.

The names and details of the parameters calibrated from the components listed above are listed in the Appendix. Figure 2 shows the relationships between the components shown in Modelica model in Figure 1. The generator outputs an active (P) and reactive (Q) power, as well as speed derivation ( $\Delta\omega$ ), mechanical power ( $P_{mech}$ ), and electrical power ( $P_{elec}$ ). The PSS uses  $P_{elec}$  as an input to obtain an additional tracking signal ( $V_{OTHSG}$ ) for the AVR's input. The AVR also needs the generator's exciter field voltage ( $E_{fd0}$ ) and terminal machine voltage ( $E_{comp}$ ) to determine what adjustments need to be made to  $E_{fd}$ . The governor, which is a speed controller for the generation unit, utilizes  $\Delta\omega$  and the reference mechanical power ( $P_{mech0}$ ) to control the turbine behavior in terms of speed and mechanical power. The turbine will then provide a mechanical power signal to the generator. These components are also mapped to a one line diagram of an SMIB system to provide understanding of where these components would be in a power system.

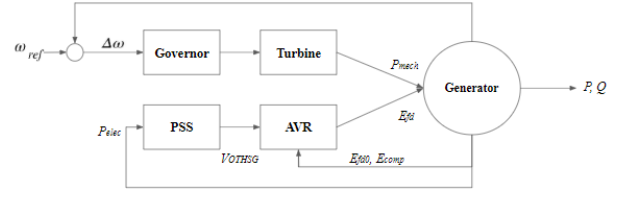


Figure 2. Relationship between system components.

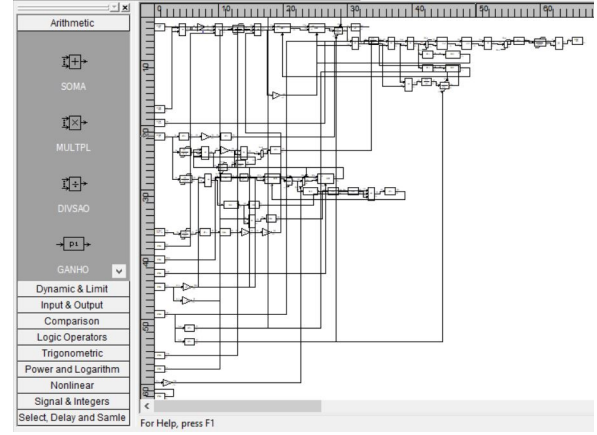


Figure 3. Itaipu User-Defined AVR model in CduEdit Software.

### 2.1.2 Re-implementation of User-defined Models

The user-defined models have been developed from models provided by Itaípu's engineers. They were created using the modeling software CduEdit. An example of how the models are set up in CduEdit is shown in Figure 3.

Some additional functions were created for the Anatem-Modelica equivalent user-defined models, such as the 'pulso' function (meaning 'pulse' in English), which was used in the modeling of the PSS and AVR:

```

model Pulso
  parameter Real p1 = 0;
  parameter Real p2 = 1;
  parameter Real p3 = 1e10;
  parameter Real p4 = 1;
  Modelica.Blocks.Interfaces.RealInput u;
  Modelica.Blocks.Interfaces.RealOutput y;
equation
  if u < p1 then y = 0;
  elseif (u > p1 and u < p3) then y = p2;
  else y = p4;
  end if;
end Pulso;

```

The Modelica implementation for the AVR model is shown in Figure 4. It includes a main AVR loop, overexcitation limiter, underexcitation limiter, VHz limiter, and a scaling factor. The main loop contains the transfer functions to regulate  $E_{fd}$ , the field voltage. The scaling factor scales the initial  $E_{fd}$  of the generator to the initial output of the AVR to ensure that the models are operating on the same base. The definition for each of the parameter abbreviations are located in the Appendix.

### 2.1.3 Brazilian software

The engineers at Itaípu used industry-specific software to implement their models. We re-implemented these models

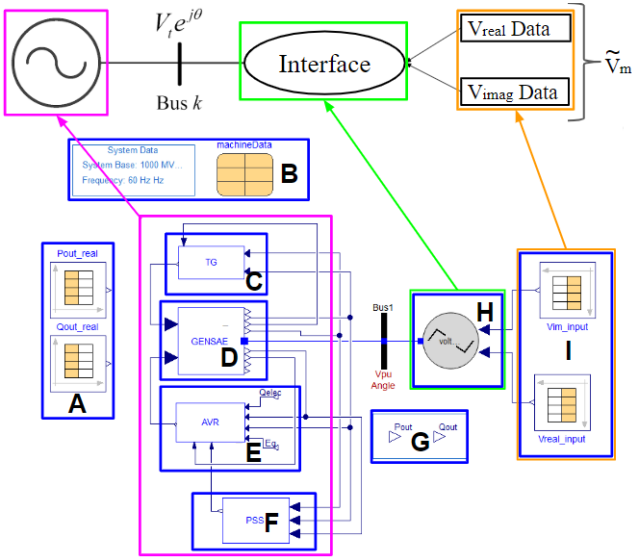
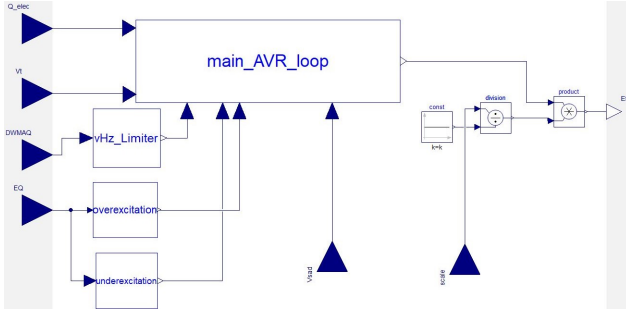
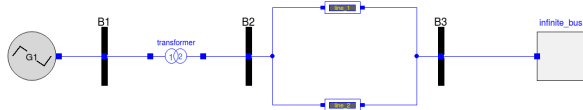


Figure 1. Modelica model of the generator, TG, AVR, and PSS in Dymola. This is compared to an SMIB one line diagram where all of the relationships are shown using magenta boxes.

<sup>1</sup>In previous work, the Itaípu plant has been modeled with a simpler generator model with parameters  $X_d$ ,  $H$ ,  $R_a$ , and  $X'_d$  (M. Podlaski, L. Vanfretti, J. Pesente and P. H. Galassi, 2019). The order of parameter selection was derived from guidelines in (Kundur). The equations used to model the generator are also obtained from (Kundur).



**Figure 4.** User-defined AVR model functions implemented in Modelica.



**Figure 5.** SMIB system set up in Dymola. Equivalent system is also implemented in Anatem.

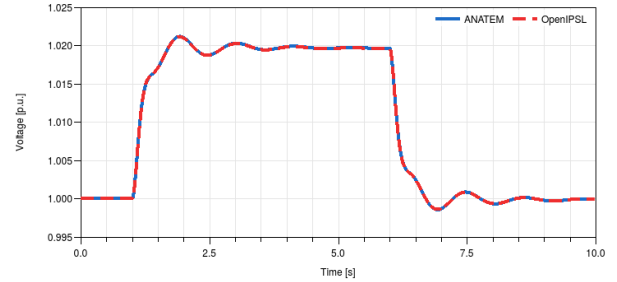
in Dymola. The software CduEdit (Cepel, c), which is pictured in Figure 3, is used to create control system diagrams for components. It is proprietary software used in Brazil in the utility sector for engineers to maintain a database of user-defined controller models for power systems (Cepel, c). It is a graphical interfaces used to create and edit user defined controls (CDUs). The CDUs can be simulated in ANATEM (Cepel, b), which is an industry-specific tool used for electromechanical transient analysis. These softwares also interact with ANAREDE (Cepel, a), which is a program that assists in the analysis of power system networks, such as power flow, network equivalents, and contingency analysis. These tools are limited in analysis capability, so we must re-implement the CDUs for the Itaipu plant in Dymola for parameter calibration.

#### 2.1.4 Software-to-software verification with Anatem

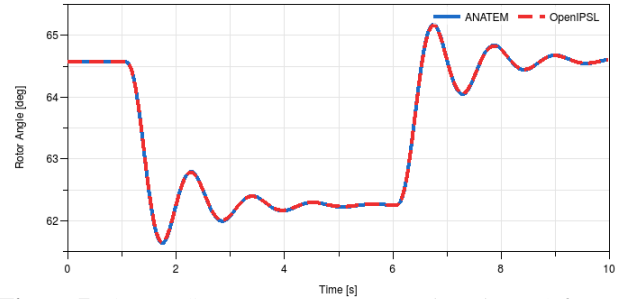
The verification will be carried out by simulating the same system in both software packages and comparing the obtained results, with the goal of showing that the models are equivalent in both software programs. This is necessary because it is challenging to prove to the users of domain-specific tools that they can obtain the same results as those tools in Modelica as long as the models are correctly re-implemented. This brings confidence to the models in Modelica that the results are going to be just as good as, if not better than, the domain-specific tool.

The system chosen to be implemented is the single-machine infinite-bus (SMIB) system, due to its simplicity. The SMIB system implemented using OpenIPSL is shown in the Figure 5.

It is important to mention that some parameter conversion need to be carried out. For example, the saturation curves need to be converted from an exponential representation to a polynomial one. In addition, exciter and power system stabilizers were adapted. The per unit parameters from the circuit were converted to ANAREDE in the Anatem software for the power-flow calculation, resulting in the adequate initial guess values to the system.



**Figure 6.** OpenIPSL vs Anatem voltages at bus B1 for software validation.



**Figure 7.** OpenIPSL vs Anatem rotor angle at bus B1 for software validation.

The event tested is a step change in the terminal voltage reference for the exciter system. The reference increase by 0.02 at time  $t=1s$  and it decreases, also as a step change, back to the original value at instant  $t=6s$ . The results from OpenIPSL and ANATEM are displayed in figures for easy comparison. The resultant voltage at bus B1 given by both software packages is displayed in Figure 6. It is possible to observe that the curves from ANATEM and OpenIPSL overlap throughout the entire simulation.

The rotor angle behavior given by both software packages is shown in Figure 7. Again, it is possible to observe the superposition of the results given by both software packages.

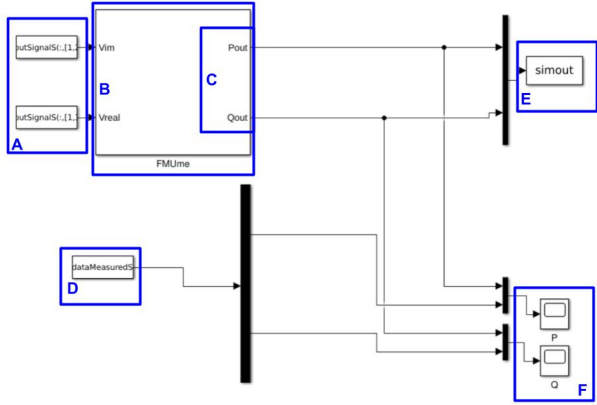
#### 2.1.5 Using PMU data with Modelica

The voltage data obtained from the PMUs is injected into the system at the site of the infinite bus, shown in Box I in Figure 1. Inside of the 'combiTimeTable' blocks in Block I, the real and imaginary components of the voltage are listed in tables over the period of the system event. Those voltage signals are then converted to flow variables (real and imaginary currents) to be injected at the machine's point of interconnection. Those flow variables then control the generator power output according to the voltage input.

Block A in Figure 1 contains the tables of data for the active and reactive power. The purpose for including these measurements in the model is to observe the fit of the simulation to the measurements in the plotting window.

## 2.2 Preparing models for RaPid

The models were optimized using RaPid (L. Vanfretti, M. Baudette, A. Amazouz, T. Bogodorova, T. Rabuzin, J. Lavenius, F. Jose Gomez-Lopez, 2016), a MATLAB toolbox used for parameter validation, calibration, and opti-



**Figure 8.** Itaipu system in Simulink using the FMI Toolbox for use in RaPID.

mization that uses models exported using FMI standard for Model Exchange, called Functional Mock-up Units (FMUs). RaPID uses both Simulink and MATLAB functions from the FMI Toolbox for MATLAB (Modelon, 2018) to simulate and perform the computations with the model. The FMUs need to be loaded and configured in a Simulink block from the FMI Toolbox library; in this paper, the block for model exchange from the FMI Toolbox was used to simulate the model. A MATLAB script is used to specify measurements, define an optimization problem, and to provide an initial guess of the desired parameters. The complete system used for all experiments discussed in this paper is configured in Simulink is shown in Figure 8, labeled as follows:

- A. Input voltage measurements split into a real and imaginary component. Measurements are from PMUs.
- B. FMU containing the Modelica model.
- C. Output of the FMU (created from system in Figure 1); simulated P and Q.
- D. Measurements of P and Q for graphical comparison, used in software for validation.
- E. Output P and Q results to the work space. This is updated every iteration.
- F. Scopes to monitor the simulated response against the measurements during each simulation run.

These models were calibrated using the toolbox with the methods outlined in Section 3. This was necessary to determine the change in the parameters due to the aging components in the system.

### 3 Parameter Identification and Calibration Methods

The `fmincon` and particle swarm optimization (PSO) solvers are used in the parameter estimation experiments for the user defined models. The PSO solver is used to find a global solution for the parameters when calibrating each

component in the model. Once a global solution is found, `fmincon` is used to optimize the local solution.

In Equation (1), the lower and upper limits to the optimization problem are defined as  $p_{min}$  and  $p_{max}$ .

$$\min f(x) \text{ such that } \{ p_{min} \leq x \leq p_{max} \}. \quad (1)$$

During each optimization run using PSO and `fmincon`, the estimated parameter vector  $\bar{p}$  is continuously optimized and updated to simulate the response of the system. The absolute difference between the simulation and measurements is calculated at each time step, as follows:

$$\epsilon_1 = \begin{bmatrix} P_{out}^{simulated} - P_{out}^{reference} \\ Q_{out}^{simulated} - Q_{out}^{reference} \end{bmatrix}^T \quad (2)$$

where  $P_{out}^{simulated}$  is the output from the simulation of the FMU for the active power,  $P_{out}^{reference}$  is the active power measurement from the PMU data;  $Q_{out}$  is the reactive power and follows the same process. The objective function is then determined by computing the Frobenius norm from the mismatch  $\epsilon_1$  between the simulation and PMU measurements for active and reactive power of the generator. The sum of mismatches is calculated from the norms of the measurement/simulation pair at each time step, returning the fitness of the simulated model to the measurements. The fitness of the active and reactive power are weighted equally when optimizing the parameters by minimizing:

$$f(x) = \sum_{i=1}^m \sum_{j=1}^n (\epsilon_{ij} * \epsilon_{ij}) \quad (3)$$

This is repeated for each individual parameter calibrated in this paper listed in the Appendix.

The model shown in Figure 1 was exported from Dymola as a Functional Mock-up Unit (FMU) (Dassault, 2018) to calibrate the models' parameters using MATLAB's optimization solver, `fmincon` (Mathworks), a particle swarm optimization (PSO) solver, and Simulink. The `fmincon` optimization runs were executed for up to 5000 iterations using an error tolerance of  $1 \times 10^{-5}$ . The PSO optimization runs were executed for up to 200 iterations with an error tolerance of  $1 \times 10^{-3}$ . The parameter values were changed each iteration with the goal of providing the best optimization fit to the reference measurements using RaPID (L. Vanfretti, M. Baudette, A. Amazouz, T. Bogodorova, T. Rabuzin, J. Lavenius, F. Jose Gomez-Lopez, 2016), a MATLAB toolbox for rapid parameter identification.

The optimization process used in this paper for the comparison with the generic IEEE power system models is described in (M. Podlaski, L. Vanfretti, J. Pesente and P. H. Galassi, 2019). Both the user-defined and generic models are calibrated using the same process.

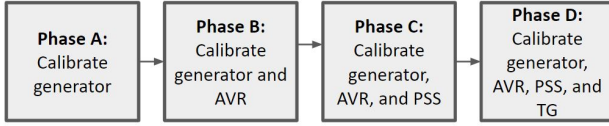


Figure 9. Process for calibrating user-defined components.

### 3.1 Sequential Parameter Estimation Methodology for User-Defined Models

The optimization process used in this paper is shown in Figure 9. First the generator parameters are calibrated without any of the control system included in the model. All of these parameters are calibrated simultaneously using PSO. The results from the PSO are then used as the initial guess to calibrate the parameters sequentially. The sequential parameter estimation process follows the method shown in Algorithm 1.

---

**Algorithm 1** Sequential parameter calibration using heuristic solver, then gradient solver

---

**Use PSO algorithm:**

$x$  = vector for all parameters to be calibrated  
 $x_{min}$  = vector of lower limit of all parameters calibrated  
 $x_{max}$  = vector of upper limit of all parameters calibrated  
 $x = \min f(x)$  such that  $x_{min} \leq x \leq x_{max}$

**Use fmincon solver:**

$p$  = empty vector  
 $p_{min}$  = empty vector  
 $p_{max}$  = empty vector  
 For parameter in  $x$ :  
 $p = p.append(x)$  % Use results of PSO as starting guess  $x$   
 $p_{min} = p_{min}.append(x_{min})$   
 $p_{max} = p_{max}.append(x_{max})$   
 $\min f(x)$  such that  $p_{min} \leq p \leq p_{max}$

---

This method follows the sequence defined in Figure 9. After calibrating the generator parameters, the final solution of the generator `fmincon` optimization is used as the initial guess of generator parameters for calibrating the AVR. Similarly to the calibration of the generator model, Algorithm 1 is used to calibrate the parameters of the generator and AVR model. A PSO routine is run for all of the parameters in the generator and AVR, then its solution is used as the initial guess for the calibration of the individual AVR parameters with the `fmincon` solver. This process is then repeated again for the PSS and TG, adding them into the system sequentially.

### 3.2 User-defined models vs. Generic Models

In this paper, two different models for calibration are studied: generic and user-defined. The generic models are re-implemented from the model description provided in IEEE's Recommended Practice for Excitation System Models for Power System Stability Studies (IEEE, 2016). These models in (IEEE, 2016) are also implemented in other power system analysis tools, including the OpenIPSL library. In this study, the GENSAE syn-

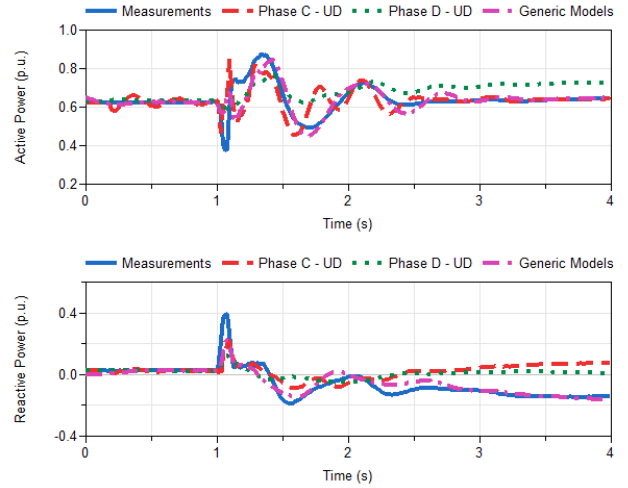


Figure 10. Calibration results for User-defined AVR/PSS calibration (red), AVR/PSS/TG (green), and IEEE generator AVR/PSS/TG (purple) for September 22, 2015 event.

chronous generator, SEXS AVR, and STAB3 PSS are used to model the Itaipu system for the generic model studies. The generic TG model is the IEEE HYGOV model (NEPLAN AG, b), which follows the model definition for an IEEE standard turbine-governor in a hydro plant. The control diagrams and equations defined by IEEE to create these models are include in the Appendix.

The user-defined models used to model the Itaipu system were created by the plant engineers in CDUEdit, which is explained in Section 2.1.3. These models in Dymola are the exact same as the model description in the CDUEdit software. They have the same performance in both ANATEM and in Dymola as shown in Section 2.1.4.

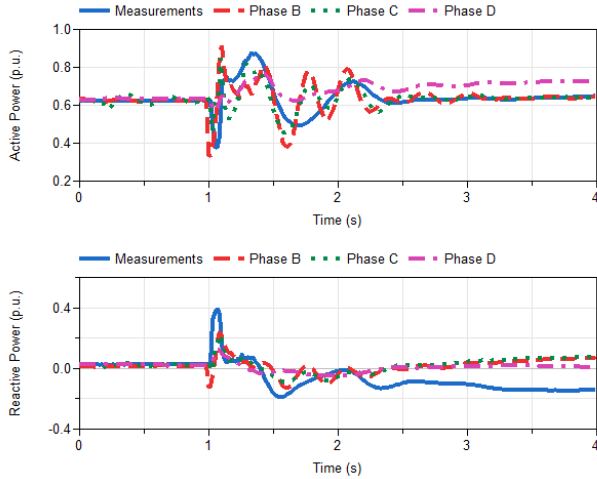
## 4 Case Studies

Data from two different fault events occurring at a generator at Itaipu were used for this model validation and calibration process. The models with calibrated parameters were simulated in Dymola with a variable time step solver, Dassl with a tolerance of  $10^{-3}$ .

### 4.1 Results - Case 1: September 22, 2015

The model was calibrated using both user-defined components and IEEE generic components (M. Podlaski, L. Vanfretti, J. Pesente and P. H. Galassi, 2019) for the AVR, PSS, and TG models. Figure 10 shows the results of the AVR and PSS calibration for this data set using both the user-defined and IEEE generic models. The generic models show graphically that they perform better at modeling the fault response than the user-defined models. The Euclidean norm according to Equation 3 of the user-defined models is 1.2; the IEEE generic models have a Euclidean norm of 1.1017. The IEEE generic models have a slightly better fit than the user-defined models. The results for all of the calibration steps are shown in Table 1.

The AVR/PSS calibration shows a better fit than the AVR only calibration according to Figure 11. The accu-



**Figure 11.** All calibration phases for user-defined models compared for September 22, 2015 data set.

**Table 1.**  $\|x\|$  fitness of results per model for September 22, 2015 and November 2, 2016 events

Model Setup/Date	9/22/2015	11/2/2016
<b>GENSAE generator</b>	1.9548	1.0884
<b>Generic AVR</b>	1.5767	2.2833
<b>Generic AVR/PSS</b>	1.1017	1.1017
<b>Generic AVR/PSS/TG</b>	1.3192	1.3288
<b>UD AVR</b>	2.2875	2.2827
<b>UD AVR/PSS</b>	1.2	1.1055
<b>UD AVR/PSS/TG</b>	2.0944	5.6683

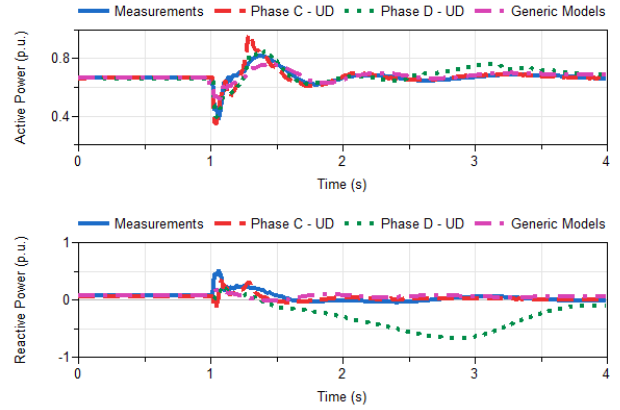
racy of the model decreases when the TG is added to the system, causing distrust in that model. The TG model significantly damps the active power output from the generator and provides too much reactive support. The reference signals, such as reference voltages, were not calibrated in the TG and may need to be adjusted in the future to develop a more accurate model.

## 4.2 Results - Case 2: November 2, 2016

The model was calibrated using both user-defined AVR, PSS, and TG models and IEEE generic AVR and PSS models (M. Podlaski, L. Vanfretti, J. Pesente and P. H. Galassi, 2019). Figure 12 shows the results of the AVR and PSS calibration for the data set using the generic and user-defined control system models. The user-defined models have a better fit with the measurements than the generic models.

Figure 15 shows the results of the calibration at each step. When the TG is added to the model, the simulation creates a poor fit to the measurements. The TG damps the power output from the generator and causes a large dip in reactive power when the system returns to steady state around 1.5 seconds.

The user-defined TG model has an issue similar to the results of the previous data set where the power from the



**Figure 12.** Calibration results for User-defined AVR/PSS calibration (red), AVR/PSS/TG (green), and IEEE generator AVR/PSS/TG (purple) for for November 2, 2016 event.

generator is significantly damped. The addition of the TG in the model causes an over correction of reactive power after the fault, as shown in Figure 15. After the fault, the TG does not let the system return to a steady state; there is a slow oscillation most evidently seen in the reactive power response of the model. When the mechanical power of the generator is used to control the input (shown in Figure 1 with block C removed), the system recovers to near steady state within 1-2 seconds after the fault, as shown in Figure 16. The mechanical turbine power is held constant due to the absence of the turbine-governor.

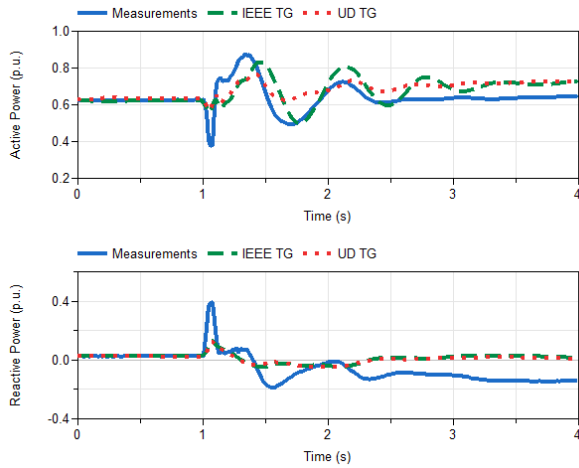
The fitness of the model calibration to the measurements are shown in Table 1. The fitness of the AVR/PSS user-defined model and the generic model are comparable, but the fitness significantly decreases when the TG is included in the model.

## 4.3 Errors with the User-Defined TG Model

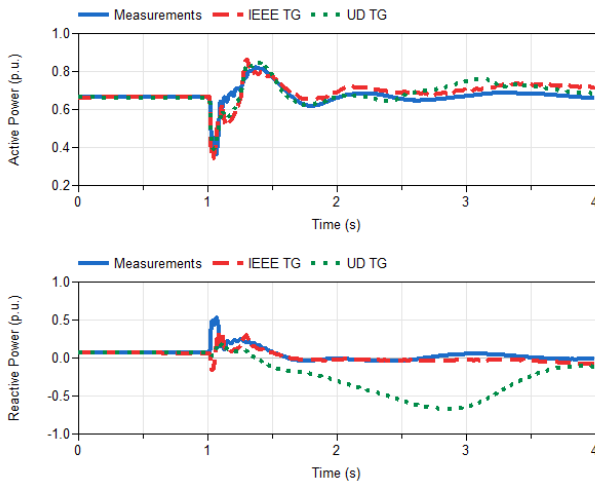
The user-defined TG causes such a high error. Figures 13 and 14 shows the results for the power system containing models of the re-implemented user-defined AVR and PSS. The TG was varied between a calibrated model for the user-defined TG and the IEEE HYGOV TG model. The IEEE model is more accurate than the user-defined model in this case, showing that the user-defined TG model has some error that causes the machine to absorb large amounts of reactive power. Figure 14 shows a drastic change between the user-defined Itaipu TG and the IEEE HYGOV TG response. There seems to be an error in the transfer function of the user-defined TG that causes the machine to consume a large amount of reactive power under certain conditions instead of going back to steady-state like the actual system response.

## 5 Discussion

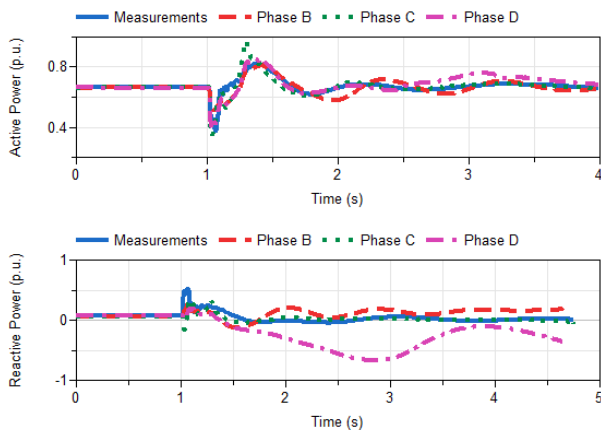
While carrying out this work, it was expected that the user-defined model description would produce a more accurate result than the generic models. The model of the Itaipu



**Figure 13.** Final phases for user-defined AVR and PSS models with both a user-defined and IEEE standard TG model compared for September 22, 2015 data set.



**Figure 14.** Final phases for user-defined AVR and PSS models with both a user-defined and IEEE standard TG model compared for November 2, 2016 data set.



**Figure 15.** All calibration phases for user-defined models compared for November 2, 2016 data set.

plant was most likely originally described in the 1990's. There have been changes in the models since, but they have not updated to be reflected in the model. The user-defined models were created back before the IEEE standard models were originally defined. Although the IEEE standard models are simplified, they have a broader application scope and are able to capture the actual system response and dynamics with better accuracy.

These observations raise the importance of model maintenance and model validation. Thanks to the availability of PMU measurements, this is becoming more possible. This was not possible before, where the only measurements were taken during commissioning tests that do not fully reflect the entire spectrum of the system response.

## 6 Conclusion

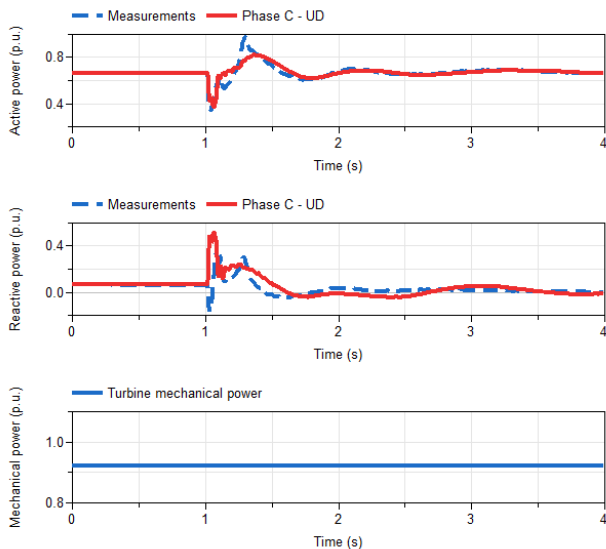
The user-defined models did not perform as well as the generic models for the control system of the plant after calibrating the parameters for the two faults analyzed. Although the fitnesses of the two modeling methods are comparable when the AVR and PSS are included in the system, the generic models are consistently more accurate than the user-defined models. In both cases studied, the fitness of the model increases when the user-defined AVR is added to the system from the basic generator only model; however, in the generic case, the models show an improvement in fitness to the PMU measurements after each parameter is calibrated. In the future, the models used for the Itaipu power plant need to be corrected to better to fit the actual response of the plant to the simulation.

These results show that the models currently used for power systems cannot be blindly trusted without the type of analysis shown in this paper. The approximations of the models do not capture all of the behaviors in the physical system, causing distrust in the models. For example, there is a 20 Hz oscillation seen in the measurements that the models cannot replicate, as shown in Figure 17. This implies for both the user-defined and generic models that more detailed representations of certain components need to be developed. This implies that the models need to be revisited to be able to capture the behavior, but also the traditional modeling approach may be insufficient.

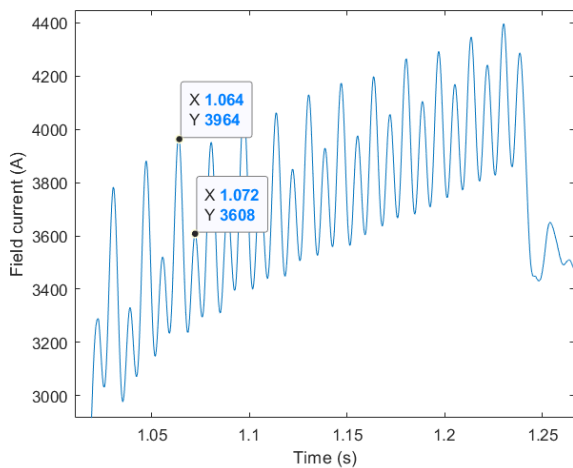
## Acknowledgements

This work was funded in whole or part by by the National Aeronautics and Space Administration through the University Leadership Initiative award for the Center for Cryogenic High-Efficiency Electrical Technologies for Aircraft (CHEETA), the New York State Energy Research and Development Agency (NYSERDA) through the Electric Power Transmission and Distribution (EPTD) High Performing Grid Program under agreement number 137940, the Engineering Research Center Program of the National Science Foundation and the Department of Energy under Award EEC-1041877, in part by the CURENT Industry Partnership Program, in part by Dominion En-





**Figure 16.** Response of the system consisting of generator, AVR, and PSS during the event on November 2, 2016.



**Figure 17.** 20 Hz oscillation in field current from measurements.

ergy, and in part by the ECSE Department at RPI.

The first author was supported by Rensselaer Polytechnic Institute through the Rensselaer Graduate Fellowship 2019-2020.

## References

Stefan Andersson and Jonatan Strömner. Model calibration of a vertical wind power plant using Dymola/Modelica. 2013.

Cepel. ANAREDE - Network Analysis Program, a.

Cepel. ANATEM - Analysis of Electromechanical Transients, b.

Cepel. CDUEdit - User-Defined Controller Editor, c.

Dassault. Dymola user manual vol 2. 6.10. Technical report, Dassault Systemes AB, March 2018.

IEEE. IEEE Recommended Practice for Excitation System Models for Power System Stability Studies. Technical report, IEEE Power and Energy Society, 2016. URL <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7553421>.

Itaipu Binacional. Generating units.

J. Chen, P. Shrestha, S. Huang, N. D. R. Sarma, J. Adams, D. Obadina, and J. Ballance. Use of synchronized phasor measurements for dynamic stability monitoring and model validation in ERCOT. In *2012 IEEE Power and Energy Society General Meeting*, pages 1–7, July 2012. doi:10.1109/PESGM.2012.6345152.

P. Kundur. *Power System Stability and Control*. McGraw-Hill.

L. Vanfretti, M. Baudette, A. Amazouz, T. Bogodorova, T. Rabuzin, J. Lavenius, F. Jose Gomez-Lopez. RaPIId: A modular and extensible toolbox for parameter estimation of Modelica and FMI compliant models. 2016. doi:10.1016/j.softx.2016.07.004.

L. Vanfretti, W. Li, T. Bogodorova and P. Panciatici. Unambiguous power system dynamic modeling and simulation using modelica tools. *IEEE Power Energy Society General Meeting*, pages 1–5, 2013. doi:10.1109/PESMG.2013.6672476.

M. Baudette, L. Vanfretti, J. Rabuzin, M. Murad. iTesla Power Systems Library (iPSL): A Modelica Library for Phasor Time-Domain Simulations. doi:10.1016/j.softx.2016.05.001.

M. Podlaski, L. Vanfretti, J. Pesente and P. H. Galassi. Automated parameter identification and calibration for the itaipu power generation system using modelica, fmi, and rapid. *7th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, pages 1–6, 2019. doi:10.1109/MSCPES.2019.8738793.

Mathworks. fmincon Interior-Point Algorithm with Analytic Hessian. Technical report.

Modelon. FMI Toolbox User’s Guide 2.6.4. Technical report, Modelon AB, July 23 2018.

NEPLAN AG. EXCITER MODELS: Standard Dynamic Excitation Systems in NEPLAN Power System Analysis Tool. Technical report, NEPLAN AG, a. URL [https://www.neplan.ch/wp-content/uploads/2015/08/Nep\\_EXCITERS1.pdf](https://www.neplan.ch/wp-content/uploads/2015/08/Nep_EXCITERS1.pdf).

NEPLAN AG. TURBINE-GOVERNOR MODELS: Standard Dynamic Turbine-Governor Systems in NEPLAN Power System Analysis Tool. Technical report, NEPLAN AG, b. URL [https://www.neplan.ch/wp-content/uploads/2015/08/Nep\\_TURBINES\\_GOV.pdf](https://www.neplan.ch/wp-content/uploads/2015/08/Nep_TURBINES_GOV.pdf).

NEPLAN AG. POWER SYSTEM STABILIZER MODELS: Standard Dynamic Power System Stabilizers in NEPLAN Power System Analysis Tool. Technical report, NEPLAN AG, c. URL [https://www.neplan.ch/wp-content/uploads/2015/08/Nep\\_PSSs.pdf](https://www.neplan.ch/wp-content/uploads/2015/08/Nep_PSSs.pdf).

P. Pourbeik, G. Chown, James Feltes, F. Modau, S. Sterpu, R. Boyer, K. Chan, L. Hannett, D. Leonard, L.T.G. Lima, W. Hofbauer, L. Gerin-Lajoie, S. Patterson, J. Undrill, and F. Langenbacher. Technical report, IEEE, 01 2013.

Z. Huang, P. Du, D. Kosterev, and S. Yang. Generator dynamic model validation and parameter calibration using phasor measurements at the point of connection. *IEEE transactions on power systems*, 28(2):1939–1949, 2013.

## APPENDIX: Variables and Parameters GENSAE generator

Parameter	Details
Generator - IEEE GENSAE Model	
$T'_{d0}$	d-axis transient open circuit time constant
$T''_{d0}$	d-axis sub transient open circuit time constant
$T''_{q0}$	q-axis sub transient open circuit time constant
H	Inertia constant
D	Speed damping
$X_d$	d-axis reactance
$X'_d$	d-axis transient reactance
$X''_d$	d-axis sub transient reactance
$X''_q$	q-axis sub transient reactance
$X_q$	q-axis reactance
$X_l$	leakage reactance
AVR - Itaipu User-Defined Model	
$K_v$	AVR integrator gain
$K_{ei}$	AVR gain
$K_{min}$	Underexcitation limiter gain
$K_{point}$	AVR gain
$T_i$	Overexcitation limiter time constant
$T_a$	Overexcitation limiter time constant
$T_b$	Overexcitation limiter time constant
$T_{ai}$	Overexcitation limiter time constant
PSS - Itaipu User-Defined Model	
$K_f$	
$K_{f1}$	AVR time constant
$T_f$	AVR gain
$T_p$	PSS time constant
$K_1$	PSS time constant
$T_1$	PSS time constant
$K_2$	PSS gain
$T_2$	PSS gain
TG - Itaipu User-Defined Model	
$T_n$	Accelerometer time constant
$NT_v$	Adjustment of accelerometer time constant
$T_d$	Integrator time constant
$T_{f1}$	Time of closing distributor fast part
$T_{f2}$	Time of closing distributor slow part
$T_v$	Equivalent time of distributor valve
$T_w$	Water starting time
$T_{ya}$	Time of opening of the distributor

## APPENDIX: Equations for IEEE standard components

### IEEE HYGOV Turbine Governor

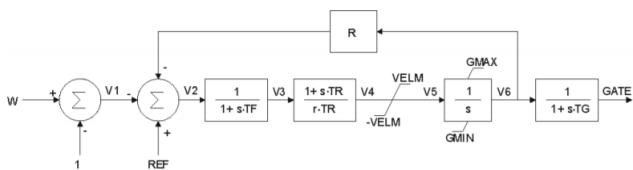


Figure 18. Control diagram for IEEE HYGOV TG (NEPLAN AG, b)(Pourbeik et al., 2013)

### GENSAE generator

$$K_{1d} = \frac{(X'_d - X''_d)(X_d - X'_d)}{(X'_d - X_l)^2}$$

$$K_{2d} = \frac{(X'_d - X_l) * (X''_d - X_l)}{(X'_d - X''_d)}$$

$$K_{3d} = \frac{X''_d - X_l}{X'_d - X_l}$$

$$K_{4d} = \frac{X'_d - X''_d}{X'_d - X_l}$$

$$\frac{dE_{pq}}{dt} = \frac{1}{T_{pd0}(E_{fd} - X_{ad}I_{fd})}$$

$$\frac{d\Psi_{kd}}{dt} = \frac{1}{T''_{d0}(E'_q - \Psi_{kd} - X'_d - X_l) * id}$$

$$\frac{d\Psi''_q}{dt} = \frac{1}{T''_{q0}(-\Psi''_q + (X_q - X''_q)) * id}$$

$$\Psi_{d'} = E'_q + K_{3d} + \Psi_{kd}K_{4d}$$

$$\Psi_d = \Psi''_d - X''_d * i_d$$

$$\Psi_q = -\Psi''_q - X''_q * i_q$$

$$X_{ad}I_{fd} = K_{1d} * (E'_q - \Psi_{kd} - (X'_d - X_l) * i_d) + (X_d - X'_d) * i_d + (SE_{exp} + 1) * E'_q$$

$$T_e = \Psi_d * i_q - \Psi_q * i_d$$

$$u_d = (-\Psi_q) - R_a * i_d$$

$$u_q = \Psi_d - R_a * i_q$$

### SEXS AVR

$$V_{REF} = E_{fd0}/K + E_{COMP0}$$

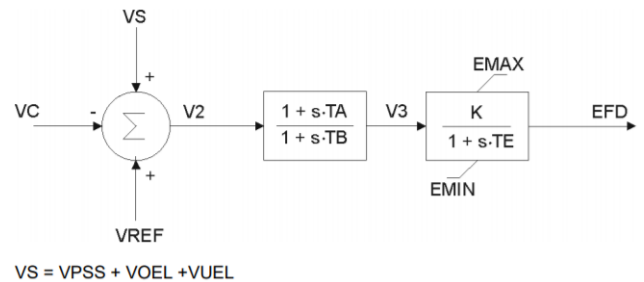


Figure 19. Control diagram for SEXS AVR(IEEE, 2016)(NEPLAN AG, a)

### STAB3 PSS

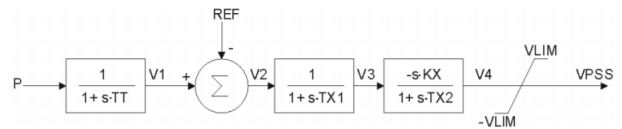


Figure 20. Control diagram for STAB3 PSS(IEEE, 2016)(NEPLAN AG, c)