RPI

ALSETlab

Enhancing Large-Scale Power Systems Simulations through Functional Mock-up Unit-based Grid-Forming Inverter Models

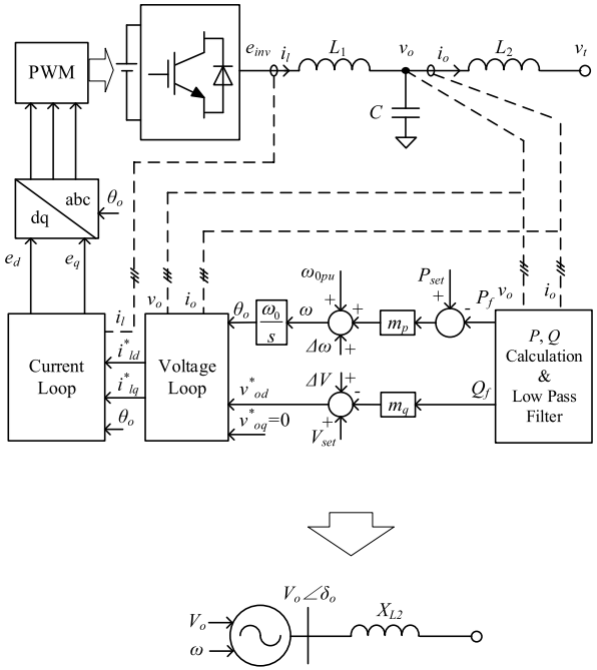
September 8-10, 2025, Lucerne, Switzerland

Luigi Vanfretti
Professor, ECSE Dept., RPI, Troy, NY, United States
Luigi.Vanfretti@gmail.com, https://alsetlab.github.io/

16th modelica&fmi Conference

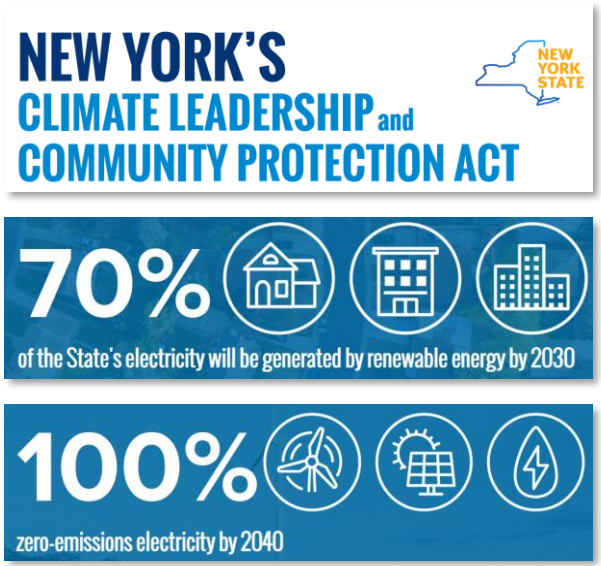
Outline

- Motivation and Background
  - Motivation
  - Background – Grid Forming Converters (GFM)
- GFM Library
  - Overview
  - Voltage Sources
  - Active Power-Frequency Droop-based Control (P-f)
  - Reactive Power-Voltage Control (Q-v)
  - Testing
- FMU Export
  - Modelica FMU Wrapper
  - FMU Generation for ePHASORSIM
- Data Exchange Between ePHASORSIM and FMUs
- Example
- Advantages of FMU-based Approach
- Conclusions and Further Work



Grid Forming Converter Schematic and Positive Sequence Representation

- **NYS Climate Goals:** 70% renewable by 2030, decarbonize grid by 2040
- **Challenges:** Extensive simulations needed for renewable impact assessment, high modeling and computational efforts
- **Solution:** Develop FMI v1.0 Compliant Functional Mock-up Unit (FMU) for Grid-Forming (GFM) Inverters
- **Benefits:** Improved simulation, reduced manual modeling
- **GFM Technology:** Supports voltage, frequency, fault ride-through
- **Implementation:** Studied impacts of 6,200+ MVA renewable on 5000-bus NYS grid using Modelica and OPAL-RT's ePHASORSIM™



- **Grid-ForMing converters (GFMs)** are power electronic devices that attempt to act similarly to a synchronous generator, creating their own voltage and frequency to establish an AC grid.
- **How they differ:** Unlike traditional grid-following converters which are current sources that need an existing grid to operate, GFCs are voltage sources that can establish their own terminal voltage → useful for “black-start”, i.e., restoration.
- **Why they're needed:**
  - Solar and wind are interfaced with power converters to the grid,
  - Adding to many converters leads to loss of "inertia" and therefore the stability provided by large, spinning synchronous generators.
  - The hypothesis is that GFCs may be able to support stability and to enable a future grid with 100% renewables.

- Real-World Test of a BESS with GFM [A]:

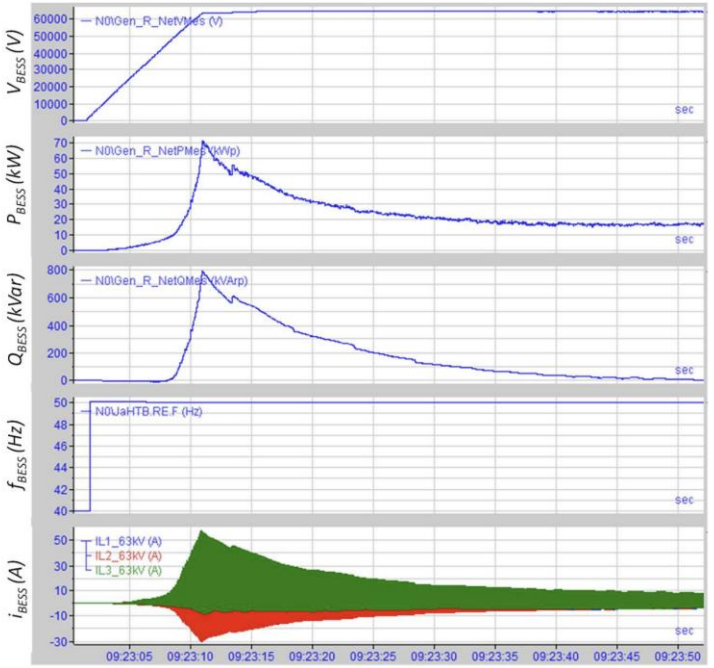
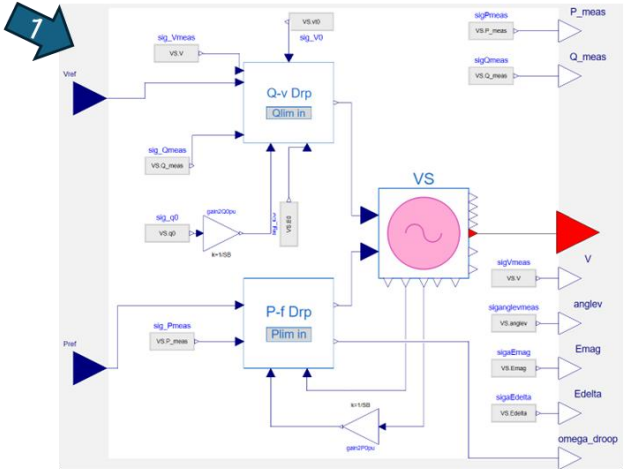


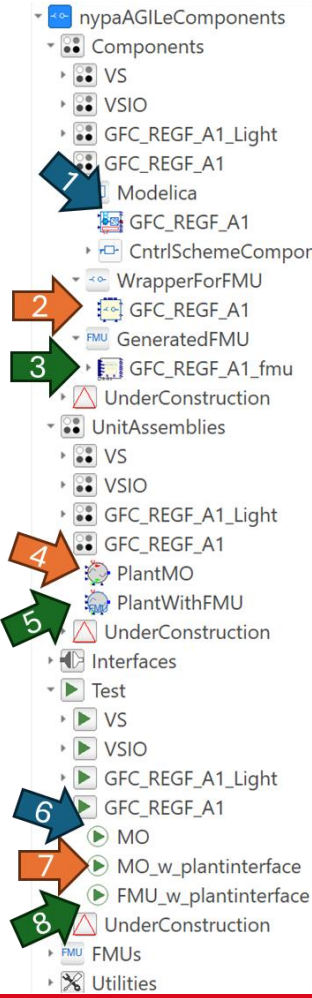
Figure 7 : TEST 1 - BESS soft start-up energization

[A] H. Saad et al., "Experience and EMT study for onsite grid forming test using BESS - RINGO black start project," 21st Wind & Solar Integration Workshop (WIW 2022), Hybrid Conference, The Hague, Netherlands, 2022, pp. 241-249, doi: 10.1049/icp.2022.2760.

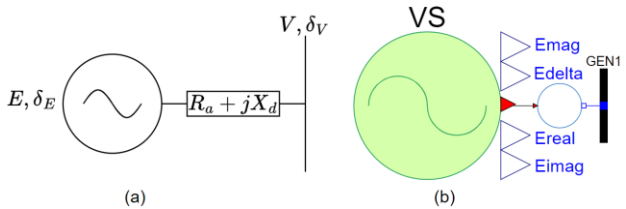
GFM Library - Overview

- Component Models:
  - 1: GFM Component
- Models with Wrapper:
  - 2: Wraps 1 to generate FMU
  - 4: Wraps 2 in a “plant” interface to simulated within a power system
- Models to test FMU:
  - 3: FMU generated
  - 5: Wraps 3 in a “plant” interface to simulate within a power system
- Test system models:
  - 6: Tests the Modelica model.
  - 7: Tests the Modelica model with wrapper interface.
  - 8: Tests the FMU within a Modelica grid model.





- A simple VS with an internal impedance of $R_a + jX_d$ connected to a bus:



- The objective is to find the internal voltage $E\angle\delta_E$ of the source from the active and reactive power (P and Q) at the terminal bus.
- The terminal voltage is denoted by $V\angle\delta_V = V_r + jV_i$
- The $\Re\{\tilde{E}\}$ and $\Im\{\tilde{E}\}$ part of the internal voltage are:

$$\Re\{\tilde{E}\} = E_{\text{real}} = V_r - I_i X_d + I_r R_a$$
$$\Im\{\tilde{E}\} = E_{imag} = V_i + I_i R_a + I_r X_d$$

Component is initialized as:

```
// Initialization of internal VS parameters
Er0 = vr0 + CoB*R_a*ir0 - CoB*X_d*ii0 "Initial
    value of E_real";
Ei0 = vi0 + CoB*R_a*ii0 + CoB*X_d*ir0 "Initial
    value of E_imag";
E0  = sqrt(Er0^2+Ei0^2);
delta0 = atan2(Ei0, Er0);
```

Parameters for VS

nypaAGILeComponents.Components.VS.Modelica.VoltageSource

General | Voltage Source parameters | Attributes

Component

Name

Comment

Icon

Model

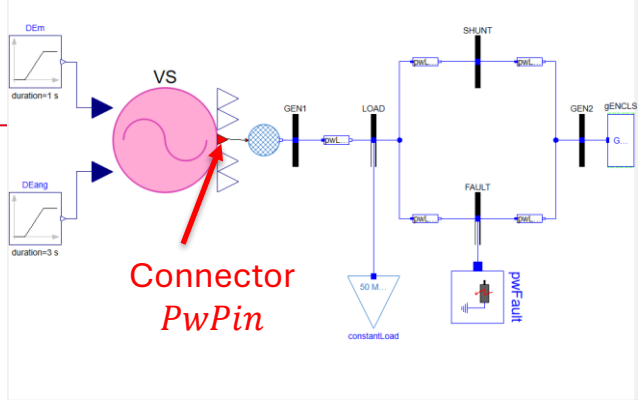
Path nypaAGILeComponents.Components.VS.Modelica.VoltageSource

Comment Simple voltage source with internal impedance

Opal-RT PF and Other Params.

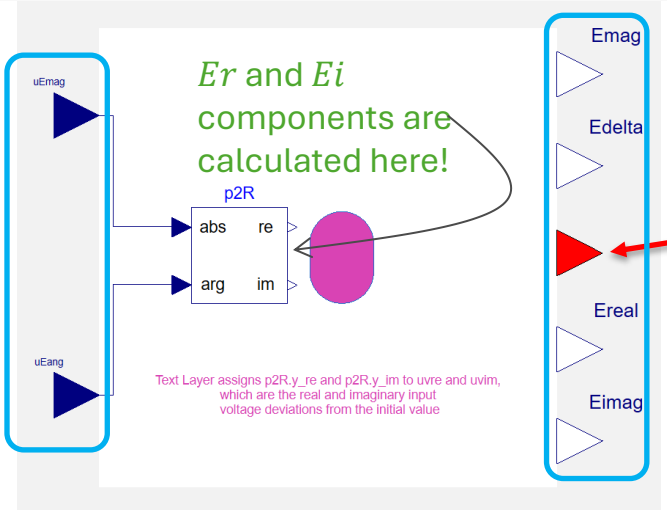
partType	1	
IBUS	100	Located system bus
P_gen	40.0	Bus Active Power, MW
Q_gen	5.416582	Bus Reactive Power, MVAR
Vt_abs	1.0	Bus Voltage Magnitude, p.u.
Vt_ang	4.038907	Bus Voltage Angle, deg.
SB	100	Machine Base Power, MVA
fn	60	Nominal frequency
M_ID	"M1"	Machine Identifier

Voltage Source Component - Interfaces and Connectors



- The VS model within a power system is shown in pink:
- The diagram layer of the VS model shows several interfaces and connectors that couple it to input data and the physical connection to the network:

Interfaces of type "Real" with pre-defined signal flow:
`RealInput` (input) `RealOuput` (output)



The `PwPin` component is an "Acasual Interface" with NO Pre-defined Direction

```
connector PwPin
  Real vr "real part of the voltage";
  Real vi "imaginary part of the voltage";
  flow Real ir "real part of the current";
  flow Real ii "imaginary part of the current";
end PwPin;
```

- The connector defines 4 variables, 2 "potential" (`vr` and `vi`), and two flow (`ir` and `ii`) that need to be solved.
- This implies that there must be equations inside VS that help in that solution.

Coupling PwPin Variables with the internal model equations

- The voltage and current variables of the `PwPin` component, named "**p**", define the terminal voltage variables.
- We link the internal voltage source equations with those variables, which is done as:

```
Er = p.vr + R_a*p.ir - X_d*p.ii;
Ei = p.vi + R_a*p.ii + X_d*p.ir;
```

- These gives us 2 equations; *however, we have 4 unknowns,* the next two equations provide the additional 2 equations needed to make the system "balanced", which are the **power injections**:

```
P = p.vr*p.ir + p.vi*p.ii;
Q = p.vi*p.ir - p.vr*p.ii;
```

- This model can now be coupled with another component that uses the `PwPin` connector, and the equations will be solved.
- However, we are interested in exporting the component as an FMU to work with other tools, so we need to couple the external world to the internal model.

- ePhasorSim uses the **FMI standard version 1**, and in this version, only `RealInput` and `RealOutput` interfaces are used to export the FMU.

Controllable Voltage Source

- The controllable VS is developed based on the equations of the simple VS.
- It has two inputs **uEmag** and **uEang** to vary the voltage magnitude and angle of the VS.
- The input to the controllable VS voltage source can be provided from two sources:
  - **Case1:** inject a voltage phasor to define the internal voltage of the source from an external tool (e.g., a power system simulator like ePHASOSRSIM),
  - **Case 2:** inject a signal that is a deviation from the initial value of E0 anddelta0 for internal testing.

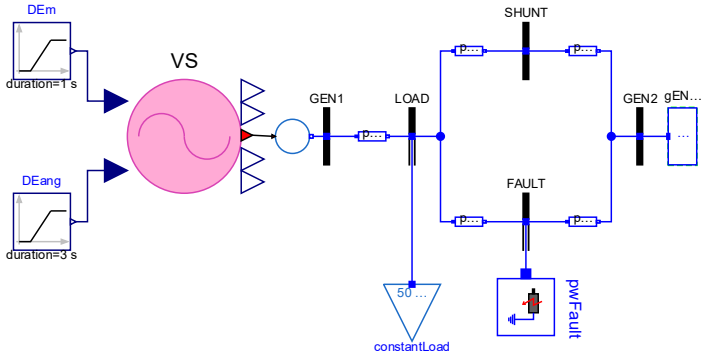
```
if Case 1 then "input from external tool"
    E = uEmag;
    delta = uEang;
    Er = p2R.y_re "Real part of E";
    Ei = p2R.y_im "Imaginary part of E";
else "deviation from initial value"
    E = E0 + uEmag;
    delta = delta0;
    Er = Er0 + p2R.y_re;
    Ei = Ei0 + p2R.y_im;
```

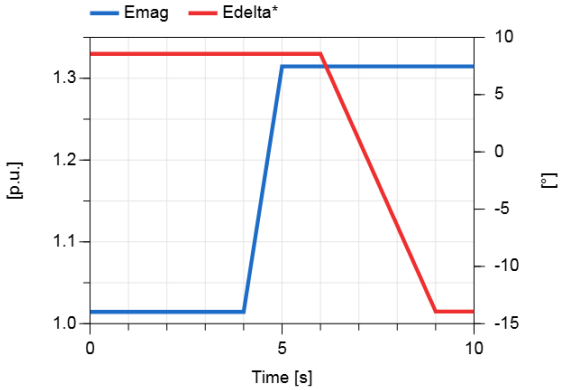
The controllable VS is used as a basic building block to build the GFM IBR model.

The GFM inverter model has two inputs: **uEmag** and **uEang**.
Input **uEang** controls the voltage angle → P-f control loop
Input **uEmag** controls the voltage magnitude → Q-V control loop.

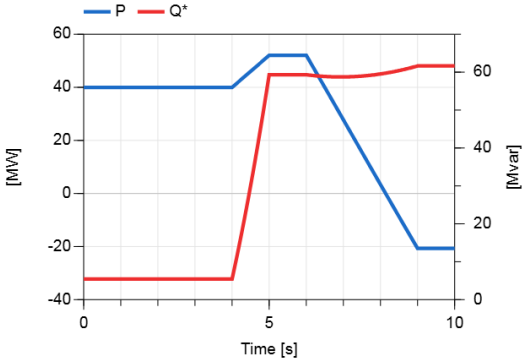
Controllable Voltage Source Example

Models are interfaced to the OpenIPSL SMIB for testing within the Dymola environment.





For Emag, the ramp was activated at 4 s., changing the value from 1.01439 p.u. to 1.03439 p.u. within 1 s.

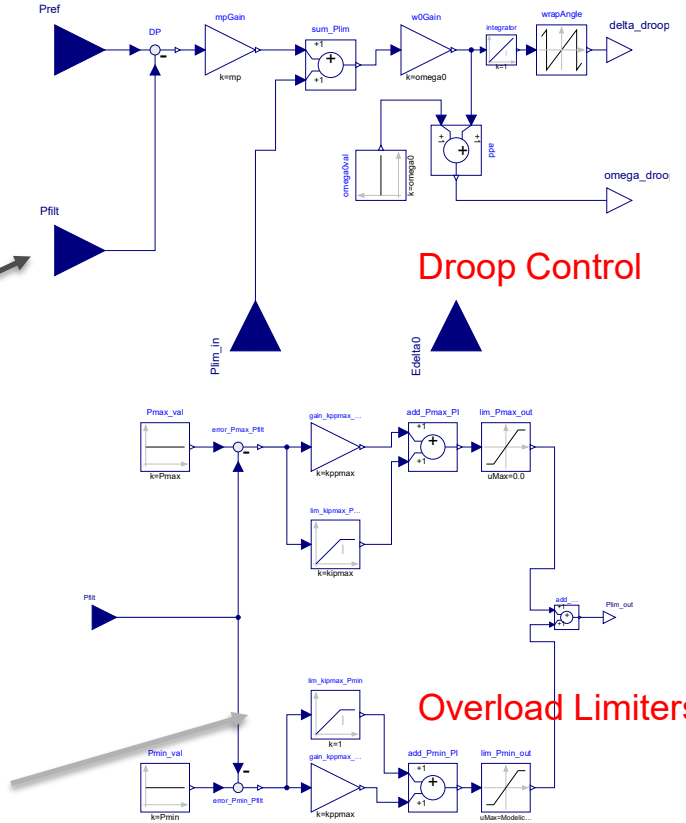


For Eang, the ramp was activated at 6s changing the value from 0.149386 rad to −0.243364 rad within 3 s.

- The P-f droop control adjusts the output frequency of the inverter in response to changes in active power output.
- The active power output of the inverter can be written as:

$$P \approx \frac{EV}{X_L} sin(\delta_E - \delta_v) \approx \frac{EV}{X_L}(\delta_E - \delta_v)$$

- A deviation from P_{ref} will make the inverter active power to change, thus changing the angular frequency Ꞷ of the internal voltage to reduce the inverter power output, which is modeled as a change in δ_E.
- Limiter block prevents the active power output of the inverter from exceeding its limits.



Initializing δ_{droop}

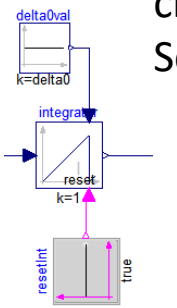
- During initialization we must define an initial equation for the angle
- We also need to set the angle to be exactly equal to that of the VS, and not be modified to the solver
- We want to avoid using `fixed=true` which can create other issues

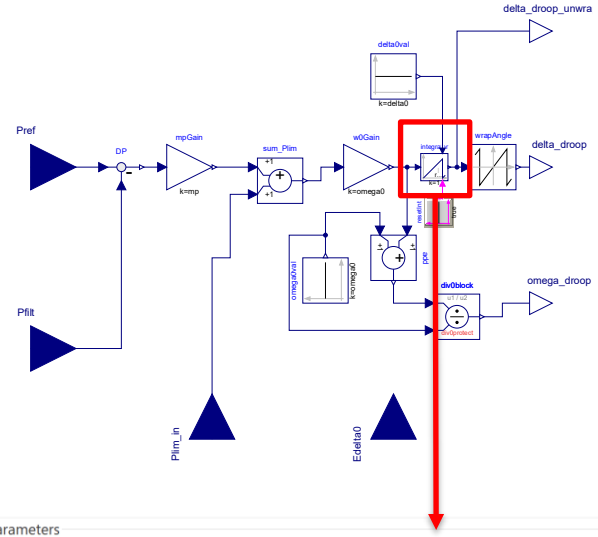
Solution:

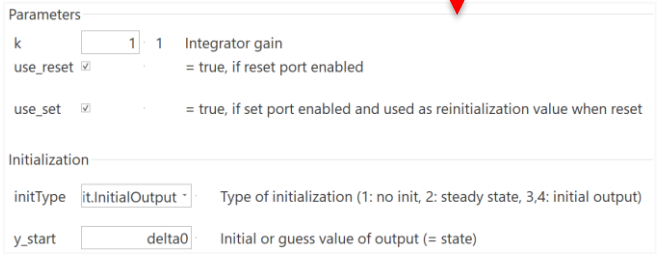
- Use a reset on the integrator
- Set the value to which the integrator is to be reset
- Use that of the VS and not the one found through the initialization→ i.e., use the value assigned during the initial equation

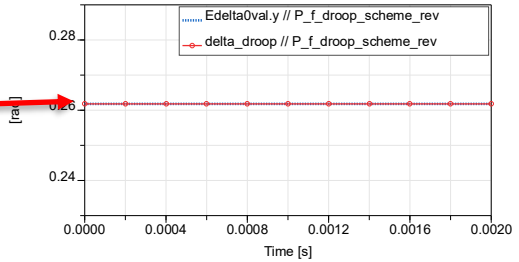
```
initial equation
    delta0 = Edelta0
```
← delta0 = to the VS angle

- After initialization, the voltage source angle and the P-f droop angle are the same.





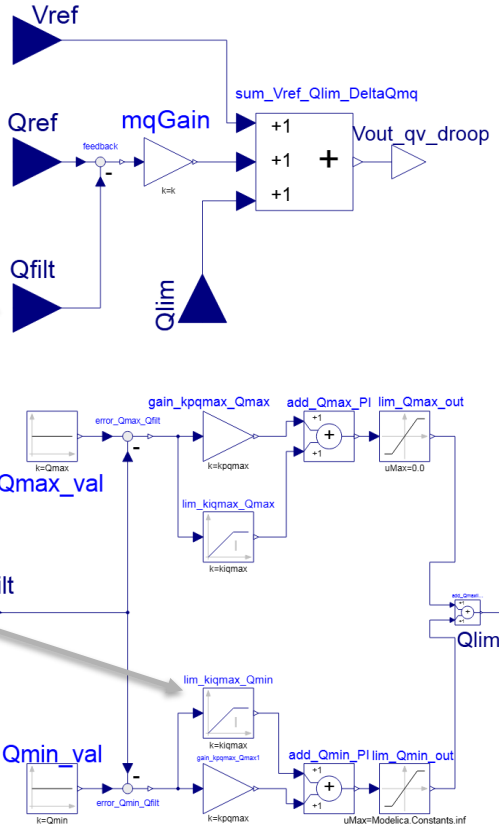




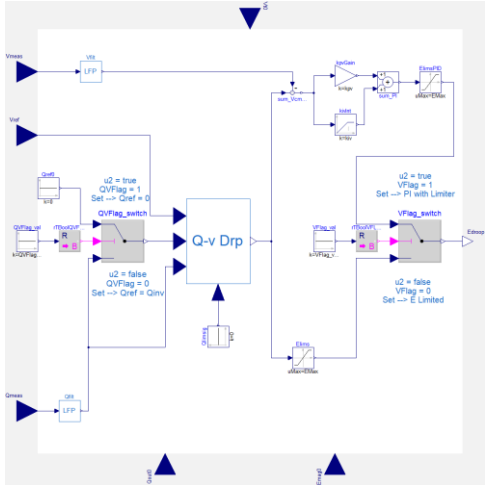
- The reactive power-voltage (or the Q-V) droop-based control adjusts the inverter reactive power output in response to changes in the voltage at its terminals.
- The reactive power output of the inverter can be written as:

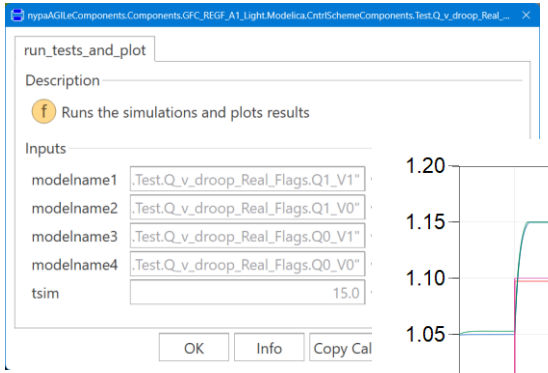
$$Q \approx \frac{E^2 - EV\cos(\delta_E - \delta_v)}{X_L} \approx \frac{E(E-V)}{X_L}$$

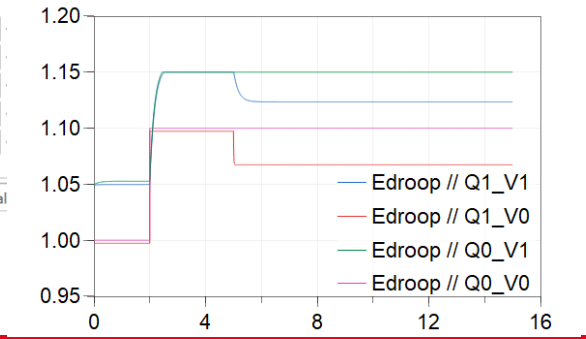
- When there is a change in the terminal voltage, the Q-V droop manipulates the magnitude of the internal voltage to adjust the reactive power output accordingly.
- Output power limiter block prevents the reactive power output from exceeding the converter's limits
- Two reactive power control modes that are set by `QVFlag`: plant controller strategy of adjusting either the reactive power or voltage setpoint.

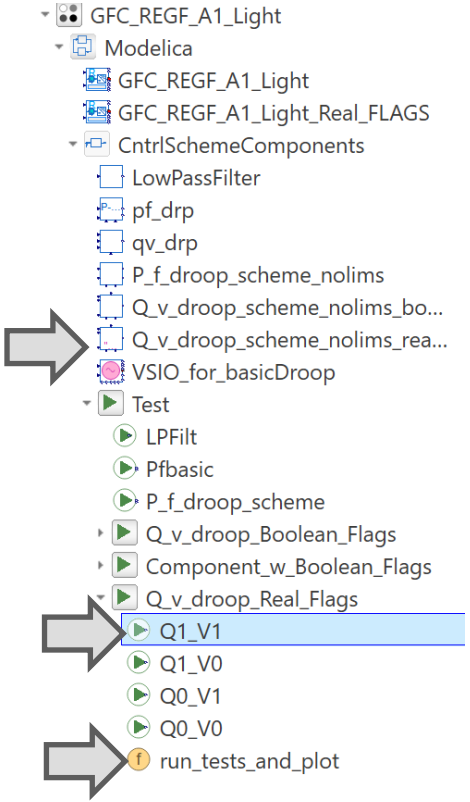


- Q-v control scheme has options for QVFLAG and VFLAG
- Modelica implementation done with Boolean parameters, and these values were propagated.
- Dymola simulations were able to capture the change of parameters.
- Introduced a “Constant” block and a Real-to-Boolean conversion block to minimize the number of changes.



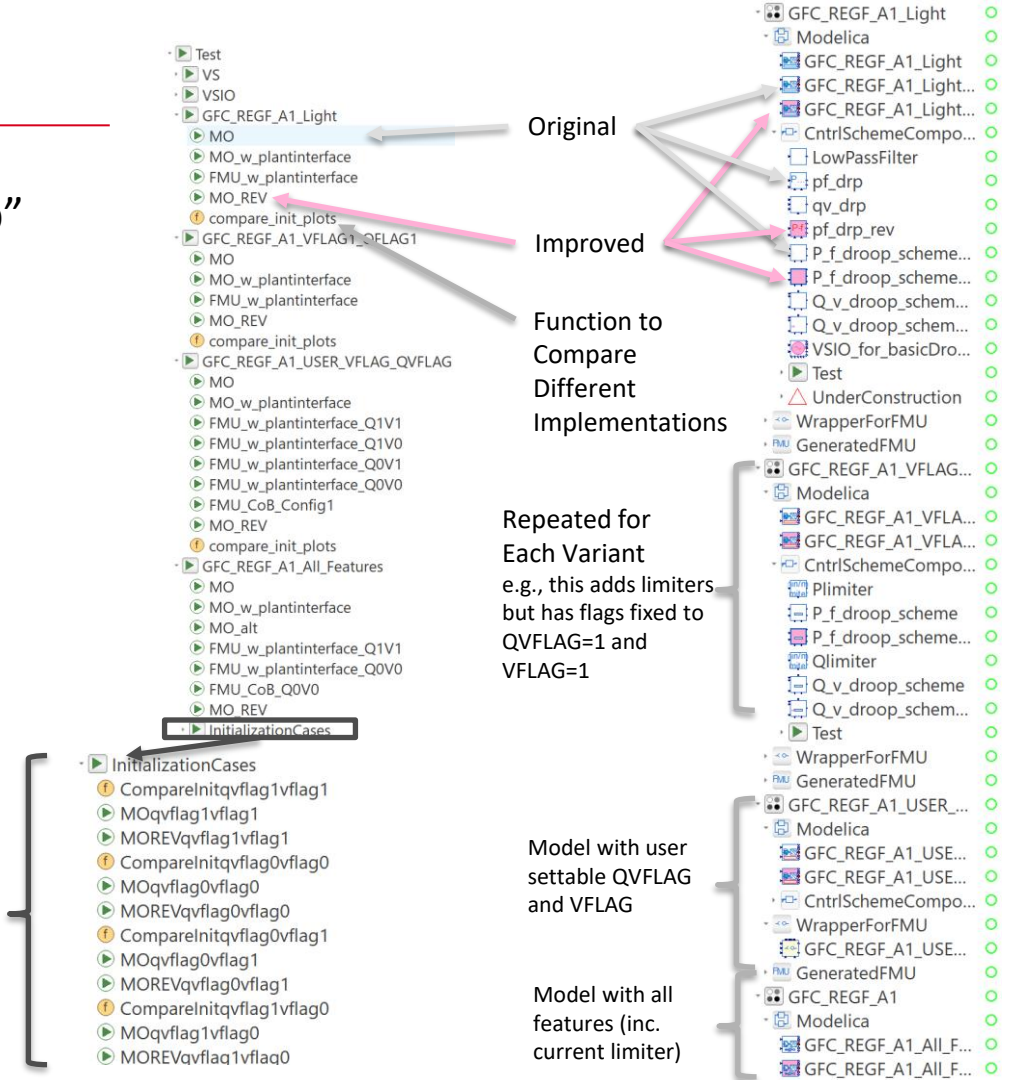






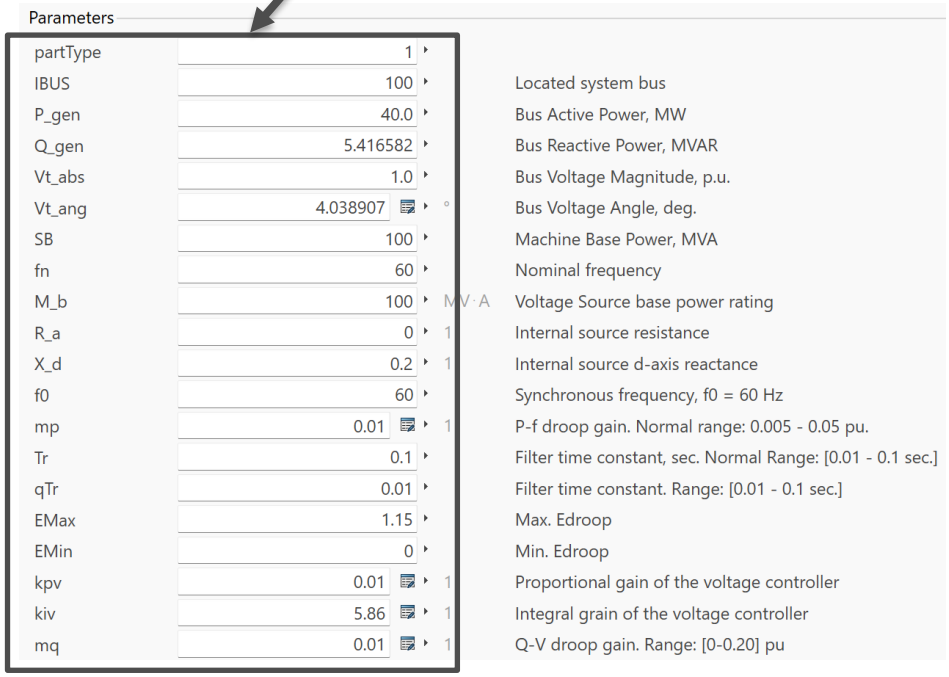
Testing

- We propagate the P-f droop block “up” to the higher levels
- To be able to compare, we have different versions of:
  - P-f Droop / Q-V Loop
  - P-f / Q-V Control Scheme
  - Converter
- All variants tested for QVFLAG = 1 and VFLAG = 1
- Model with all features tested for all combinations of QVFLAG and VFLAG
- Simulates and plots



RPI

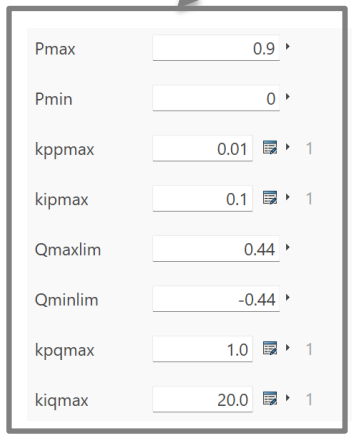
- Basic Parameters:



Parameters

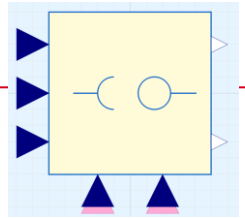
partType	1		
IBUS	100		Located system bus
P_gen	40.0		Bus Active Power, MW
Q_gen	5.416582		Bus Reactive Power, MVAR
Vt_abs	1.0		Bus Voltage Magnitude, p.u.
Vt_ang	4.038907	°	Bus Voltage Angle, deg.
SB	100		Machine Base Power, MVA
fn	60		Nominal frequency
M_b	100	MV·A	Voltage Source base power rating
R_a	0	1	Internal source resistance
X_d	0.2	1	Internal source d-axis reactance
f0	60		Synchronous frequency, f0 = 60 Hz
mp	0.01	1	P-f droop gain. Normal range: 0.005 - 0.05 pu.
Tr	0.1		Filter time constant, sec. Normal Range: [0.01 - 0.1 sec.]
qTr	0.01		Filter time constant. Range: [0.01 - 0.1 sec.]
EMax	1.15		Max. Edroop
EMin	0		Min. Edroop
kpv	0.01	1	Proportional gain of the voltage controller
kiv	5.86	1	Integral grain of the voltage controller
mq	0.01	1	Q-V droop gain. Range: [0-0.20] pu

- Additional Parameters for limiters:

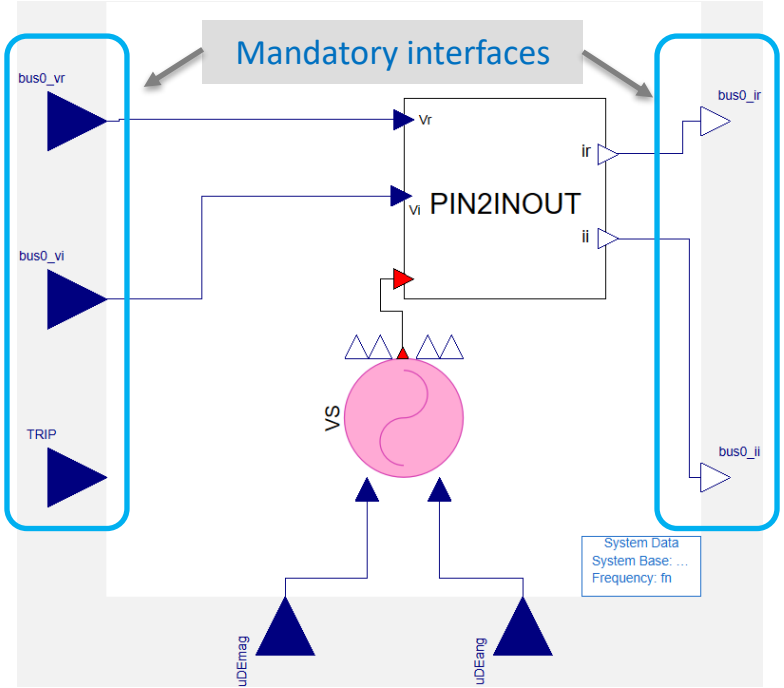


Pmax	0.9		Upper limit of the inverter active power output. Normal Range: [0.1 - 1] pu
Pmin	0		Lower limit of the inverter active power output. The value should be negative when representing energy storage systems.
kppmax	0.01	1	Proportional gain of the overload mitigation controller. Normal Range: [0.005 - 0.05] pu.
kipmax	0.1	1	Integral gain of theoverload mitigation controller. Nomral Range: [0.05-0.2] pu/s
Qmaxlim	0.44		Upper limit of the inverter reactive power output. Normal Range: [0.44-1.0] pu
Qminlim	-0.44		Lower limit of the inverter reactive power output. Normal Range: [-0.44 - -1.0] pu
kpqmax	1.0	1	Proportional gain of the Qmax and Qmin controller. Range when VFLAG=1 is [1-5] pu. Range when VFLAG = 0, [0-0.5] pu.
kiqmax	20.0	1	Integral gain of the Qmax and Qmin Controller. Range when VFLAG=1 is [3-20] pu. Range when VFLAG = 0, [3-30] pu

FMU Export - Modelica to FMU Wrapper



- Created a “Wrapper” that interfaces the **component** to be exported with the **mandatory** interfaces required by ePhasorSim
- It interfaces with the “**block**” from the OpalRT-dev library called “PIN2INOUT”.
- That means that the PIN2INOUT:
  - ***Converts the acasual variables in PwPin*** (p.vr, p.vi, p.ir and p.ii) ***into casualized variables*** (bus0_vr, bus0_vi, bus0_ir, and bus0_ii)
  - ***It assigns the value of variables*** (p.vr, p.vi, p.ir and p.ii) ***in the PwPin to the inputs*** ((bus0_vr, bus0_vi) **and outputs** (bus0_ir, and bus0_ii)
  - This results in the voltage and current having a specific direction, i.e., as input or output, respectively.
  - Consequently, the “bus0_ir” and “bus0_ii” are the currents that will be injected into the network solution of ePHASORSIM

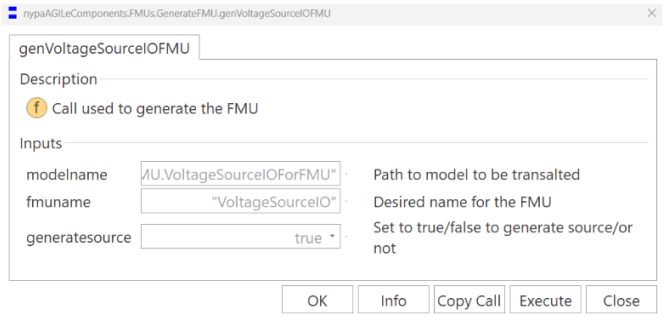


FMU Generation

Dependency: Windows Subsystem for Linux (WSL) 2 with Linux kernel update package

FMU with and without Source Codes

- FMUs can be generated with or without the C source code.
- With Dymola, this capability depends on whether or not the user has a license that enables code export.
- A user with a local “vanilla” Dymola license can regenerate the FMUs and execute off-line local simulations.
- Export of source code is only required when running the ePHASORSIM in real time.



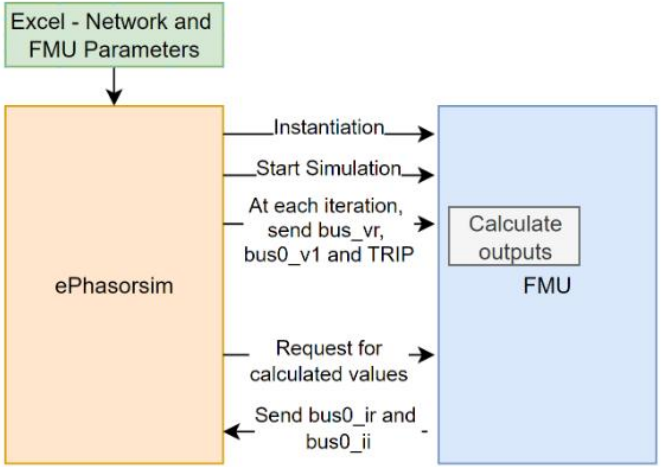
Custom function to generate FMUs with/without source code

```
  extends Modelica.Icons.Function;
  input String modelname = "IBR_VS_Droop_FMU"
  input String fmuname = "GENSAL";
algorithm
  // Flags used in "..\FMUCreator\dymolaFlag.cfg
      "
  Advanced.CheckPackageRestriction :=false;
  Advanced.FMI.xmlIgnoreProtected :=false;
  Advanced.FMI.xmlIgnoreLocal :=false;
  Advanced.FMI.BlackBoxModelDescription :=false;
  Advanced.Define.AimForHighAccuracy :=false;
  Advanced.Define.NewJacobian :=false;
  Advanced.EnableCodeExport := true;
  Advanced.SourceCodeExportNormal :=true;
  Advanced.FMI.CrossExport :=true;
  Advanced.FMI.FMUIncludeSource :=true;
  Advanced.FMI.FMUSourceCodeUniqueNaming :=true;
  // Create FMU
  translateModelFMU(modelname, false, fmuname, "
      1", "all", true, 1);
end generateFMUWithSourceCode;
```

OPAL-RT ePHASORSIM currently supports FMI version 1. We use FMI for **Model Exchange** (ME).

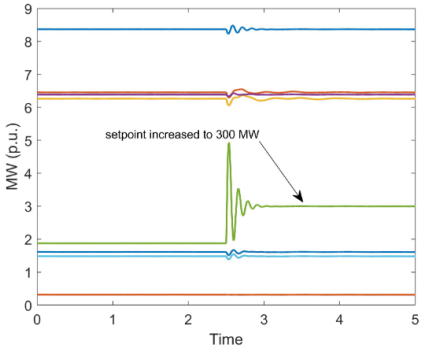
Data Exchange between ePHASORSIM and FMU

- In ePHASORSIM, all parameters for a power system model are defined in an Excel file.
- Once ePHASORSIM reads the network and FMU parameters from the Excel file, it instantiates the FMU and passes the parameter data to it.
- ePHASORSIM uses a fixed time-step solve, at each simulation step, it sends the terminal bus voltage real and imaginary values, bus0_vr and bus0_vi, and any inputs defined in the wrapper to the FMU.
- The FMU equations are evaluated with the received values, and the results are sent back to ePHASORSIM in the form of real and imaginary current injection values bus0_ir and bus0_ii, and any of the output values included in the FMU wrapper.
- ePHASORSIM then solves for all variables within its internal models and assembles the overall network solution.

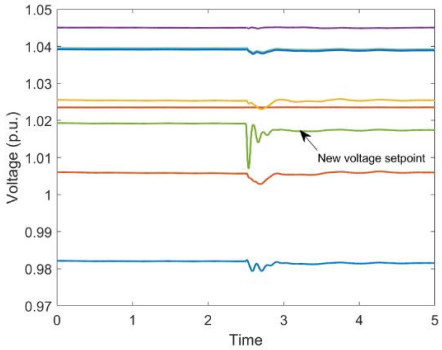


Example - GFM inverter FMU on NYS Transmission

- GFM IBRs, modeled as FMU, were integrated into the NYS 5000-bus, 800-machine transmission system in ePhasorsim.
- A total of 9 GFM IBR with 6272.5 MVA capacity was configured to substitute traditional generators within the NYS transmission system.



GFM response to changes in MW setpoint



GFM response to changes in voltage setpoint

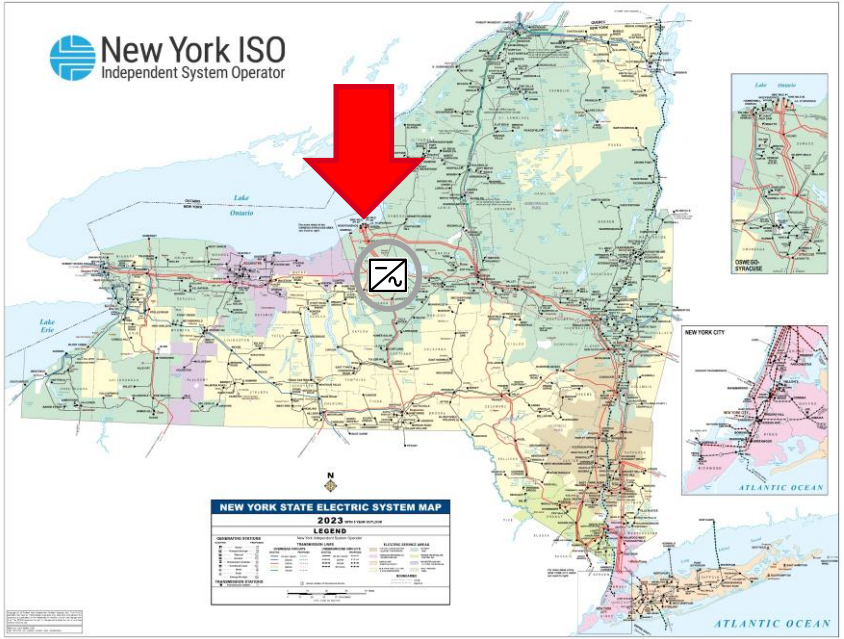


Table 1. Comparison between Simulink-Based and FMU-Based GFM IBR Deployment in ePHASORSIM

Aspect	Simulink-Based Deployment	FMU-Based Deployment
Controller Integration	Manual copy-paste of droop-based GFM controller block each time	Controller is pre-embedded in FMU; no need for repetition
Signal Wiring	Manual setup for each new GFM (all inputs/outputs to/from solver)	Minimal signal mapping (e.g., `Pset`, `Vset`) required
Model Scalability	Tedious and error-prone with more GFMs	Easily scalable to large numbers of GFMs (e.g., hundreds)
Transformer Setup	Details must be manually specified in Transformer sheet	Can be included directly in FMU or in the additional FMU-named sheet
Error Probability	High, especially with manual wiring and naming	Lower, due to fewer steps and more automation
Time Efficiency	Time-consuming due to repetitive steps and complex wiring	Very time-efficient and streamlined for high GFM penetration
Maintenance & Debugging	Difficult to trace and fix due to signal complexity	Easier maintenance

- Demonstrated advantages of FMUs for large-scale power system simulations.
- Developed grid-forming inverter model in Modelica, integrated with ePHASORSIM to evaluate renewable energy impact.
- Utilized Modelica's object-oriented features to create controllable voltage source and GFM-controlled inverter.
- FMUs offer significant time savings in modeling, debugging, and parametrization.

- Future work:
  - GFM Model:
    - Use better object orientation to reuse the Modelica model in other power system tools, e.g., for ePHASORSIM-specific parameters and interfaces.
    - Integrate GFM model in the OpenIPSL library, in progress – link to PR on Github, here
  - Build Modelica models and FMU models for virtual synchronous machines, explore FMU interoperability, investigate IBR parameter uncertainties.

References

[1] "IEEE Guide for Control of Small (100 kVA to 5 MVA) Hydroelectric Power Plants," in IEEE Std 1020-2011, vol., no., pp.1-56, 12 April 2012.

[2] F. P. De Mello and C. Concordia, “Concepts of synchronous machine stability as affected by excitation control,” IEEE Trans. Power App. Syst., vol. PAS-88, pp. 316–329, 1969.

[3] C. Mishra, L. Vanfretti, M. Baldwin, J. de la Ree Jr., and K. D. Jones, “Analysis of Generator Forced Oscillations during MOD 25 Testing Exploiting Wavelets,” Hawaii International Conference on System Sciences 2024 (HICSS), Hilton Hawaiian Village Waikiki, January 3-6, 2024.

[4] E. Larsen and D. Swann, “Applying power system stabilizers, parts I, II and III,” IEEE Trans.Power App. Syst., vol. PAS-100, pp. 3017–3046, Jun. 1981 .

[5] ENTSO-E, “Oscillation Event 03.12.2017,” 2018, Available online: https://tinyurl.com/ENTSOE2017

[6] P. X. Verdugo, A. B. De La Torre, J. C. Cepeda and K. A. Paucar, "Testbed for PSS tuning using synchrophasor measurements and a real-time digital simulator," 2017 IEEE PES Innovative Smart Grid Technologies Conference - Latin America (ISGT Latin America), Quito, Ecuador, 2017, pp. 1-6, doi: 10.1109/ISGT-LA.2017.8126744..

[7] V. S. Perić, X. Bombois and L. Vanfretti, "Optimal Signal Selection for Power System Ambient Mode Estimation Using a Prediction Error Criterion," in IEEE Transactions on Power Systems, vol. 31, no. 4, pp. 2621-2633, July 2016, doi: 10.1109/TPWRS.2015.2477490.

[8] Xavier Bombois and Luigi Vanfretti, “Performance monitoring and redesign of power system stabilizers based on system identification techniques,” Sustainable Energy, Grids and Networks, Volume 38, 2024, 101278, ISSN 2352-4677.

[9] L. Vanfretti and C.R. Laughman, “Power System Modeling for Identification and Control Applications using Modelica and OpenIPSL,” in Proc. of the 2024 8th IEEE Conference on Control Technology and Applications (CCTA), Newcastle upon Tyne, UK, August 21-23, 2024.

[10] L. Vanfretti and J. H. Chow, "Analysis of power system oscillations for developing synchrophasor data applications," 2010 IREP Symposium Bulk Power System Dynamics and Control - VIII (IREP), Rio de Janeiro, Brazil, 2010, pp. 1-17, doi: 10.1109/IREP.2010.5563289.

[11] L. Vanfretti and X. Bombois, “Power System Oscillation Monitoring and Damping Control Redesign under Ambient Conditions and Multiple Operating Points,” 20th IFAC Symposium on System Identification, July 17-19, 2024, Boston, MA, USA. IFAC-PapersOnLine, Volume 58, Issue 15, 2024, Pages 526-531, ISSN 2405-8963.

[12] Torre, Aharon & Cepeda, Jaime & Herrera, J.. (2013). Implementación de un sistema de monitoreo de área extendida WAMS en el Sistema Nacional Interconectado del Ecuador SNI. Ingenius. 10.17163/ings.n10.2013.04.

RPI

ALSET Lab