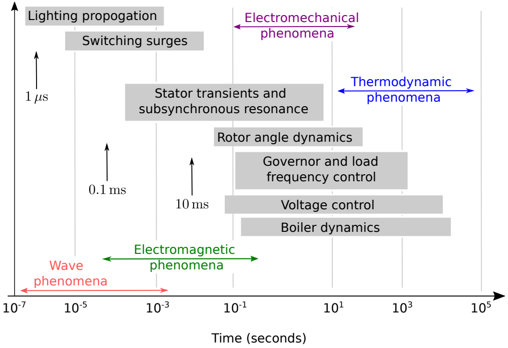
Investigating options to accelerate power system dynamic simulations

Duncan Callaway, with Matt Bossart, Gabriel Colon-Reyes, Rodrigo Henriquez, Ruth Kravis, Jose Daniel Lara, Bri-Mathias Hodge, Ciaran Roberts, Sunash Sharma

October 11th 2023 || Supported by DOE's Advanced Grid Modeling Program and Solar Energy Technologies Office (UNIFI)

Time scale coupling complicates dynamic simulation

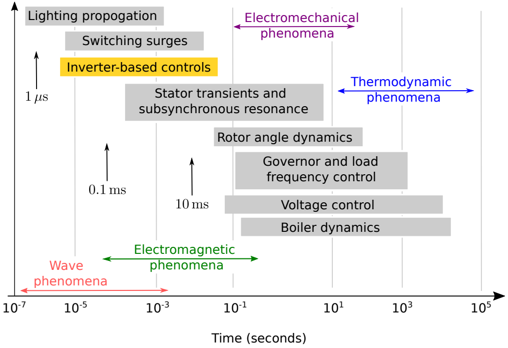
- Power system dynamic analysis is built on principles of timescale separation and singular perturbations
  - **Phasor Domain (PD) simulations**: fast solutions by modeling slower time scale dynamics only



(Hatziargyriou *et al*)

Time scale coupling complicates dynamic simulation

- Power system dynamic analysis is built on principles of timescale separation and singular perturbations
  - **Phasor Domain (PD) simulations**: fast solutions by modeling slower time scale dynamics only
- Active control on faster timescales $\longrightarrow$ timescale separation questioned
  - **EMT simulations**: model phenomena across all time scales.

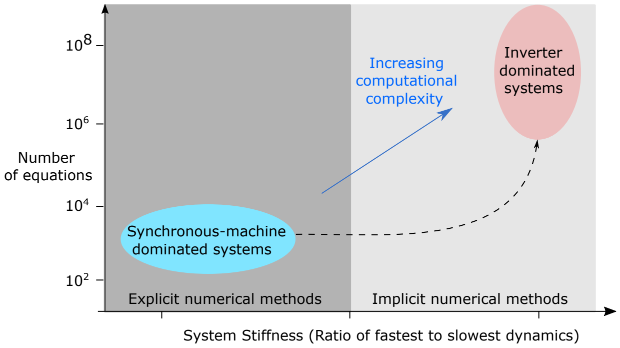


(Hatziargyriou *et al*)

Orders of magnitude complicate dynamic simulation

1. We are grappling with orders of magnitude more state variables
   - Can no longer **universally** neglect fast dynamics
   - Centralized synchronous units are replaced with 10s, or 100s, of inverter based resources
2. System stiffness increasing by orders of magnitude (dynamics span more time scales)

   $\longrightarrow$ Need implicit numerical solvers



Existing simulation tools struggle to solve the problem

- Current commercial tools do not accurately capture fast dynamics do not easily scale
  - **Accuracy**: PD unable to accurately model network dynamics or other electromagnetic scale phenomena
  - **Solution time**: A 30 second EMT simulation of Maui takes $\approx$ 5 hours to run on a 64 core computer
  - Note, some tools (e.g. DIgSILENT) can simulate portions of network as EMT. Still hard to do system-level interaction studies

$\rightarrow$ **Critical need for new tools to accurately and quickly model and simulate these complex systems**

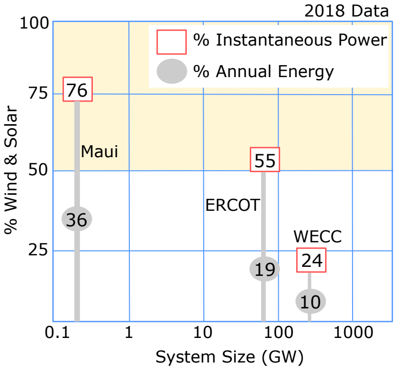
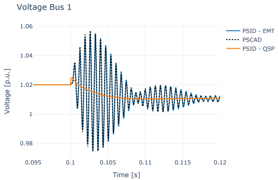


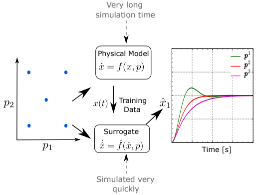
Figure 1: System scale, from [1]

Today's talk: Addressing these problems

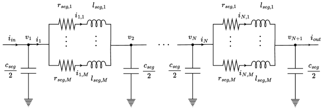
1. `PowerSimulationsDynamics.jl`
   - A software package to enable acceleration, research and development



2. Scientific machine learning
   - How much can we speed up simulation by replacing parts of a model with ML?
   - Is the result accurate?

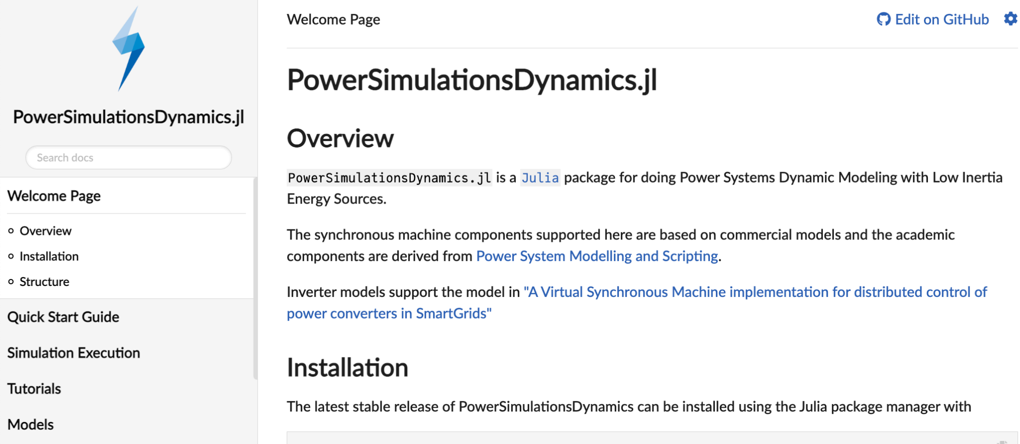


3. Examining line dynamics
   - How many states do we need to simulate?



Enabling software: PowerSimulationsDynamics.jl

- **PowerSimulationsDynamics.jl** (PSID) is a Julia-based software package that grew out of Scalable Integrated Infrastructure Planning project at NREL and further developed at Berkeley.


PowerSimulationsDynamics.jl

Search docs

Welcome Page
- Overview
- Installation
- Structure

Quick Start Guide

Simulation Execution

Tutorials

Models

Welcome Page

Edit on GitHub

PowerSimulationsDynamics.jl

Overview

`PowerSimulationsDynamics.jl` is a `Julia` package for doing Power Systems Dynamic Modeling with Low Inertia Energy Sources.

The synchronous machine components supported here are based on commercial models and the academic components are derived from Power System Modelling and Scripting.

Inverter models support the model in "A Virtual Synchronous Machine implementation for distributed control of power converters in SmartGrids"

Installation

The latest stable release of PowerSimulationsDynamics can be installed using the Julia package manager with

Enabling software: PowerSimulationsDynamics.jl

- **PowerSimulationsDynamics.jl** (PSID) is a Julia-based software package that grew out of Scalable Integrated Infrastructure Planning project at NREL and further developed at Berkeley.

- **Enables power system modeling / simulation innovation**
  - Balanced dq formulation; enables electromagnetic fidelity
  - Rapid prototyping of new converter models
  - Enables development and large-scale testing of state of the art numerical integration methods
  - Leverages Julia’s tremendous computing power
  - Supports surrogate integration, including training-to-simulation workflow
  - Open source: 600+ installs in last 12 mos; all our work gets pushed there.

Enabling software: PowerSimulationsDynamics.jl

- **PowerSimulationsDynamics.jl** (PSID) is a Julia-based software package that grew out of Scalable Integrated Infrastructure Planning project at NREL and further developed at Berkeley.
- **Enables power system modeling / simulation innovation**
  - Balanced dq formulation; enables electromagnetic fidelity
  - Rapid prototyping of new converter models
  - Enables development and large-scale testing of state of the art numerical integration methods
  - Leverages Julia's tremendous computing power
  - Supports surrogate integration, including training-to-simulation workflow
  - Open source: 600+ installs in last 12 mos; all our work gets pushed there.
- **Key recent outputs**
  - Large library of power system test cases for computational experiments, including 240 bus WECC
  - Integration with MIT's DifferentialEquations.jl libraries
  - Implement a range of speedup features such as automatic differentiation for Jacobian construction

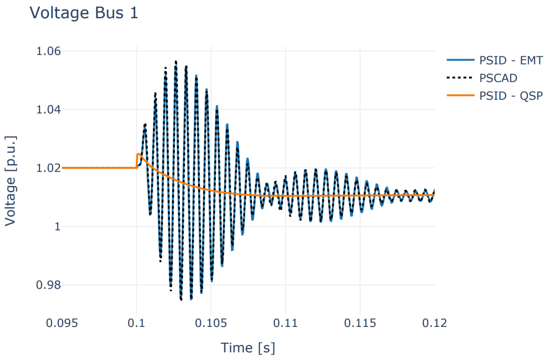
PSID is accurate vs. PSS/E and PSCAD

PD Validation in 240 Bus WECC Case	Generator Trip	Line Trip
Average Angle RMSE [deg]	3.172×10^{-2}	1.569×10^{-4}
Average Voltage RMSE [pu]	1.082×10^{-5}	7.831×10^{-7}
Average Speed RMSE [pu]	1.599×10^{-6}	5.225×10^{-7}

EMT Validation in 9 Bus WECC case	Line Trip
Average Voltage RMSE [pu]	2.31×10^{-2}
Average Speed RMSE [pu]	9.92×10^{-4}

- All the models to execute and validate the models are available in the repository: `https://github.com/Energy-MAC/AGM-Benchmarks`

PSID is 20x faster than PSCAD (w/ π line models)

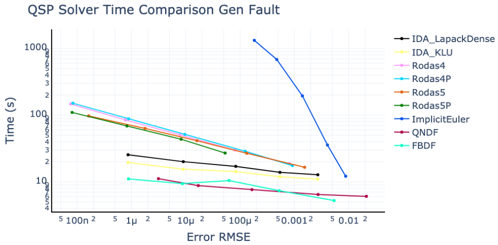


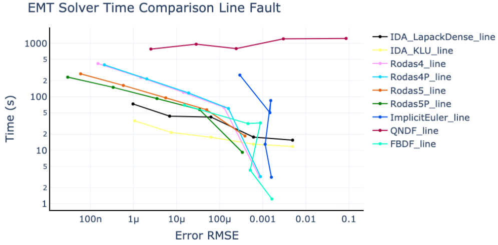
- Adaptive time-stepping captures fast dynamics and maintains significantly faster solution times
- PSID 10x-20x faster than EMT simulation considering GFM, GFL and synchronous machines devices for a 9-Bus test case.

PSCAD Init	PSCAD run	FBDF	IDA
39.72 s	57.10 s	8.61 s	3.62 s

- In the table, both simulations use same dedicated hardware. PSCAD used $\Delta t = 0.005$ and PSID solver used adaptive time stepping with atol $= 10^{-9}$ and rtol $= 10^{-9}$.

PSID enables a diversity of solvers





- Top plot: PD inverter and machine models.
- Bottom plot: EMT cases are significantly stiffer than the PD case; changes the “fastest” solver.
- The choice for the ”best” solver depends on the application and the type of contingency being studied.
- These results show that there are avenues to reduce time per step for solution methods that require fewer steps and also increasing the accuracy of implicit methods

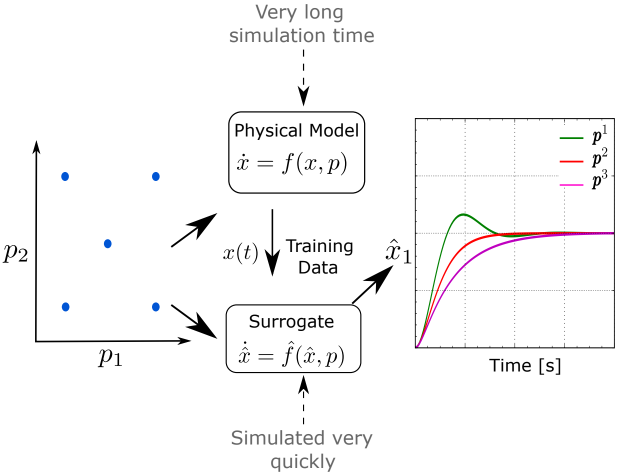
Can we further accelerate dynamic simulations?

- Surrogates are a computationally cheap accurate approximation of the true system trained using data-driven methods
- **Two basic approaches**
  1. Learn the solution

$$x(t) \approx \hat{F}(t, p)$$

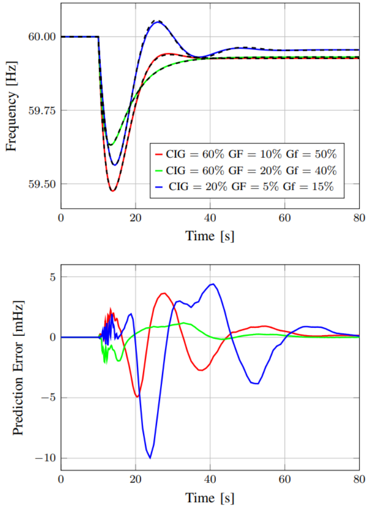
  2. Learn the dynamics

$$\dot{x} \approx \hat{f}(x, p)$$



Learning the solution is accurate

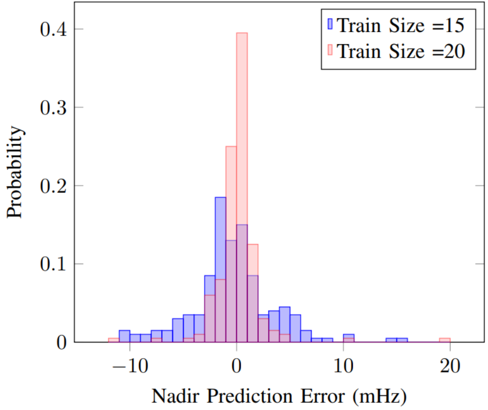
- Example use case: frequency dynamics, following the loss of generation, with grid-following (Gf) or grid-forming (GF) converters
- Continuous time echo state networks (CTESNs):
  - Developed with MIT partners and adapted for power systems at LBL, learn how the solution changes as a function of system parameters
  - Core idea: recurrent neural network trainable on adaptively-stepped simulation data



Learning the solution enables significant acceleration

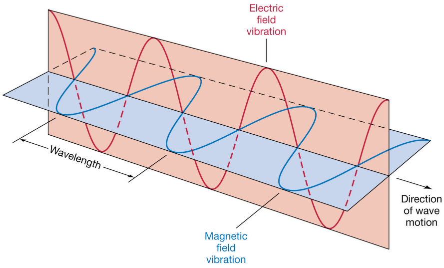
System Size	CTESN	Full model	accel
18 Bus	0.312 s	0.849 s	2.71x
36 Bus	0.797 s	4.37 s	5.47x
72 Bus	1.492 s	29.19 s	19.56x
144 Bus	2.9 s	109.83 s	37.83x

Implementation and acceleration benchmark systems publicly available at `https://github.com/Energy-MAC/CTESN_PSCC`

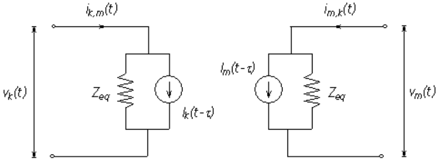


What's the role of transmission line dynamics?

- Phasor domain models: couple buses via algebraic line models that ignore the EM dynamics.
- EMT: captures the detailed physics of TLs, including wave propagation and frequency dependence.
  - EM time scale phenomena can accurately propagate from one network location to another.
- If we need line dynamics to study IBR interactions at high penetrations, we need much more computing power
- In the following, we'll examine both small signal and transient dynamics

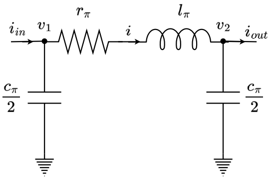


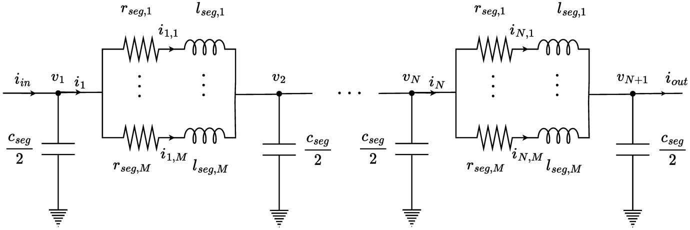
(https://physics.stackexchange.com/)



(PSCAD)

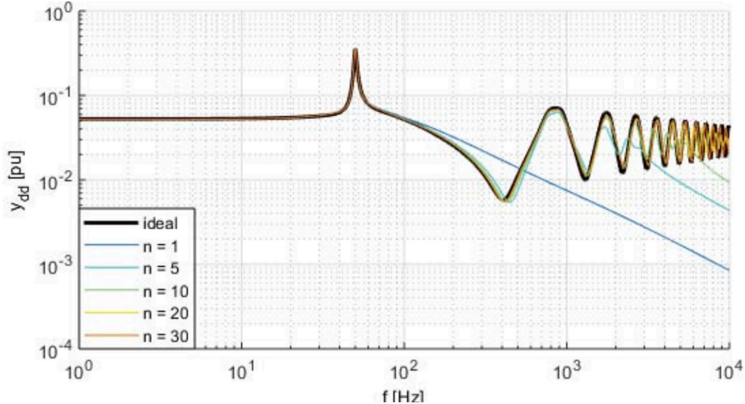
Examining the role of line dynamics





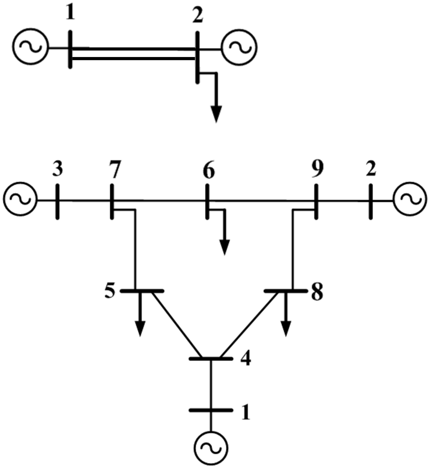
- Multi-segment multi-branch model adapted from D'Arco 2015 (originally developed for dc cables)
- Advantage of these line model forms:
  - Enables dq representation, integration with PSID $\rightarrow$ full resolution of (balanced) EMT
  - Facilitates comparison of dynamics, eigenvalues, one assumption at a time
  - Can achieve any desired resolution (wave propagation, frequency dependence) by adding segments and branches
- Disadvantage:
  - Increases number of states, simulation time

Multi-segment multi-branch frequency response



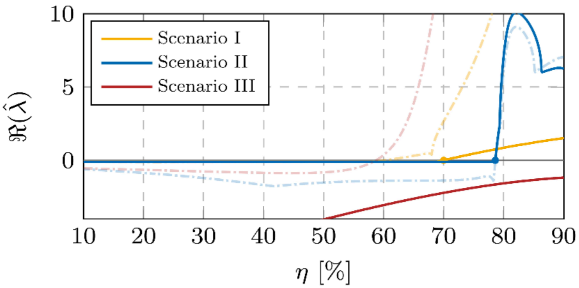
- Frequency response in domain of inner control loops depends strongly on line model
- (Figure from D'arco *et al* 2016)

Simulation cases



- We'll focus on interactions between GFM and synchronous machines
- Scenarios investigated:
  - Line length
  - Loading
  - For small signal: location of GFM, SM
  - For time domain sim: Line trip (line 4-5 in 9 bus)

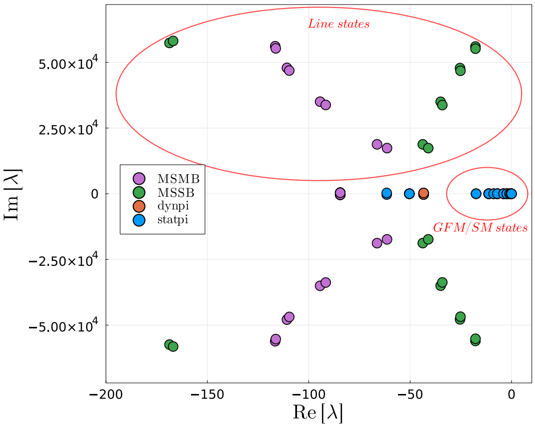
Small signal setup: Markovic *et al* two bus model



- "Scenario II" has GFM at one bus, SM at other
- η is the fraction of generation from GFM
- Solid line is largest eigenvalue using RL branch
- Dashed line is largest eigenvalue for static line model

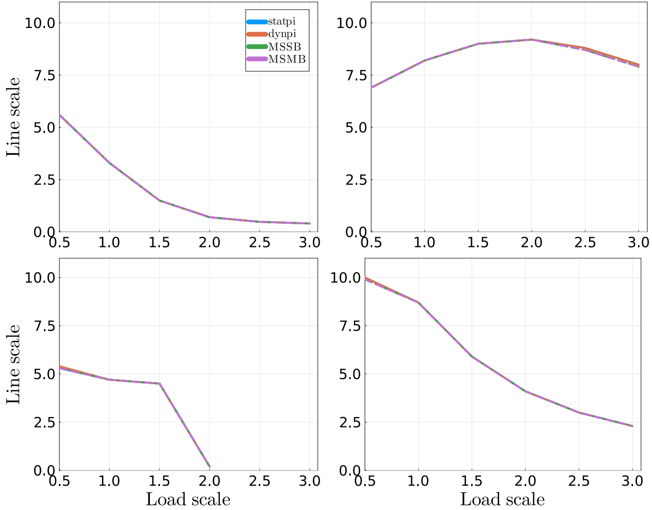
- Results show that adding line dynamics with GFM has limited impact on small signal stability
- Open question: does this hold for higher fidelity line models?

Small signal results, two bus (GFM v. SM)



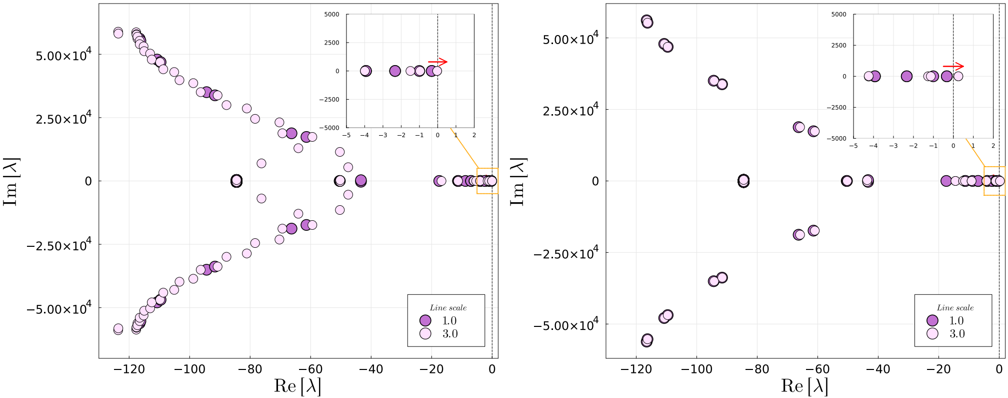
- System eigenvalues under different line models, for a GFM v SM test case, with the load at the GFM bus, and *load scale* $= 1.0$, *line scale* $= 1.0$.
- Line dynamics are fast (many kHz) with very negative real part.
- Least stable modes have very low participation factors from line states.
- Reminder: This is the GFM case; GFL case, other configurations differ.
- **Takeaway**: in this case, line state modes are far from imaginary axis

Small signal results: More two bus cases



- Line length where the system loses stability as a function of system loading
  - Line scale = 1 $\longrightarrow$ 100km line
  - Load scale = 1 $\longrightarrow$ Load = SIL ($\sim$ 100MW)
- Top row is GFM vs SM, load at the GFM bus (left), load at the SM bus (right).
- Bottom row is SM v SM (left) and GFM v GFM (right)
- **Takeway**: Line models don't influence small signal results across wide range of cases

Small signal results: Two bus eigenvalue movement



MSMB eigenvalues under a sweep of (a) line lengths (with *load scale* $= 1.0$) and (b) loading (with *line scale* $= 1.0$).

Dynamics setup: High inverter-based infeed

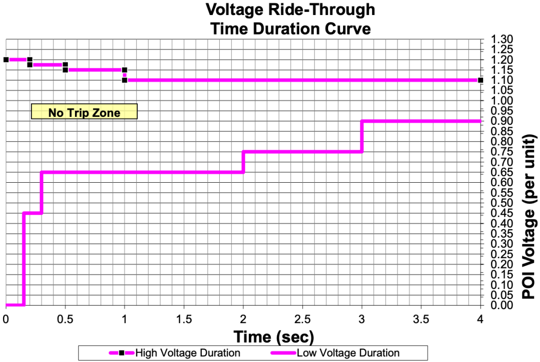
NERC
NORTH AMERICAN ELECTRIC RELIABILITY CORPORATION

2022 California Battery Energy Storage System Disturbances

California Events: March 9 and April 6, 2022
Joint NERC and WECC Staff Report

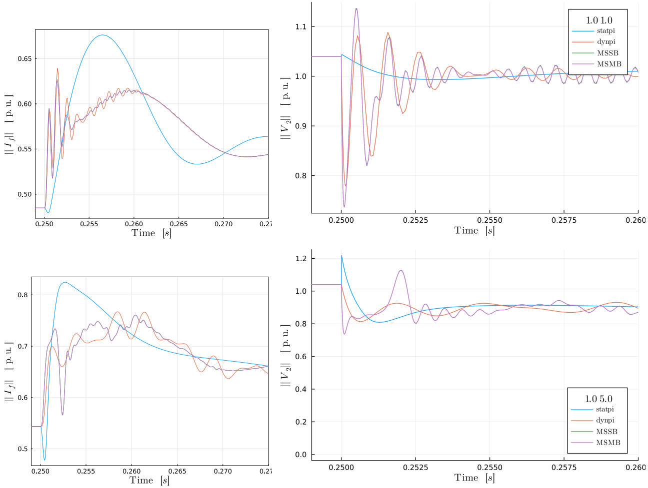
September 2023

“Poor commissioning practices are a significant contributor to the unreliable performance of IBRs… BESS ride-through performance is not adequately assessed during the interconnection process.”



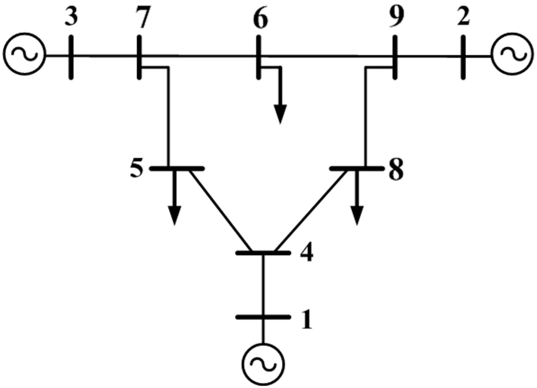
(from NERC PRC-024-2)

Dynamic simulations: two bus branch trip



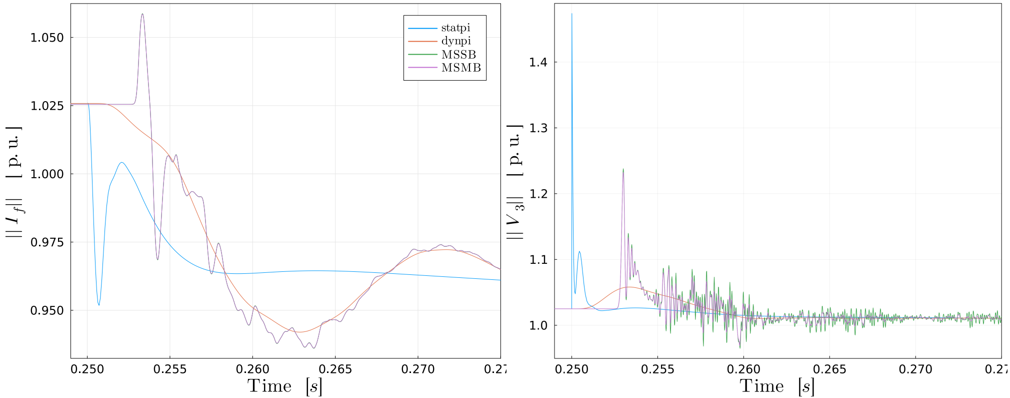
- Top row: 100 km line. Bottom row: 500 km line.
- Takeaways:
  - Larger differences for *short line*.
  - Differences dissipate quickly
  - Frequencies not high enough for MSMB / MSSB to differ
  - Dynamic models differ wrt overcurrent, voltage tripping
  - Note modeling overcurrent saturation on inverter
  - SM-SM dynamics similar – but consequences are smaller

9 bus branch trip



- GFM at Bus 1, 3; SM at bus 2
- Trip branch 4-5
- Measure states as Bus 3

Dynamic simulations: 9 bus branch trip



- Bus 3 inverter filter current for *line scale* $= 3.0$, and *load scale* $= 1.0$.
- Key takeaway: higher fidelity line produces filter current spike missed by other models
- This effect is more pronounced with *longer lines*

Line dynamics takeaways

Small signal:

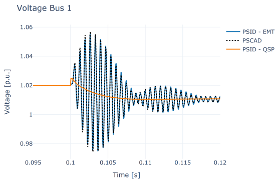
- Each line model generates different sets of eigenvalues, but
- line states have very low participation factors in the least stable eigenvalues
  - consistent with and expand on earlier work
- Future work:
  - Different inverter controls
  - Different gain parameters

Dynamic simulations

- Dynamic interaction studies show similar dynamics in many cases
- But! In some cases high filter currents and bus voltages may be missed by simpler line models
- Frequency dependent line models do not have an impact on results
- Future work:
  - Different line parameters
  - Bigger sweep of loading, line lengths
  - Different types of faults

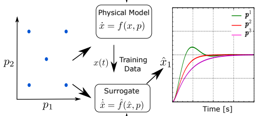
Today's talk: Tackling the problem

1. `PowerSimulationsDynamics.jl`
   - A software package to enable acceleration, research and development



Very long simulation time

2. Scientific machine learning
   - How much can we speed up simulation by replacing parts of a model with ML?
   - Is the result accurate?



Simulated very quickly

3. Examining line dynamics
   - How many states do we need to simulate?