OpenDSS Public Webinar 2023

EPRI
Distribution Operations and Planning

27 September 2023

Agenda

- OpenDSS - 15 Years of Shaping the Future of Energy
- Overview of Available Training Materials
- Update and Demonstration on New Features and Functionality
- DSS User Applications and Presentations

Housekeeping

You'll get all resources (presentations and scripts) in the OpenDSS repository.

You'll know when the video recordings are available via email.

We'll post it in the SourceForge discussion as well.

You'll earn Professional Development Hours (PDH).

You've have required it during the registration.

Any questions, please contact Arin Nichols (anichols@epri.com).

You can interact with us during the meeting by using the Webex Q&A feature.

We'll try to answer as many questions as possible.

OpenDSS - 15 Years of Shaping the Future of Energy

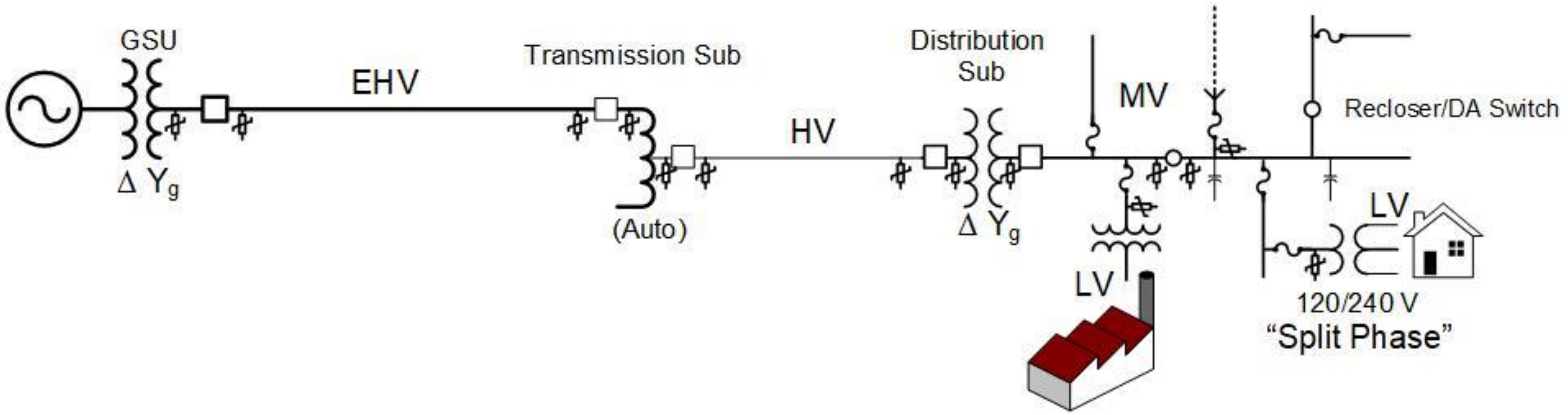
What is the OpenDSS?

- Script-driven, frequency-domain electrical circuit simulation tool
  - Script with text commands
  - Script with another program (Python, MATLAB, C++, Delphi, etc.)
    - Use Windows COM interface
    - Use Direct DLL interface
    - Drive with OpenDSS-G

- Specific models for:
  - Supporting **utility distribution system** analysis
  - Designed for the unbalanced, multi-phase North American power distribution systems
    - Can model European-style systems also
      - These typically have a simpler structure

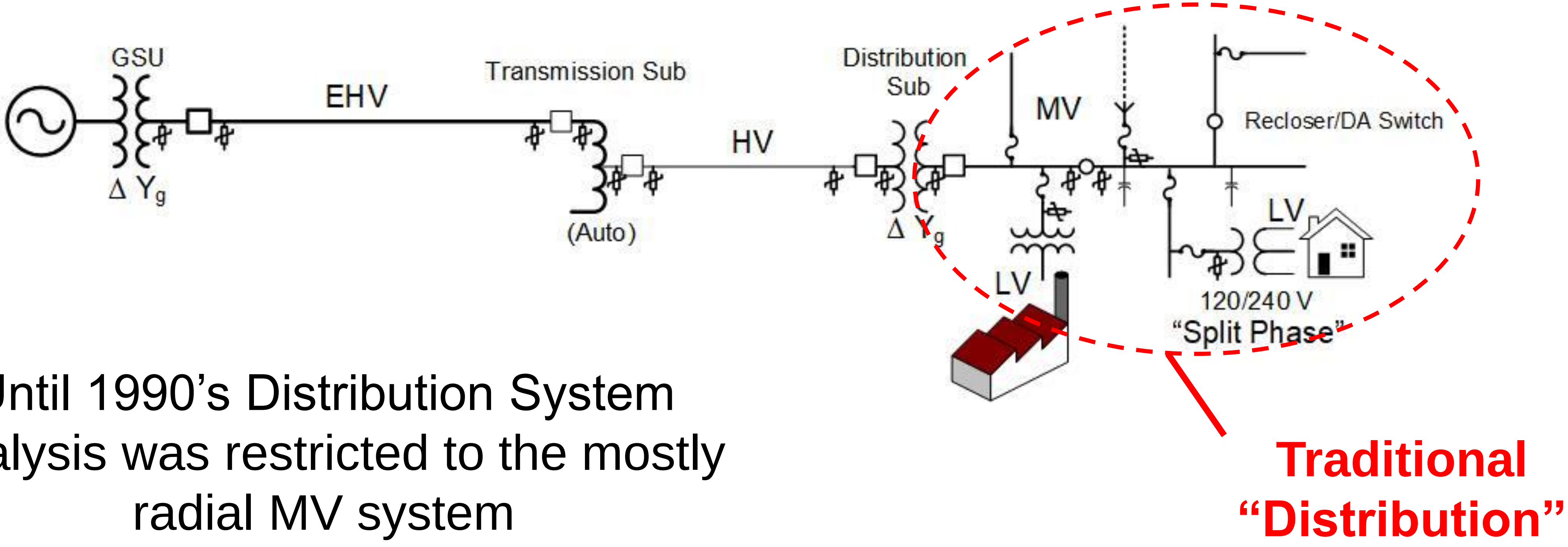
What Problem was OpenDSS Designed to Solve?

[One-Line Diagram of the Power Delivery System from Generator to Load]



What Problem was OpenDSS Designed to Solve?

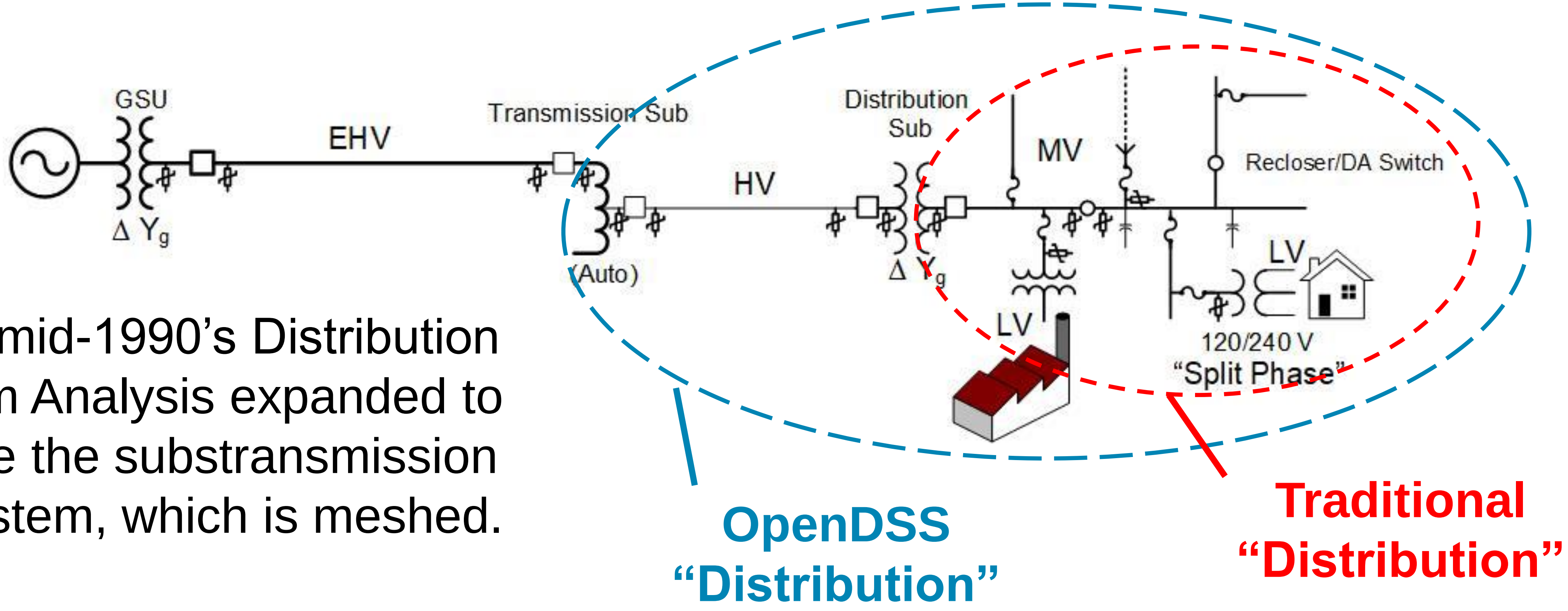
[One-Line Diagram of the Power Delivery System from Generator to Load]



Until 1990's Distribution System Analysis was restricted to the mostly radial MV system

What Problem was OpenDSS Designed to Solve?

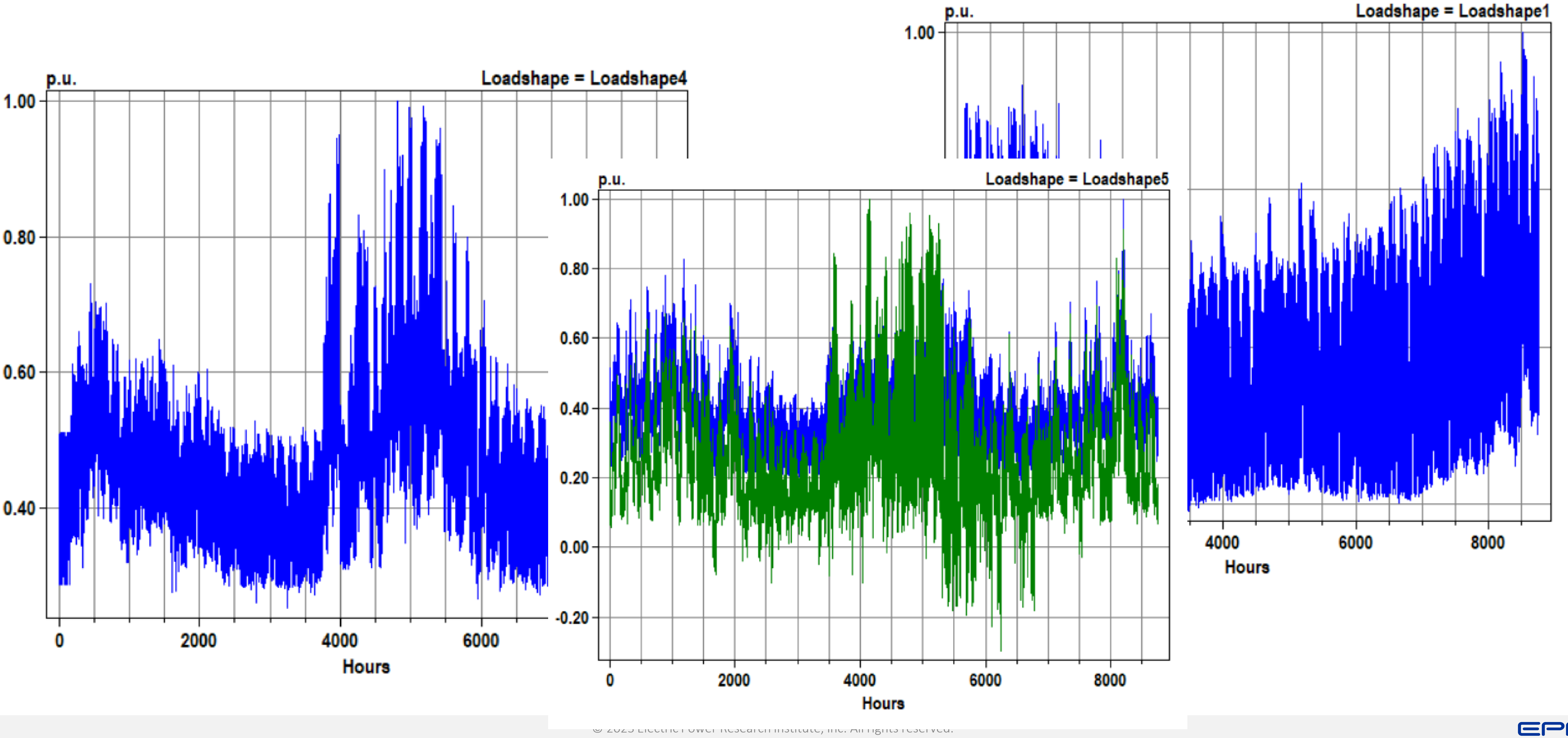
[One-Line Diagram of the Power Delivery System from Generator to Load]



In the mid-1990's Distribution System Analysis expanded to include the substransmission HV system, which is meshed.

DER was one of the key drivers

Then the load started to move …



OpenDSS Development

- Heritage
  - **Harmonics solvers** rather than **power flow**
    - Gives OpenDSS extraordinary distribution system modeling capability
  - Simpler to solve power flow problem with a harmonics solver than vice-versa
  - More like EMT or Dynamics model than typical Distribution Power Flow

- Supports all **rms steady-state** (i.e., frequency domain) analyses commonly performed for utility distribution system planning
  - And many other types of analyses
  - Original purpose: **DG interconnection analysis**

OpenDSS Development

- Development began in 1997 at Electrotek Concepts, Inc in Knoxville, Tennessee
- Key Authors
  - Roger Dugan
  - Tom McDermott
- Acquired by EPRI in 2004
- Made Open Source in September 2008 to speed the development of the "Smart Grid"
- Well over 165 000 downloads from [Sourceforge]

Overview of Available Training Materials

Virtual Trainings

2020

- **Session 1:** Distribution system basics, Intro to OpenDSS, OpenDSS basics and scripting
- **Session 2:** Introduction to OpenDSS-G, APIs, COM interface
- **Session 3:** PVSystem, InvControl, Storage, Storagecontroller
- **Session 4:** Advanced topics & Applying DSS in R&D

2021

- **Session 1:** Distribution system basics, OpenDSS basics and scripting
- **Session 2:** Introduction to OpenDSS-G, APIs, COM interface
- **Session 3:** Controls in OpenDSS and PVSystem + InvControl
- **Session 4:** Advanced topics & Applying DSS in R&D

2022

- **Session 1:** Distribution system basics, OpenDSS basics and scripting
- **Session 2:** Introduction to OpenDSS-G, New Functionality in DSS, Advanced Topics
- **Session 3:** Extending Capabilities of OpenDSS via a Programing Language
- **Session 4:** Applying DSS in R&D

YouTube Resources

EPRI OpenDSS	OpenDSS-G	Paulo Radatz
• This tutorial focuses on the modeling of the primary elements of OpenDSS, as well as covering some of its plotting capabilities.	• This channel provides instructions on how to use OpenDSS-G and some of its applications.	• This playlist focuses on how to simulate snapshot and QSTS modes, including an understanding of the main OpenDSS reports. • This playlist covers the basics of controlling OpenDSS using Python.

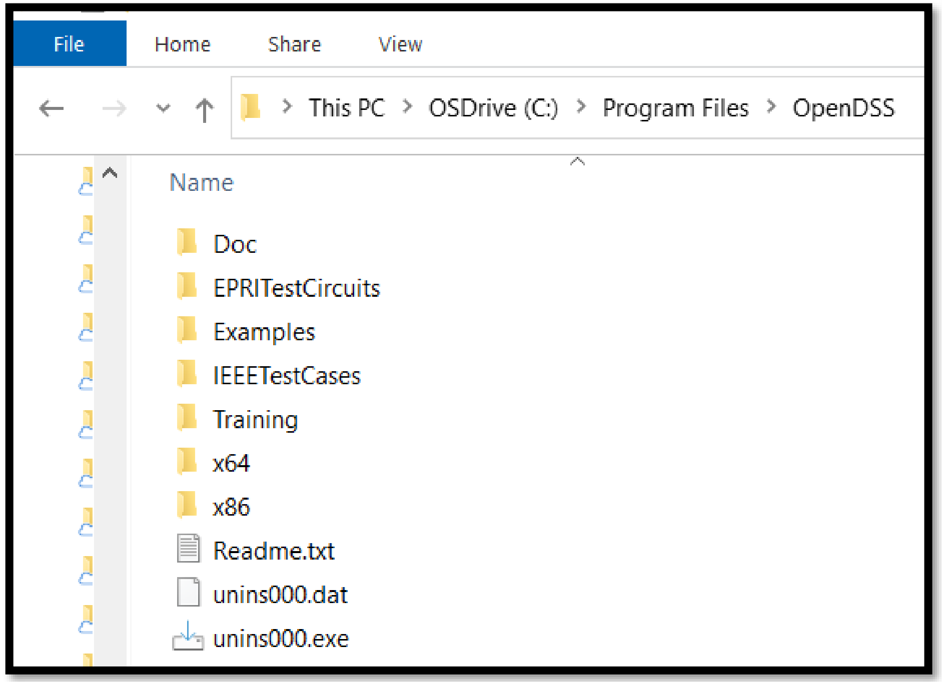
Grid Hosting Capacity Public Webinar

Hosting Capacity Webinar recording:

OpenDSS files and Python scripts provided in the shared folder:

OpenDSS Local Folder Resources

- *C:\Program Files\OpenDSS*



Doc - Documentation, Questions, and TechNotes. The OpenDSSPrimer.pdf is a good resource for getting an overview of OpenDSS, while the OpenDSSManual.pdf is useful for addressing any questions.

EPRITestCircuits - Three models of actual electric power distribution circuits.

Examples - This section provides examples of different aspects of OpenDSS, including modeling, simulation, COM interface, and user examples, among others.

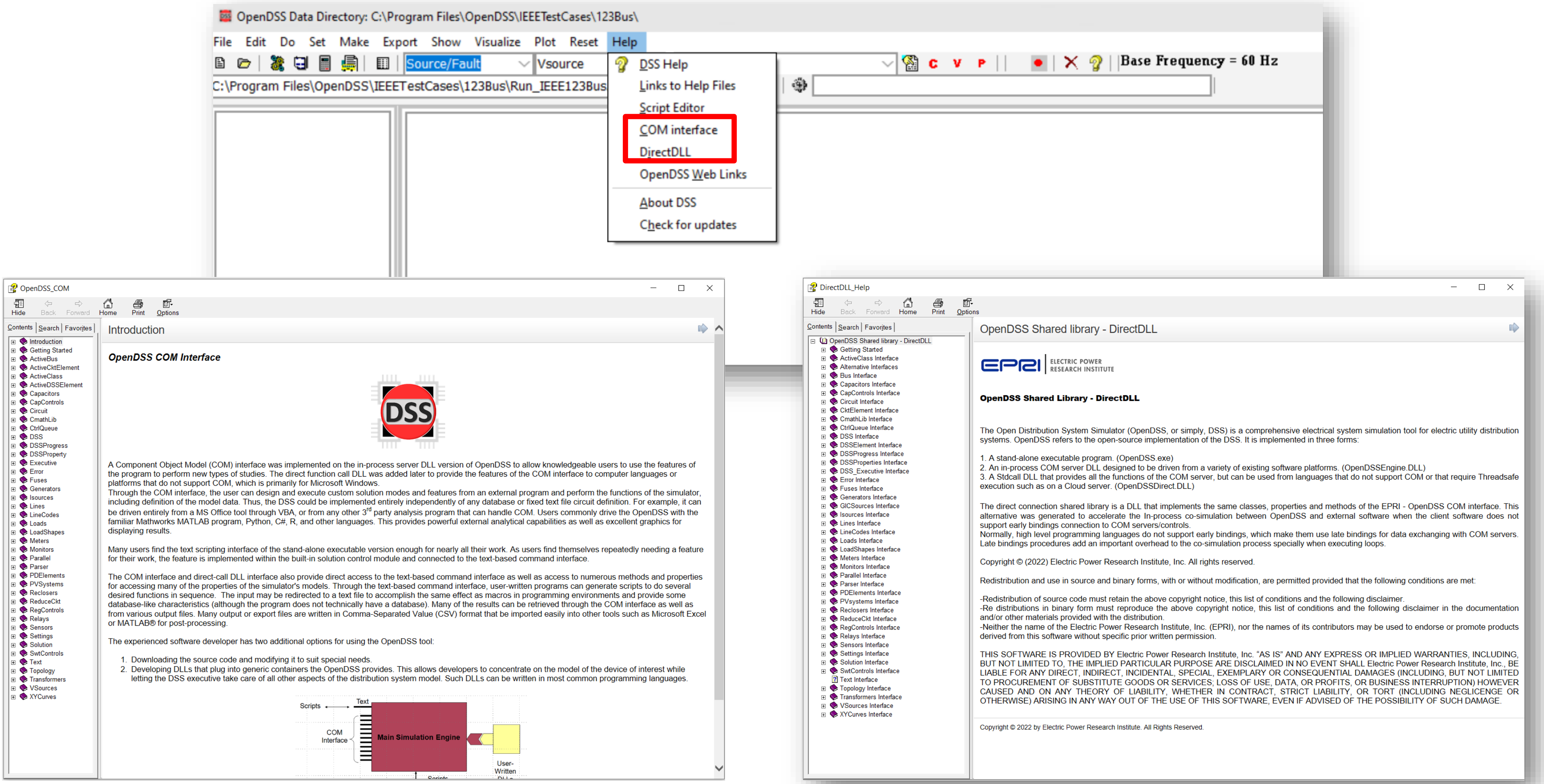
IEEETestCases - Contains IEEE test cases.

Training - For additional resources, please visit this link

x64 and x86 - Contains the different versions of OpenDSS:

- Standalone version (OpenDSS.exe)
- COM interface version (OpenDSSengine.dll)
- Direct DLL version (OpenDSSDirect.dll)
- Command prompt version (OpenDSScmd.exe)

COM Interface and Direct DLL OpenDSS Documentations

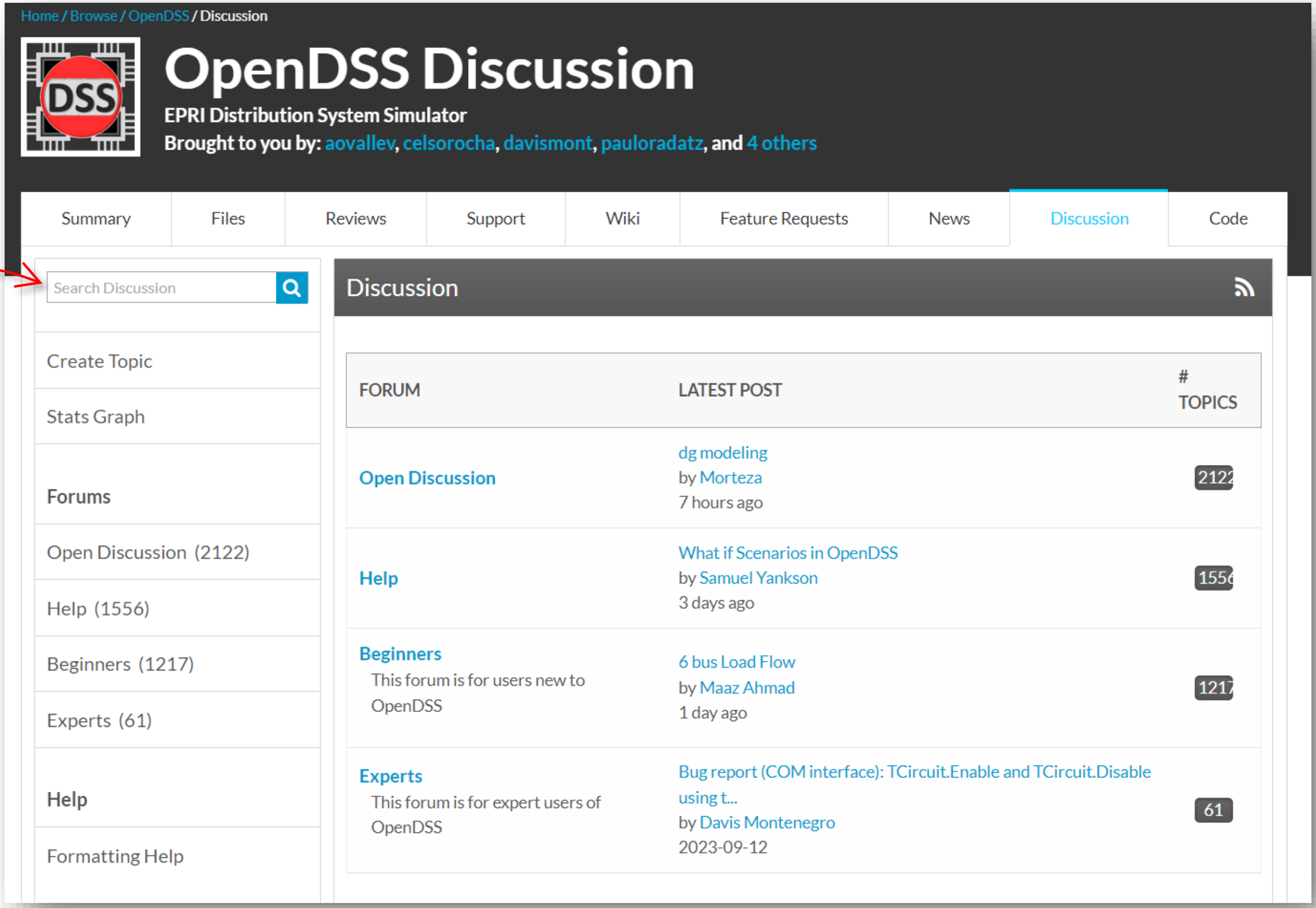


Getting Help with OpenDSS

Our community forum is the ideal platform for posting your OpenDSS questions, regardless of whether you're a beginner or an advanced user.

"Search Discussion" Feature

- This tool allows you to search for preexisting discussions related to your topic. You may find that your question has already been answered, or you can gain valuable insights from the existing content.
- By exploring these discussions, you can save time and improve your understanding of OpenDSS and related subjects.



Update and Demonstration on New Features and Functionality

Dynamic expressions

What are dynamic expressions

There is a current interest in the industry for integrating dynamic studies into large scale models for different applications.

OpenDSS has such functionality, however, the dynamic behavior of PCE such as generators is scripted within the model code, suggesting that for representing a different dynamic (swinging equation), the user needs to modify the source code for the element or create a new one.

Dynamic expressions is an object type in OpenDSS for entering a customizable swinging equation describing a dynamic behavior that can be adopted by other elements in the model

Dynamic expressions

$$\dot{V} = -\frac{D}{M}V + \frac{Pshaft - Pterm}{M}$$
$$\dot{\theta} = V$$

```
ClearAll

// Set the base frequency to 60Hz
Set DefaultBaseFrequency=60

Var @Zbase=53.615

// quasi-ideal source for inf. bus at SourceBus with initial conditions Vpu=0.90081 and Vangle = 0
New Circuit.SimpleDemo
~ BasekV=345 pu=0.90081 phases=3
~ Angle = 0.0 Model=ideal puZideal=[ 1.0e-7, 0.00001]  BaseMVA=2220

// New parallel lines from SourceBus to high side of transformer
New Line.Source_HT_1 Bus1=SourceBus Bus2=HT  R1=0 X1=(0.5 @Zbase *) R0=0 X0=(0.5 @Zbase *) C1=0
C0=0 length=1 Units=mi
New Line.Source_HT_2 Bus1=SourceBus Bus2=HT  R1=0 X1=(0.93 @Zbase *) R0=0 X0=(0.93 @Zbase *)
C1=0 C0=0 length=1 Units=mi

// New Transformer with reactance j0.15
New Transformer.Step_Up Phases=3 Windings=2 XHL=15 ppm=0
~ buses=(HT LT) conns='wye wye' kvs="345 24" kvas="2220000 2220000" %Loadloss=0

// Constant kW at specified power factor
New Generator.G1 Bus1=LT kV=24 kW=(2220000 0.9 *) kvar=(2220000 0.436 *)   Model=1 vminpu= 0.80
Vmaxpu=1.4  MVA=2220  XRdp=1e12  Xdp=0.3  Xdpp=0.25 H=3.5 D=0

// Set all voltage bases in model
```

Dynamic expressions

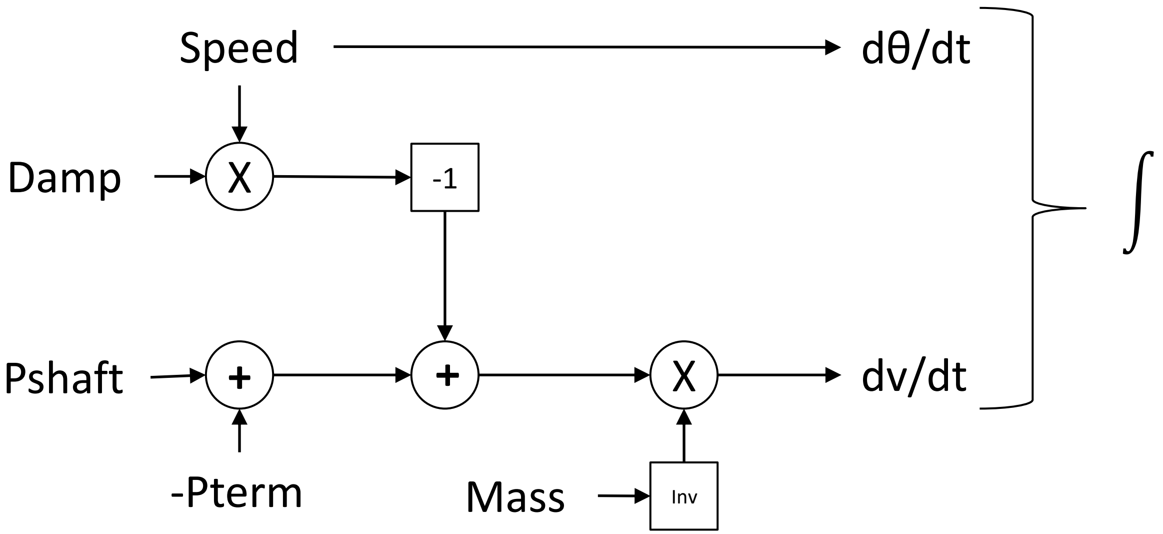
$$\dot{V} = -\frac{D}{M}V + \frac{Pshaft - Pterm}{M}$$
$$\dot{\theta} = V$$

```
ClearAll

// Set the base frequency to 60Hz
Set DefaultBaseFrequency=60
Var @Zbase=53.615
// quasi-ideal source for inf. bus at SourceBus with initial conditions
Vpu=0.90081 and Vangle = 0
New Circuit.SimpleDemo
~ BasekV=345 pu=0.90081 phases=3
~ Angle = 0.0 Model=ideal puZideal=[ 1.0e-7, 0.00001]  BaseMVA=2220

// New parallel lines from SourceBus to high side of transformer
New Line.Source_HT_1 Bus1=SourceBus Bus2=HT  R1=0 X1=(0.5 @Zbase *) R0=0
X0=(0.5 @Zbase *) C1=0 C0=0 length=1 Units=mi
New Line.Source_HT_2 Bus1=SourceBus Bus2=HT  R1=0 X1=(0.93 @Zbase *)
R0=0 X0=(0.93 @Zbase *) C1=0 C0=0 length=1 Units=mi

// New Transformer with reactance j0.15
New Transformer.Step_Up Phases=3 Windings=2 XHL=15 ppm=0
~ buses=(HT LT) conns='wye wye' kvs="345 24" kvas="2220000 2220000"
%Loadloss=0

// Creates the diffrential equation for the generator's dynamics
New DynamicExp.myDiffEq nvariables=6 varnames=[Speed Mass PShaft Pterm
Damp theta]
~ expression=[Speed dt = 1 Mass / (Pshaft Pterm - Damp Speed * -) *;
theta dt = Speed]

New Generator.G1 Bus1=LT kV=24 kW=(2220000 0.9 *) kvar=(2220000 0.436 *)
Model=1 vminpu= 0.80  Vmaxpu=1.4 DynamicEq=myDiffEq MVA=2220  XRdp=1e12
Xdp=0.3  Xdpp=0.25
// Initializes the dynamic equation's state variables
~ Damp = 0 PShaft = P0 Pterm = P Speed = 0 theta = Edp
~ Mass = (3.5 2 * 2220000000 376.99112 / *)
~ DynOut = [Speed theta]…
```

Dynamic expressions

$$\dot{V} = -\frac{D}{M}V + \frac{Pshaft - Pterm}{M}$$
$$\dot{\theta} = V$$

[Speed dt = 1 Mass / (Pshaft Pterm - Damp Speed * -) *;
theta dt = Speed]



Dynamic expressions

"[v dt = 1 M / (Pshaft Pterm - D v * -) *; theta dt = v]"

Load var 1

Load var 2

Operate

Load var 3

Operate

Load var 4

Load var 5

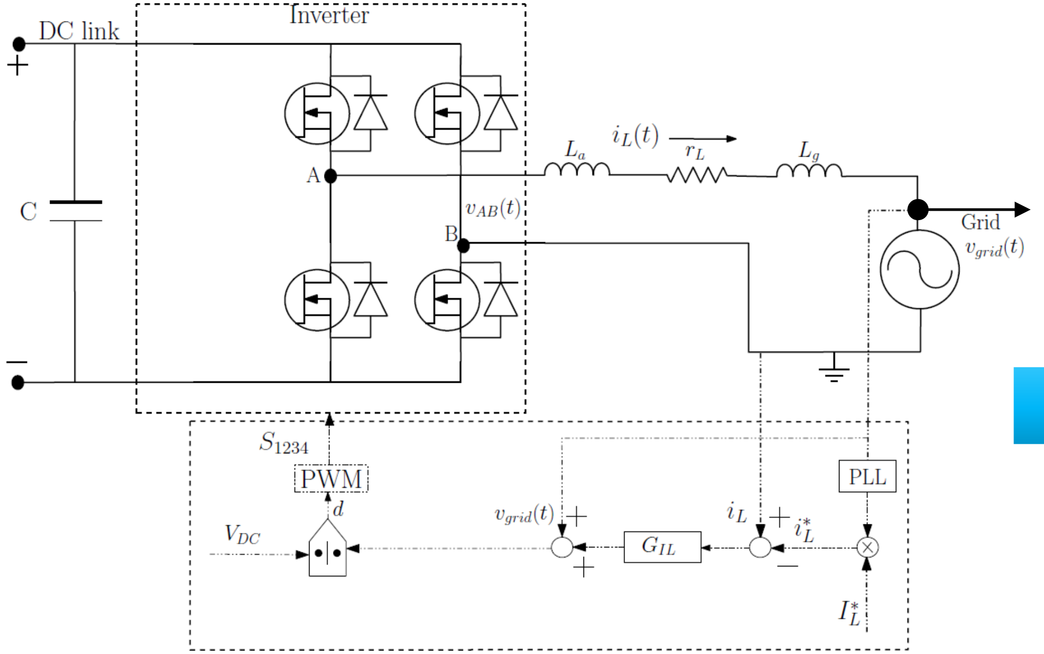
Operate

…

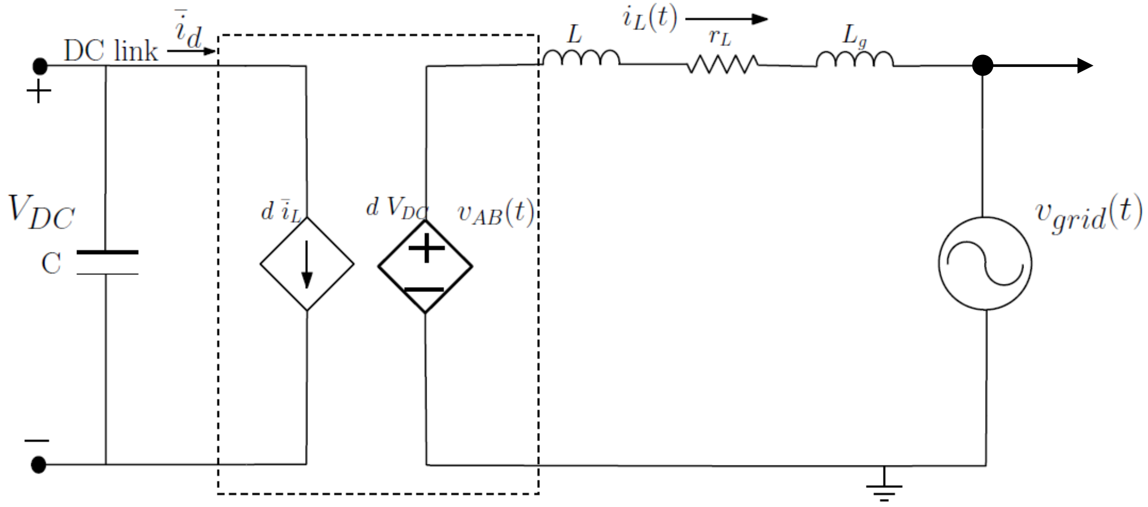
Load var 1
Load var 2
Load var 3
Load var 4
Load var 5
…

Adopting obj
“Generator”

Dynamic expressions



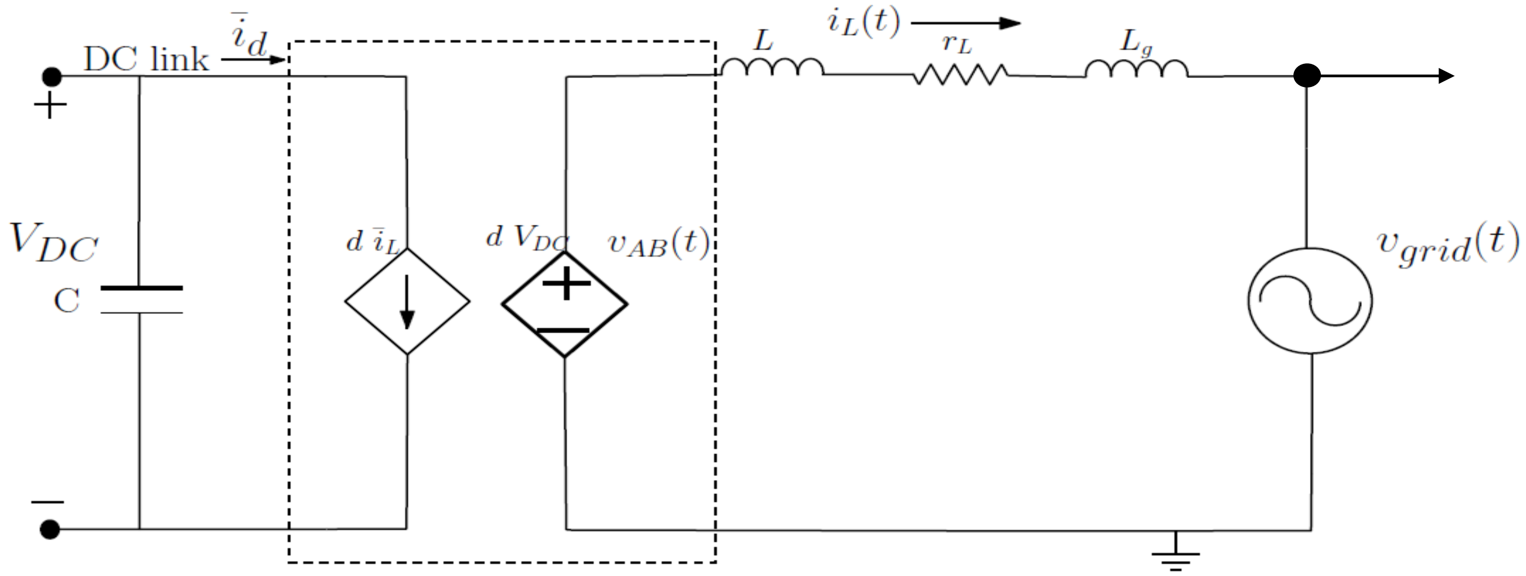
Detailed model



Approximated average model in dynamic phasor domain

Dynamic expressions

Approximated average model in dynamic phasor domain



$$\frac{d\langle i_L\rangle_1^R}{dt} = \omega\ \langle i_L\rangle_1^I + \frac{1}{L}\left(-r_L\ \langle i_L\rangle_1^R - \langle x_1\rangle_1^R\ V_{DC}\right)$$

$$\frac{d\langle i_L\rangle_1^I}{dt} = -\ \omega\ \langle i_L\rangle_1^R + \frac{1}{L}\left(-r_L\ \langle i_L\rangle_1^I - \frac{\langle V_{gm}\rangle_1^I}{2} - \langle x_1\rangle_1^I\ V_{DC}\right)$$

Dynamic expressions

$$\frac{d\langle i_L\rangle_1^R}{dt} = \omega\ \langle i_L\rangle_1^I + \frac{1}{L}\ \left(-r_L\ \langle i_L\rangle_1^R\ -\ \langle x_1\rangle_1^R\ V_{DC}\right)$$

$$\frac{d\langle i_L\rangle_1^I}{dt} = -\ \omega\ \langle i_L\rangle_1^R + \frac{1}{L}\ \left(-r_L\ \langle i_L\rangle_1^I\ -\ \frac{\langle V_{gm}\rangle_1^I}{2} - \langle x_1\rangle_1^I\ V_{DC}\right)$$

Approximated average model in dynamic phasor domain

```
! First, dynamics expressions for defining the inverter behavior
New DynamicExp.myDiffEq nvariables=4 varnames=[it vdc modul vac]
~ expression=[it dt = 1 0.61059E-3 / ( -0.230187 it * modul vdc * + vac - ) *]

New DynamicExp.myDiffEq2 nvariables=4 varnames=[it vdc modul vac]
~ expression=[it dt = 1 0.50882E-3 / ( -0.1918225 it * modul vdc * + vac - ) *]

New "Storage.mystorage" phases=3 conn=delta bus1=StoBus kV=0.48 kva=800 kWrated=800 kWhrated=6000
%stored=100 %reserve=20
~ %EffCharge=90 %EffDischarge=90 %IdlingkW=1 %R=50 %X=50 State=IDLING kP=0.01 KVDC=0.700 PITol=0.1
SafeVoltage=0
~ DynamicEq=myDiffEq2
~ it = imag vdc = kvdc modul=mod vac = vmag
~ DynOut = [it]

New "PVSystem.myPV" phases=3 conn=delta bus1=PVBus kV=0.48 kva=800 pmpp=800 daily=pvshape %R=50 %X=50
kP=0.01 KVDC=0.700 PITol=0.1 SafeVoltage=0
~ DynamicEq=myDiffEq
~ it = imag vdc = kvdc modul=mod vac = vmag
~ DynOut = [it]
```

Dynamic expressions

Examples:

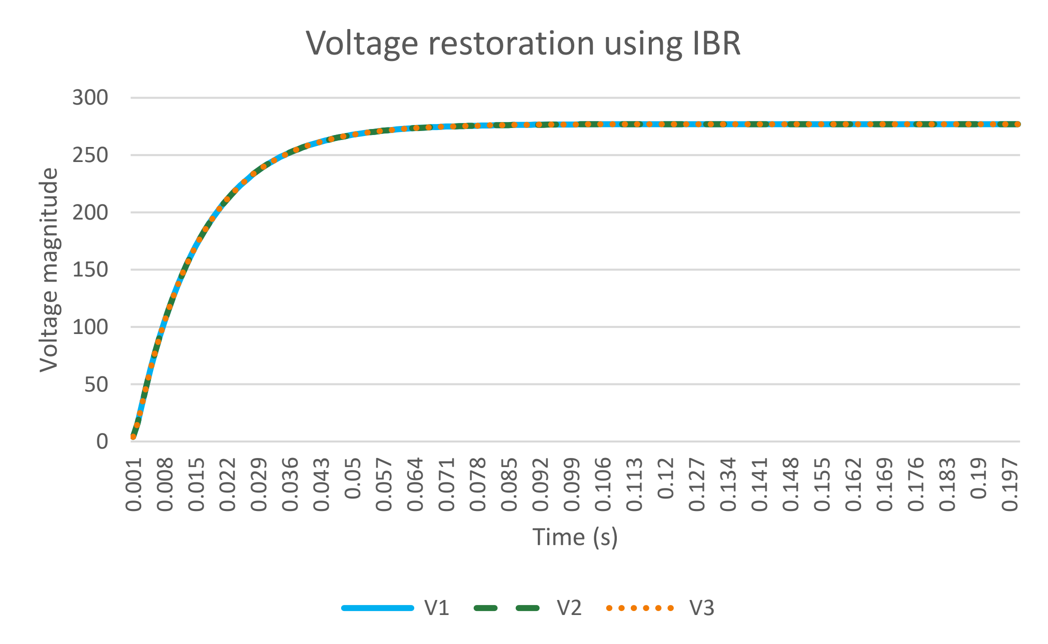
https://sourceforge.net/p/electricdss/code/HEAD/tree/trunk/Version8/Distrib/Examples/Dynamic_Expressions/

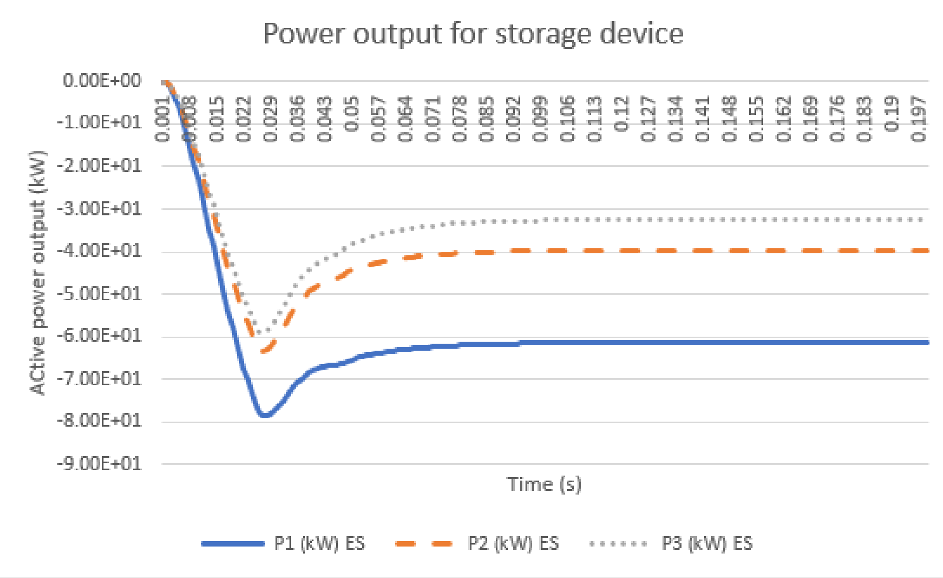
https://sourceforge.net/p/electricdss/code/HEAD/tree/trunk/Version8/Distrib/Examples/IBRDynamics_Cases/GFL_IEEE123/

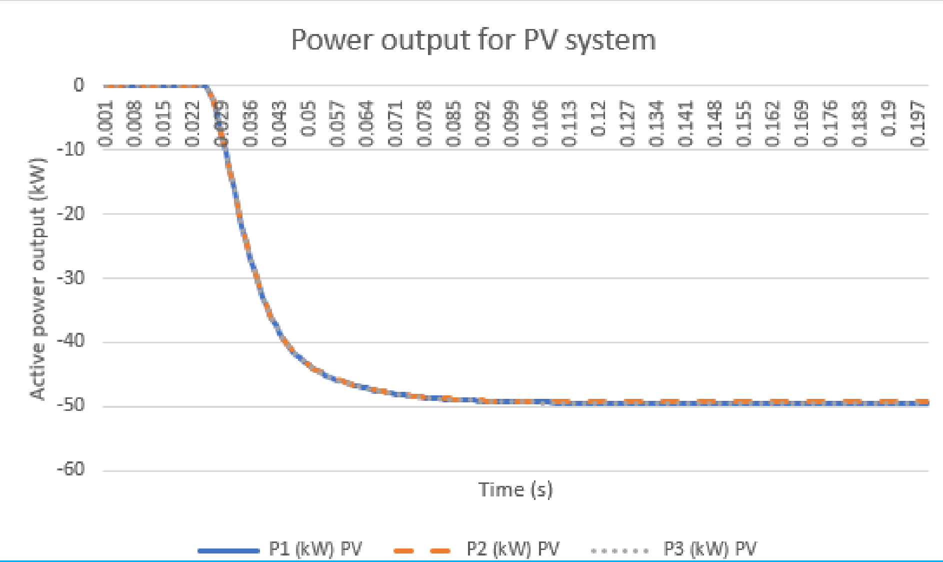
https://sourceforge.net/p/electricdss/code/HEAD/tree/trunk/Version8/Distrib/Examples/IBRDynamics_Cases/

Grid forming inverter for IBR

GFM for IBR

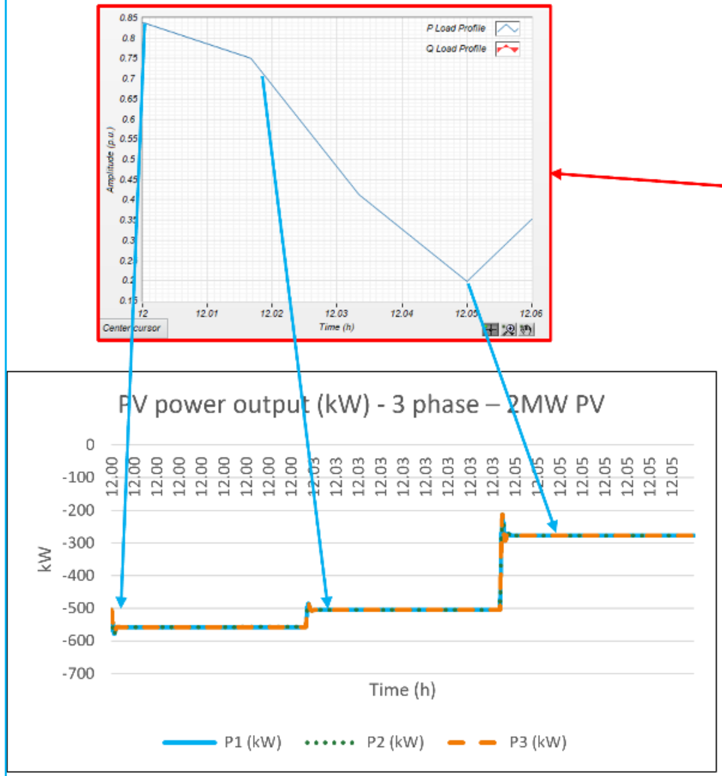


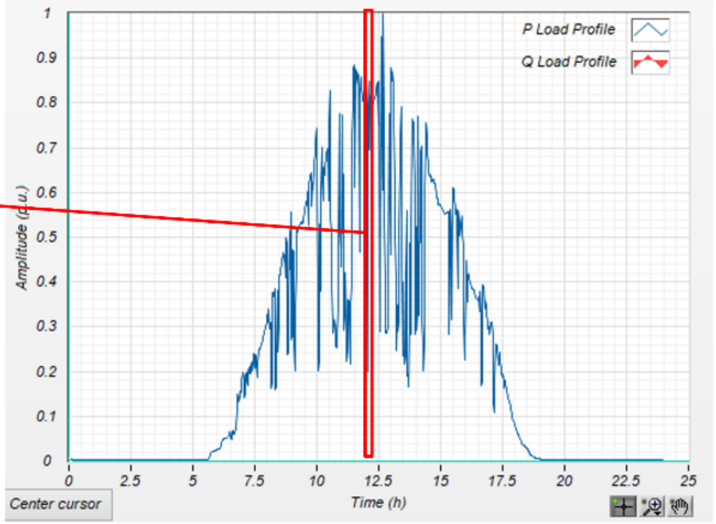




When configured in Grid Forming inverter control mode (GFM), an IBR provides the services traditionally provided by synchronous generators. For this, GFM inverters differ from the operational features of Grid Following (GFL) inverters in several areas. One aspect is that GFL affords the opportunity to energize a section of the grid (which would have been typically provided by a GFM inverter) and it can also coexist and facilitate energy transactions with other GFL inverters within the same circuit.

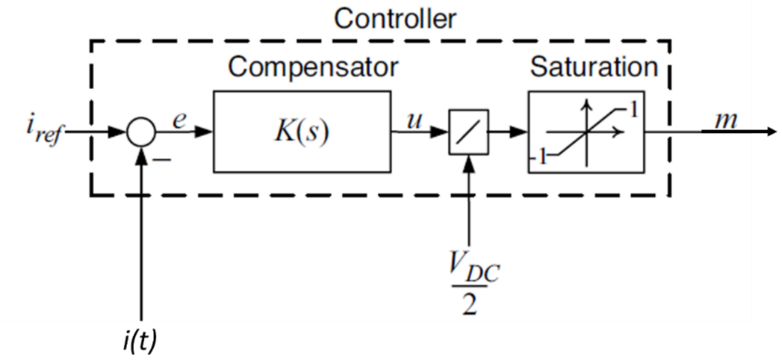
GFM for IBR





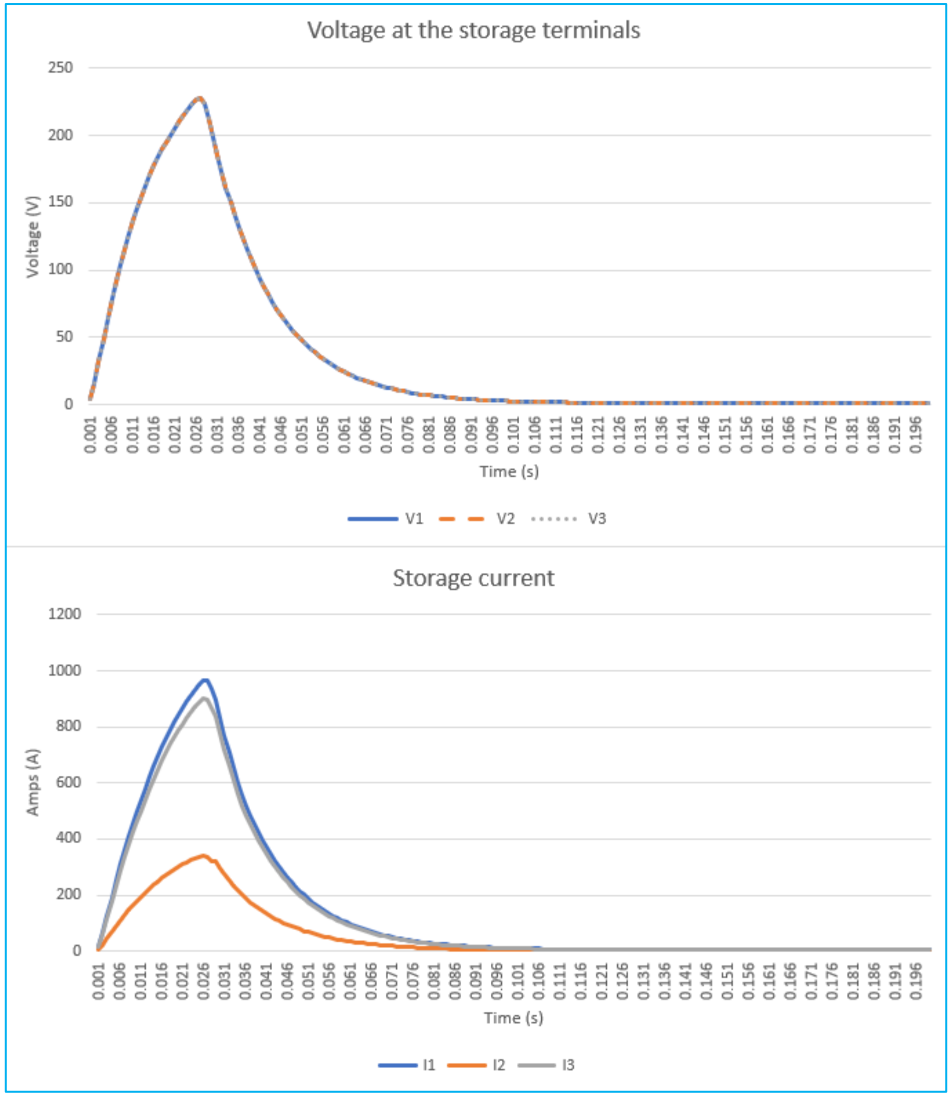
The PV system dynamics depends directly on the PV output, which is dictated by the irradiance profile.

Depending on the time of the day the dynamics simulation takes place, the response for the PV will change.



GFM for IBR

This operation mode requires adjustments in the IBR but also support from an InvControl for current limiting purposes.




```
! This case considers the default load shape to represent
the load demand in time for daily simulations
! Adding DER for supporting the microgrid below switch 5

New "Storage.mystorage" phases=3 conn=delta bus1=StoBus
kV=0.48 kva=800 kWrated=800 kWhrated=6000 %stored=100
%reserve=20
~ %EffCharge=90 %EffDischarge=90 %IdlingkW=1 %R=50 %X=50
State=DISCHARGING kP=0.5 KVDC=0.700 PITol=0.01 AmpLimit=500
AmpLimitGain=0.836

New "PVSystem.myPV" phases=3 conn=delta bus1=PVBus kV=0.48
kva=200 pmpp=200 daily=pvshape %R=50 %X=50 kP=0.1 KVDC=0.700
PITol=0.1

! Grid forming inverter mode requires an InvControl for
monitoring the current
New InvControl.StoCtrl DERList=[Storage.mystorage] mode=GFM
```

GFM for IBR

Examples:

https://sourceforge.net/p/electricdss/code/HEAD/tree/trunk/Version8/Distrib/Examples/IBRDynamics_Cases/

https://sourceforge.net/p/electricdss/code/HEAD/tree/trunk/Version8/Distrib/Examples/Microgrid/GridFormingInverter/GFM_AmpsLimit_123/

https://sourceforge.net/p/electricdss/code/HEAD/tree/trunk/Version8/Distrib/Examples/Microgrid/GridFormingInverter/GFM_IEEE123/

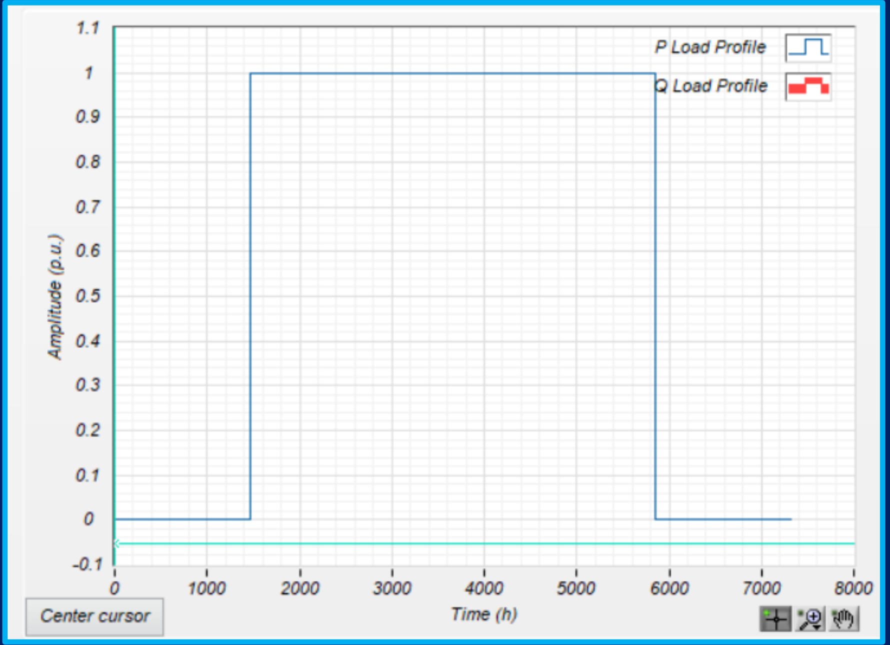
https://sourceforge.net/p/electricdss/code/HEAD/tree/trunk/Version8/Distrib/Examples/Microgrid/GridFormingInverter/GFM_IEEE8500/

CapControl follow mode and interpolation modes for LoadShapes

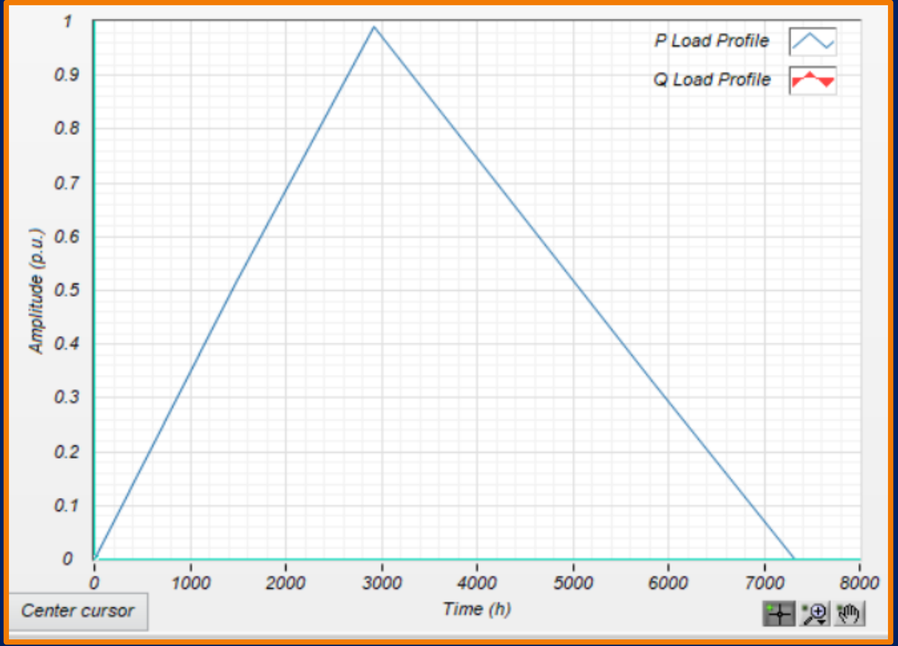
CapControl Follow mode

```
New "Capacitor.myCap" phases=1
bus1=myBus.3 kvar=[ 100] conn=wye
kv=7.20533135948653

! Cap controls New
"LoadShape.CapCtrlTime" npts=4
mult=[0 1 0 0] hour=[0 2881 7297
8760] interpolation=EDGE

New "CapControl.myCapCtrl"
capacitor=myCap type=Follow
terminal=1 ControlSignal=CapCtrlTime
delay=0 delayOFF=0
```

EDGE

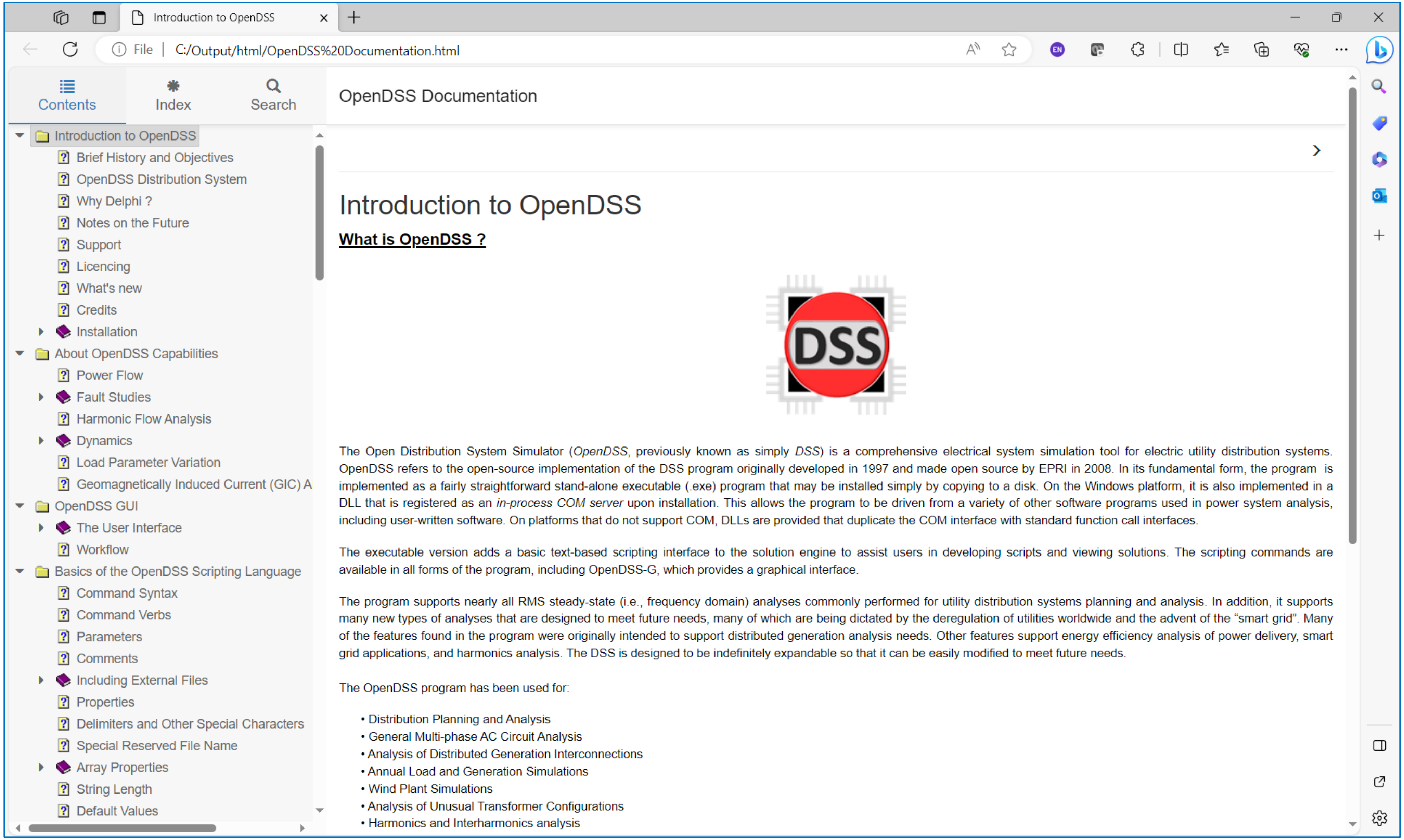


AVG



Documentation in the cloud

The documentation is moving into the cloud



More flavors for DSS at the base

The C++ version mimics the original in Delphi, will be publicly available in Summer 2024. The base version will remain Delphi, new features in C++ will be migrated to Delphi. No version will replace the other.

LL Faults in Fault Study Mode

LL Faults in Fault Study Mode - Background

Compile *"Path/Master.dss"*
Set mode=Faultstudy
Solve
Show faults
Export faults

- Fault Study Mode
  - Apply "All-Node" (LLL), "One-Node to Ground" (LG), "Adjacent Node-Node" (LL) to **all buses and nodes** in the model
  - Results can be reported in two ways:
    - *Export Faults*: **maximum** fault current for each fault type and bus
    - *Show Faults:* **all** fault currents at each bus for each type

IEEE13Nodeckt_EXP_FAULTS.CSV - Notepad

```
Bus,  3-Phase,  1-Phase,  L-L
SOURCEBUS     ,    100419,    105433,     86965
650           ,   2101888,   2728208,   1801815
RG60          ,   1468375,   1738305,   1246291
633           ,      8189,      4712,      7121
634           ,     19438,     16245,     16902
671           ,      5705,      3096,      5042
645           ,      5967,      4329,      5717
646           ,      5004,      3656,      4741
692           ,      5705,      3096,      5042
675           ,      5026,      2853,      4482
611           ,      2374,      2374,         0
652           ,      2294,      2294,         0
670           ,      8555,      4682,      7589
632           ,     11403,      6264,     10135
680           ,      4510,      2446,      4005
684           ,      4043,      2740,      3910
```

IEEE13Nodeckt_FaultStudy.Txt - Notepad

```
ONE-Node to ground Faults

                                    pu Node Voltages (L-N Volts if no base)
Bus          Node  Amps          Node 1     Node 2     Node 3    ...

"SOURCEBUS"    1        105435        0.000      0.955      0.998
"SOURCEBUS"    2        105437        0.998      0.000      0.955
"SOURCEBUS"    3        105435        0.955      0.998      0.000
"650"          1       2810837        0.000      0.894      0.868
"650"          2       2810900        0.868      0.000      0.894
"650"          3       2810909        0.894      0.868      0.000
"RG60"         1       1760498        0.000      0.907      0.999
"RG60"         2       1792349        0.999      0.000      0.923
"RG60"         3       1760533        0.923      0.981      0.000
"633"          1          4712        0.000      1.319      1.254
"633"          2          4632        1.277      0.000      1.217
"633"          3          4596        1.262      1.248      0.000
"634"          1         16291        0.000      1.111      1.066
"634"          2         16274        1.072      0.000      1.065
"634"          3         16029        1.080      1.079      0.000
"671"          1          3096        0.000      1.335      1.254
```

IEEE13Nodeckt_FaultStudy.Txt - Notepad

```
FAULT STUDY REPORT

ALL-Node Fault Currents

Bus              Node 1  X/R        Node 2  X/R        Node 3  X/R    ...  (Amps)

"SOURCEBUS"      100419   4.0      100419   4.0      100419   4.0
"650"           2101888   4.6     2101888   4.6     2101888   4.6
"RG60"          1442309   2.5     1468375   2.5     1442309   2.5
"633"              8189   1.8        7933   2.9        7111   1.9
"634"             19431   1.7       19438   2.0       18698   1.8
"671"              5705   1.9        5552   4.3        4822   2.3
"645"              5890   1.0        5967   3.6
"646"              4851   0.9        5004   2.8
"692"              4822   2.3        5705   1.9        5552   4.3
"675"              5026   1.6        5014   3.2        4318   2.1
"611"              2374   1.8
"652"              2294   1.4
"670"              8555   2.0        8341   4.4        7257   2.4
"632"             11403   2.1       11133   4.4        9691   2.5
"680"              4510   1.9        4427   4.5        3814   2.5
"684"              4043   5.3        3994   1.1
```

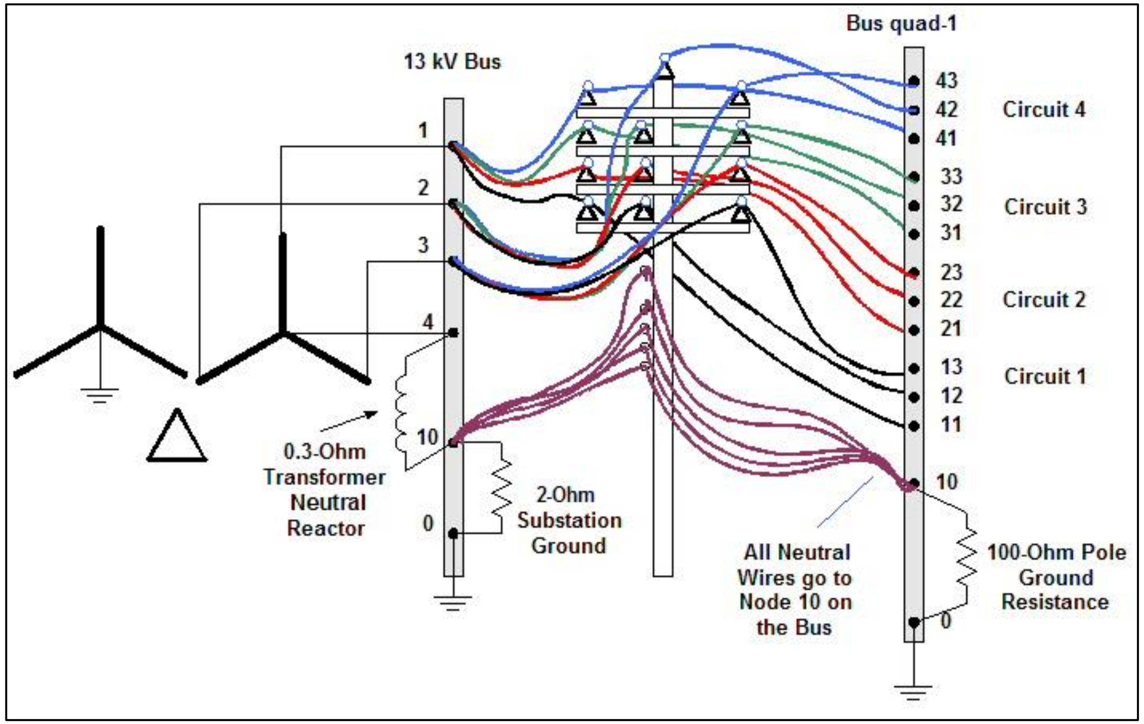
IEEE13Nodeckt_FaultStudy.Txt - Notepad

```
Adjacent Node-Node Faults

                                       pu Node Voltages (L-N Volts if no base)
Bus          Node-Node       Amps       Node 1     Node 2     Node 3    ...

"SOURCEBUS"   1    2        86964       0.500      0.500      1.000
"SOURCEBUS"   1    3        86962       0.500      1.000      0.500
"SOURCEBUS"   2    3        86965       1.000      0.500      0.500
"650"         1    2      1801750       0.536      0.464      1.000
"650"         1    3      1801764       0.464      1.000      0.536
"650"         2    3      1801815       1.000      0.536      0.464
"RG60"        1    2      1246239       0.547      0.500      1.057
"RG60"        1    3      1235486       0.505      1.037      0.552
"RG60"        2    3      1246291       1.056      0.547      0.500
"633"         1    2         7121       0.493      0.492      0.988
"633"         1    3         6714       0.513      1.069      0.513
"633"         2    3         6174       0.995      0.513      0.513
"634"         1    2        16902       0.498      0.493      0.981
"634"         1    3        16596       0.493      1.017      0.499
"634"         2    3        16255       0.983      0.501      0.496
"671"         1    2         5042       0.479      0.478      0.977
"671"         1    3         4529       0.492      1.073      0.492
"671"         2    3         4232       0.956      0.517      0.517
"645"         2    3         5717       0.512      0.512
```

LL Faults in Fault Study Mode

$C_{13,2} = 78$ LL faults @ *Bus quad-1*

- IEEE NEV Test Case
  - Buses = 95
  - Nodes = 378



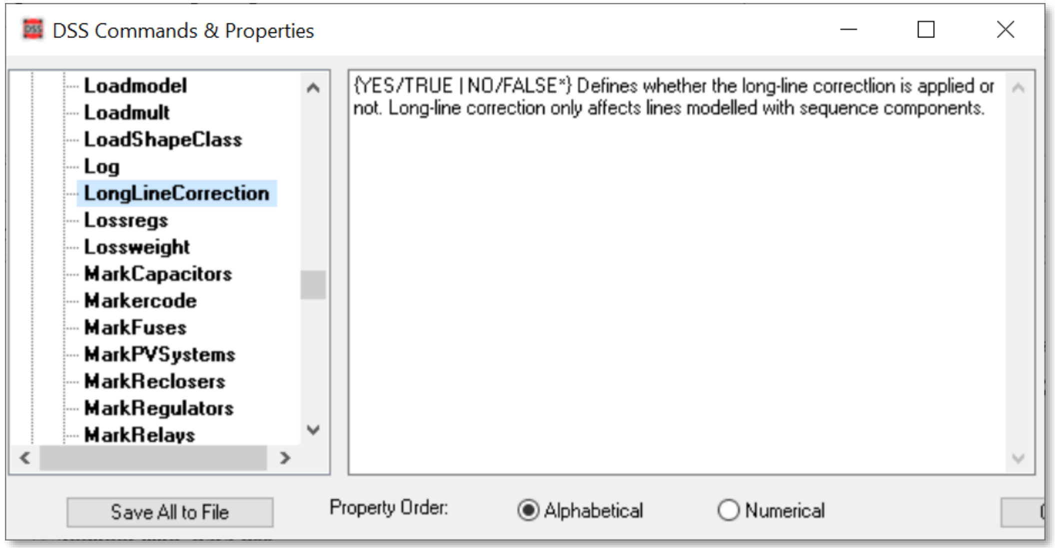
Adjacent Node-Node Faults

Bus	Node-Node	Amps	pu Node Voltages (L-N Volts if no base) Node 1	Node 2	Node 3	...											
"QUAD-1"	11 12	13759	0.512	0.512	1.002	0.524	0.504	1.001	0.524	0.504	1.001	0.524	0.504	1.000	0.001
"QUAD-1"	11 13	13615	0.495	1.025	0.495	0.486	1.025	0.507	0.488	1.025	0.507	0.488	1.024	0.507	0.003
"QUAD-1"	11 21	142	1.004	1.025	1.001	1.004	1.024	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	11 22	13708	0.514	0.507	1.001	0.525	0.513	1.000	0.525	0.505	1.000	0.525	0.506	0.999	0.003
"QUAD-1"	11 23	13606	0.494	1.026	0.507	0.486	1.025	0.494	0.487	1.025	0.507	0.488	1.025	0.506	0.004
"QUAD-1"	11 31	143	1.003	1.025	1.001	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	11 32	13634	0.514	0.510	1.000	0.526	0.507	1.000	0.526	0.514	1.000	0.525	0.506	0.999	0.004
"QUAD-1"	11 33	13579	0.493	1.027	0.506	0.486	1.026	0.506	0.488	1.026	0.494	0.488	1.025	0.506	0.005
"QUAD-1"	11 41	231	1.003	1.025	1.001	1.003	1.025	1.000	1.003	1.025	1.000	1.003	1.024	0.999	0.003
"QUAD-1"	11 42	13569	0.514	0.513	0.999	0.527	0.511	0.999	0.526	0.508	1.000	0.526	0.514	0.999	0.005
"QUAD-1"	11 43	13553	0.493	1.028	0.506	0.486	1.027	0.506	0.488	1.026	0.506	0.489	1.024	0.493	0.005
"QUAD-1"	11 10	10870	0.025	1.179	1.187	0.062	1.178	1.187	0.066	1.177	1.188	0.067	1.175	1.187	0.025
"QUAD-1"	12 13	13756	1.002	0.508	0.507	1.001	0.522	0.498	1.002	0.522	0.499	1.001	0.521	0.498	0.002
"QUAD-1"	12 21	13706	0.523	0.511	1.003	0.511	0.504	1.002	0.523	0.505	1.002	0.523	0.505	1.001	0.001
"QUAD-1"	12 22	66	1.004	1.025	1.001	1.003	1.025	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	12 23	13733	1.001	0.508	0.500	1.001	0.522	0.508	1.001	0.522	0.498	1.001	0.522	0.499	0.003
"QUAD-1"	12 31	13636	0.522	0.511	1.004	0.522	0.504	1.003	0.511	0.508	1.002	0.523	0.507	1.001	0.002
"QUAD-1"	12 32	37	1.004	1.025	1.001	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	12 33	13682	1.000	0.509	0.503	1.000	0.523	0.500	1.001	0.523	0.509	1.001	0.523	0.498	0.003
"QUAD-1"	12 41	13583	0.521	0.510	1.004	0.521	0.504	1.003	0.522	0.507	1.002	0.510	0.508	1.001	0.003
"QUAD-1"	12 42	61	1.004	1.025	1.001	1.003	1.025	1.000	1.003	1.025	1.000	1.002	1.025	0.999	0.003
"QUAD-1"	12 43	13642	1.000	0.509	0.505	1.000	0.523	0.503	1.001	0.523	0.500	1.001	0.523	0.509	0.004
"QUAD-1"	12 10	11009	1.192	0.029	1.160	1.191	0.065	1.159	1.190	0.068	1.158	1.189	0.070	1.156	0.029
"QUAD-1"	13 21	13609	0.488	1.024	0.496	0.496	1.024	0.508	0.487	1.025	0.508	0.488	1.024	0.507	0.003
"QUAD-1"	13 22	13728	1.004	0.520	0.506	1.002	0.506	0.497	1.002	0.520	0.499	1.002	0.520	0.498	0.002
"QUAD-1"	13 23	88	1.004	1.025	1.000	1.003	1.025	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	13 31	13583	0.491	1.023	0.496	0.489	1.023	0.509	0.496	1.025	0.508	0.487	1.024	0.508	0.003
"QUAD-1"	13 32	13676	1.005	0.519	0.506	1.003	0.519	0.497	1.003	0.506	0.500	1.002	0.519	0.500	0.003
"QUAD-1"	13 33	83	1.004	1.025	1.000	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	13 41	13559	0.493	1.022	0.497	0.491	1.022	0.509	0.489	1.024	0.509	0.496	1.024	0.508	0.004
"QUAD-1"	13 42	13625	1.006	0.518	0.505	1.004	0.518	0.497	1.003	0.519	0.500	1.002	0.505	0.501	0.004
"QUAD-1"	13 43	186	1.004	1.025	1.000	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.024	1.000	0.003
"QUAD-1"	13 10	10831	1.157	1.210	0.027	1.158	1.211	0.063	1.158	1.212	0.067	1.157	1.211	0.068	0.027
"QUAD-1"	21 22	13763	0.525	0.504	1.002	0.512	0.512	1.001	0.525	0.504	1.001	0.524	0.504	1.000	0.002
"QUAD-1"	21 23	13622	0.487	1.025	0.508	0.495	1.025	0.495	0.487	1.025	0.508	0.488	1.024	0.507	0.003
"QUAD-1"	21 31	27	1.004	1.025	1.001	1.003	1.025	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	21 32	13706	0.526	0.507	1.001	0.513	0.506	1.000	0.525	0.513	1.001	0.525	0.504	1.000	0.003
"QUAD-1"	21 33	13611	0.487	1.026	0.507	0.494	1.026	0.507	0.487	1.025	0.494	0.488	1.024	0.507	0.004
"QUAD-1"	21 41	121	1.004	1.025	1.001	1.003	1.025	1.000	1.003	1.025	1.000	1.003	1.024	0.999	0.003
"QUAD-1"	21 42	13615	0.526	0.510	1.000	0.513	0.510	1.000	0.525	0.508	1.000	0.525	0.513	1.000	0.004
"QUAD-1"	21 43	13588	0.486	1.026	0.507	0.494	1.026	0.507	0.487	1.026	0.507	0.488	1.024	0.494	0.004
"QUAD-1"	21 10	10823	0.068	1.176	1.186	0.026	1.180	1.188	0.063	1.179	1.189	0.067	1.177	1.188	0.026
"QUAD-1"	22 23	13761	1.003	0.521	0.498	1.002	0.507	0.507	1.002	0.521	0.497	1.002	0.520	0.498	0.002
"QUAD-1"	22 31	13709	0.524	0.504	1.002	0.524	0.512	1.002	0.512	0.506	1.001	0.524	0.506	1.000	0.001
"QUAD-1"	22 32	69	1.004	1.025	1.001	1.003	1.025	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	22 33	13732	1.002	0.521	0.500	1.001	0.507	0.499	1.002	0.521	0.507	1.002	0.521	0.497	0.002
"QUAD-1"	22 41	13639	0.523	0.503	1.003	0.523	0.511	1.002	0.523	0.505	1.002	0.511	0.508	1.000	0.001
"QUAD-1"	22 42	23	1.004	1.025	1.001	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	22 43	13687	1.002	0.521	0.502	1.001	0.507	0.501	1.001	0.521	0.499	1.002	0.521	0.507	0.003
"QUAD-1"	22 10	10939	1.191	0.073	1.158	1.193	0.030	1.160	1.193	0.067	1.160	1.191	0.072	1.158	0.030
"QUAD-1"	23 31	13611	0.489	1.024	0.508	0.487	1.024	0.495	0.495	1.025	0.508	0.486	1.024	0.507	0.003
"QUAD-1"	23 32	13732	1.004	0.520	0.497	1.003	0.520	0.506	1.002	0.507	0.498	1.002	0.520	0.499	0.002
"QUAD-1"	23 33	10	1.004	1.025	1.001	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.024	0.999	0.003
"QUAD-1"	23 41	13590	0.492	1.023	0.509	0.490	1.023	0.496	0.488	1.024	0.508	0.495	1.024	0.507	0.003
"QUAD-1"	23 42	13676	1.005	0.519	0.497	1.004	0.519	0.506	1.003	0.520	0.498	1.002	0.506	0.500	0.003
"QUAD-1"	23 43	127	1.004	1.025	1.001	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.024	1.000	0.003
"QUAD-1"	23 10	10789	1.157	1.209	0.068	1.158	1.212	0.028	1.158	1.213	0.064	1.158	1.213	0.068	0.028
"QUAD-1"	31 32	13762	0.525	0.505	1.001	0.525	0.504	1.001	0.513	0.512	1.001	0.525	0.504	1.000	0.002
"QUAD-1"	31 33	13623	0.489	1.025	0.507	0.487	1.025	0.507	0.494	1.025	0.495	0.486	1.024	0.507	0.003
"QUAD-1"	31 41	120	1.004	1.025	1.001	1.003	1.025	1.000	1.003	1.025	1.000	1.003	1.024	0.999	0.003
"QUAD-1"	31 42	13675	0.525	0.508	1.000	0.525	0.508	1.000	0.513	0.507	1.000	0.525	0.513	1.000	0.003
"QUAD-1"	31 43	13615	0.488	1.025	0.507	0.486	1.025	0.507	0.494	1.026	0.507	0.486	1.024	0.494	0.003
"QUAD-1"	31 10	10782	0.076	1.173	1.186	0.068	1.177	1.188	0.027	1.181	1.189	0.064	1.179	1.189	0.027
"QUAD-1"	32 33	13761	1.003	0.521	0.499	1.002	0.521	0.498	1.002	0.507	0.507	1.001	0.520	0.497	0.002
"QUAD-1"	32 41	13708	0.524	0.504	1.002	0.524	0.503	1.002	0.524	0.512	1.001	0.512	0.505	1.000	0.001
"QUAD-1"	32 42	85	1.004	1.025	1.001	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.025	0.999	0.003
"QUAD-1"	32 43	13737	1.003	0.521	0.501	1.002	0.521	0.500	1.001	0.507	0.499	1.001	0.520	0.507	0.002
"QUAD-1"	32 10	10890	1.189	0.081	1.156	1.192	0.072	1.159	1.194	0.031	1.161	1.193	0.070	1.160	0.031
"QUAD-1"	33 41	13617	0.490	1.024	0.508	0.489	1.024	0.508	0.488	1.024	0.495	0.495	1.024	0.507	0.003
"QUAD-1"	33 42	13735	1.004	0.520	0.498	1.003	0.520	0.497	1.003	0.520	0.506	1.001	0.506	0.498	0.002
"QUAD-1"	33 43	135	1.004	1.025	1.001	1.003	1.024	1.000	1.003	1.025	1.000	1.002	1.024	1.000	0.003
"QUAD-1"	33 10	10749	1.155	1.208	0.076	1.157	1.211	0.068	1.159	1.214	0.029	1.159	1.214	0.065	0.029
"QUAD-1"	41 42	13727	0.525	0.505	1.001	0.525	0.505	1.001	0.525	0.505	1.001	0.512	0.512	1.000	0.002
"QUAD-1"	41 43	13633	0.489	1.024	0.508	0.488	1.024	0.508	0.487	1.025	0.508	0.495	1.024	0.495	0.003
"QUAD-1"	41 10	10753	0.081	1.171	1.185	0.076	1.175	1.187	0.068	1.179	1.189	0.028	1.181	1.189	0.028
"QUAD-1"	42 43	13792	1.004	0.520	0.499	1.003	0.520	0.498	1.002	0.520	0.497	1.001	0.506	0.506	0.002
"QUAD-1"	42 10	10846	1.187	0.088	1.154	1.189	0.082	1.156	1.192	0.075	1.160	1.194	0.032	1.162	0.032
"QUAD-1"	43 10	10723	1.154	1.207	0.081	1.156	1.210	0.076	1.158	1.214	0.069	1.159	1.216	0.029	0.029

Long-Line Correction

- Correction is needed when modeling relatively long lines with nominal π models.
- Effects tend to be noticeable at higher frequencies.
- OpenDSS applies long-line corrections in harmonics mode for each attempted harmonic order.
- Corrections are applied only when enabled AND
  - We have positive sequence models (Corrections to R1, X1 and C1) OR
  - We have lines modeled with symmetrical components (Corrections to R1, X1, C1, R0, X0 and C0).

```
Clear
New Circuit.newone ...
Set LongLineCorrection = true
...
solve mode = harmonics
...
```



[A] J. Arrillaga and N. R. Watson, Power System Harmonics. Wiley, November 2003

Long-Line Correction

- Correction is needed when modeling relatively long lines with nominal π models.
- Effects tend to be noticeable at higher frequencies.
- OpenDSS applies long-line corrections in harmonics mode for each attempted harmonic order.
- Corrections are applied only when enabled AND
  - We have positive sequence models (Corrections to R1, X1 and C1) OR
  - We have lines modeled with symmetrical components (Corrections to R1, X1, C1, R0, X0 and C0).

```
Clear
New Circuit.newone ...
Set LongLineCorrection = true
...
solve mode = harmonics
...
```

Total series impedance

$$Z = R + j\omega L$$

Total shunt admittance

$$Y = G + j\omega C$$

Propagation constant

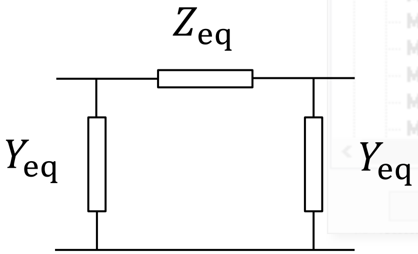
$$\gamma = \sqrt{ZY}$$

Characteristic impedance

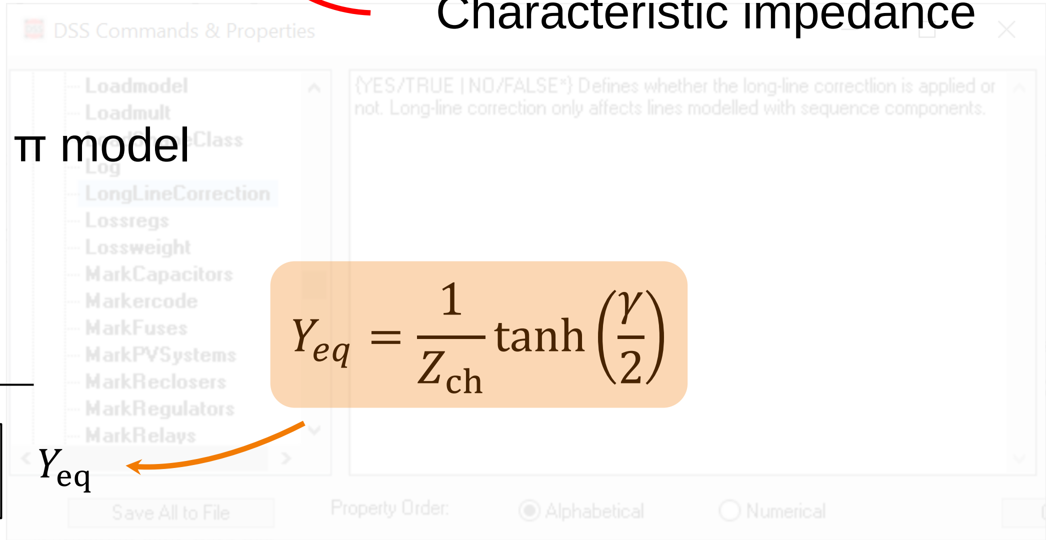
$$Z_{\mathrm{ch}} = \sqrt{Z/Y}$$

Corrected equivalent π model

$$Z_{\mathrm{eq}} = Z_{\mathrm{ch}} \sinh(\gamma)$$



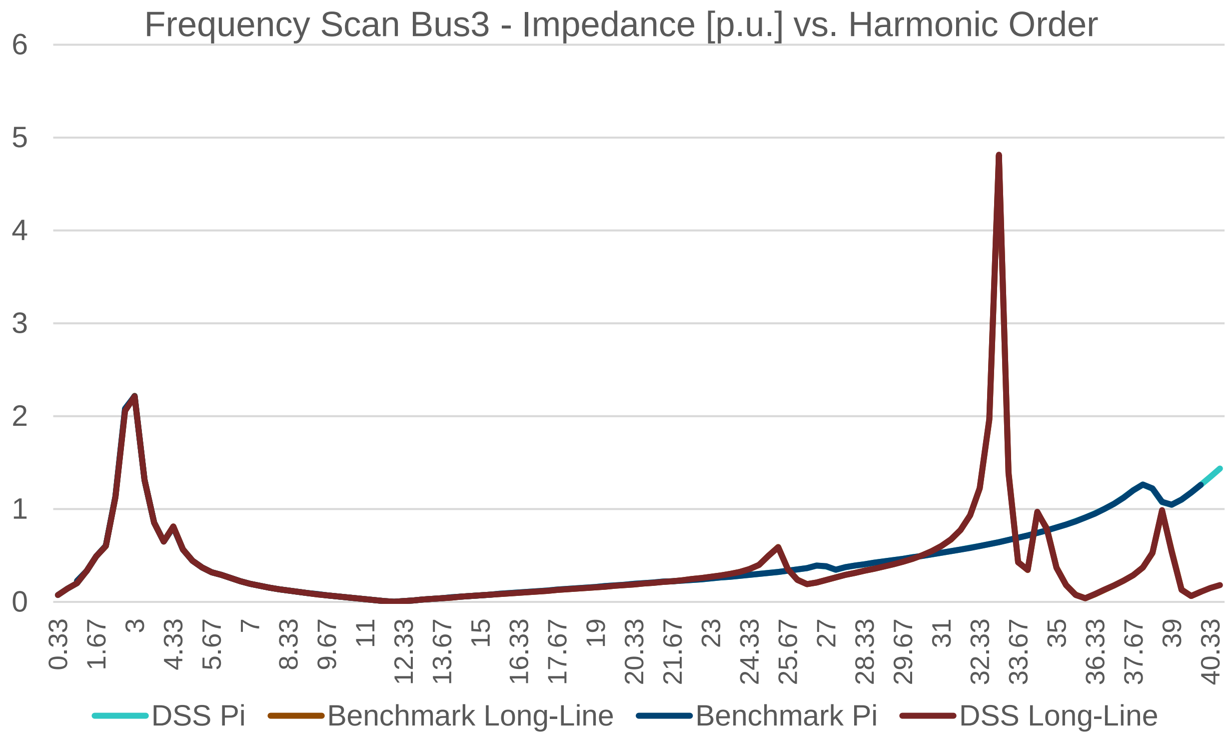
$$Y_{eq} = \frac{1}{Z_{\mathrm{ch}}} \tanh\left(\frac{\gamma}{2}\right)$$



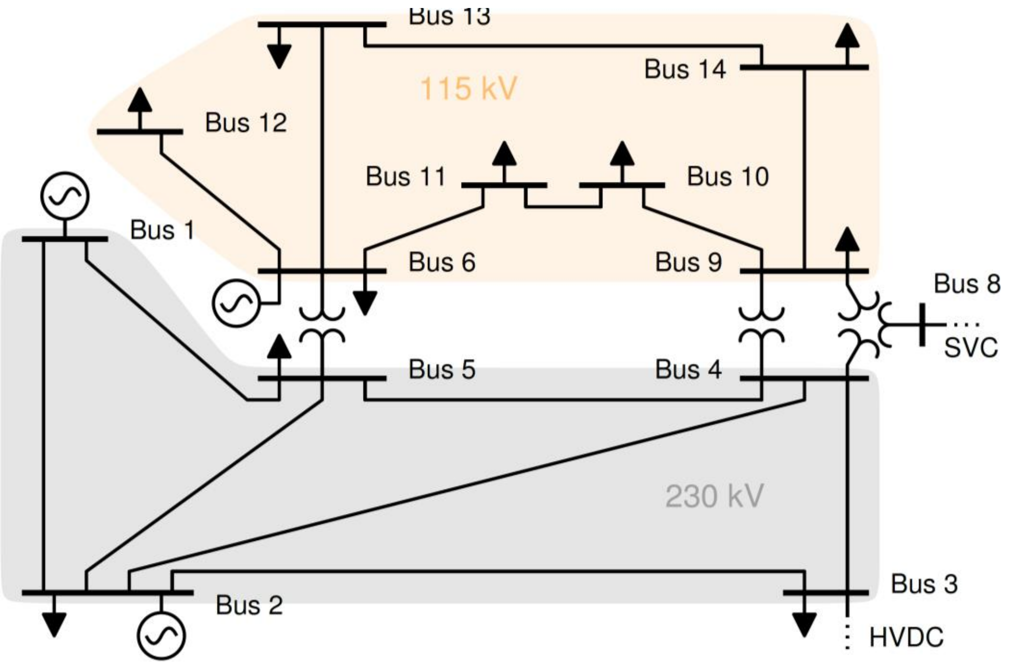
[A] J. Arrillaga and N. R. Watson, Power System Harmonics. Wiley, November 2003

Long-Line Correction

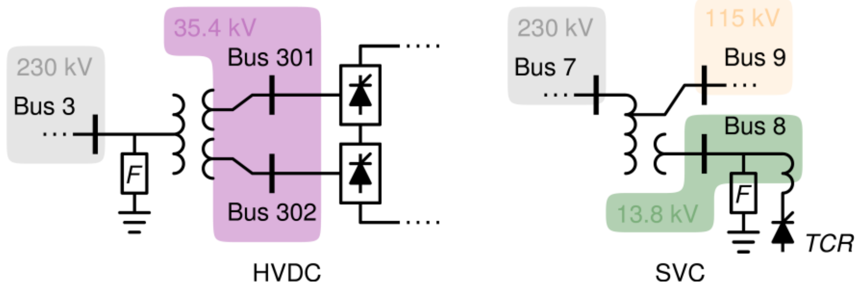
- IEEE Harmonics Task Force benchmark circuit [A]
- Journal paper submitted and under review. Compares harmonics analysis of benchmark against OpenDSS and a commercial Transmission System Planning tool.



Benchmark case: IEEE 14-bus case



Harmonic Sources in IEEE 14-bus case



[A] R. Abu-Hashim, R. Burch, G. Chang, M. Grady, E. Gunther, M. Halpin, C. Harziadonin, Y. Liu, M. Marz, T. Ortmeyer, V. Rajagopalan, S. Ranade, P. Ribeiro, T. Sim, and W. Xu, "**Test systems for harmonics modeling and simulation**," IEEE Transactions on Power Delivery, no. 2, pp. 579–587, April 1999

DSS User Applications and Presentations

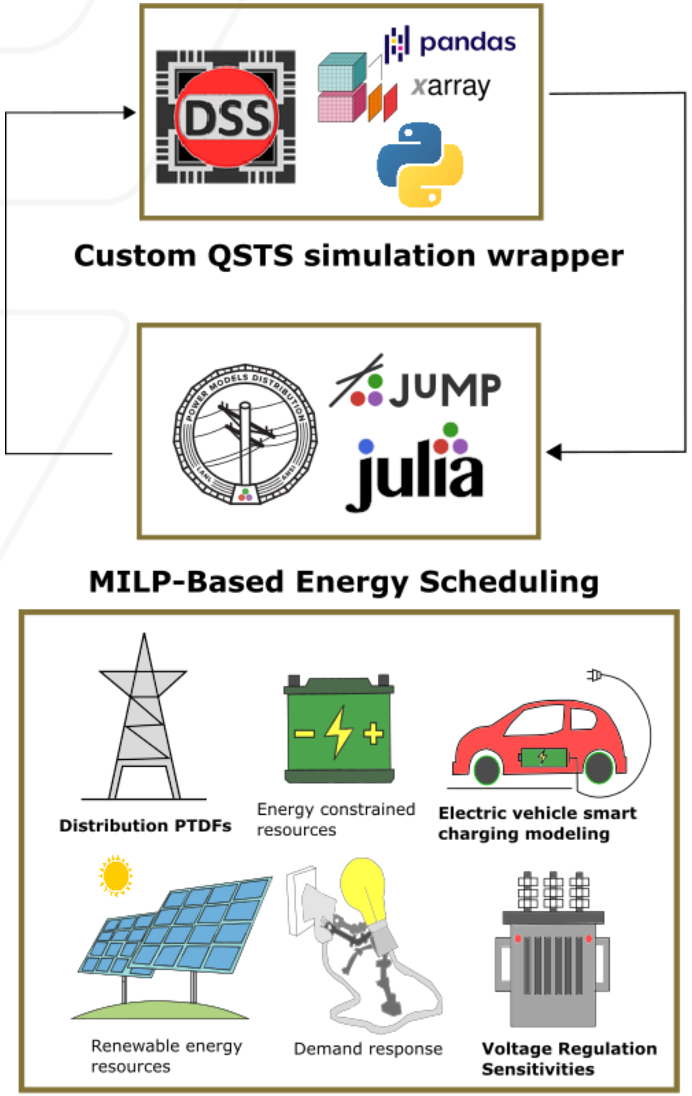
External Speakers

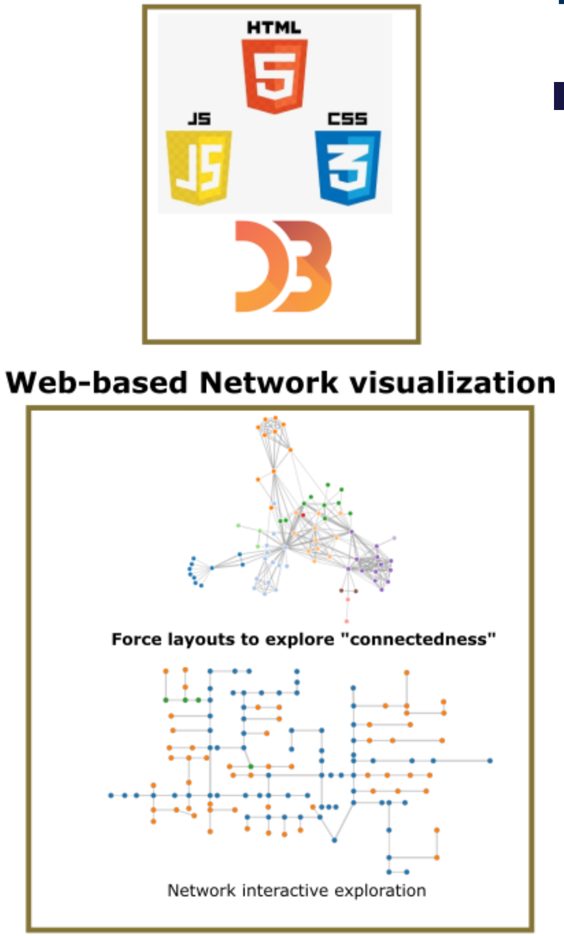
- Jorge Fernandez
  - Energy Scheduling of Distributed Energy Resources in Distribution Feeders
- Ramón Emilio De Jesús Grullón
  - Modeling and Simulation of Distributed Network with High Renewable Penetration in Open-Source Software: QGIS and OpenDSS
- Ana Camila F. Mamede, Ph.D.
  - Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks
- Professor Dr. Zhaoyu Wang
  - OpenDSS for Student Training

Jorge Fernandez

- Jorge Fernandez is a Ph.D. student in the School of Electrical and Computer Engineering (ECE) at GaTech, advised by Prof. Santiago Grijalva. Before joining GaTech, he was a digital services engineer at XM (the Colombian ISO). He started using OpenDSS for his PhD research, which is mainly about optimization/control through time. In Power Systems terms, he works in Unit Commitment at the Transmission level and Energy Scheduling at the Distribution level.

Energy Scheduling of Distributed Energy Resources in Distribution Feeders





Highlights:

- Custom QSTS simulation wrapper for **OpenDSS** (Python).
  - JSON parsing capabilities
  - Custom management of energy meters and monitors.
  - Data manipulation using Pandas and Xarrays.

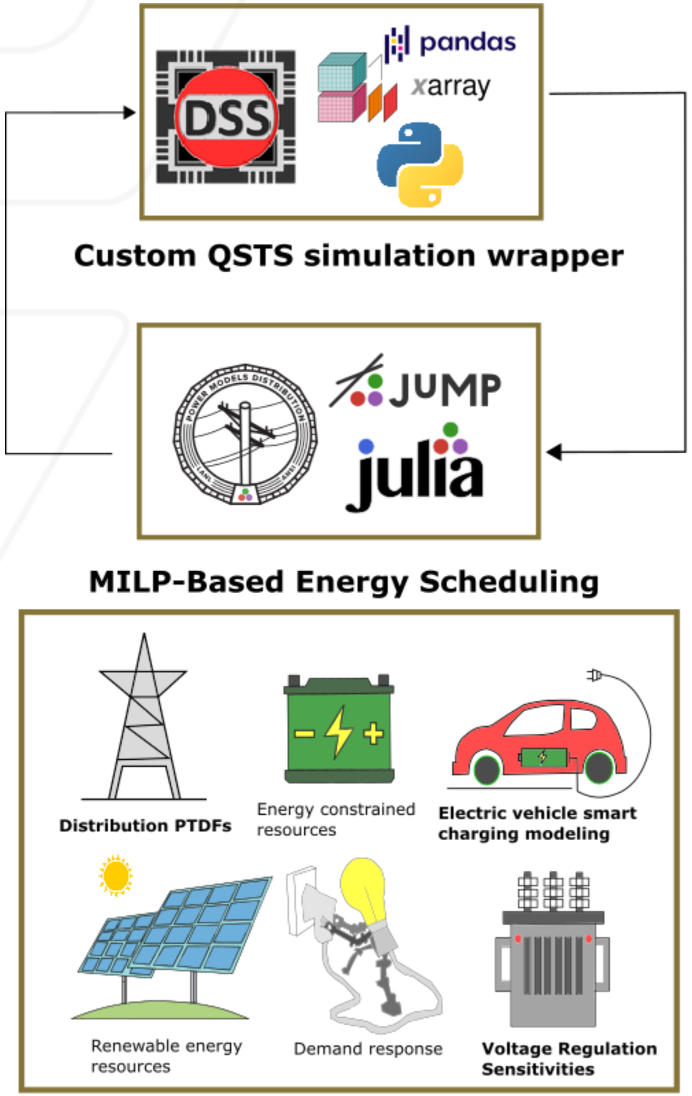
Python is great for manipulating data. Among the best libraries for this task are Pandas, Xarrays, and Numpy. Furthermore, it has early binding capabilities which enable fast COM interfacing with OpenDSS.

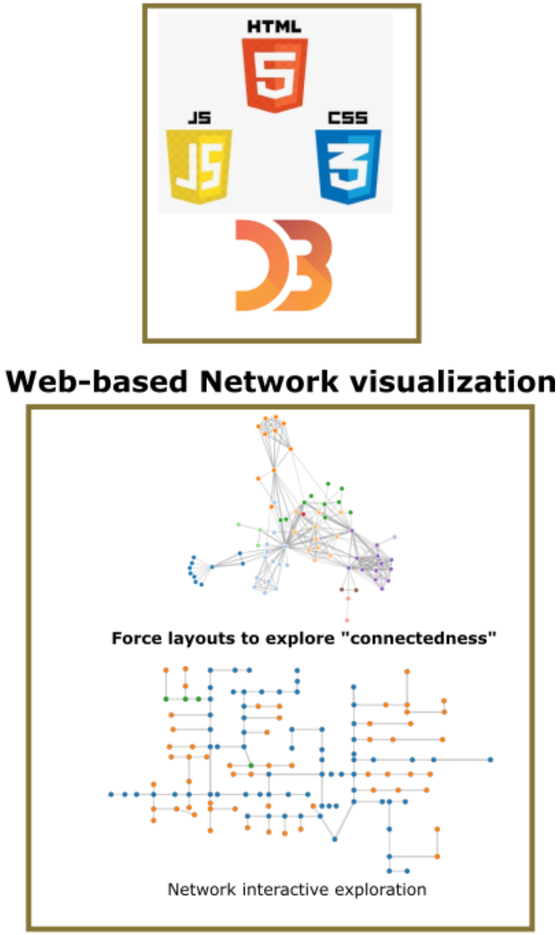
Goals:

1. let OpenDSS handle the non-linear network and provide a warm-start to the optimization (compute voltages at every bus-node and power flows at every line/transformer-phase for every time-step).
2. Return optimization results to OpenDSS and to recover AC feasibility. The algorithm becomes a sequential loop.

Georgia Tech

Energy Scheduling of Distributed Energy Resources in Distribution Feeders





Highlights:

- MILP-based Energy Scheduling (Julia).

  - Hours-ahead or Day-ahead scheduling horizon.
  - Distribution PTDF and Voltage Regulation Sensitivities modeling.
  - Intertemporal constraints and EV Smart Charging modeling.

Julia specializes in optimization programming. Its Mathematical Optimization Interface (JuMP) facilitates coding large optimization programs and connecting to different solvers. Moreover, there are dedicated JuMP Power System wrappers (PowerModels) maintained by two national labs which provide great functionalities and utilities.

Goal:

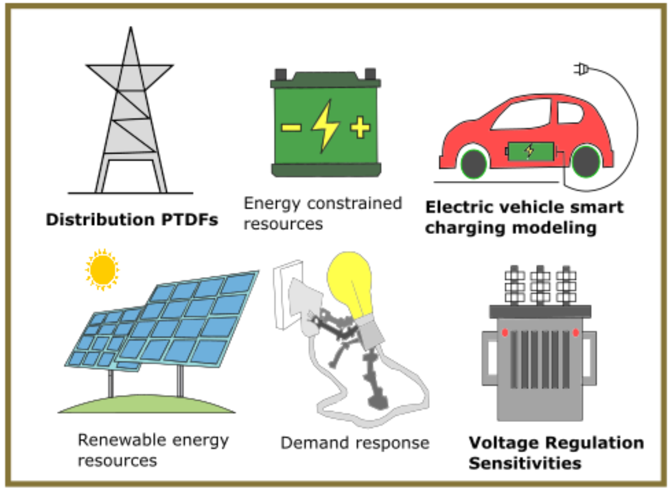
1. Schedule energy resources along an optimization horizon (consideration for past and future states).
2. Include maximum number of switches for devices and optimally schedule them.
3. Consider intertemporal constraints such as battery charging and discharging.

Georgia Tech

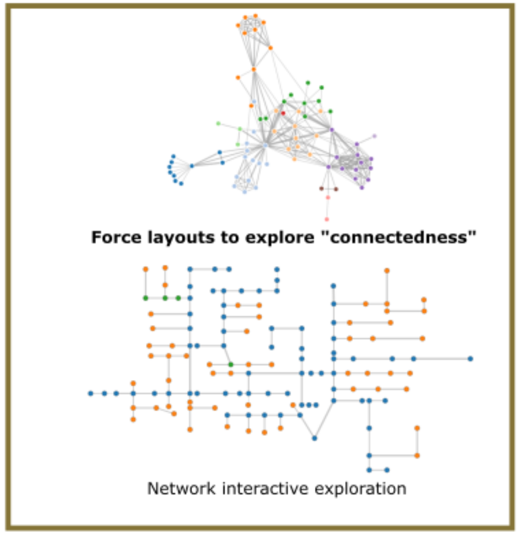
Energy Scheduling of Distributed Energy Resources in Distribution Feeders

Custom QSTS simulation wrapper

MILP-Based Energy Scheduling



Web-based Network visualization



Highlights:

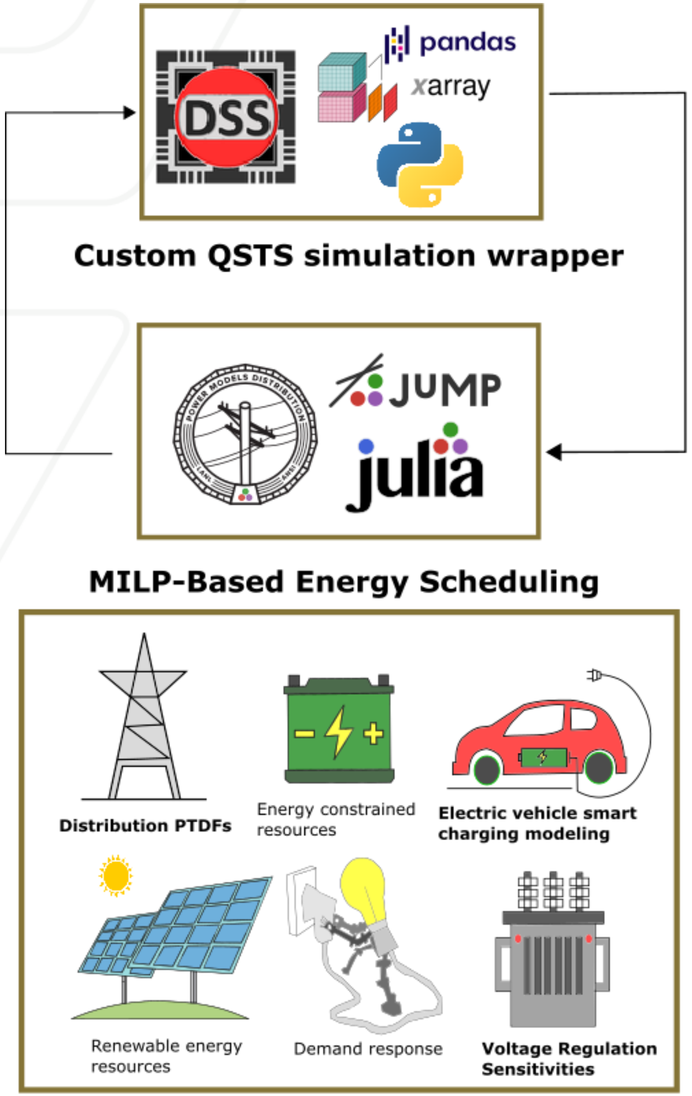
- Web-based Network visualization (JavaScript).
    - Distribution network exploration using interactive visualizations.
    - Force layouts to explore notions of “connectedness”.

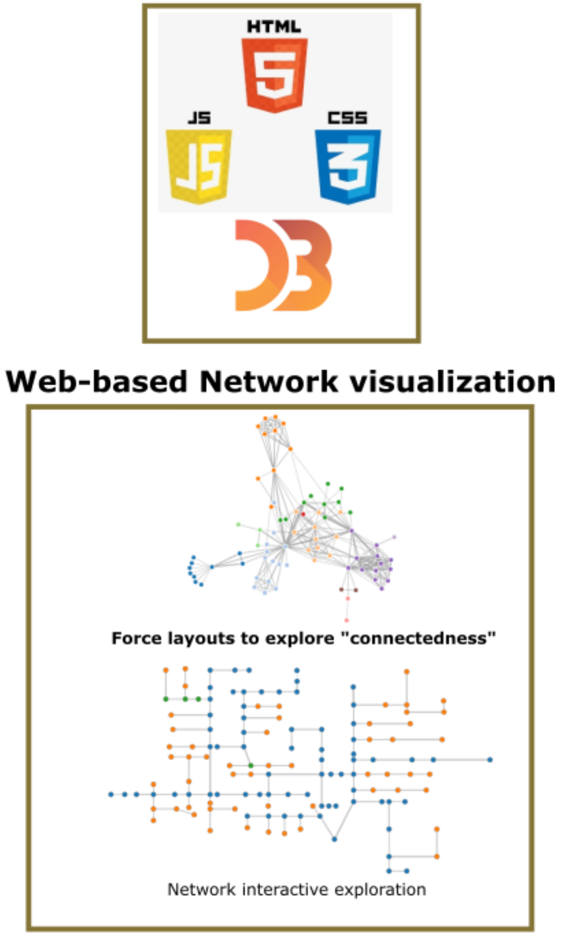
JavaScript (together with CSS and HTML) is the standard for creating expressive visualizations for data exploration and storytelling. There are multiple frameworks and a large support community. The D3 library stands out as the most flexible framework for information visualization.

Goals:

1. Explore different notions of “connectedness”. Power systems are constrained by bus locations. If we relax this constraint, we can use the Ybus to represent device connectivity and the Zbus to represent electrical connectivity between buses. This has applications to optimal transmission switching problems and optimal islanded microgrid operation.
2. Leverage Moder front-end architectures to explore results in an interactive manner.

Energy Scheduling of Distributed Energy Resources in Distribution Feeders





Applications:

- Optimization-based operating envelopes.
- Dynamic pricing strategies for EV charging integration.
- Voltage regulation control.
- Force layouts for network connectivity exploration.

Published research on topic:

[1] “Distribution System EV-hosting Capacity Assessment Considering Decentralized Smart Charging and Static Pricing Rates”, PESGM 2023.

[2] “Conditions for Estimation of Sensitivities of Voltage Magnitudes to Complex Power Injections”, TPWRS 2023.

[3] “Sequential Energy Scheduling Approach to Support Decentralized Smart Charging of Electric Vehicles”, NAPS 2022.

[4] “Implicit Reactive Power Flow Representation to Support Distribution System DER Energy Scheduling”, NAPS 2022.

[5] “Penalty factor-based formulation to support energy scheduling under highly constrained lines in distribution systems”, TPEC 2022.

Georgia Tech

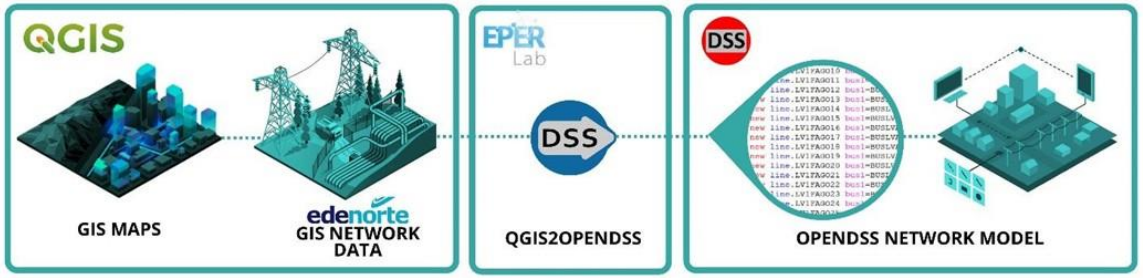
Ramón Emilio De Jesús Grullón

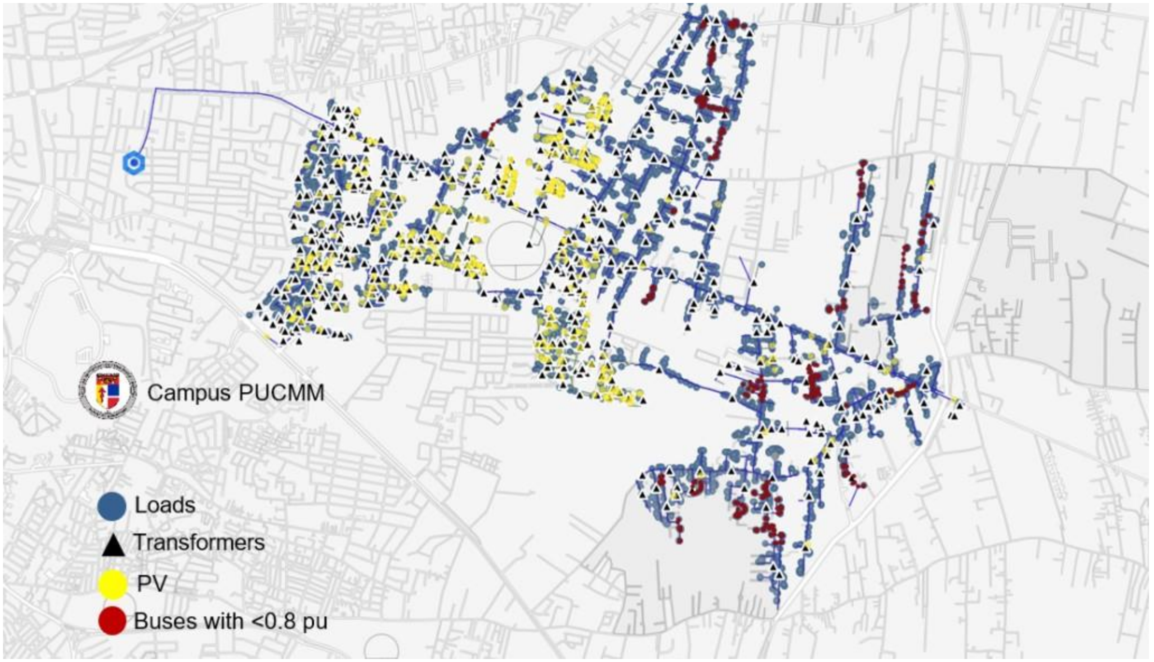
- Ramón Emilio De Jesús-Grullón, MEng, is a researcher specializing in energy resilience and microgrids at the Pontificia Universidad Católica Madre y Maestra (PUCMM), Dominican Republic. He completed his B.S in Electromechanical Engineering in 2015 and later specialized in Renewable Energies and Energy Markets at the School of Industrial Organization (EOI), Madrid, in 2017. Ramón's professional background spans seven years in the energy and water sectors. He began in the Design and Construction of Distribution Networks before moving to Water Leakage Detection in District Metered Areas (DMA) in Water Distribution Networks. Subsequently, he worked on Renewable Energy and Electricity Markets, conducting research on Demand Response and Energy Hedging Markets for wind and solar power in Spain, Poland, the U.K., and Italy. His active research focuses on Energy Resilience and Microgrids under the Partnerships for Enhanced Engagement in Research (PEER) program of the United States National Academy of Sciences (NAS) and the National Fund for Innovation and Scientific and Technological Development (FONDOCYT) of the Dominican Republic. Besides his research at PUCMM, Ramón is involved with Energy Journal, a portal dedicated to the latest in Energy and Sustainability within Dominican Republic

MODELING AND SIMULATION OF DISTRIBUTION NETWORKS WITH HIGH RENEWABLE PENETRATION IN OPEN-SOURCE SOFTWARE: QGIS AND OPENDSS

Ramón Emilio De Jesús Grullón - Microgrid Research Lab - PUCMM

r.dejesus@ce.pucmm.edu.do





From GIS Layers to Power Simulation

Implemented **QGIS2OPENDSS**, a plugin designed to automatically generate distribution network models for **OpenDSS**, which data comes directly from an open-source Geographic Information System (GIS) software environment, **QGIS.**

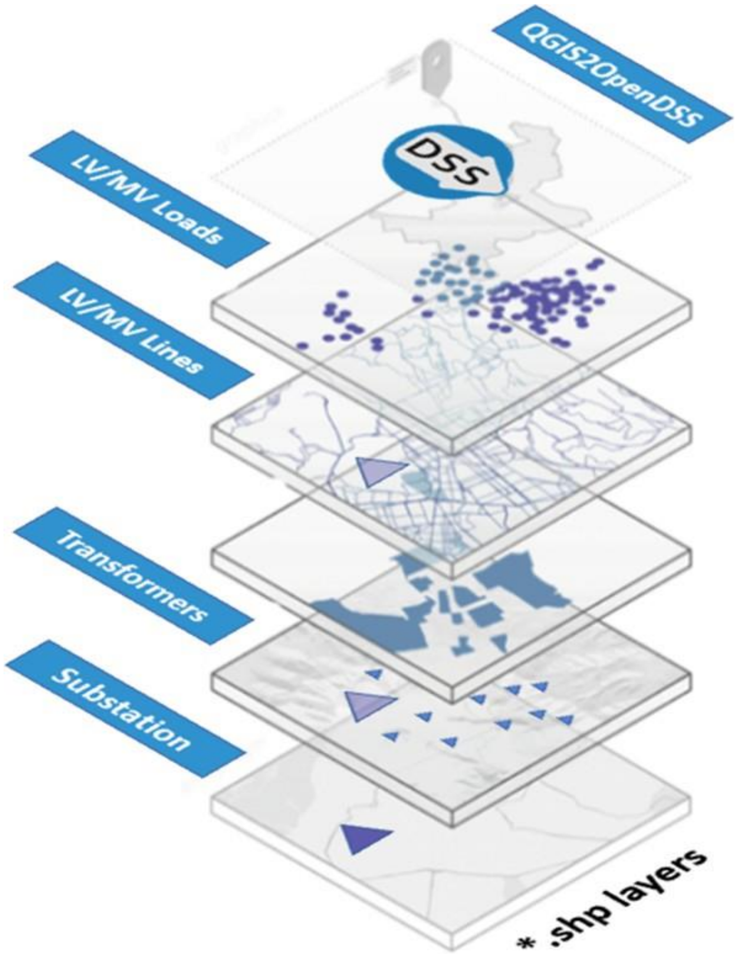
Ongoing studies

Quantifying the impacts of variable sources and loads in the voltage profiles of the feeder and in network losses.

Analysis of the Impact of Distributed Energy Generation on voltage stability

This research is funded by the NAS and USAID under USAID parent award number AID-OAA-A-11-00012

MODELING AND SIMULATION OF DISTRIBUTION NETWORKS WITH HIGH RENEWABLE PENETRATION IN OPEN-SOURCE SOFTWARE: QGIS AND OPENDSS



DATA REQUIREMENTS (EXAMPLES)

Table 1 presents the list of attributes in the shape files of **overhead Medium Voltage (MV), Low Voltage (LV) lines.**

Table 1 - Overhead MV/LV Attributes	
Required Attributes	**Optional Attributes**
NEUTMAT	LENGTH
NEUTSIZ	LENUNIT
PHASEMAT	X1
PHASESIZ	Y1
LINEGEO	X2
PHASEDESIG	Y2
NOMVOLT	

Mandatory and optional attributes in the shapefiles of the **distributed generators**

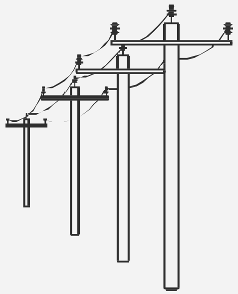
Table 2 - Distributed Generators Attributes	
Required Attributes	**Optional Attributes**
Tech	X1
KVA	Y1
PHASEMAT	
Curve1 (Irradiance)	
Curve 2 (temperature)	

*DSS FILE CREATOR

THE PLUGGIN IN NUMBERS (*LINES OF CODE*)

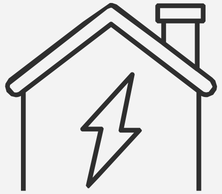
```
TestCircuit-VOLG101/
   ├── DG.dss
   ├── MVLines.dss
   ├── LVLines.dss
   ├── LVLoads.dss
   ├── MVLoads.dss
   ├── Loadshapes.dss
   ├── Substation.dss
   ├── Transformers.dss
   └── Library
         ├── Wiredata.dss
         └── ConfigLInes
               ├── LineSpacing
               ├── LineGeometry
               └── LineCode
```

NETWORK



BT –9,991
MT –2,498

LOADS



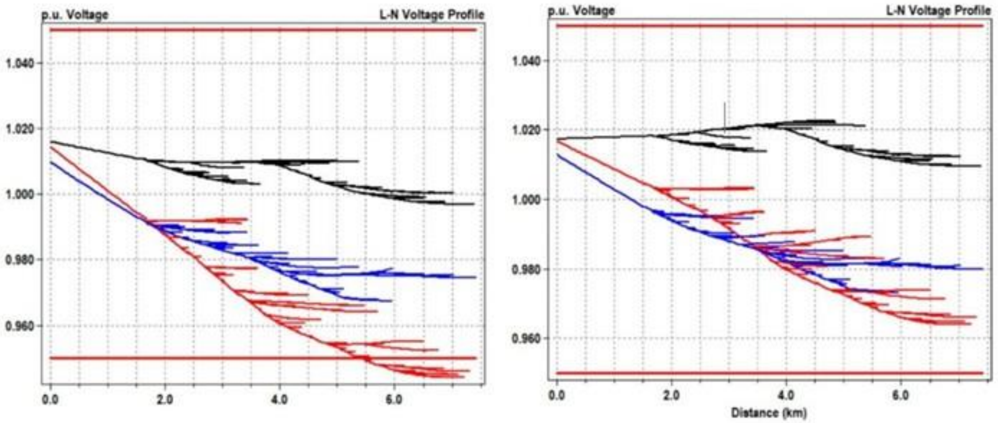
LOADSHAPES –441
LOADS –6,559

XFXs & BUSES

LV Buses –10,206
MV Buses - 2,470
XFX - 579

INITIAL ANALYSIS CARRIED OUT TO STUDY THE IMPACT OF PHOTOVOLTAIC (PV) GENERATION

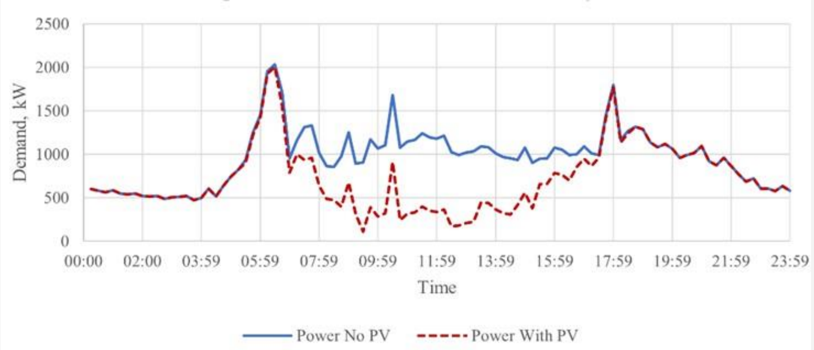
PV penetration heat map (% voltage rise)



L-N Voltage profiles (pu) - (a) Without PV (b) With PV

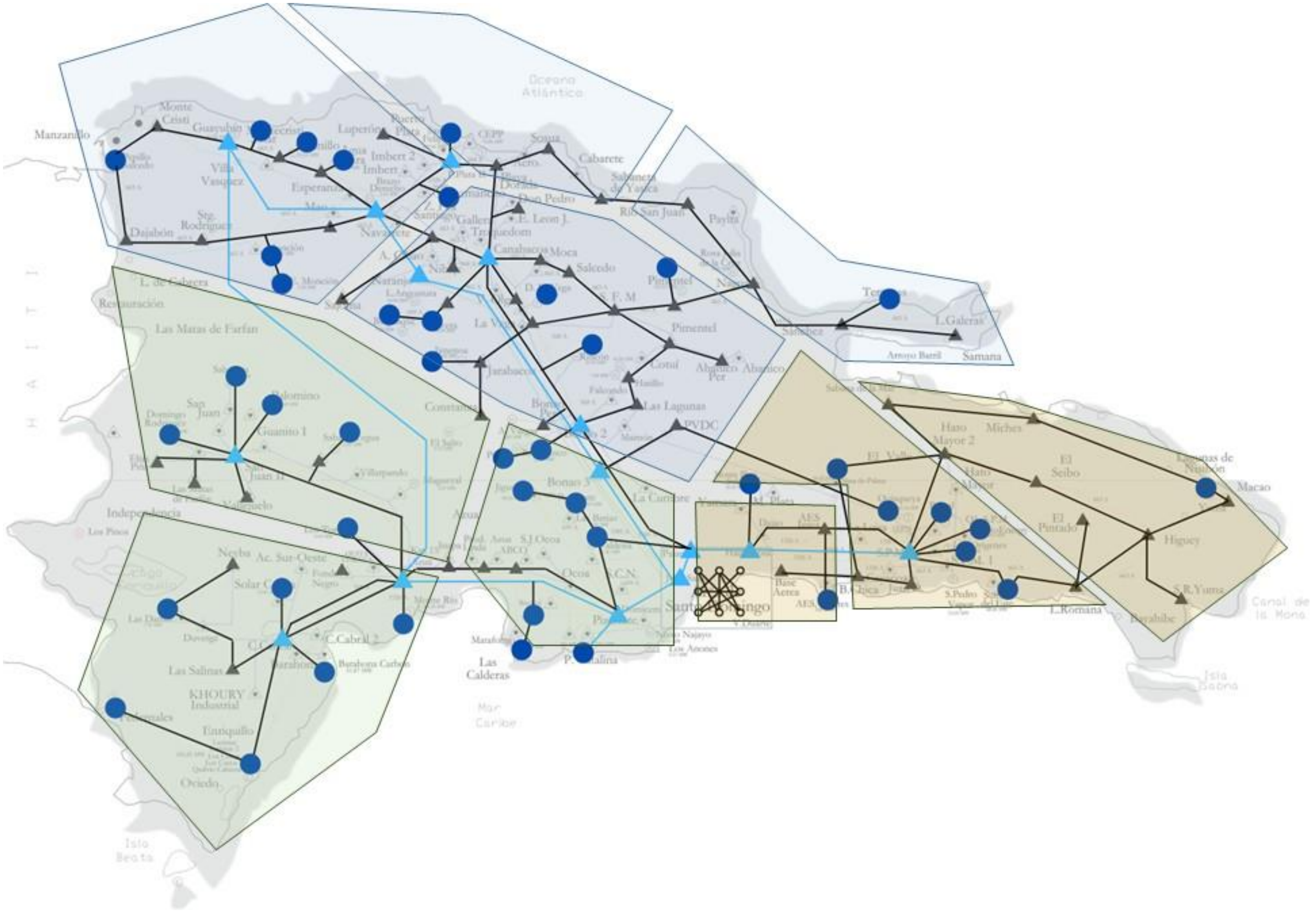
Feeder characteristics

VOLG101 characteristics	
Circuit Alias	VOLG101
System Voltage (kV):	12.47
Number of customers:	7,428
Sub transmission Voltage (kV):	69
Circuit MV lines (kM):	78
Circuit LV and Services lines (kM):	145
Number of transformers:	578
Number of PV Installations:	394
Reported Technical Losses (2021)	6.4%
Renewable Penetration:	60%



Feeder load profile with and without PV - daily time series simulation.

Q&A



THANK YOU!

r.dejesus@ce.pucmm.edu.do

microgrid.pucmm.edu.do

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4523911

Ana Camila F. Mamede, Ph.D.

Ana Camila Mamede completed her bachelor's degree in electrical engineering at the Federal University of Uberlândia in 2014, followed by a master's degree in 2016 and a doctoral degree in 2021 at the same institution. During her doctoral studies, she participated in a sandwich Ph.D. program, spending one year at the Faculty of Engineering of the University of Porto in Portugal. Currently, she is conducting postdoctoral research at the Electric Distribution Laboratory, focusing on modeling and analyzing distribution systems, photovoltaic distributed generation, energy storage systems, power quality, technical loss calculations, data analysis, and program development for electrical engineering studies.

Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks

Ana Camila F. Mamede, Ph.D.
Electrical Distribution Laboratory
Federal University of Uberlândia
Brazil

Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks

R&D – ANEEL / UFU / Energisa (2022 – 2024)

“Development of remote photovoltaic distributed generation systems and distributed energy storage along feeders, with price arbitration functions and provision of local ancillary services to improve the quality of electrical energy in secondary distribution circuits”

- Geographical Visualization of the Feeder.
- Integration of Distributed Energy Resources (PVSystem and Energy Storage Systems) at various locations.
- Six different energy storage system dispatch modes.
- Graphical and numerical results, such as transformer loading, technical losses, steady-state voltage variation, battery state variables, among others.

Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks

File selection

Path to the folder with .dss files:

D:/repos/gui_dss_v5_ingles/Alimentado

Select a Master:

MasterDuPVS863730_M3.dss

Storage Dispatch Modes

Select the dispatch mode:

- (●) PeakShave (Discharging) and Time (Charging)
- ( ) Follow (Discharging) and Time (Charging)
- ( ) Schedule (Discharging) and Time (Charging)
- ( ) PeakShave (Discharging) and PeakShaveLow (Charging)
- ( ) Loadshape (Charging and Discharging)
- ( ) Time (Charging and Discharging)

☐ Export .csv results

File: `D:/repos/gui_dss_v5_ingles/Alimentador_863730_PVS\MasterDuPVS863730_M3.dss`

Dispatch Mode: `PeakShave (Discharging) and Time (Charging)`

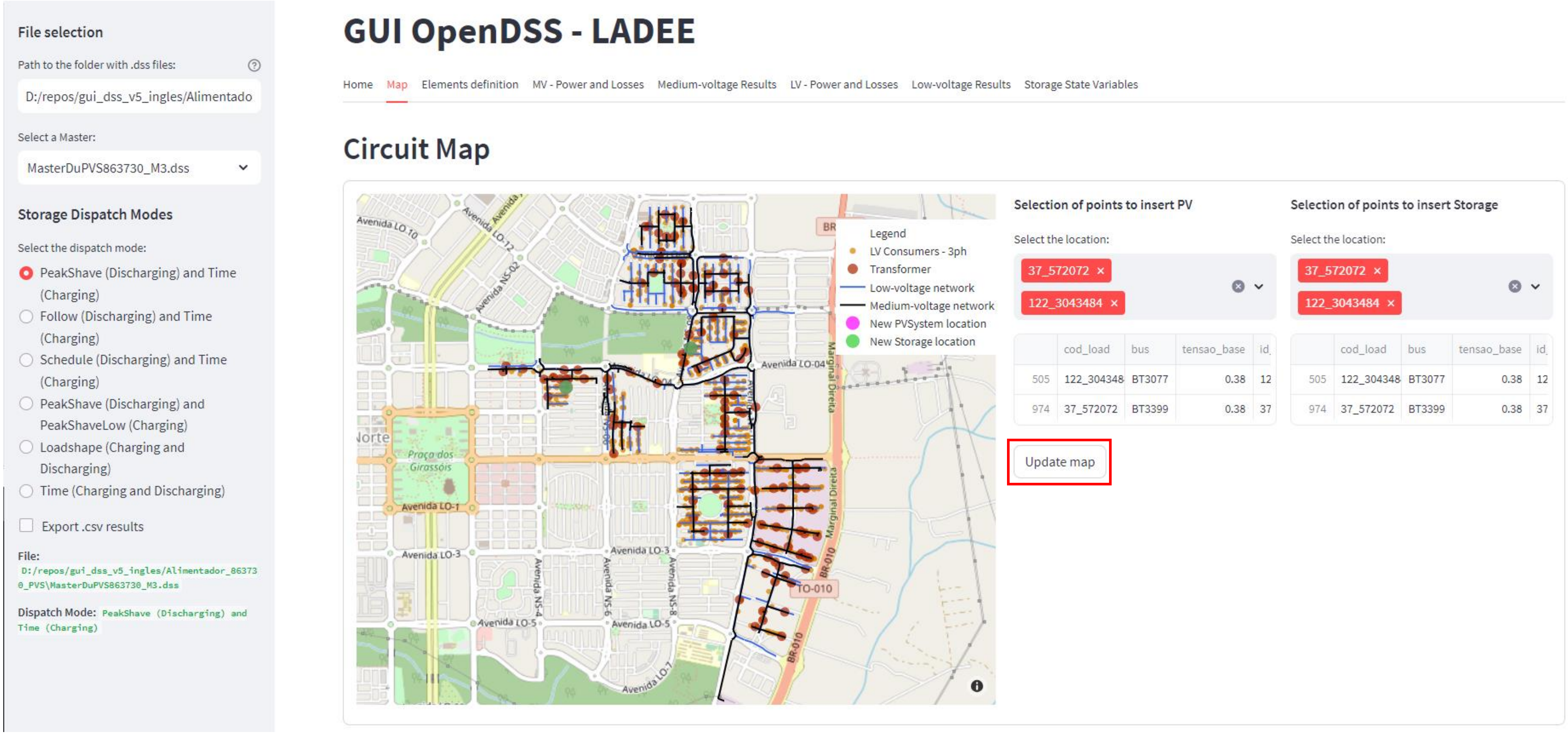
GUI OpenDSS - LADEE

Home | Map | Elements definition | MV - Power and Losses | Medium-voltage Results | LV - Power and Losses | Low-voltage Results | Storage State Variables

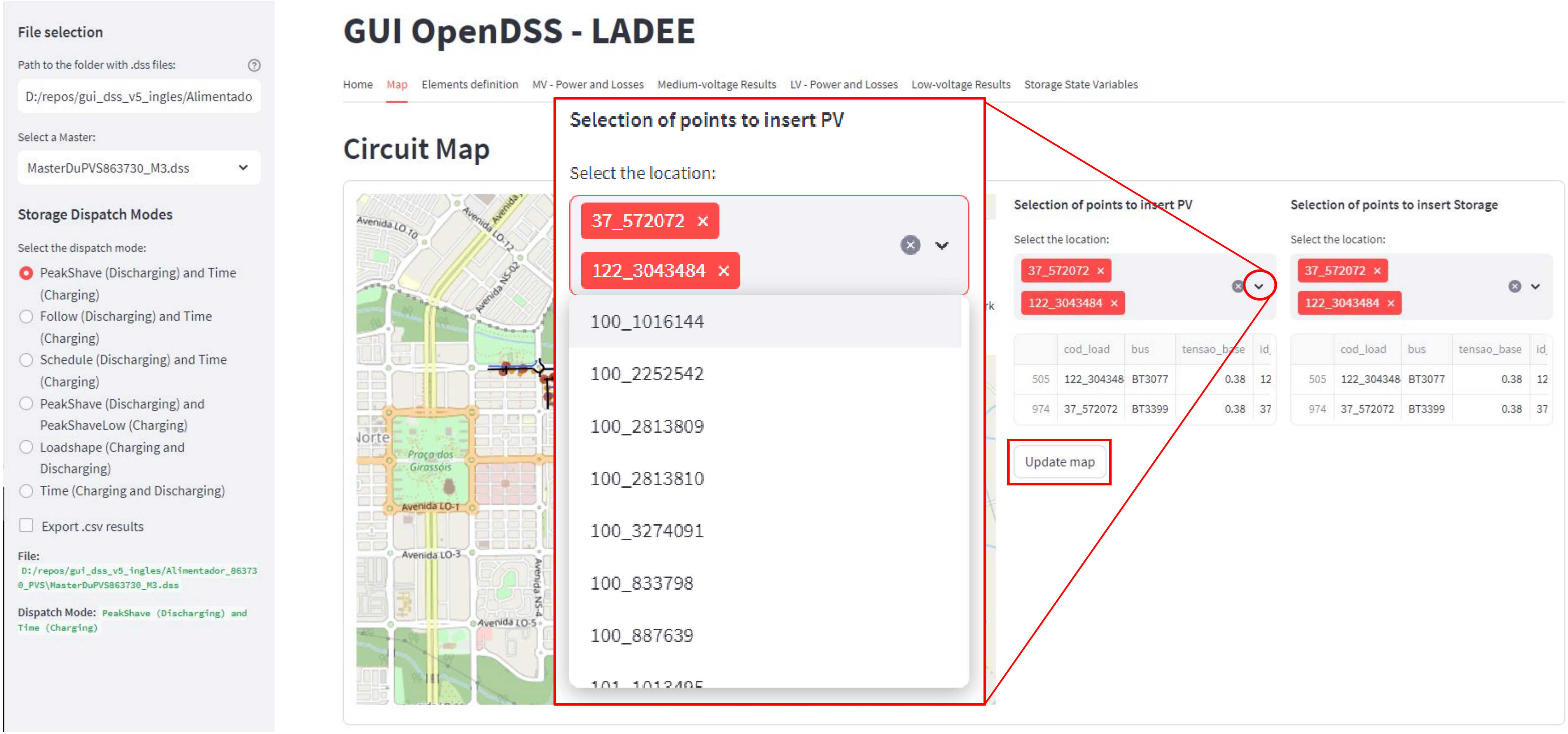
Responsible: Ana Camila Mamede

Contact: anacamilamamede@gmail.com

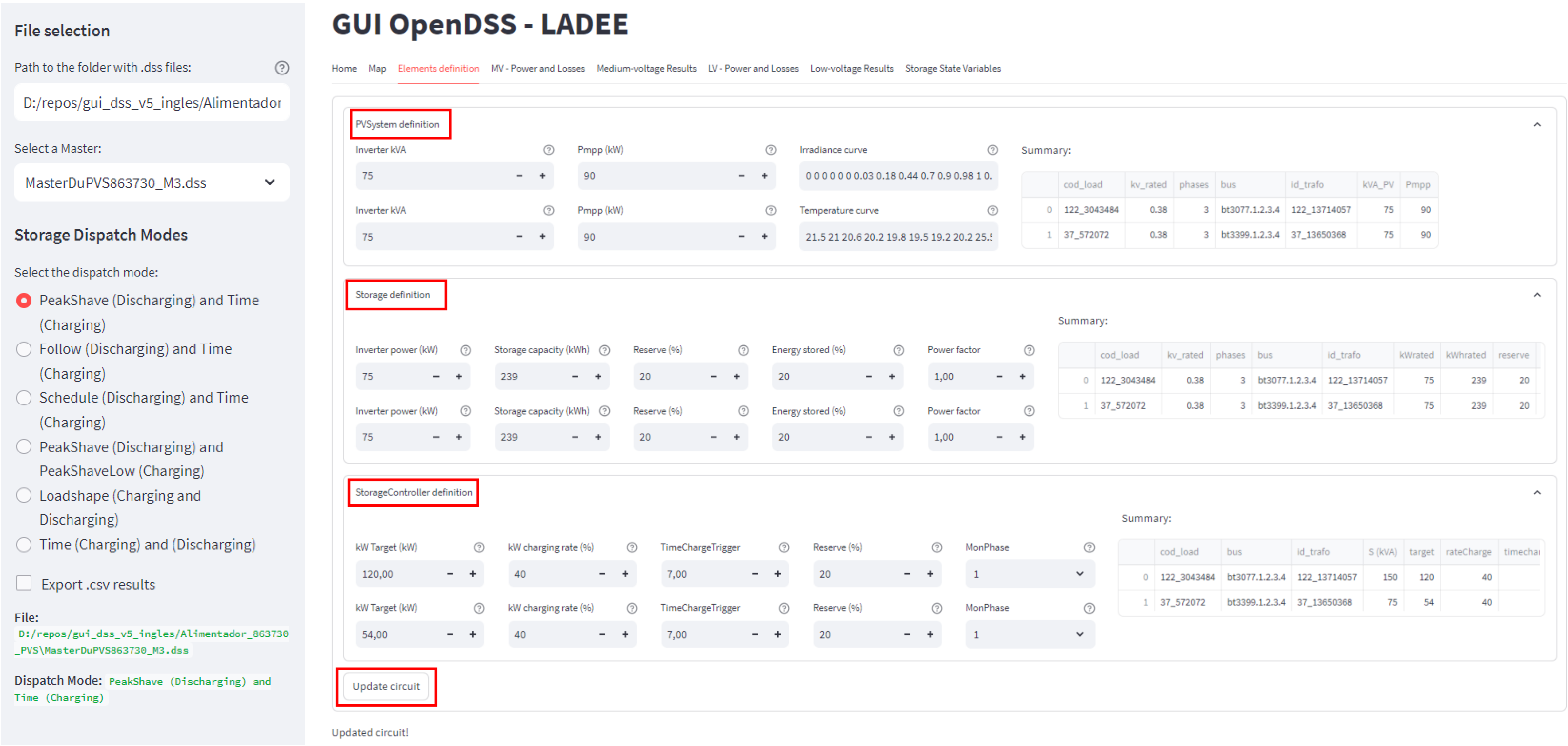
Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks



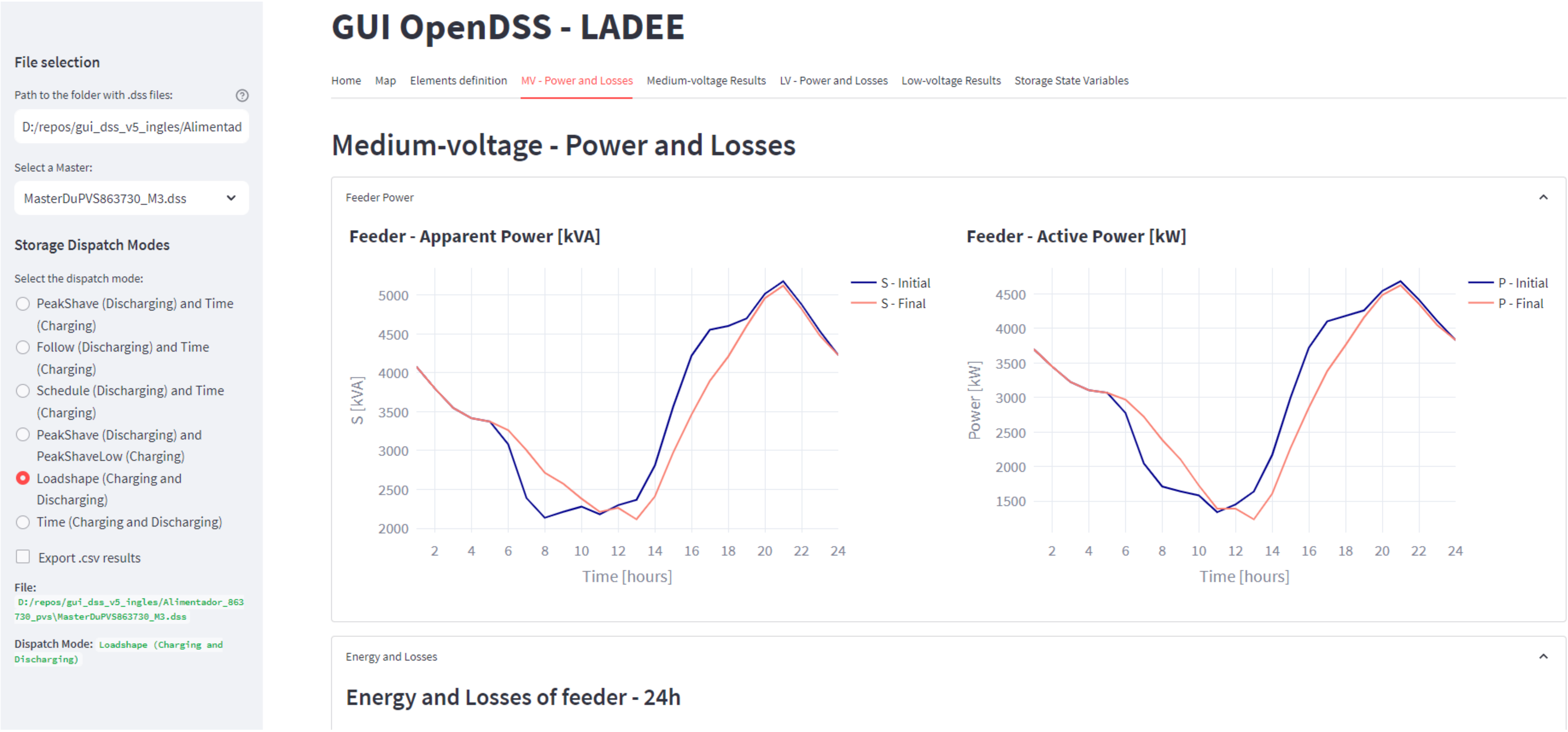
Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks



Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks



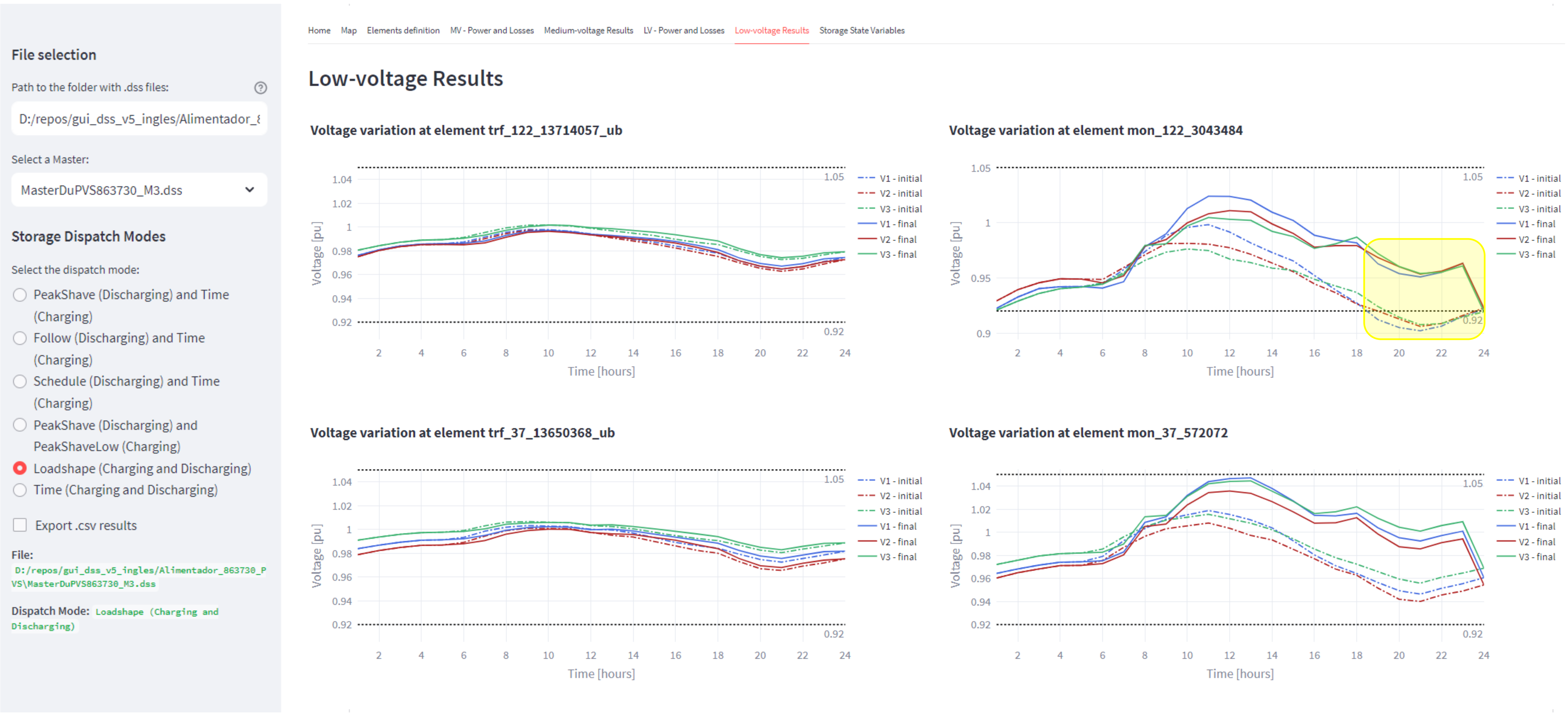
Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks



Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks

Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks

Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks



Development of a graphical interface to analyzing the impact of distributed generation and energy storage systems on distribution networks

Thank You

anacamilamamede@gmail.com

Professor Dr. Zhaoyu Wang

Zhaoyu Wang received the B.S. and M.S. degrees in electrical engineering from Shanghai Jiao Tong University, and the M.S. and Ph.D. degrees in electrical and computer engineering from Georgia Institute of Technology. He is the Northrop Grumman Endowed Associate Professor with Iowa State University. His research interests include optimization and data analytics in power distribution systems and microgrids. He was the recipient of the National Science Foundation CAREER Award, the IEEE Power and Energy Society (PES) Outstanding Young Engineer Award, the Northrop Grumman Endowment, College of Engineering's Early Achievement in Research Award, and the Harpole-Pentair Young Faculty Award Endowment. He is the Technical Committee Program Chair (TCPC) of IEEE Power System Operation, Planning and Economics (PSOPE) Committee, the Chair of PSOPE Awards Subcommittee, the Vice Chair of IEEE Distribution System Operation and Planning Subcommittee.

09/13/2023

OpenDSS for Student Training

Dr. Zhaoyu Wang
Northrop Grumman Associate Professor

Outline

- Power Distribution Courses at Iowa State University
- OpenDSS Training Modules in Courses
- Senior Design Projects using OpenDSS
- Using OpenDSS to Anlayze Utility Data
- Branch Flow v.s. Bus Injection
- Research Inspired by OpenDSS – Transformer Modeling
- Iowa Time-Series Distribution Test System in OpenDSS

Distribution System Courses at ISU

Term	Course number	Course Title	Credits	Number of students
Fall 2023	EE653	Power Distribution System Protection, Outage Management, and Reliability	3	13
Spring 2023	EE455	Introduction to Energy Distribution Systems	3	34
Spring 2021	EE455	Introduction to Energy Distribution Systems	3	41
Spring 2020	EE455	Introduction to Energy Distribution Systems	3	42
Fall 2019	EE653	Power Distribution System Modeling, Optimization and Simulation	3	14
Spring 2019	EE555	Advanced Energy Distribution Systems	3	9
Fall 2018	EE455	Introduction to Energy Distribution Systems	3	32
Fall 2017	EE555	Advanced Energy Distribution Systems	3	12
Spring 2017	EE455	Introduction to Energy Distribution Systems	3	36

Using OpenDSS in ISU Courses

- We have offered OpenDSS tutorials at several undergraduate/graduate courses.
- These courses also include project assignments using OpenDSS.

Undergraduate Course

EE 455: Introduction to Energy Distribution Systems

- Introduction to OpenDSS
  - Introduce online resources
  - Guide for installations
  - Introduce user interface
- Modeling distribution circuits
- Simulating built-in test cases
- Time series power flow

Graduate Courses

EE 555: Advanced Energy Distribution Systems

EE 653: Power Distribution System Protection, Outage Management, and Reliability

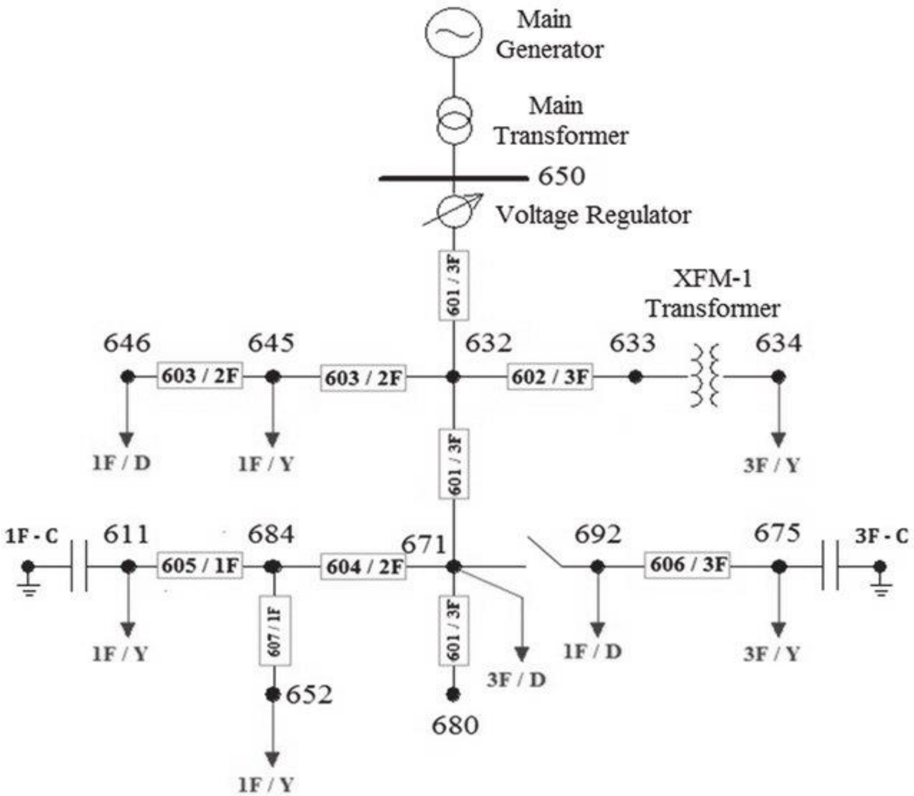
EE 653: Power Distribution System Modeling & Optimization

- Modeling of various distribution system components, such as line regulators, load tap changers, IBRs, and loads
- Details on the formation of primitive and network admittance matrix of the distribution system
- Detailed power flow algorithms used in OpenDSS
- Comparison of OpenDSS power flow with forward-backward algorithm
- Interfacing OpenDSS with MATLAB and Python
- Fault analysis
- Result visualization

Senior Design Projects using OpenDSS

Extracting admittance matrices of unbalanced distribution test systems using OpenDSS

- Building a Python program to extract the system admittance matrix using OpenDSS python API.



IEEE 13 bus system

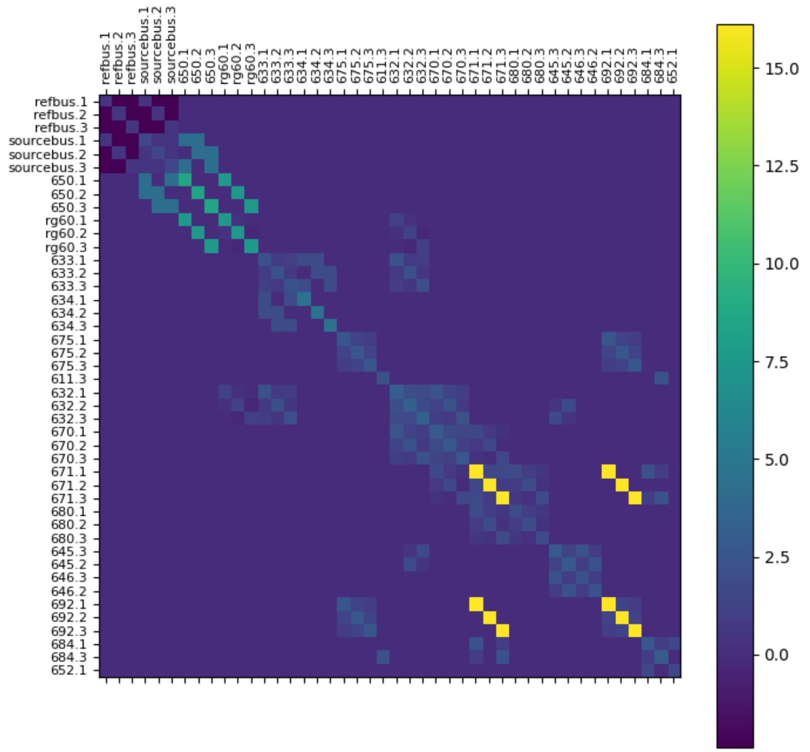
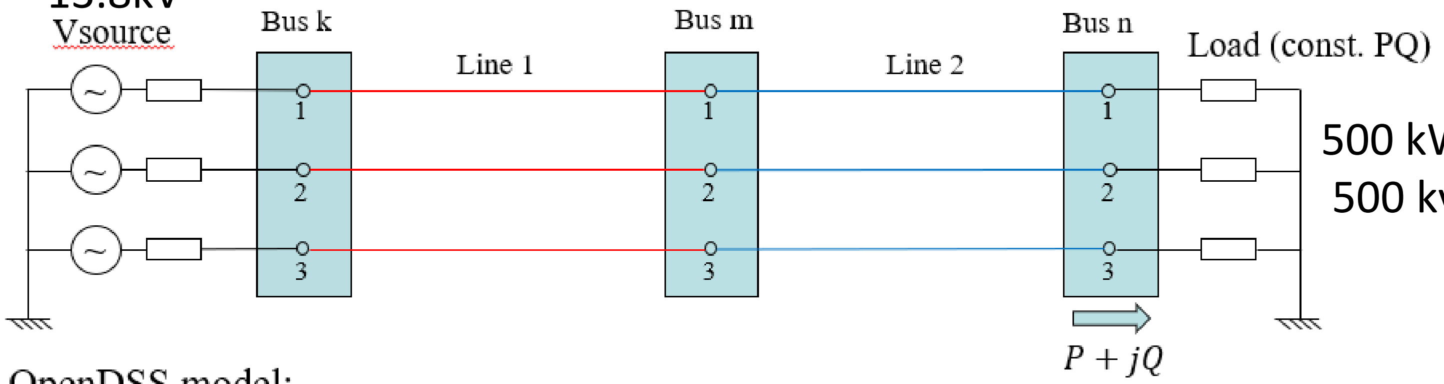


Fig.: Visualization of absolute value of Admittance matrix of IEEE 13 bus system

Senior Design Projects using OpenDSS

Comparison of OpenDSS's power flow method with backward forward method



500 kW and
500 kvar

OpenDSS model:

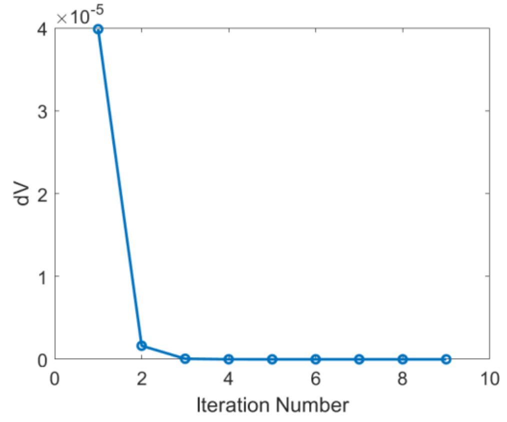
$$dV = \frac{||V_i| - |V_{i-1}||}{V_{nominal}}$$

	Real	Imag.	Abs.	p.u.
Vk1	8020.78	103.35	8021.4	1.01
Vk2	-3920.89	-6997.88	8021.4	1.01
Vk3	-4099.90	6894.53	8021.4	1.01
Vm1	7995.12	96.12	7995.7	1.00
Vm2	-3914.32	-6972.04	7995.7	1.00
Vm3	-4080.80	6875.92	7995.7	1.00
Vn1	7995.12	96.12	7960.5	0.99
Vn2	-3914.32	-6972.04	7960.5	0.99
Vn3	-4080.80	6875.92	7960.5	0.99

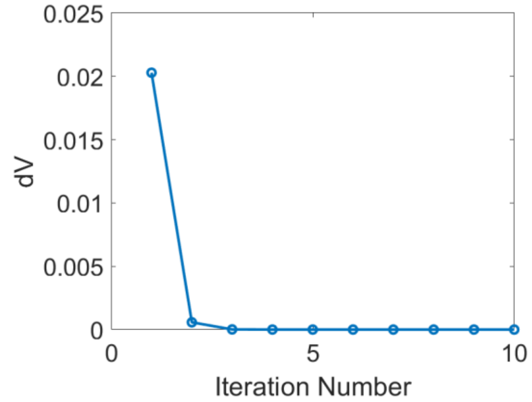
Computed Voltages (volts) by Matlab Code Using Backward-Forward Method

	Real	Imag.	Abs	p.u.
Vk1	8020.8	103.4	8021.4	1.01
Vk2	-3920.9	-6998.00	8021.4	1.01
Vk3	-4100.11	6894.40	8021.4	1.01
Vm1	7995.13	96.29	7995.7	1.00
Vm2	-3914.18	-6972.13	7995.7	1.00
Vm3	-4080.95	6875.84	7995.7	1.00
Vn1	7959.91	100.03	7960.5	0.99
Vn2	-3893.33	-6943.50	7960.5	0.99
Vn3	-4066.59	6843.47	7960.5	0.99

Computed Voltages (volts) by Matlab code using OpenDSS's method



MATLAB's code using OpenDSS's Method



MATLAB code using Backward-Forward Method

Senior Design projects using OpenDSS

BESS sizing for peak load reduction in a utility's feeder

- Build a utility feeder model in OpenDSS.
- Integrated AMI data to OpenDSS model for time-series power flow.
- A BESS is placed at the feeder head for peak load reduction in coordination with distributed wind generation.
- The BESS's discharge power rate is set to be 0.5, and its capacity is set to vary from 100 kWh to 1,000 kWh. The peak load reduction performance on the peak day of every month in one year is evaluated.
- The percentages of peak power reduction of the peak day in each month are summarized in Fig (b). With increased BESS capacity, the peak load reduction performance improves.

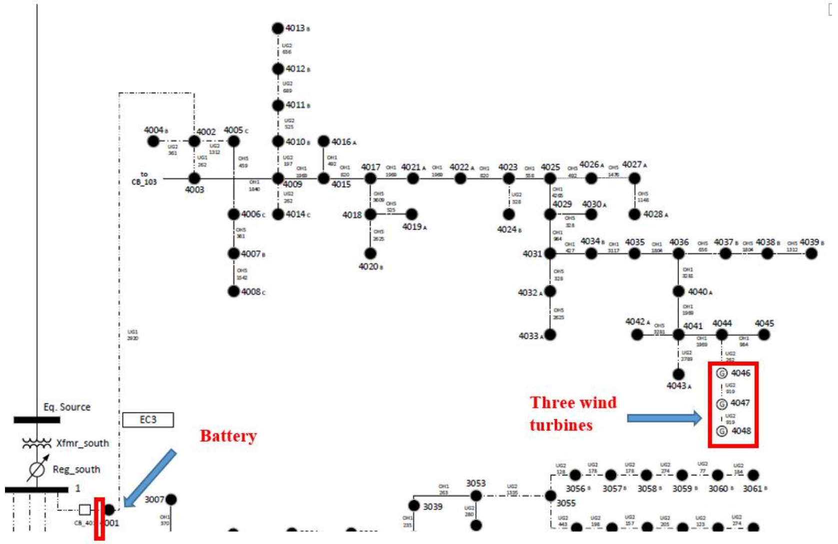


Fig (a). Topology of a real distribution feeder with wind turbines.

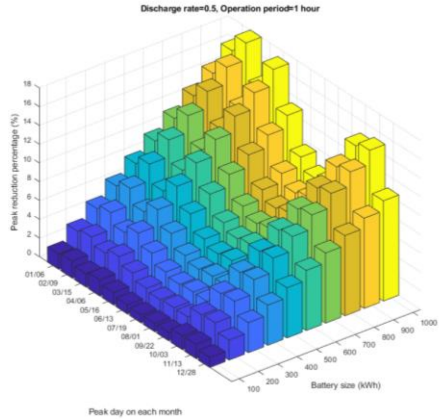
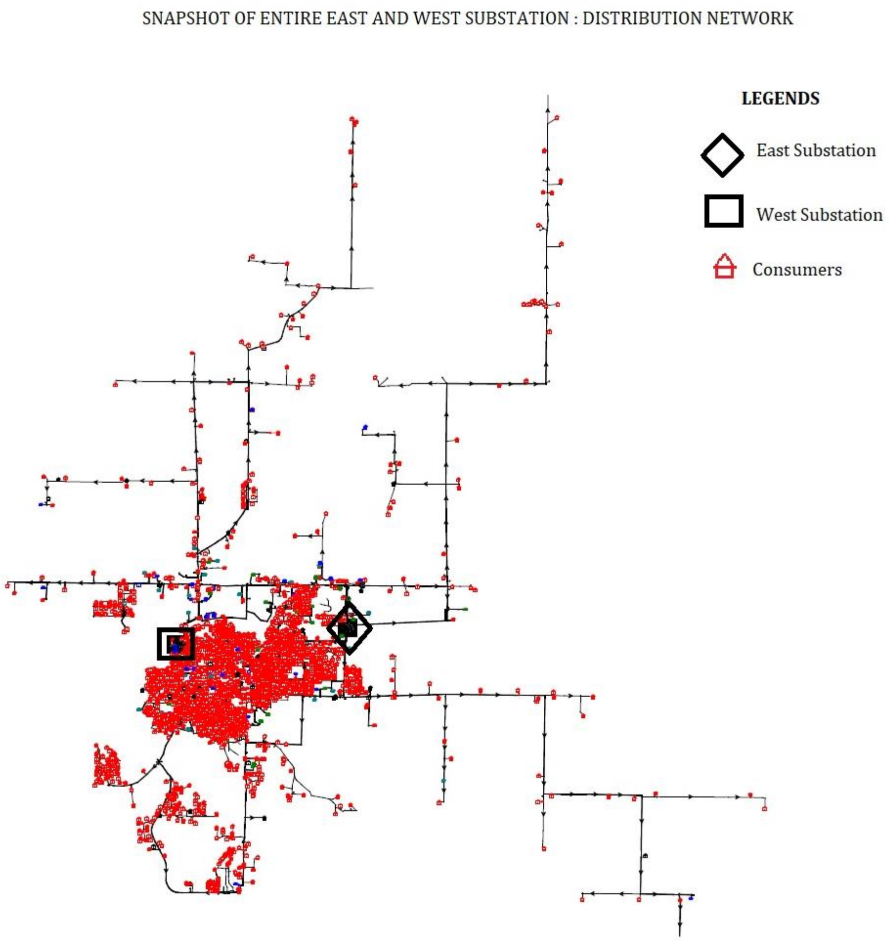


Fig (b). Peak power reduction percentages with different battery sizes during peak load hour with discharge power rate 0.5

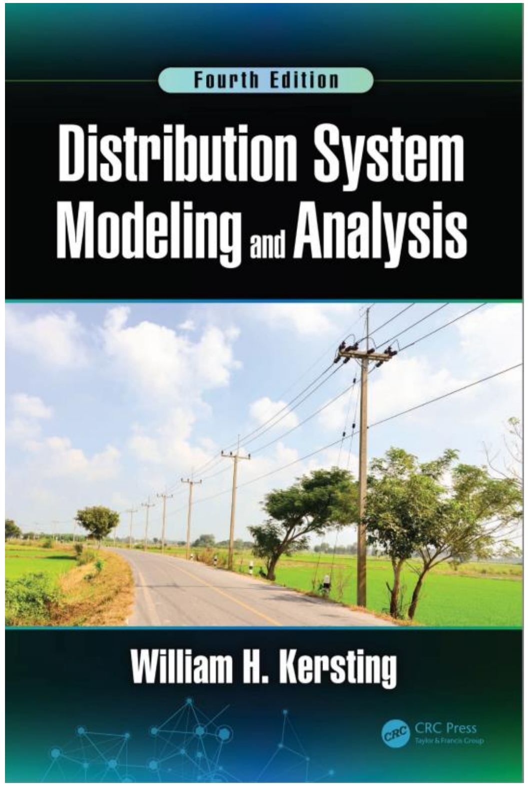
Using OpenDSS to Analyze Utility Data

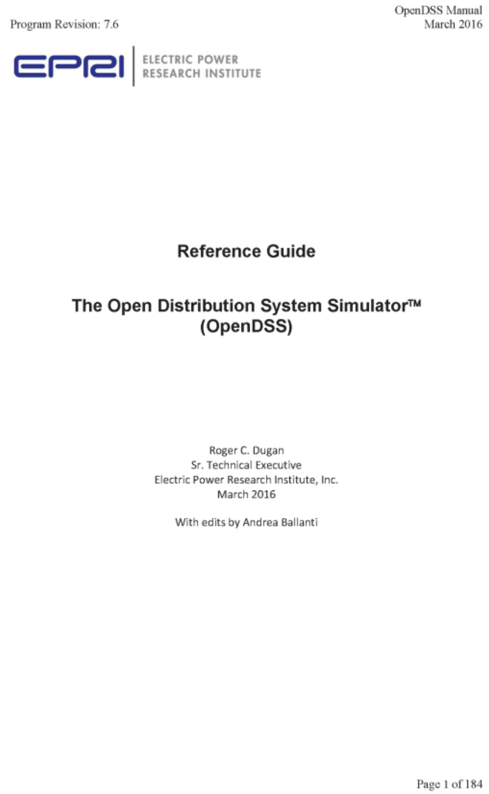
Migrating a utility's distribution system from Milsoft to OpenDSS



	East Substation	West Substation
North Transformer Details:		
Nominal Voltage of Transformer	69.0 kV	69.0 kV
Capacity	10000 kVA	10000 kVA
Windings	2	2
Primary Winding Rated Voltage	69.0 kV	69.0 kV
Secondary Winding Rated Voltage	13.8 kV	13.8 kV
Connection Type:		
Winding 1	Delta	Delta
Winding 2	Wye- Grounded	Wye- Grounded
%R	0.75602	0.74426
%X	7.5602	7.44260
Feeders Connected	5 (EB2, EB3, EB4, EB5, EB6)	2 (B2, B3)
South Transformer Details:		
Nominal Voltage of Transformer	69.0 kV	69.0 kV
Capacity	10000 kVA	10000 kVA
Windings	2	2
Primary Winding Rated Voltage	69.0 kV	69.0 kV
Secondary Winding Rated Voltage	13.8 kV	13.8 kV
Connection Type:		
Winding 1	Delta	Delta
Winding 2	Wye- Grounded	Wye- Grounded
%R	0.62659	0.54235
%X	6.26591	5.42350
Feeders Connected	4 (EC2, EC3, EC4, EC5)	3 (C3, C4, C5)

Power Flow Analysis – Branch Flow and Bus Injection




Program Revision: 7.6

OpenDSS Manual
March 2016

EPRI ELECTRIC POWER RESEARCH INSTITUTE

Reference Guide

The Open Distribution System Simulator™ (OpenDSS)

Roger C. Dugan
Sr. Technical Executive
Electric Power Research Institute, Inc.
March 2016

With edits by Andrea Ballanti

Page 1 of 184

- In Kersting's book, the branch flow model (BFM) is used for power flow calculation.
- OpenDSS uses the bus injection model (BIM) for power flow analysis.

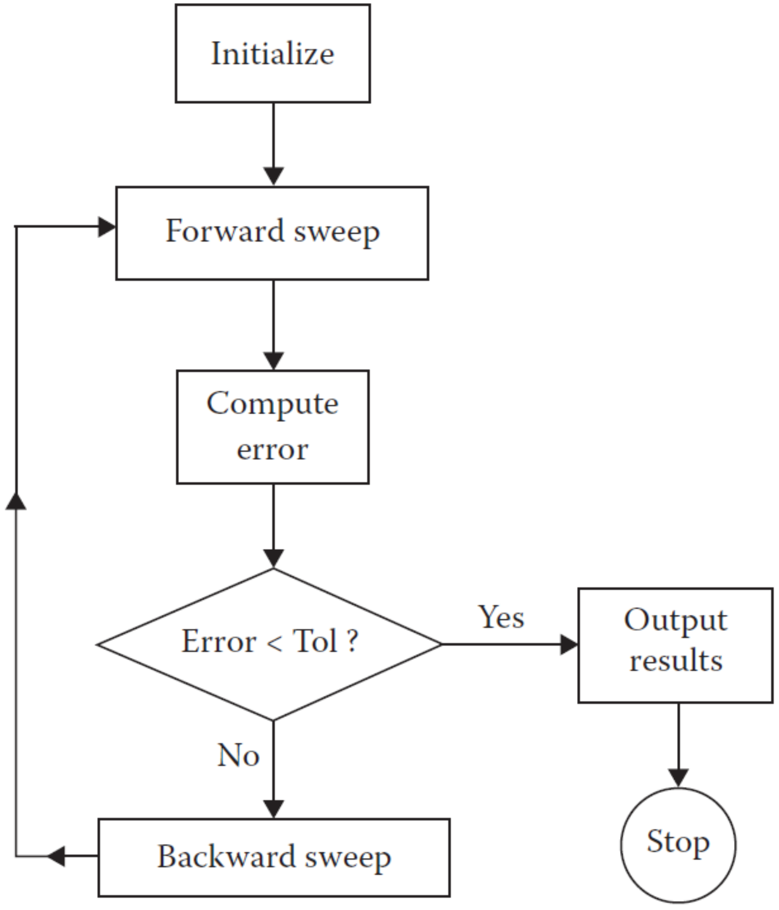
Power Flow Analysis – Branch Flow & Bus Injection

Branch Flow Method

- Step 1: Model series components in the following format:

$$[VLN_{abc}]_m = [A] \cdot [VLN_{abc}]_n - [B] \cdot [I_{abc}]_n \quad (Eq.1)$$
$$[I_{abc}]_n = [c] \cdot [VLN_{abc}]_m + [d] \cdot [I_{abc}]_m \quad (Eq.2)$$

- Step 2: Set the load currents [*Iabc*] to be zero and run the first forward sweep by applying *Eq.1.*
- Step 3: Update the load/shunt device currents using the most recent voltages.
- Step 4: Compute the currents from the load back to the source using the most recently computed voltages from the forward sweep by applying *Eq.2.*
- Step 5: Compute the downstream voltages from the source by applying *Eq.1.*
- Step 6: Check whether the voltage errors are within the tolerance. If not, go to Step 3.



Flow chart for the forward-backward sweep method.

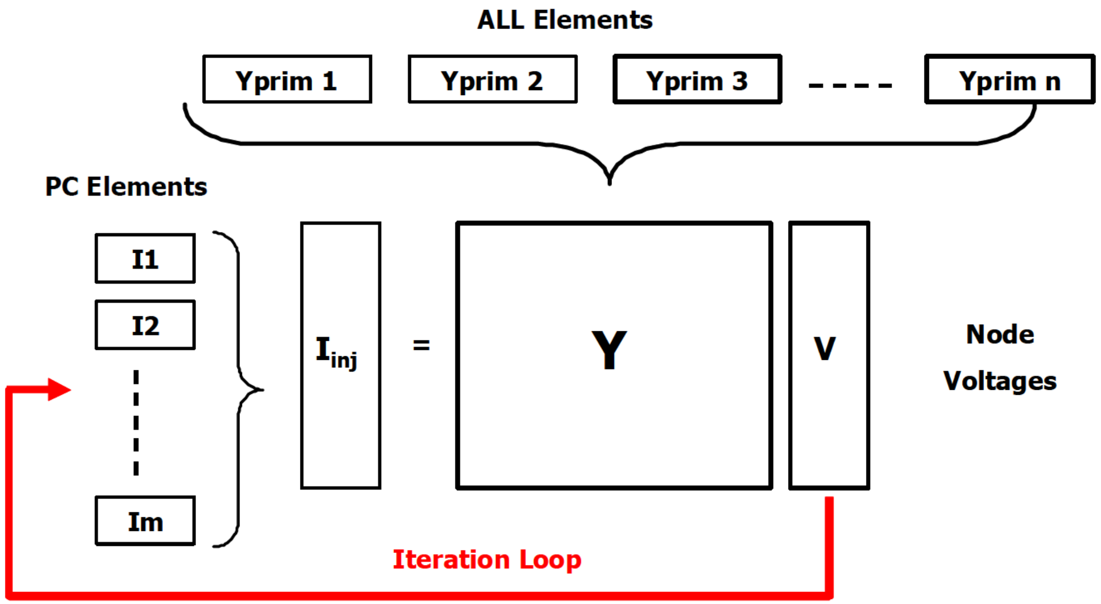
Power Flow Analysis – Branch Flow & Bus Injection

Bus Injection Method - OpenDSS

- Step 1: Represent Each element of the system by a "primitive" nodal admittance (Y) matrix and then coalesce into one large system Y matrix.
- Step 2:Calculate the injection currents of the Vsource:

$$\boldsymbol{I}_{source} = \boldsymbol{Y}_{source} * \boldsymbol{V}_{source}$$

- Step 3: Calculate the initial nodal voltages using $\boldsymbol{I}_{source}$ with a other injections currents set to 0: $\mathbf{V}_0 = \mathbf{Y}^{-1} * \mathbf{I}_{\mathrm{inj},0}$
- Step 4: Calculate the injection (compensation) currents for all t power conversion (PC) elements to obtain $\mathbf{I}_{\mathrm{inj,n}}$, where, n den the n^{th} iteration.
- Step 5: Update voltage vector: $\mathbf{V}_{n+1} = \mathbf{Y}^{-1} * \mathbf{I}_{\mathrm{inj},n}$
- Step 6: Check whether the voltage errors are within the tolerance. If not, go to Step 4.



Flow chart for bus injection method in OpenDSS.

Three-Phase Transformer Models

- In Kersting's book, the modeling of three-phase transformers follows

$$[VLN_{abc}] = [A_t] \cdot [VLN_{ABC}] - [B_t] \cdot [I_{abc}] \ \ (Eq.3)$$
$$[VLN_{ABC}] = [a_t] \cdot [VLN_{abc}] + [b_t] \cdot [I_{abc}] \ \ (Eq.4)$$
$$[I_{ABC}] = [c_t] \cdot [VLN_{abc}] + [d_t] \cdot [I_{abc}] \ \ (Eq.5)$$

- $Eq.3$ is used in the forward sweep to compute new node voltages downstream from the source using the most recent line currents. In the backward sweep, $Eq.5$ is used to compute the source-side line currents using the newly computed load-side line currents.
- Different transformer connection types have different generalized matrices.

	Δ – Grounded Y Step-down	Δ – Grounded Y Step-up	Ungrounded Y – Δ Step-down	Grounded Y – Grounded Y
$[a_t]$	$-\frac{n_t}{3}\begin{bmatrix}0 & 2 & 1\\1 & 0 & 2\\2 & 1 & 0\end{bmatrix}$	$\frac{n_t}{3}\begin{bmatrix}2 & 1 & 0\\0 & 2 & 1\\1 & 0 & 2\end{bmatrix}$	$n_t\begin{bmatrix}1 & -1 & 0\\0 & 1 & -1\\-1 & 0 & 1\end{bmatrix}$	$\begin{bmatrix}n_t & 0 & 0\\0 & n_t & 0\\0 & 0 & n_t\end{bmatrix}$
$[b_t]$	$-\frac{n_t}{3}\begin{bmatrix}0 & 2Z_{t_b} & Z_{t_c}\\Z_{t_a} & 0 & 2Z_{t_c}\\2Z_{t_a} & Z_{t_b} & 0\end{bmatrix}$	$\frac{n_t}{3}\begin{bmatrix}2Z_{t_a} & Z_{t_b} & 0\\0 & 2Z_{t_b} & Z_{t_c}\\Z_{t_a} & 0 & 2Z_{t_c}\end{bmatrix}$	$\frac{n_t}{3}\begin{bmatrix}Z_{t_{ab}} & -Z_{t_{ab}} & 0\\Z_{t_{bc}} & 2Z_{t_{bc}} & 0\\-2Z_{t_{ca}} & Z_{t_{ca}} & 0\end{bmatrix}$	$\begin{bmatrix}n_tZ_{t_a} & 0 & 0\\0 & n_tZ_{t_b} & 0\\0 & 0 & n_tZ_{t_c}\end{bmatrix}$
$[c_t]$	$\begin{bmatrix}0 & 0 & 0\\0 & 0 & 0\\0 & 0 & 0\end{bmatrix}$	$\begin{bmatrix}0 & 0 & 0\\0 & 0 & 0\\0 & 0 & 0\end{bmatrix}$	$\begin{bmatrix}0 & 0 & 0\\0 & 0 & 0\\0 & 0 & 0\end{bmatrix}$	$\begin{bmatrix}0 & 0 & 0\\0 & 0 & 0\\0 & 0 & 0\end{bmatrix}$
$[d_t]$	$\frac{1}{n_t}\begin{bmatrix}1 & -1 & 0\\0 & 1 & -1\\-1 & 0 & 1\end{bmatrix}$	$\frac{1}{n_t}\begin{bmatrix}1 & 0 & -1\\0 & 1 & -1\\-1 & 0 & 1\end{bmatrix}$	$\frac{1}{3n_t}\begin{bmatrix}1 & -1 & 0\\1 & 2 & 0\\-2 & -1 & 0\end{bmatrix}$	$\frac{1}{n_t}\begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}$
$[A_t]$	$\frac{1}{n_t}\begin{bmatrix}1 & 0 & -1\\-1 & 1 & 0\\0 & -1 & 1\end{bmatrix}$	$\frac{1}{n_t}\begin{bmatrix}1 & 0 & -1\\0 & 1 & -1\\-1 & 0 & 1\end{bmatrix}$	$\frac{1}{3n_t}\begin{bmatrix}2 & 1 & 0\\0 & 2 & 1\\1 & 0 & 2\end{bmatrix}$	$\frac{1}{n_t}\begin{bmatrix}1 & 0 & 0\\0 & 1 & 0\\0 & 0 & 1\end{bmatrix}$
$[B_t]$	$\begin{bmatrix}Z_{t_a} & 0 & 0\\0 & Z_{t_b} & 0\\0 & 0 & Z_{t_c}\end{bmatrix}$	$[Z_{t_{abc}}]$	$\frac{1}{9}\begin{bmatrix}2Z_{t_{ab}} + Z_{t_{bc}} & 2Z_{t_{bc}} - 2Z_{t_{ab}} & 0\\2Z_{t_{bc}} - 2Z_{t_{ca}} & 4Z_{t_{bc}} - Z_{t_{ca}} & 0\\Z_{t_{ab}} - 4Z_{t_{ca}} & -Z_{t_{ab}} - 2Z_{t_{ca}} & 0\end{bmatrix}$	$[Z_{t_{abc}}]$
n_t	$\frac{VLL_{rated\ primary}}{VLN_{rated\ secondary}}$	$\frac{VLL_{rated\ primary}}{VLN_{rated\ secondary}}$	$\frac{VLN_{rated\ primary}}{VLL_{rated\ secondary}}$	$\frac{VLN_{rated\ primary}}{VLN_{rated\ secondary}}$

- In OpenDSS, a more generalized method is used to represent *Yprim* matrices for three-phase transformers of various connection types:

$$\boldsymbol{Y}_{prim} = \boldsymbol{ANBY}_{sc(1V)}\,\boldsymbol{B}^{\mathrm{T}}\boldsymbol{N}^{T}\,\boldsymbol{A}^{T}$$

where:

  - Matrix *A* is determined by the winding connection.
  - Matrix *N* is related to the windings' rated voltages.
  - Matrix *B* is a constant matrix, given by:

$$\boldsymbol{B} = \begin{bmatrix}1 & 0 & 0\\-1 & 0 & 0\\0 & 1 & 0\\0 & -1 & 0\\0 & 0 & 1\\0 & 0 & -1\end{bmatrix}$$

  - $\boldsymbol{Y}_{sc(1V)}$ represents the nodal admittance matrix of the transformer on one voltage base, with the format of:

$$\boldsymbol{Y}_{sc(1V)} = \begin{bmatrix}y_{Aa(1V)} & 0 & 0\\0 & y_{Bb(1V)} & 0\\0 & 0 & y_{Cc(1V)}\end{bmatrix}$$

$$y_{Aa(1V)} = y_{Bb(1V)} = y_{Cc(1V)} = \frac{1}{[r(pu) + jx(pu)] * \frac{1^2}{S_t/3}}$$

Sum of windings' percent resistance

Percent reactance high-to-low

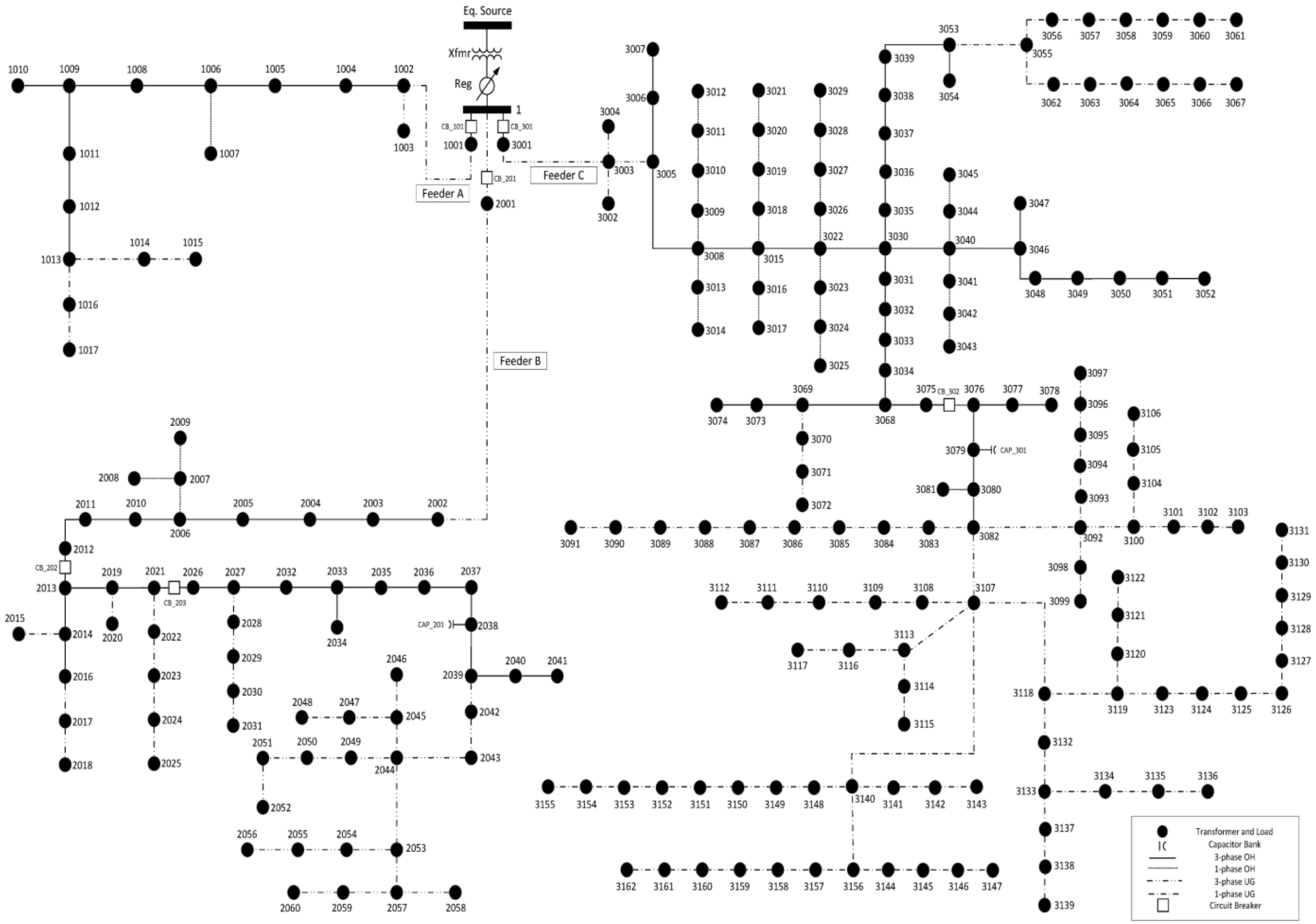
- The transformer's *Yprim* can be derived in a more generalized way, based on the short-circuit impedance of the pair of windings, their voltage ratings, and types of connection.

Iowa Time-Series Distribution Test System

With permission from our utility partner, we share a real distribution grid model with one-year smart meter measurements. This dataset provides an opportunity for researchers and engineers to perform validation and demonstration using real utility grid models and field measurements.

- The system consists of 3 feeders and 240 nodes and is located in the Midwest U.S.
- The system has 1120 customers (i.e., commercial and residential customers), and all of them are equipped with smart meters. These smart meters measure hourly energy consumption (kWh). We share the one-year real smart meter measurements for 2017.
- The system has standard electric components such as overhead lines, underground cables, substation transformers with LTC, line switches, capacitor banks, and secondary distribution transformers. The real system topology and component parameters are included.
- You may download the dataset at: http://wzy.ece.iastate.edu/Testsystem.html , including system description (in .doc and .xlsx), smart meter data (in .xlsx), OpenDSS model, and Matlab code for quasi-static time-series simulation.

Iowa Time-Series Distribution Test System

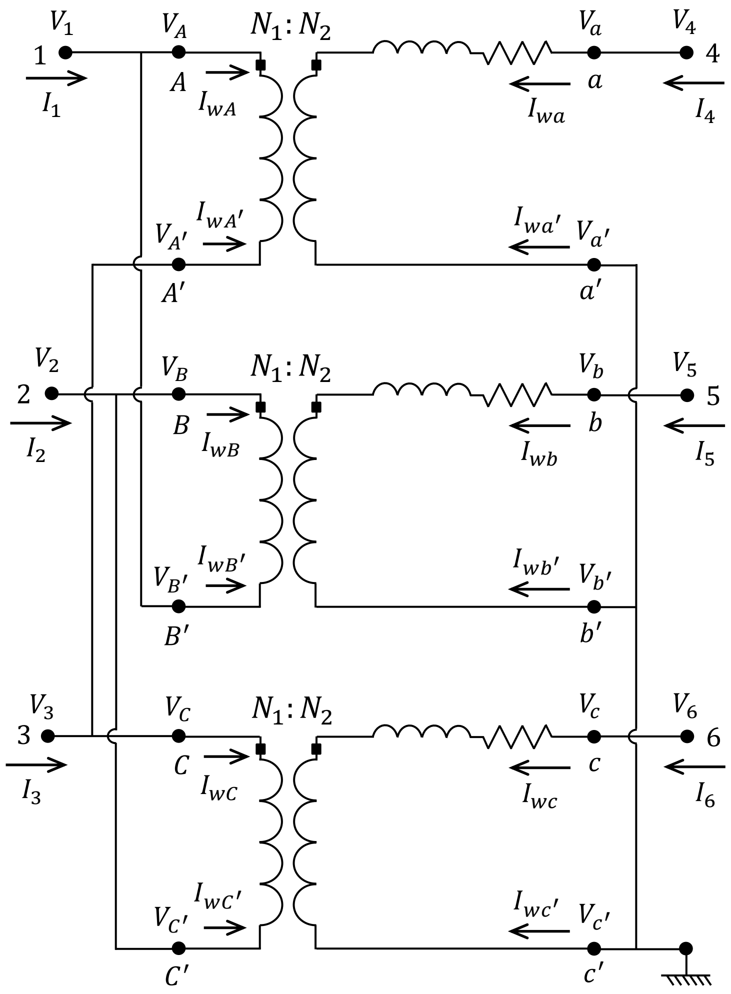


F. Bu, Y. Yuan, Z. Wang, K. Dehghanpour, and A. Kimber, "A Time-series Distribution Test System based on Real Utility Data." 2019 North American Power Symposium (NAPS), Wichita, KS, USA, 2019, pp. 1-6.

Thank You!
Q&A

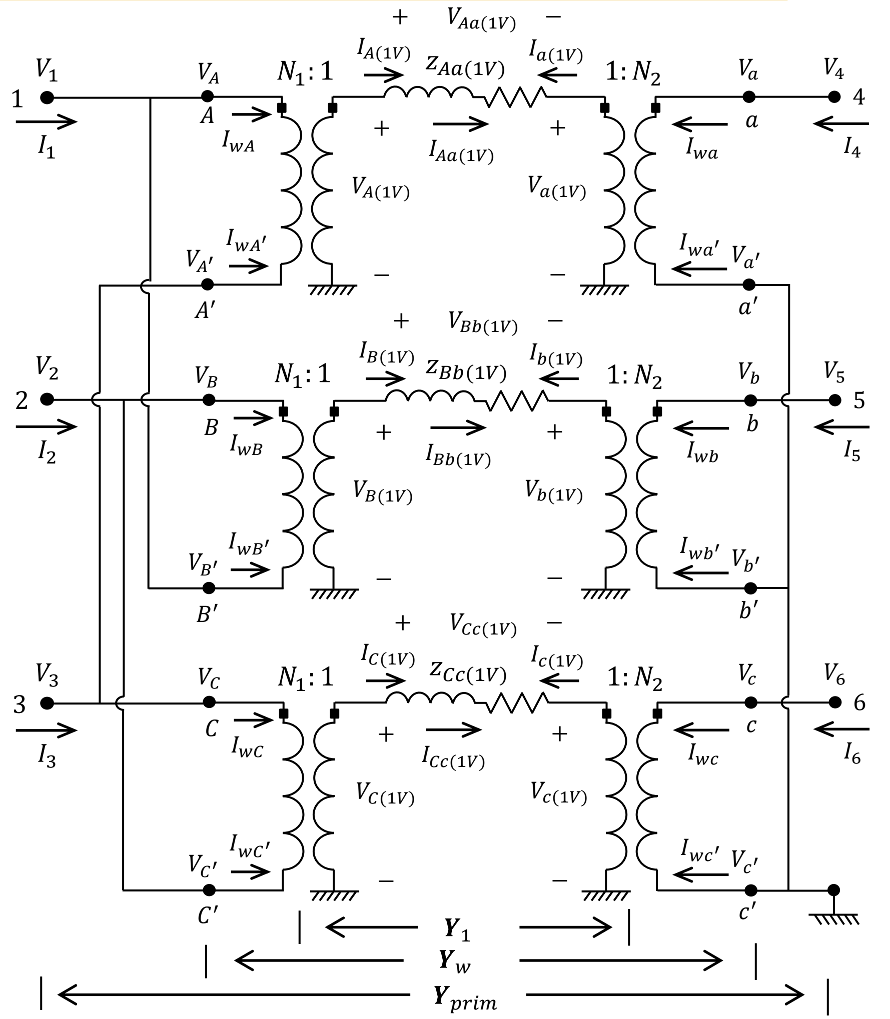
IOWA STATE UNIVERSITY

Three-Phase Transformer Models: Backup Slides



Delta–Grounded Wye Transformer with Dyn1 Connection

Consider the short-circuit impedance in a one voltage base:



Three-Phase Transformer Models: Backup Slides

Example:

// Define a three-phase transformer.

Transformer name | Number of phases | Number of windings | Percent reactance high-to-low

```
New Transformer.xfmr phases=3 windings=2 xhl=5
~ wdg=1 bus=K.1.2.3    conn=delta kV=115  kva=500 %r=1
~ wdg=2 bus=M.1.2.3.0  conn=wye   kV=4.16 kva=500 %r=1
```

Specify which winding will be edited | Winding connection | Rated voltage | Base kVA rating | Percent resistance

- *A* is determined by the winding connection. For the Delta-Grounded Wye Step-down transformer with Dyn1 connection, we have:

$$\boldsymbol{A}=\begin{bmatrix}1&0&0&0&0&1&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0&1&0&0\\0&1&0&0&0&0&0&0&1&0&0&0\\0&0&1&0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0&0&0&1&0\end{bmatrix}$$

- Matrix *B* is a constant matrix, given by:

$$\boldsymbol{B}=\begin{bmatrix}1&0&0\\-1&0&0\\0&1&0\\0&-1&0\\0&0&1\\0&0&-1\end{bmatrix}$$

- We have $\boldsymbol{Y}_{sc(1V)}=\begin{bmatrix}y_{Aa(1V)}&0&0\\0&y_{Bb(1V)}&0\\0&0&y_{Cc(1V)}\end{bmatrix}$

where

$$y_{Aa(1V)}=y_{Bb(1V)}=y_{Cc(1V)}=\frac{1}{[r(pu)+jx(pu)]*\frac{1^2}{S_t/3}}=\frac{500000}{3*(0.02+j0.05)}=$$

$$1149425.2873\ -j2873563.2184\ S$$

- *N* is related to the winding turn ratio:

$$\boldsymbol{N}^T=\begin{bmatrix}\frac{1}{N_1}&-\frac{1}{N_1}&0&0&0&0&0&0&0&0&0&0\\0&0&\frac{1}{N_2}&-\frac{1}{N_2}&0&0&0&0&0&0&0&0\\0&0&0&0&\frac{1}{N_1}&-\frac{1}{N_1}&0&0&0&0&0&0\\0&0&0&0&0&0&\frac{1}{N_2}&-\frac{1}{N_2}&0&0&0&0\\0&0&0&0&0&0&0&0&\frac{1}{N_1}&-\frac{1}{N_1}&0&0\\0&0&0&0&0&0&0&0&0&0&\frac{1}{N_2}&-\frac{1}{N_2}\end{bmatrix}$$

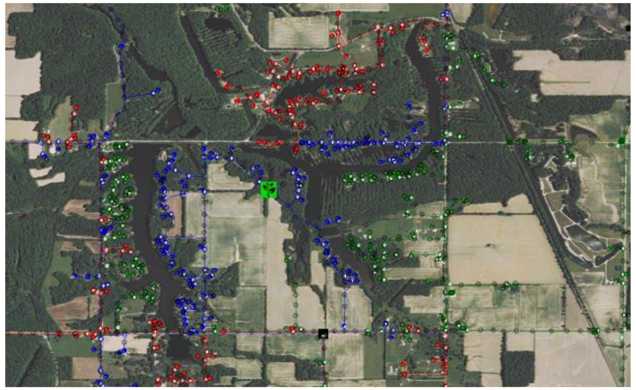
where N_1=115000, N_2= $4160/\sqrt{3}$

(Note: For a delta-connected winding, $N=VLL$; for a Y-connected winding, $N=\frac{VLL}{\sqrt{3}}$)

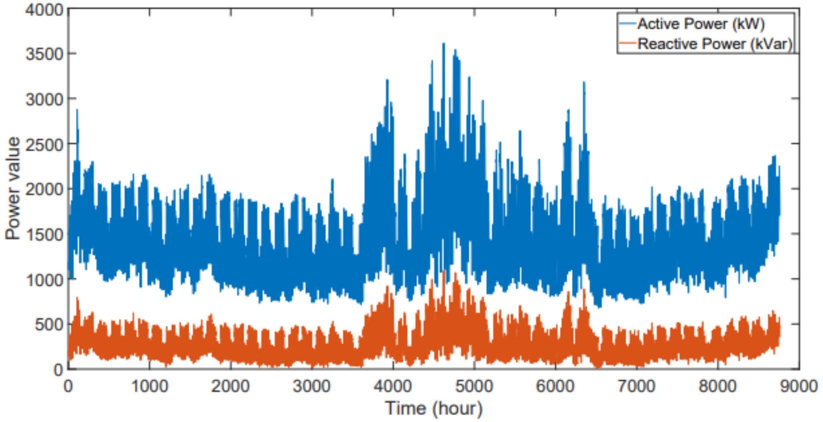
- Finally, we can calculate *Yprim*:

$$\boldsymbol{Y}_{prim}=\boldsymbol{ANBY}_{sc(1V)}\,\boldsymbol{B}^{\mathrm{T}}\boldsymbol{N}^{T}\,\boldsymbol{A}^{T}$$

OpenDSS: Connecting Industry Data and Research



A real distribution system in Milsoft



Yearly load profile

Real distribution system data usually includes:

- System Information (GIS information, device information).
- Raw AMI Data (customer load information, DER information).

OpenDSS enables us to integrate the information for further analysis (e.g., power flow analysis, DER integration, peak shaving).

Together…Shaping the Future of Energy®