

# HiPAS GridLAB-D

Software Implementation Presentation (May 2020)

EPC 17-043

GLOW

EPC 17-046

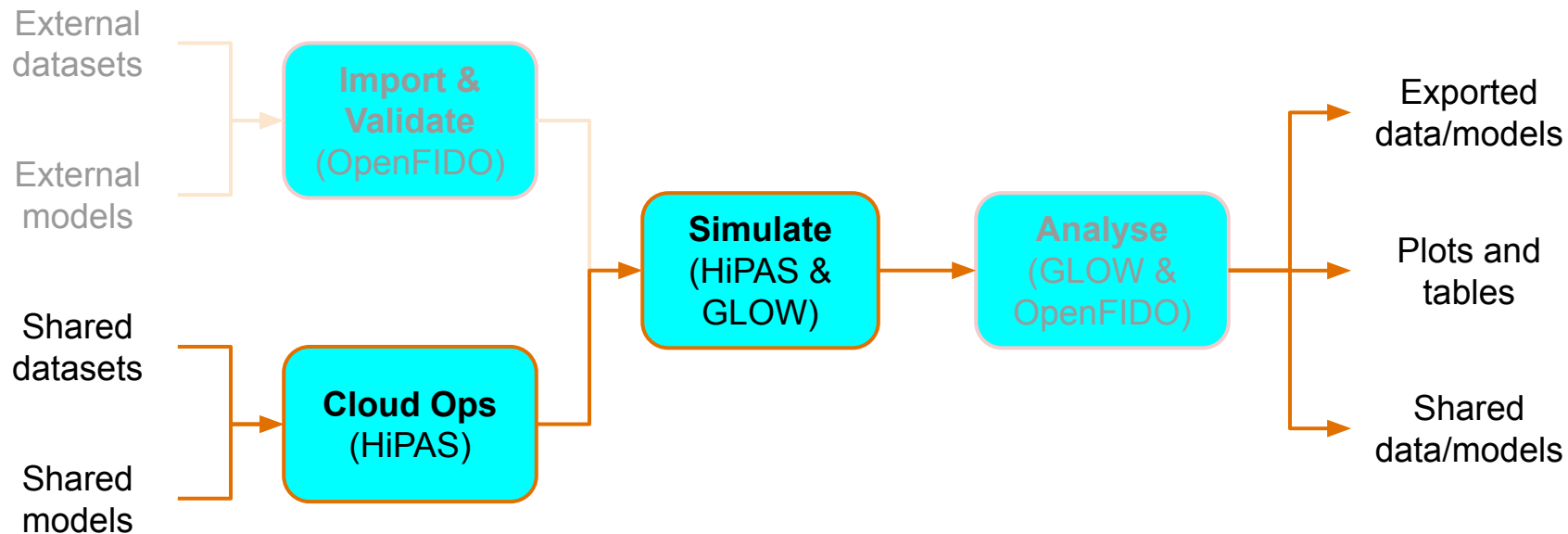
HiPAS

EPC 17-047

OpenFIDO

This presentation was prepared with funding from the California Energy Commission under grant EPC-17-046. SLAC National Accelerator Laboratory is operated for the US Department of Energy by Stanford University under Contract No. DE-AC02-76SF00515

# Project Focus Area



## Enhance GridLAB-D to support principle California use-cases

1. Integration Capacity Analysis - Integrate improvements from GLOW (CEC)
2. Resilience - Integrate improvements from GRIP (DOE)
3. Tariff design - Integrate improvements from Powernet With Market (CEC)
4. Electrification - Enable decarbonization simulations

# HiPAS Capabilities Currently in Development

## High-performance simulation

- Machine learning powerflow solver

## Improved data processing tools

- General file input/output (“any input, any output, anytime, anywhere”)
- High-performance database module (InfluxDB)

## New simulation modules/classes

- Industrial loads (NAICS facilities from NERC LMFT)
- Residential loads (RBSA data, multi-family residences)
- Commercial loads (CEUS data, retail/office/schools/health/etc. buildings)
- Revenue module (tariff and billing)

# Highlights: Cloud deployment

## Github deployment

- Project repository: <http://source.gridlabd.us/>
- Integrated online documentation at <http://docs.gridlabd.us/>

## Docker containers maintained/updated automatically

- Docker hub repository located at <http://docker.gridlabd.us/>

## CircleCI free tier operations

- GitHub projects can run HiPAS GridLAB-D on CircleCI for free
  - Latest master release: *slacgismo/gridlabd:latest*
  - Release candidate: *slacgismo/gridlabd:develop*
  - Tagged version: *slacgismo/gridlabd:beauharnois-03*

# Cloud operations: GitHub projects with GridLAB-D

The screenshot shows the GitHub interface for the repository 'dchassin / gridlabd-ci-test'. At the top, there's a navigation bar with links for Pull requests, Issues, Marketplace, and Explore. Below this, the repository name is displayed along with statistics: 2 Pull requests, 1 Star, 1 Fork, and 10 Insights. The main content area shows the repository title 'Test of the GridLAB-D modeling using CircleCI' and a list of files and folders. The files include .circleci, .gitattributes, .gitignore, config.glm, my\_test.glm, output.glm, and plotcsv.py. Each file has a commit message and a timestamp. At the bottom, there's a prompt to add a README file.

Search or jump to... Pull requests Issues Marketplace Explore

dchassin / gridlabd-ci-test

Unwatch 2 Star 1 Fork 10

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security 0 Insights Settings

Test of the GridLAB-D modeling using CircleCI Edit

Manage topics

62 commits 1 branch 0 packages 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

dchassin Delete output.glm	Latest commit 5024d8d on Sep 20, 2019
.circleci	Delete output.glm 8 months ago
.gitattributes	Initial commit 13 months ago
.gitignore	Merge branch 'initial-setup' 9 months ago
config.glm	Update config.glm 9 months ago
my_test.glm	Merge branch 'initial-setup' 9 months ago
output.glm	Update output.glm 9 months ago
plotcsv.py	Updated test 9 months ago

Help people interested in this repository understand your project by adding a README. Add a README

# Cloud operations: Running GridLAB-D in CircleCI

**dchassin**  
David P. Chassin

Pipelines

Add Projects

Organization Settings

Plan

Old Experience

?

 Help

 Display a menu

gridlabd-ci-test > master > workflow > build (65)

build

SUCCESS

Rerun

Duration 33s

Finished 5 days ago

Docker Medium

2 CPU / 4 GB RAM

5024d8d

master

David P. Chassin

STEPS

TESTS

ARTIFACTS

Parallel Run 0

root/project/.gitattributes

root/project/.gitignore

root/project/config.glm

root/project/my\_test.csv

root/project/my\_test.glm

root/project/my\_test.png

root/project/output.glm

root/project/plotcsv.py

root/project/.circleci/.DS\_Store

root/project/.circleci/config.yml

root/project/.git/FETCH\_HEAD

root/project/.git/HEAD

root/project/.git/ORIG\_HEAD

Air temperature [degF]

Indoor

2019-01-01 00:00:00 PST

2019-01-01 03:20:00 PST

2019-01-01 06:40:00 PST

2019-01-01 10:00:00 PST

2019-01-01 13:20:00 PST

2019-01-01 16:40:00 PST

2019-01-01 20:00:00 PST

2019-01-01 23:20:00 PST

73.0

72.5

72.0


71.5


71.0



Date/Time

# Results: Electrification Study



HectorTR

UpdatesSupport



Jobs » HectorTR » electrification\_study

By project

My branchesAll branches

My jobsAll jobs

> electrification_study	✓ SUCCESS	master #107 Sacramento year round simulation...	workflow build	24 days ago	1:27:48 3959dbb
> Gridworks_example	✓ SUCCESS	master #106 Riverside year round simulation...	workflow build	24 days ago	1:14:02 e2a7d5a
> tariff_design	✓ SUCCESS	master #105 Los Angeles year round simulati...	workflow build	24 days ago	58:50 b250246
> gridlabd-ci-test	✓ SUCCESS	master #104 China lake year round simulatio...	workflow build	24 days ago	1:45:13 8c3e17f
> ICA	✓ SUCCESS	master #103 Burbank year round simulation 2...	workflow build	24 days ago	1:13:10 1d7b368
> gridlabd-ci-test-1	✓ SUCCESS	master #102 San Jose year round simulation	workflow build	28 days ago	1:08:10 44f54de



# Next Steps: Deployment and operations infrastructure

- **Restructuring of GridLAB-D resources in GitHub**
  - Weather data: <http://weather.gridlabd.us/> (done)
  - Component libraries: <http://library.gridlabd.us/> (in progress)
  - Project templates: <http://template.gridlabd.us/> (coming soon)
  - Examples solution: <http://example.gridlabd.us/> (coming soon)
  - Training videos: <http://training.gridlabd.us/> (coming soon)
  - Tutorials videos: <http://tutorials.gridlabd.us/> (coming soon)
  - Online documentation: <http://docs.gridlabd.us/> (done)
  - Reference models: <http://models.gridlabd.us/> (in progress)
  - Docker images: <http://docker.gridlabd.us/> (done)
- **Security/scaling AWS CloudFront (https support)**

# Highlights: Machine Learning Powerflow

## Powerflow simulation in GridLAB-D

- 3-phase, unbalanced, quasi-steady power flow
- Finds voltages given power injections

## Standard approach

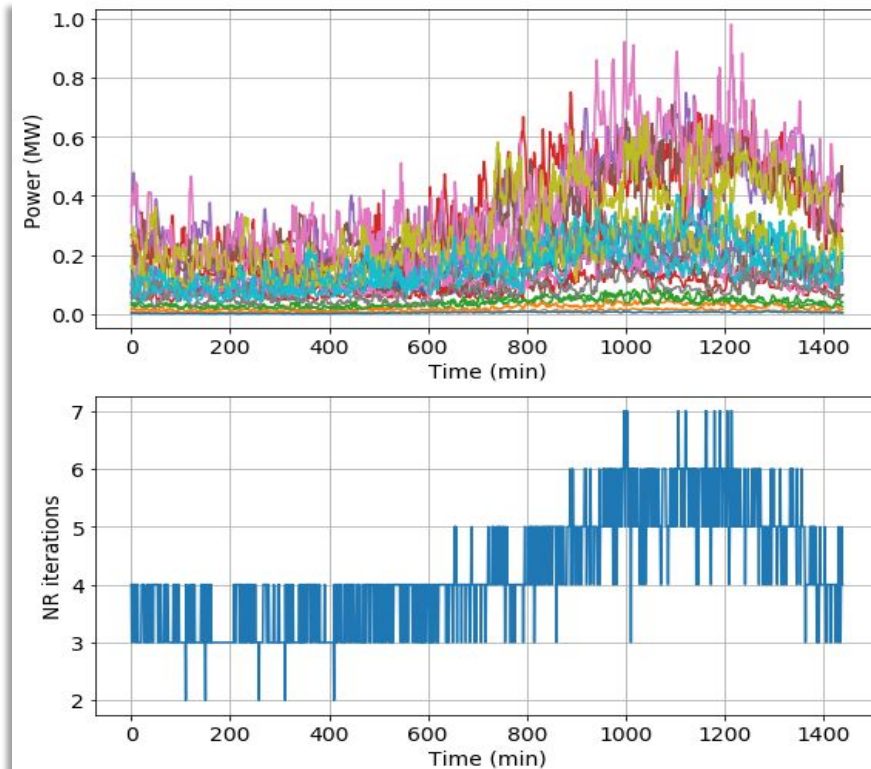
- Newton Raphson with the current injection method
- Computationally expensive for large networks

## Machine learning approach

- Convolutional neural network (CNN)
- Training using previous NR solutions

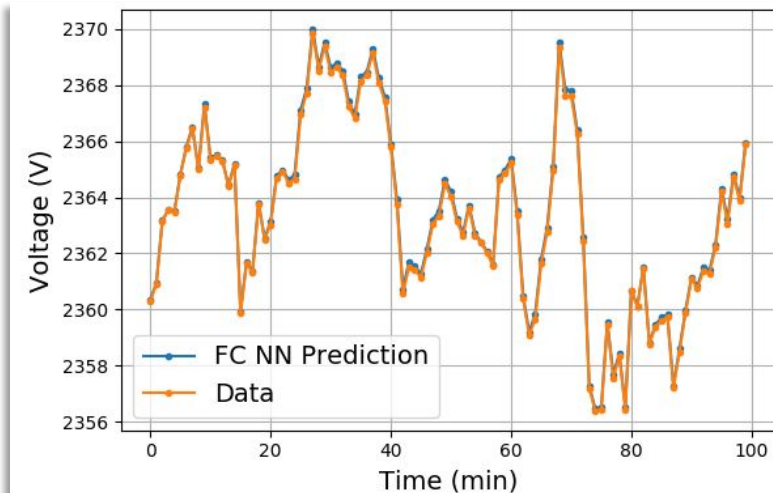
## Solution performance and validation

- Training duration ~ 1 week of simulation
- Voltage error typically < 0.05%



# Results: ML powerflow solver prediction accuracy

Voltage magnitude prediction for the IEEE 123 bus network (bus 92 phase A) using a fully-connected neural network.



Test errors for the convolutional neural network using *tanh* activation.

Power network	Test error
IEEE 4	1.0e-4
IEEE 13	1.7e-4
IEEE 123	9.9e-5
R1-12.47-3	2.6e-4
R2-12.47-2	4.8e-4

## Notes

- Accuracy achieved by the convolutional model is suitable for many simulation applications
- Optimal hyperparameters can be estimated based on known network characteristics (e.g. size)

# Next steps: Machine learning powerflow solver

## Explore improved CNN architecture

- Network model contain useful topology information

## Optimize training duration

- Automatically detect when training is done/required

## Integrate ML solver into GridLAB-D solvers

- Need to know when to fall back to NR solver
- Opportunity to hot-start NR solver using ML result

## Evaluate performance of deployed solver

- NR solver performance may be affected by ML solver

**Are any new use cases emerging that we consider?**

- PSPS impact analysis is an emerging use-case

**What is your view on cloud operations?**

- Utility comfort with cloud operations increasing steadily
- Enable support of multiple cloud vendors

**Thank You**

Contact: [dchassin@slac.stanford.edu](mailto:dchassin@slac.stanford.edu)

# ML Powerflow: Case Studies

## Populate grid models with residential load data

- Individual house end-use metering dataset

## Generate training, validation, and test sets

- NR-based GridLAB-D powerflow solver input & output
- IEEE and PNNL taxonomy feeders

## Analyze various loading conditions

- Power injection magnitude
- Power factor
- Load composition (voltage dependency)

## Power network models vary in terms of

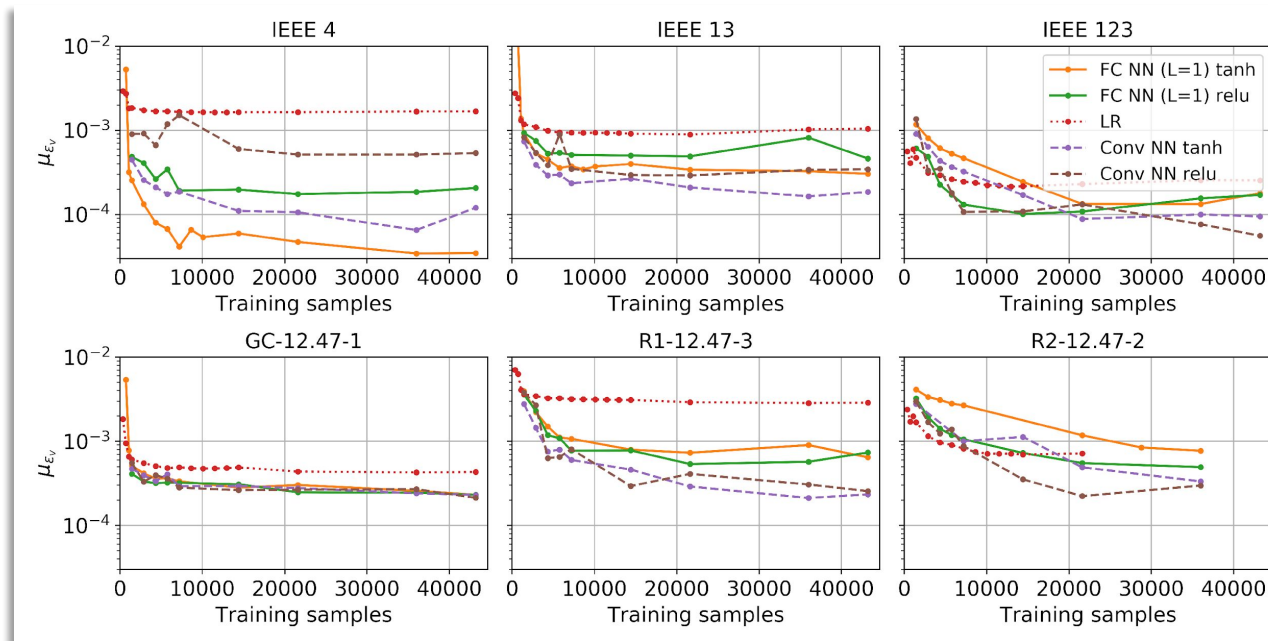
- Input and output dimension
- Linearity

Power network dimensions

Network	Voltage dimension	Power dimension
IEEE 4	12	3
IEEE 13	48	22
IEEE 123	402	95
GC-12.47-1	108	9
R1-12.47-3	297	37
R2-12.47-2	2553	214

# ML Powerflow: Performance Results

Voltage magnitude errors

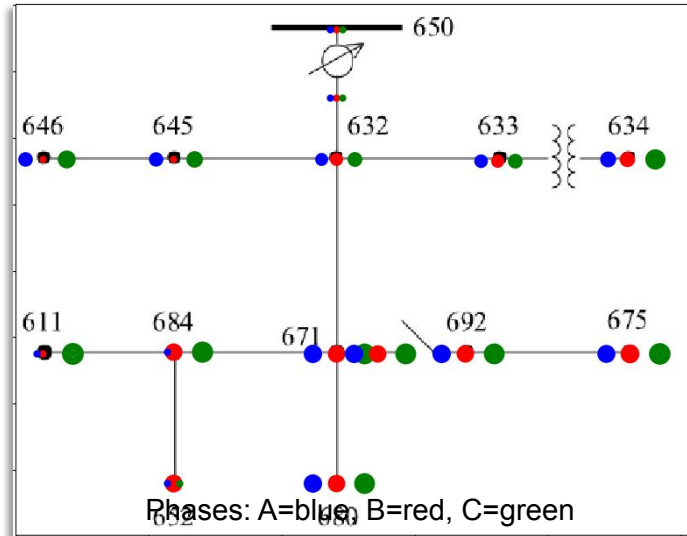


- The convolution model outperforms other models for most networks and is more scalable with network size
- Training data requirements depend on both the network size, characteristics, and how much the network is loaded
- Tanh NN activation generally results in the best and most consistent performance

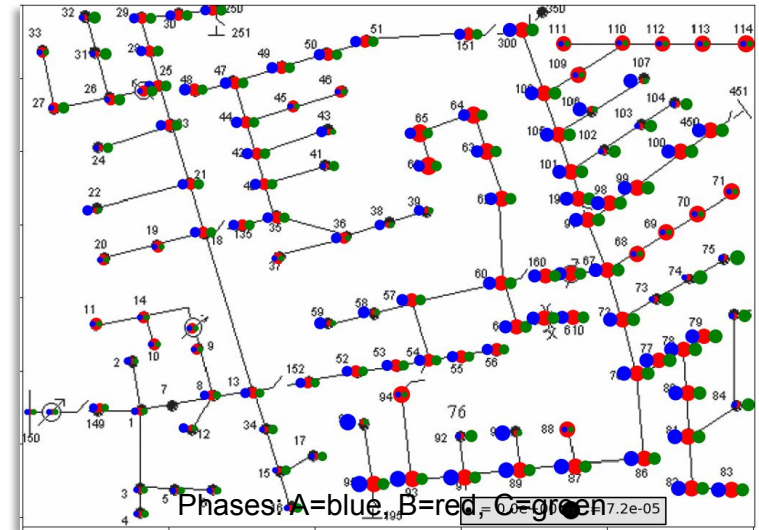


# ML Powerflow: Spatial Dependency of Errors

IEEE 13 bus network



IEEE 123 bus network



- Marker size indicates prediction error magnitude
- Voltage magnitude prediction errors increase further from the slack bus where there is larger deviation from the nominal voltage

# Results – Computational Cost

## Linear Regression

- Fast training time (<15 s on personal laptop)
- Small training set requirements (low memory requirements)

## Neural network approaches

- Training time is 1-3 orders of magnitude larger than linear regression
  - Additional computational gains may be achieved using improved optimization and weight initialization methods
- Larger training set requirements (larger memory requirements) compared to linear regression
  - Training set size could potentially be minimized through better design of the training set (the sampling of power injections)

Linear regression training time (using a laptop Intel Core i7 processor)

