SANDIA REPORT
SAND2022-12669
Printed September 2022

IMoFi (Intelligent Model Fidelity): Physics-Based Data-Driven Grid Modeling to Accelerate Accurate PV Integration Updated Accomplishments

Matthew J. Reno, Logan Blakely, Rodrigo D. Trevizan, Bethany Pena, Matthew Lave, Joseph A. Azzolini, Jubair Yusuf, C. Birk Jones, Alvaro Furlani-Bastos, Rohit Chalamala
Sandia National Laboratories

Mert Korkali, Chih-Che Sun, Jonathan Donadee, Emma M. Stewart, Vaibhav Donde
Lawrence Livermore National Laboratory

Jouni Peppanen, Miguel Hernandez, Jeremiah Deboever, Celso Rocha, Matthew Rylander, Piyapath Siratarnsophon
Electric Power Research Institute

Santiago Grijalva, Samuel Talkington, Cristian Gomez-Peces, Karl Mason, Sadegh Vejdan, Ahmad Usman Khan, Jordan Sihno Mbeleg, Kavya Ashok, Deepak Divan
Georgia Institute of Technology

Feng Li, Francis Therrien, Patrick Jacques
CYME International T&D

Vittal Rao, Cody Francis, Nicholas Zaragoza
Texas Tech University

David Nordy, Jim Glass, Derek Holman, Tim Mannon
Electric Power Board Chattanooga

David Pinney
National Rural Electric Cooperative Association

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico
87185 and Livermore,
California 94550

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology & Engineering Solutions of Sandia, LLC.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@osti.gov
Online ordering:	http://www.osti.gov/scitech

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Rd
Alexandria, VA 22312

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.gov
Online order:	https://classic.ntis.gov/help/order-methods/

ABSTRACT

This report summarizes the work performed under a project funded by U.S. DOE Solar Energy Technologies Office (SETO), including some updates from the previous report SAND2022-0215, to use grid edge measurements to calibrate distribution system models for improved planning and grid integration of solar PV. Several physics-based data-driven algorithms are developed to identify inaccuracies in models and to bring increased visibility into distribution system planning. This includes phase identification, secondary system topology and parameter estimation, meter-to-transformer pairing, medium-voltage reconfiguration detection, determination of regulator and capacitor settings, PV system detection, PV parameter and setting estimation, PV dynamic models, and improved load modeling. Each of the algorithms is tested using simulation data and demonstrated on real feeders with our utility partners. The final algorithms demonstrate the potential for future planning and operations of the electric power grid to be more automated and data-driven, with more granularity, higher accuracy, and more comprehensive visibility into the system.

ACKNOWLEDGEMENTS

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Solar Energy Technologies Office Award Number 34226. The team thanks the Electric Power Research Institute (EPRI) and CYME International for providing cost-share to the research project, and to Electric Power Board of Chattanooga for providing data and validation efforts.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

Distribution planning and operations are increasingly relying on utility distribution system models. As a result, limitations in the accuracy and detail of the models can lead to inaccurate planning and operational assessments and decisions. Errors in the models may stem from unlogged maintenance events, systems evolving over time, data entry errors, etc. The accuracy of these models is critical, however these errors often remain undetected, until issues occur, as manual field-verification is prohibitively expensive and time-consuming. Fortunately, recent grid modernization efforts, such as the widespread deployment of smart meters and other advanced metering infrastructure (AMI) devices, have dramatically improved both the quantity and quality of the data available to utilities. This project has developed a variety of physics-based, data-driven approaches that leverage AMI data to calibrate distribution system models and facilitate accurate analyses and planning tasks for integrating rooftop solar photovoltaic (PV) systems and other distributed energy resources (DERs). The following topic areas summarize the work conducted during this project.

Data Considerations

Working with real-world utility data comes with a set of data-related challenges. This portion of the project addresses those challenges. A set of definitions for the types of data challenges was created along with methodologies to systematically inject these issues into synthetic data; this includes challenges such as measurement noise, missing data, meter bias, etc. Filtering techniques were designed to filter erroneous data points from the datasets prior to use in subsequent model calibration applications. An assumption in prior model calibration efforts in literature is that the characteristics in question do not change, which is unrealistic. An algorithm to detect phase changes of customers in the dataset was developed and tested on both utility and synthetic data. The algorithm successfully detected several events in utility data with high confidence. This work also provides several recommendations related to AMI data collection in terms of meter settings and data quality. This can guide utilities in determining metering policies for time resolution, measurement resolution, noise limits, and other key metering decisions.

Phase Identification

Three phase identification algorithms were developed and tested, two based on voltage correlations and one based on state estimation. Results for the state estimation algorithm are shown on synthetic data and demonstrate the applicability of state estimation algorithm for model calibration tasks. Both voltage correlation algorithms leverage voltage time series data from customer advanced metering infrastructure (AMI) meters, and one of them also uses additional sensors placed on the medium voltage if available. Both algorithms were extensively tested using the synthetic data and further validated on the utility datasets. Utility #1 provided field validation of the results on four feeders, confirming the algorithm detection of a number of customers with incorrect phase labels. Utility #1 has also implemented a working, tested prototype of one of the algorithms into their own system. Additionally, in partnership with the National Rural Energy Cooperative Association (NRECA), one of the phase identification algorithms is now publicly available for coops and others to use as part of their Open Modeling Framework tool suite.

Secondary System Topology and Parameter Estimation

A parameter estimation algorithm based on pairwise linear regression leveraging voltage, real power, and reactive power from customer AMI meters was developed and tested on Utility #2. The algorithm successfully determines the wire type, resistance, and reactance values associated with the secondary network of the distribution system. In addition, a distribution system state estimator

(DSSE) was developed and tested using the synthetic dataset. The DSSE successfully improves visibility into the distribution system state and demonstrates results that allow for the detection of incorrect phase labels in the system.

Meter-to-transformer Pairing

A two-stage meter-to-transformer pairing algorithm was developed that first flags errors in transformer group labeling using correlation coefficients between voltage data in stage 1 and then corrects the flagged errors with a pairwise linear regression methodology using voltage, real power, and reactive power in stage 2. The algorithm was extensively tested using the synthetic data and was demonstrated as a proof of concept on examples from Utility #1 and Utility #2.

Medium-Voltage Reconfiguration Detection

An algorithm based on linear discriminant analysis (LDA) was developed to determine the position of switches with a distribution feeder. This algorithm was developed and tested for robustness using the synthetic dataset.

Regulator and Capacitor States and Settings

Two methods were developed to detect regulator and capacitor states and settings. The first method leverages the DSSE discussed in the secondary topology section. The DSSE can be used to provide visibility of regulator and capacitor states. The second method leverages measurements from both sides of the regulator, in this case from the IntellRupters® present in the Utility #1 feeders. An optimization algorithm is used characterize the physical parameters of the voltage regulator and identify the regulator tap position at each point in time. For the capacitor states, a clustering algorithm was developed that leveraged real and reactive power measurements. Results are demonstrated on data from Utility #1.

PV System Detection, Parameters, and Settings

A number of different algorithms were developed to provide enhanced visibility into behind-the-meter (BTM) photovoltaic (PV) systems. These include detecting, localizing, and estimating circuit parameters and volt-var settings. Algorithms for detecting the size of PV installations and estimating their tilt and azimuth were developed using a convolutional network method and a deep learning method, respectively. Algorithms were also developed for detecting PV control settings, power factor control settings, and volt-var control settings.

Load Modeling

Research in load modeling developed and evaluated improved, yet practical, distribution load modeling methods that leverage AMI and other modern data streams. In particular, this research developed improved spatial and temporal active and reactive power load models, methods to consider behind-the-meter PV in load modeling, representation of highly unbalanced loading conditions, and voltage-sensitive load models. This research identified numerous ways for distribution planners to enhance the conventional distribution load modeling practices to better utilize the modern data.

PV Dynamic Modeling

Viable approaches to calibrating an aggregate PV dynamic model and detecting maloperation/failure of PV inverter systems were tested in a modeling environment. More specifically, we have (1) accurately fitted the DER_A model and calibrated its parameters via a derivative-free nonlinear least-squares algorithm and (2) trained and tested some of the popular machine learning classifiers with certain pre-/post-event features, using the transient simulation data provided by CYME.

Algorithm Implementation

Two types of algorithm implementation were conducted. First, the Python implementation for both phase identification algorithms and the meter-to-transformer pairing algorithm were released as open source, hosted on the Sandia National Laboratories GitHub repository. CYME also implemented phase identification algorithms, both the one developed during this project and others for comparison, meter-to-transformer pairing algorithms, and secondary system topology algorithms as prototypes. Working in collaboration with NRECA, one of the phase identification algorithms was released as an available tool in the Open Modeling Framework (OMF) tool suite. OMF is an open-source resource of distribution modeling tools used by rural coops and others. Utility #1 implemented one of the phase identification algorithms within their system and intends it for widespread use throughout their service area.

Project Outcomes and Conclusions

This research work has resulted in a number of publications. The full publications list can be seen in Section 13.2. The ability to detect errors and calibrate distribution system models will continue to be an ongoing challenge for utilities, especially with the continued adoption of new smart grid technologies and controls. The developed algorithms, tested on real feeders, demonstrate the potential for future planning and operations of the electric power grid to be more automated and data-driven, with more granularity, higher accuracy, and more comprehensive.

ACRONYMS AND DEFINITIONS

Abbreviation	Definition
ADMS	advanced distribution management system
AMI	advanced metering infrastructure
ANSI	American National Standards Institute
AWG	American Wire Gauge
BTM	Behind-the-Meter
CC	Correlation Coefficient
CLM	Composite load model
CNN	Convolutional neural network
CYME	Company that develops the CYMDIST power system analysis tool
DER	distributed energy resource
DFT	Discrete Fourier Transform
DG	Distributed Generation
DMS	Distribution management system
DNI	direct normal irradiance
DNN	Deep neural network
DOE	Department of Energy
DSPE	Distribution System Parameter Estimation
DSSE	Distribution System State Estimation
DSTI	Distribution System Topology Identification
DWT	Discrete Wavelet Transform
EPB	Electric Power Board Chattanooga
EMT	Electromagnetic transient
EPRI	Electric Power Research Institute
ESS	energy storage system
EV	electric vehicle
FH	Feeder head
GHI	global horizontal irradiance
GIS	geographic information system
GMP	Green Mountain Power
GP	Gaussian process
GUI	graphical user interface
HC	Hosting capacity
IEEE	Institute of Electrical and Electronics Engineers
I/O	input/output

Abbreviation	Definition
k-NN	*k*-nearest neighbors
LASSO	least absolute shrinkage and selection operator
LLNL	Lawrence Livermore National Laboratory
LDA	Linear Discriminant Analysis
LDC	Line Drop Compensation
LR	linear regression
LRPE	linear regression parameter estimation
LRTE	linear regression topology and parameter estimation
LSE	Least square estimator
LTC	Load Tap Changer
MAE	Mean absolute error
MAPE	Mean absolute percentage error
MATLAB	Matrix Laboratory
ML	machine learning
MPPT	maximum power point tracking
MSE	Mean squared error
NB	Naïve Bayes
NERC	North American Electric Reliability Corporation
NREL	National Renewable Energy Laboratory
NLS	nonlinear least-squares
NLSE	Nonlinear least-squares estimator
OLS	ordinary least squares
OpenDSS	Open Distribution System Simulator™
OWA	optimally weighted average
PCA	principal component analysis
PCC	point of common coupling
PF	Powe flow
POA	plane of array
PMU	phasor measurement unit
pu	per unit
PV	photovoltaic
QSTS	Quasi-Static Time-Series
RBF	radial basis function
RDQDA	Regularized Diagonal Quadratic Discriminant Analysis
RF	random forest

Abbreviation	Definition
RMS	root mean squared
RMSE	root-mean-square error
SOC	state of charge
SNL	Sandia National Laboratories
SVD	singular value decomposition
SVM	support vector machine
TSHC	time-series hosting capacity
VRE	voltage regulating equipment
VREG	voltage regulator
WECC	Western Electricity Coordinating Council
WLS-SE	weighted least-squares static state estimator

1. INTRODUCTION

Modern distribution planning software (OpenDSS, CYME, Synergi, Milsoft, etc.) have evolved to allow accurate and detailed modeling and analysis of distribution feeders. However, while the software tools allow detailed modeling, the results are only as accurate as the models used for the simulations.

Utility distribution system models are traditionally created by importing equipment locations and connections from a utility's geographic information system (GIS). There is commonly little validation or calibration performed for these models, and all calibrations or corrections are done with manual, time-intensive testing. Historically, with low photovoltaic (PV) penetrations, model accuracy away from the substation was of little concern. However, at high PV penetrations where there will be multiple power injection points along the feeder, reverse power flows, and possible voltage and protection issues, it becomes critical to have accurate modeling to enable proper decision-making in all planning and operation stages of the distribution grid [1]. Even at the medium-voltage (e.g., 12 kV) level, the phasing of lines, topology changes, voltage regulator setpoints, and load connections are often uncertain. Very little may be known at the low-voltage level for locations of loads, connections between customers and service transformers, and the secondary system model, leading to large inaccuracies with errors >100% in the voltage drop/rise to the customer or distributed PV connected on the low-voltage.

Although utilities log PV interconnection requests, verifying all the details for hundreds of interconnections per day and keeping databases updated is a major challenge. PV penetrations on distribution feeders are increasing to become significant generation sources, but utilities lack visibility of this distributed PV. Many utilities do not record parameters for distributed PV such as their DC power rating, tilt, or azimuth, making it difficult to estimate their power. Furthermore, the actual PV system may vary from the interconnection plan, such as systems may not have been interconnected, may have been connected many months later than expected, may have changed size, have major shading issues, or may have failed. In the vast majority of cases of feeders in the United States that have been instrumented with sensors, significant differences are present between the field quantities retrieved by the sensors and the corresponding models used for planning, operations, and decision-making.

The determination of the amount of solar PV that can be reliably and safely connected at a given location of a distribution system (i.e., the PV hosting capacity) depends on a number of variables including the characteristics of the PV system, its inverter and its controls, and the model of the distribution circuit. The results of hosting capacity are known to be highly sensitive to the feeder model, where a few volts difference in the results of the power flow or quasi-static time-series (QSTS) analysis can result in estimations of PV hosting capacity varying by more than 200%. With this high level of uncertainty, planning engineers have limited confidence in distribution grid models and instead assume a conservative worst-case scenario based on rough rules of thumb rather than a detailed understanding of their system.

With the proliferation of advanced metering infrastructure (AMI), micro-inverters, and advanced sensors such as phasor measurement units (PMUs), a more granular and rigorous understanding of distribution grid operations with enhanced distribution feeder models is enabled. This will cause a fundamental change from models based on manual entry to data-driven modeling of the grid. This paradigm shift will require new learning and model calibration algorithms and interfaces between the data and modeling tools that will be developed in this project. Additionally, data-driven modeling will allow dynamically changing models that are automatically adapted to changing conditions on the grid, constantly modifying themselves based on topology reconfigurations, new PV installations, etc. This project uses the Big Data provided by the rapidly increasing distribution system sensing capabilities and physics-based machine learning using on standard power flow equations and feeder models to increase the accuracy and remove uncertainty from distribution system models to accommodate greater PV penetrations.

With high penetration of PV and other active resources, the future grid will have more complex analysis needs and new control architectures, coordinating load, storage and generation-based resources. The accuracy of distribution grid models, and availability of measurement data to validate the models will be a key component in determining overall deployment, investment, and future control strategies. Errors in models are prevalent in the distribution system, and data accuracy is a key challenge that leads to a lack of confidence in analysis and operations, and reluctance to move forward with advanced grid analyses. Barriers to modernization of the distribution system include availability and capabilities of existing planning tools, data availability, and model validation and accuracy, particularly inaccurate representation of impedances, connectivity/topology, loads, and DERs.

Inaccurate feeder models can result in under- or overestimating PV hosting capacity, highlighting the need for improvements to make models more accurate. Improved feeder models will enable accurate PV distribution impact assessment, improved distribution system efficiency, better implementation of conservancy voltage reduction, and numerous other key functions in distribution planning and operations. Model improvements have diminishing marginal returns, for example, in reducing hosting capacity error as illustrated in Figure 1-1, but this work targets the initial model improvements with very high returns. Current models are so uncertain that planners and developers have little confidence in the results–this project will provide massive improvements in PV hosting capacity accuracy and confidence by taking the first steps toward more accurate models. This enhanced utility understanding and the increased accuracy of feeder models will both substantially reduce the technical uncertainty of feeder hosting capacity calculations (eliminating the need for utilities to "hold back" feeder PV hosting capacity due to distribution model uncertainties) and allow for better utilization of inverter grid support functions.

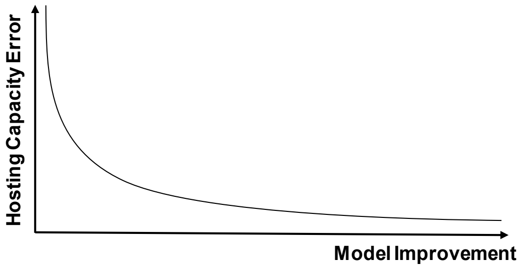


Figure 1-1. The influence of model accuracy on the hosting capacity error

1.1. Background

Distribution system models are prone to having a variety of different error types. Compared to the transmission system, distribution systems are less instrumented, less thoroughly modeled, and evolve faster. There are storm restorations, maintenance, new distributed energy resource (DER) devices, and new construction happening all the time. Each of these events can contribute to introducing errors into the distribution system model. These models are increasing critical for many desirable tasks including, grid modernization tasks, distribution system planning, resilience and reliability studies, hosting capacity analysis, and others. Errors in the distribution system models impact the accuracy of these studies and can force utilities to be, at best overly cautious, and at worst, use inaccurate results to influence their decision making. Having accurate models is critically important as DER integration increases; the zero carbon initiatives call for significant grid modernization in the coming years and accurate distribution system models are essential for success in those areas. Figure 1-2 shows a (non-exhaustive) list of distribution model errors by category. This project has developed model calibration algorithms or considered errors from nearly all the categories shown in this figure. This overview analysis of types of distribution system model errors resulted in a conference paper publication [2].

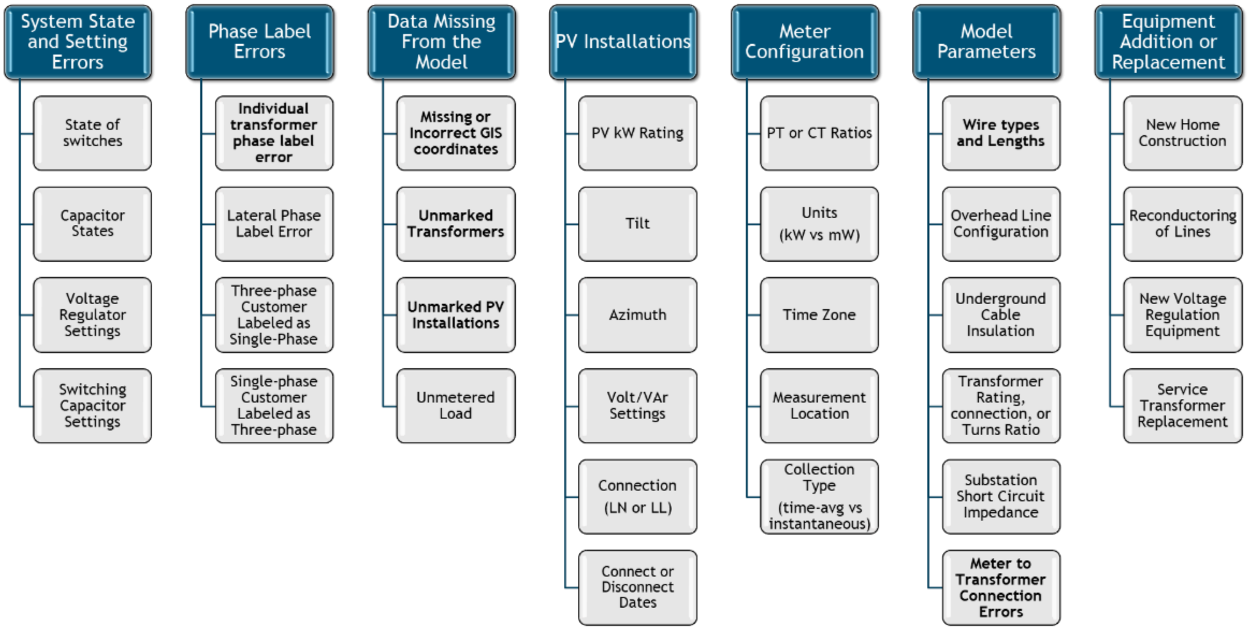

System State and Setting Errors
State of switches
Capacitor States
Voltage Regulator Settings
Switching Capacitor Settings
Phase Label Errors
Individual transformer phase label error
Lateral Phase Label Error
Three-phase Customer Labeled as Single-Phase
Single-phase Customer Labeled as Three-phase
Data Missing From the Model
Missing or Incorrect GIS coordinates
Unmarked Transformers
Unmarked PV Installations
Unmetered Load
PV Installations
PV kW Rating
Tilt
Azimuth
Volt/VAr Settings
Connection (LN or LL)
Connect or Disconnect Dates
Meter Configuration
PT or CT Ratios
Units (kW vs mW)
Time Zone
Measurement Location
Collection Type (time-avg vs instantaneous)
Model Parameters
Wire types and Lengths
Overhead Line Configuration
Underground Cable Insulation
Transformer Rating, connection, or Turns Ratio
Substation Short Circuit Impedance
Meter to Transformer Connection Errors
Equipment Addition or Replacement
New Home Construction
Reconductoring of Lines
New Voltage Regulation Equipment
Service Transformer Replacement

Figure 1-2. Common Types of Distribution System Model Errors

Table 1-1 shows a literature overview of distribution system model calibration tasks in response to the errors shown in Figure 1-2. This project has significantly advanced several of these key areas including Phase Label Errors (Section 4), Model Parameters (Section 5 and Section 6), PV Installations (Section 9), System State and Setting Errors (Section 7 and Section 8).

Table 1-1. Types of distribution system model errors and literature addressing those challenges

Error Categories	Error Types
System State and Setting Errors	[3]–[8]
	State of switches (normal open or closed)
	Capacitor states
	Voltage regulator settings
	Switching capacitor settings
Phase Label Errors	[9]–[17]
	Individual transformer phase label error
	Lateral phase label error
	Three-phase customer labeled as single-phase
	Single-phase customer labeled as three-phase
Data Missing from the Model	[6], [18]–[22]
	Missing/Incorrect GIS coordinates
	Unmarked transformers
	Unmarked PV installations

Error Categories	Error Types
	Unmetered load (unmarked customers or sources)
	Connection (LN or LL) and grounding
PV Installations	[18], [23]–[25]
	PV kW rating
	Tilt
	Azimuth
	Volt/VAr settings
	Connection (LN or LL)
	Inverter size
	Connect/disconnect dates
Meter Configuration	[8], [26]
	PT or CT ratios
	Units (kW vs mW)
	Time zone
	Measurement location
	Unknown collection type (time-avg or instantaneous)
	Unknown meter accuracy
Model Parameters	[7], [9]–[11], [22], [27], [28]
	Wire Types, overhead line configuration, underground cable insulation, lengths, and number of phases
	Transformer rating, connection (LN or LL), or turns ratio error
	Substation short circuit impedance
	Meter-to-transformer connection errors
New or Replacement Equipment	New home construction
	Reconductoring of lines
	New voltage regulation equipment
	Service transformer replacement

1.2. Project Objectives

As the cost of solar continues to decrease, grid integration becomes an increasingly important factor. The goal of the project was to develop novel methods to improve key modeling challenges that are beyond the immediate vision of utilities and regulatory agencies and to solve grid integration challenges resulting from issues in model accuracy. The research effectively utilizes modern distribution sensor data from AMI, PV systems, and supervisory control and data acquisition (SCADA) systems to calibrate distribution power flow models using physics-based Machine Learning. This will meet the following SETO objectives:

Objective 1: Develop improved modeling tools for power system planning with high PV penetration

Develop and validate advanced models and data analytical techniques (including machine learning) that improve grid modeling accuracy and robustness to support system planning and the operation for high penetrations of distributed PV. Task 1 will develop algorithms to improve modeling accuracy of distribution system model parameters for lines, topology reconfigurations, modeling errors and issues, and system phase connections. This work is summarized in Sections 2, 4, 5, 6, 7, 8, and 12.

Objective 2: Provide visibility into real-time solar generation throughout the distribution system

Develop and validate algorithms for improved observability into the state of distributed generation in the network under constantly changing conditions. This task will develop algorithms to determine locations and sizes of PV, in addition to their parameters (DC ratings, azimuth/tilt/tracking, etc.) and settings (power factor, volt-var curves, etc.). This work is summarized in Section 9.

Objective 3: Develop accurate and high-resolution (spatial and temporal) models

Create high-resolution and accurate models to reduce the cost for utilities to host higher level of solar generation on their systems. This task will develop improved spatial and temporal load modeling methods that include detailed modeling of phases, reactive power, and voltage sensitivities. This work is summarized in Section 10.

Objective 4: Enhance the understanding of PV inverter control

Identify PV inverter dynamic models to understand impacts to system inertia and stability, system protection, and capability to provide other grid services. This task will develop algorithms to improve the fidelity of dynamic models the distribution system and PV inverters and to detect maloperation, failures, or areas to improve operation. This work is summarized in Section 11.

Figure 1-3 shows an overview of the broad project tasks. Each of the blue rectangles corresponds to a chapter section in this report.

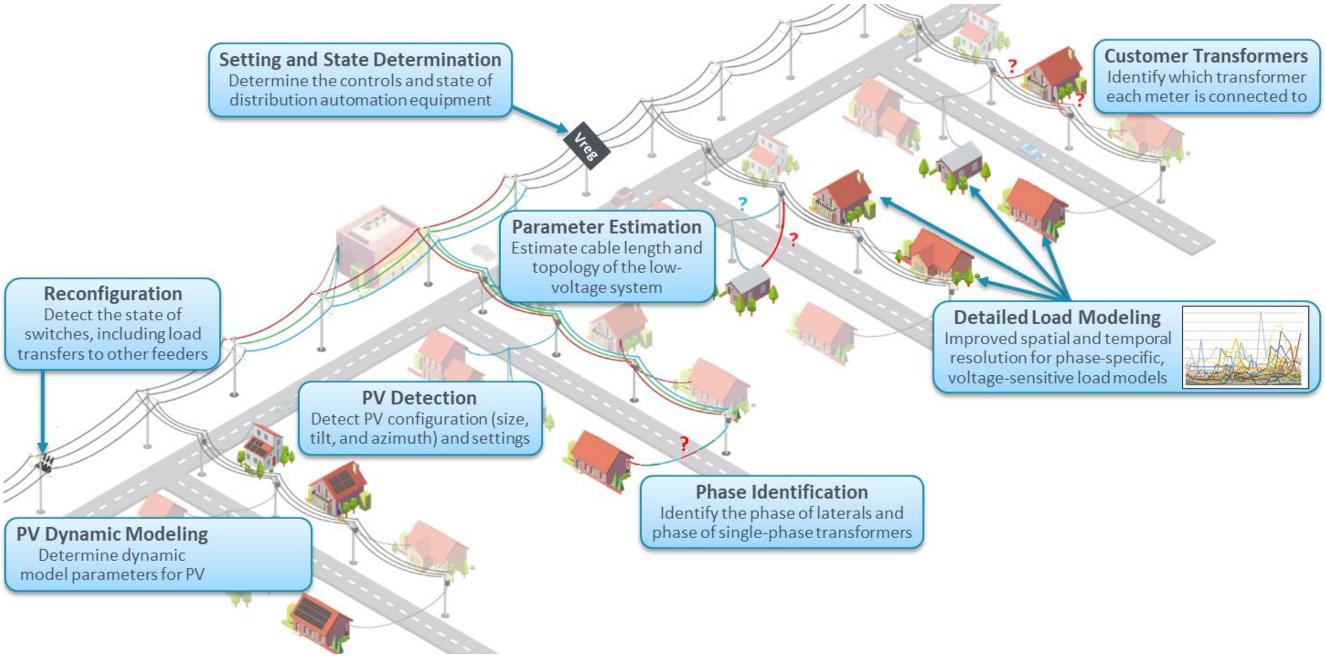

Setting and State Determination
Determine the controls and state of distribution automation equipment
Vreg
Customer Transformers
Identify which transformer each meter is connected to
?
Parameter Estimation
Estimate cable length and topology of the low-voltage system
Reconfiguration
Detect the state of switches, including load transfers to other feeders
Detailed Load Modeling
Improved spatial and temporal resolution for phase-specific, voltage-sensitive load models
PV Detection
Detect PV configuration (size, tilt, and azimuth) and settings
Phase Identification
Identify the phase of laterals and phase of single-phase transformers
PV Dynamic Modeling
Determine dynamic model parameters for PV

Figure 1-3. Project Overview

2. DATA CONSIDERATIONS

This section provides an overview of data considerations for utility AMI data. Real-world data provides a variety of challenges related to data quality that must be considered in developing model calibration algorithms. Algorithms that are not equipped to deal with these types of real-world data issues will be impossible to roll out for widespread utility usage.

2.1. Types of Data Issues

Figure 2-1 illustrates three general types of data quality issues affecting utility data. Data collection decisions are decisions made the by utility, which are the data types, measurement fields, measurement intervals, measurement resolutions, and data quantity. These decisions are influenced by data management issues, data transmission concerns, data storage, and what a particular utility feels are the use cases for the data in the future. One aspect of this project is to better inform utilities about best practices for collecting AMI data based on requirements and considerations for the model calibration algorithms developed throughout this project. Systemic errors stem from ongoing problems within the meters; these can be from meter failure issues such as meter bias or time synchronization or operator error such as incorrect time zone settings or incorrect meter units. Finally, random errors include measurement noise, missing data, bad data, etc.

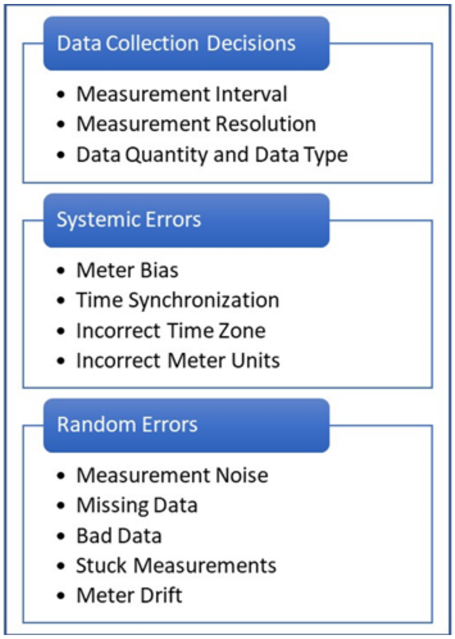


Figure 2-1. AMI data considerations chart

2.2. Data Preprocessing and Remediation

2.2.1. Definitions for AMI Data Challenges

For many of the data considerations shown in Figure 2-1, the best solution overall is to develop algorithms that are robust to those data challenges. With this objective in mind, we developed methodologies to artificially simulate these data issues in our synthetic dataset. Table 2-1 shows the formulas used to inject a wide variety of realistic data issues into the synthetic data. The data issues simulated include differing measurement intervals, differing meter resolution, meter bias, measurement noise, time synchronization, missing data, and data availability. This allowed us to rigorously test algorithms under development to quantify the effects of these data issues and design algorithms to be robust to them without incurring large preprocessing costs. The results of this data issues overview were published in a conference paper [29].

Table 2-1. AMI data quality manipulation formulas

Variable Definitions	$T_{total} = the\ total\ number\ of\ measurements\ available\ at\ the\ 1 - minute\ resolution$
	$t\ \in \{1, 2, 3, \dots, T_{total}\} - individual\ time\ step\ at\ 1 - \min resolution$
	$i\ \in I\ where\ I = \left\{1, 2, 3, \dots, \frac{T_{total}}{k}\right\} - individual\ measurement$
	$v\ \in\ V, where\ V = time\ series\ of\ voltage\ measurements$
	$C_{Total} = The\ total\ number\ of\ customers$
	$c\ \in C, where\ C = \{\ 1,2,3, \dots, C_{Total}\} - set\ of\ all\ customers$
	$\mu = ideal\ mean\ of\ the\ time\ series\ (240\ in\ this\ case)$
	$U - uniform\ distribution$
Measurement Interval	$For\ each\ c$: $v_{c,k}(i) = \frac{1}{k} \sum_{t_0 = (i-1)*k+1}^{t = k+t_0-1} v_c(t)\ , \qquad \forall i$
	$k = \{\ 1, 5, 15, 30, 60\}$ – measurement interval in minutes
Meter Resolution	$For\ each\ c$: $v_{c,k}(i) = round\big(v_{c,k}(i), d\big), \qquad \forall i$
	$d\ \in D\ where\ D = \{0, 1, 2\} - decimal\ places$
Meter Bias	$For\ Each\ c$: $V_{k,bias} =\ V_k + (b *\ \sigma_c)$
	$p\ \in \{0, 0.5, 1.0, 1.5, 2\}$ $- \max\ allowable\ percent\ bias$ $\sigma_c \sim U(-1,1) - bias\ scaling\ factor$ $b = (p/100 *\ \mu) - max\ allowable\ bias$
Measurement Noise	$For\ each\ c$: $v_{k,noise}(i) =\ v_k(i) + (n *\ \sigma_i), \qquad \forall i$
	$p\ \in \{0, 0.05, 0.25, 0.5, 0.75, 1.0, 1.25, 1.50, 1.75, 2.0\}$ $-maximum\ allowable\ percent\ noise$ $\sigma_i \sim U(-1,1) - noise\ scaling\ factor$ $n = (p/100 *\ \mu) - max\ allowable\ noise$
	$For\ each\ c$: $\forall i, \qquad v_{c,1,timeSynch}(i) =\ v_1(i +\ s_c)$ $V_{c,timeSynch} = Truncate(V_{c,timeSynch}, (2 * f))$

Time Synch	$f \in F, where\ F = \{1,2,3,4,5\}$ $-$ max $offset\ in\ minutes$ $s_c = U(-f,f)$ $-random\ scaling\ factor$ *Truncate(timeseries, truncationAmount) – truncates the beginning and end of the specified timeseries by the amount specified*
Missing Data	For Each Customer c: For ctr from 0 to h: $startPosition = U_{int}(0, \|I\|)$ $V_{c,k,Missing}\ [startPosition : (startPosition + g)] = NaN$
	$p = \{\ 0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$ $-percentages\ of\ missing\ data$ $h = floor\left(\frac{(p * \|V\|)}{g}\right)$ $-\ the\ number\ of\ missing\ data\ instances$ $g = number\ of\ samples\ missing$
Quantity of Data	$For\ Each\ c$: $V_{c,k,m} = V_{c,k,12}(1), V_{c,k,12}(2), V_{c,k,12}(3), \ldots,$ $V_{c,k,12}(30 * m)$
	$m \in M\ where\ M = \{12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1\}$ $-\ months\ of\ data$

2.2.2. Filtering Erroneous Data

It is unrealistic to expect the algorithm to handle some types of bad data. Erroneous datapoints may arise for a variety of reasons, the full description of such is outside of this project. Here we assume there is an unknown quantity of different types of bad data due to meter failure. In many cases, these outlier values are best removed prior to using the data in a model calibration algorithm. The goal of the data pre-processing task is to identify and remove errored measurements from the data set. A set of functions were created in Python for the task of removing abnormally high and low data points due to meter error from the voltage and power data set. The functions used to filter the per unit voltage data took a percentage of the mean and filtered data above or below that threshold. In the power filtering task, the objective is to identify abnormally large spikes that likely resulted from aggregate power measurements after long periods of missing data. This aggregate value appears to be expected functionality from the meter; however, for our purposes it does not reflect a "normal" timeseries value. The function created for this task used the customer's 95th percentile and a multiplier (10 or 20) to filter out these large spikes. Figure 2-2 shows examples of abnormal data points from the Utility #1, Feeder #1 that are flagged and filtered using these functions.

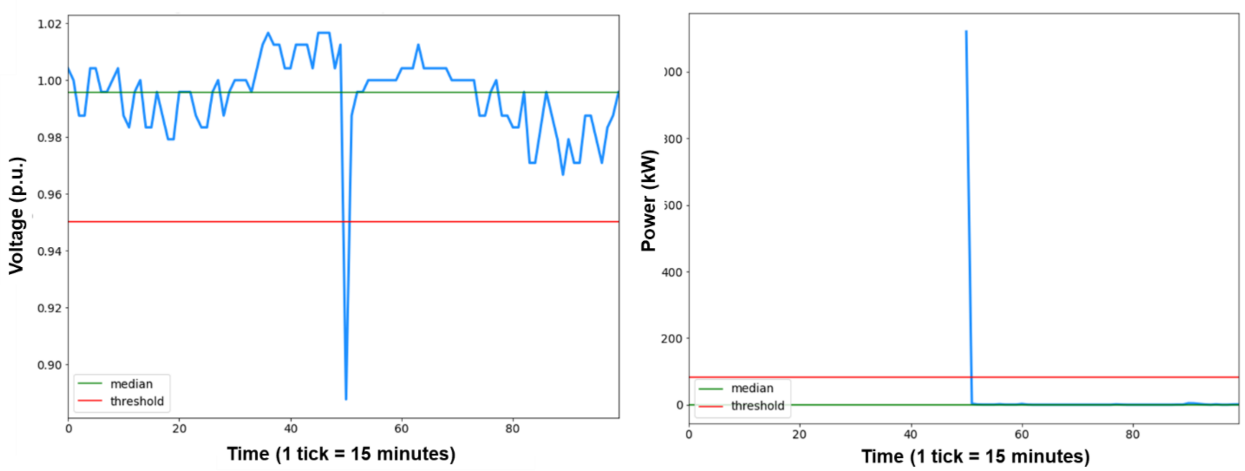


Figure 2-2. Abnormal voltage (left) and power data (right) flagged by filtering functions from Utility #1, Feeder #1.

In addition to the functions described above created to identify and filter errored/abnormal measurements, another set of functions were created to identify significant shifts in the data. Each function uses a sliding window to compare two consecutive windows of data. One function compares the mean of two windows and flags the timestamps where the percent change in mean is above a given threshold. The second function compares the standard deviation of two windows, and flags timestamps where the percent change in standard deviation is above a given threshold. Figure 2-3 shows examples of shifts in data identified in the Utility #1, Feeder #1 using these functions.

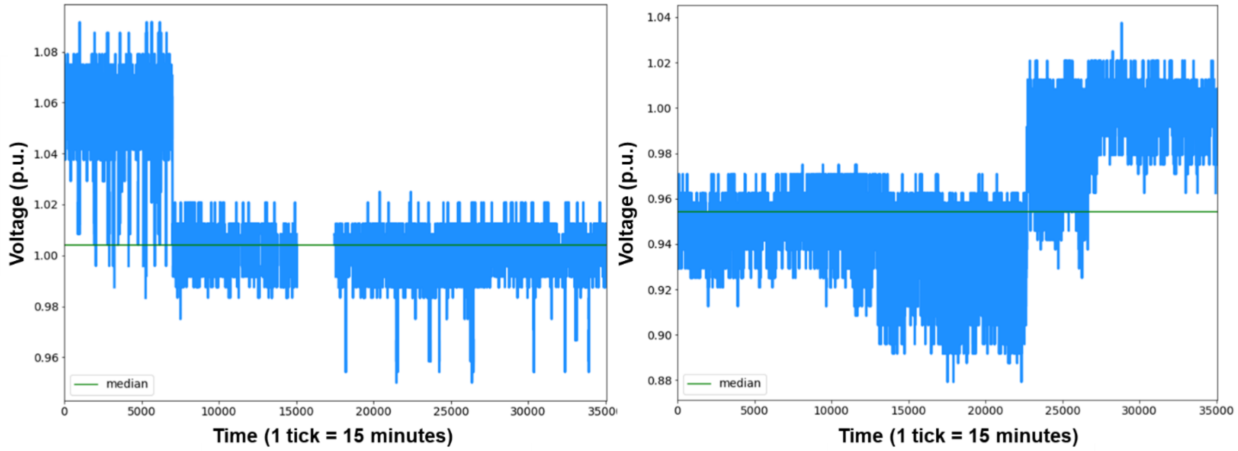


Figure 2-3. Example of shifts in data detected by sliding window functions from Utility #1, Feeder #1.

2.2.3. Data Interpolation for AMI Data

Missing data is consistently a problem for distribution system model calibration algorithms. Missing data can occur due to communication failure between the meters and the control center, outages, or filtering erroneous values as described in Section 2.2.2. Algorithms developed for model calibration must be able to deal with this issue. Many of the algorithms developed in this project such as phase identification (Section 4) and meter-to-transformer pairing (Section 6) incorporate robustness into the algorithm itself, for example by using an ensemble approach, and thus do not require perfectly complete datasets. This is one method for dealing with missing data. However, in other cases it may be desirable, or necessary, to have complete timeseries without missing values, for example in the case of state estimation (Section 4.4). In those cases, some type of data interpolation technique is necessary.

Figure 2-4 shows a straightforward linear interpolation approach for timeseries voltage data, and Figure 2-5 shows an interpolation using the optimally weighted average (OWA) approach developed during a previous project [30]. Clearly the interpolation is significantly improved using the OWA method. Testing of the OWA approach using the phase identification algorithm described in Section 4.3 demonstrated the interpolation was not accurate enough to successfully complete the phase identification task. However, in cases where having complete timeseries data is essential it could be a reasonable approach. A third method was under development when the project ended. This method was based on leveraging an interpolation of the real power datapoints to inform the interpolation of the voltage datapoints. The final development of this methodology is left to future work.

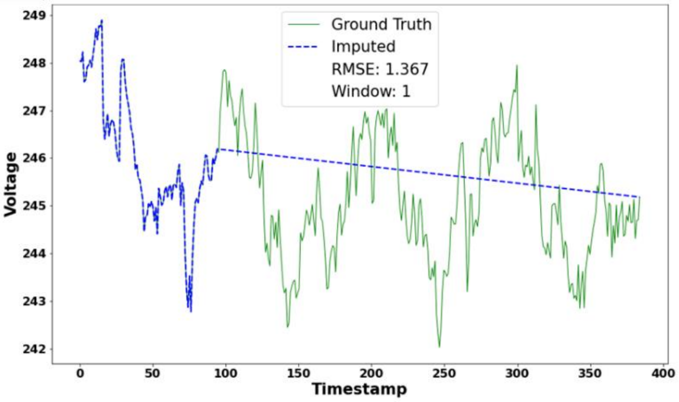


Figure 2-4. Voltage timeseries section shown with linear interpolation

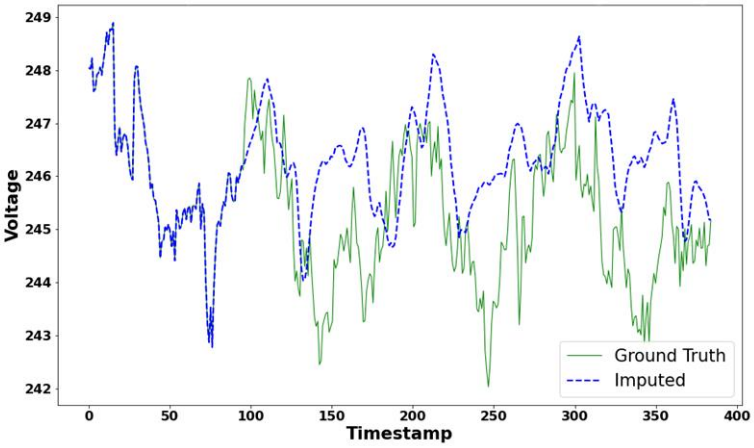


Figure 2-5. Voltage timeseries section shown with OWA interpolation

2.3. Model Evolution Through Time

2.3.1. Overview

The distribution system is changing and evolving rapidly as digital technologies are integrated with existing components, renewable energy resources are added, new construction is integrated, and as maintenance occurs, either due to upgrades or extreme events. The evolution of the distribution system over time poses some particular challenges for maintaining accurate distribution system models.

One major assumption that is made in each of those cases, and the bulk of distribution system model calibration algorithms in general, is that the attribute in question does not change *during* the period of AMI data that is being used as input. For example, the phase identification algorithms assume that the phase labels, for every customer on the feeder, are perfectly consistent during the period of AMI data being used. This of course, is an unrealistic assumption in practice. These types of model attributes are constantly changing and evolving over time in some unknown quantity and frequency; there is no way to realistically make that assumption.

2.3.2. Phase Change Detection Methodology

This work describes an algorithm to detect a change in phase for a customer during a given time period. If a customer has changed phase, this can be used to split the dataset such that the phase identification algorithms (described in Section 4) can be used to verify the correct phase for that customer.

The spectral clustering ensemble phase identification algorithm described in Section 4.3 partitions the voltage time series data into a series of windows and clusters each window using spectral clustering. This algorithm uses a window size of 384 datapoints (4 days using 15-minute interval data) as does Section 4.3. Figure 2-6 illustrates this portion of the algorithm, emphasizing the window ensemble step. Because each window of data is clustered in the ensemble step, it is possible to obtain a predicted phase for each customer for every window of data. This allows a customer's predicted phase to be observed over time.

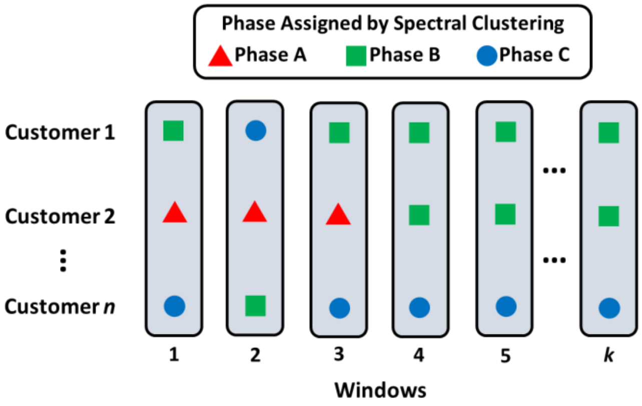


Figure 2-6. The ensemble step, which clusters each window of data and assigns a predicted phase

In the ensemble step, each window of data is clustered four times using 6, 12, 15, and 30 clusters; this step is explained in more detail in Section 4.3. From the k-vector clustering process, we can obtain a matrix with dimensions *n x m x 4* of predicted phases where *n* denotes the number of customers, *m* denotes the number of windows, and 4 is the number of values in the k-vector . We take the mode of the 4 predicted phases to condense the array to $n \times m$. In the case that there is no single value mode of the 4 predicted phases, we choose the predicted phase from the six-cluster result. We believe this prediction to be slightly more reliable than the predictions resulting from larger numbers of clusters.

Using the condensed array, the customers' predicted phase can be observed over time; therefore, it follows that a change in predicted phase over time may indicate a true phase change event. Figure 2-7 is a flow chart describing the steps taken to detect these events via the array of predicted phases per window. The key challenge is that the individual phase predictions may be quite noisy due to outside factors, measurement noise, customer behavior, etc.

To detect a change of phase event, we compare the predicted phase of each window, starting at window 20 until window – 20 is reached. If a given window *i* and the window succeeding it, window *i+1,* have different phases, this represents a potential phase change if the threshold criteria is met. We define a change point at window *i+1* for this algorithm if the following criteria is met:

- **At least X%** of windows in range [*i-20,i*] have the same phase as window *i*.
- **At least X%** of windows in range [*i+1, (i+1) + 20*] have the same phase as window *i+1*

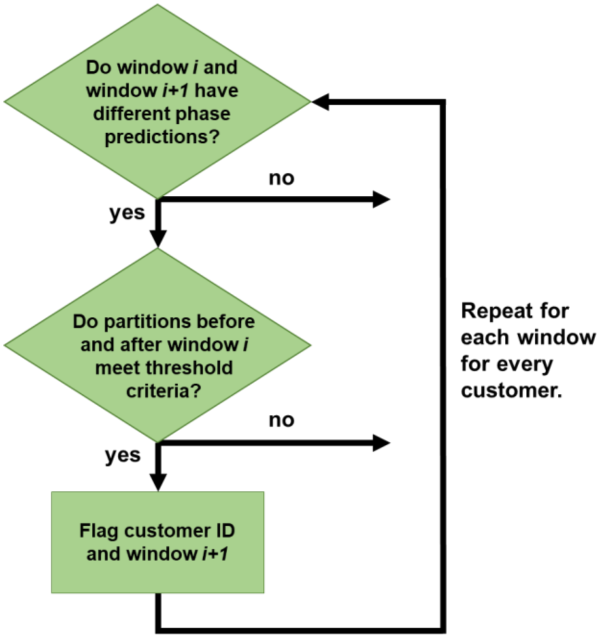

Do window *i* and window *i+1* have different phase predictions?
no
yes
Do partitions before and after window *i* meet threshold criteria?
no
yes
Flag customer ID and window *i+1*
Repeat for each window for every customer.

Figure 2-7. Flow chart of methodology for flagging possible phase change events

The comparison range of 20 windows was chosen after testing the algorithm using a number of different choices. Twenty produced the most consistent results in the synthetic dataset. This value needs to be as small as possible to reduce the data requirements for the algorithm but large enough to account for the incorrect (noisy) phase predictions in individual windows. We observed that smaller values of this parameter increased the number of false positives flagged in the synthetic dataset.

Confidence Scores

One challenge in finding change of phase events, is determining which flagged events represent true events, and which events are falsely flagged. We have developed two metrics used to help quantify how confident we are a flagged event represents a true event. These metrics can be used to help further eliminate false positive events.

Number of Segments

If we define the number of segments in a set of a window as any contiguous series of windows with the same phase, we can use the number of segments to quantify the level of consistency/variation in window predictions. Ideally, customers without events would have one segment, and true positive customers would have two segments. As the number of segments increases, meaning the variation in window predictions increases, we have less confidence that the flagged event is due to a true change point, rather than random variation in the window predictions. One note on this confidence metric is that the number of segments is highly affected by the measurement noise, and other issues, in the dataset.

K-Vector Agreement

The k-vector clustering step used in the phase identification algorithm is used to increase the robustness of the final co-association matrix. We can also base confidence in a change point occurring based on the agreement of the four k-means results by testing whether the change point occurs in the four different clustering results.

In the ideal situation, a change point would be flagged at the same location for all four, individual k-means results. This would indicate that all four clustering results in the k-vector step showed a change of phase. Therefore, we can derive a score representing the number out of four of k-means results that return a change point.

2.3.3. Phase Change Detection Results

Synthetic Data

The synthetic dataset described in Section 3.1 was used to test the algorithm for identifying phase changes. OpenDSS was used to simulate three lateral phase change events and one transformer change event affecting 46 customers in total.

Looking at Figure 2-7, the primary parameter to set is the threshold criteria shown in the middle diamond. The change point detection method was tested for a range of thresholds using the synthetic data. With the synthetic data, the ground truth is known, so we can evaluate the number of true positive and false positive customers flagged. The algorithm was tested using a range of percent thresholds from 50%-95% incrementing by 5%. In addition, we used the k-vector agreement confidence metric to further filter for customers with "high confidence." These results are shown in Figure 2-8.

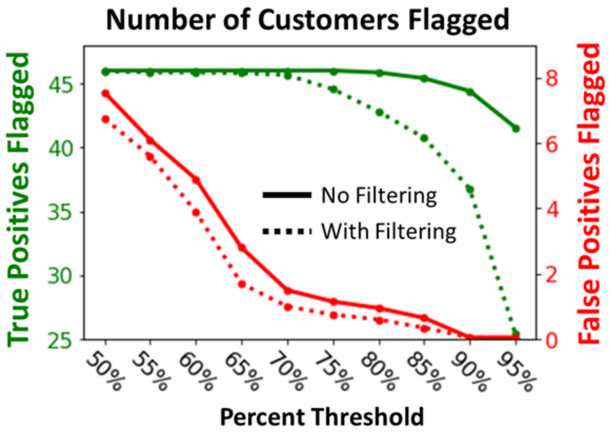


Figure 2-8. Threshold sweep showing true positives flagged and false positives flagged with and without filtering

Setting the threshold parameter involves a tradeoff between limiting the number of false positive events flagged and maintaining most if not all true positive events. This tradeoff is seen in Figure 2-8. In general, for this synthetic dataset, the number of true positive customers flagged doesn't begin to degrade until the 80% where the algorithm, flagged all true positive customers 17 out of 20 runs, and flags one false positive for 19 out of 20 runs, and flags zero false positives in the last run. In contrast, to eliminate all false positive customers, an 90% threshold was needed, at the cost of failing to flag, 2 true positive customers on average.

We can use the k-vector agreement metric to filter the number of false positive customers by only permitting customers with events that had a k-vector agreement metric greater than or equal to 3. Similar to setting a higher threshold, eliminating the number of false positive customers comes at the cost of eliminating some true positive customers. Understanding the trade-offs between allowing false events into the distribution system model and failing to detect true events will help inform how to best select the threshold parameter. Figure 2-9 shows an example of a true positive customer (customer with a phase change) and a false positive customer (no phase change); the colors represent the phase predictions for each window for the two customers.

In the synthetic dataset, we demonstrate the use of both proposed confidence metrics to filter the flagged change point results. For the synthetic dataset we assign low confidence to a flagged event represents a true event if:

- The customer has a k-vector agreement metric less than or equal to 3, AND
- The number of segments is less than the number of segments in the 95th percentile

Because the number of segments is highly affected by measurement noise, which is particular to each dataset, rather than pre-defining an "acceptable" value to filter with, we use the 95th percentile as a threshold, which is dependent on the data characteristics. The 95th percentile was chosen for the synthetic dataset because it was most effective at separating false positive customers from true positive customers.

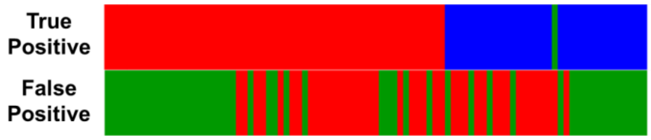


Figure 2-9. Example of false positive customer filtered with the confidence metrics, and a true positive customer.

The proposed method for detecting phase change events was additionally tested using the synthetic dataset with various levels of noise. The data was injected with normally distributed noise with a specified standard deviation. Noise injections ranged from percent standard deviation 0.01% to 0.8%. Figure 2-10 shows the results of the noise sweep. The plot contains the number of true positive and false positive customers flagged with and without filtering for customers with a k-vector agreement greater than or equal to three for each level of noise. The plot shows the average number of customers flagged for each noise level.

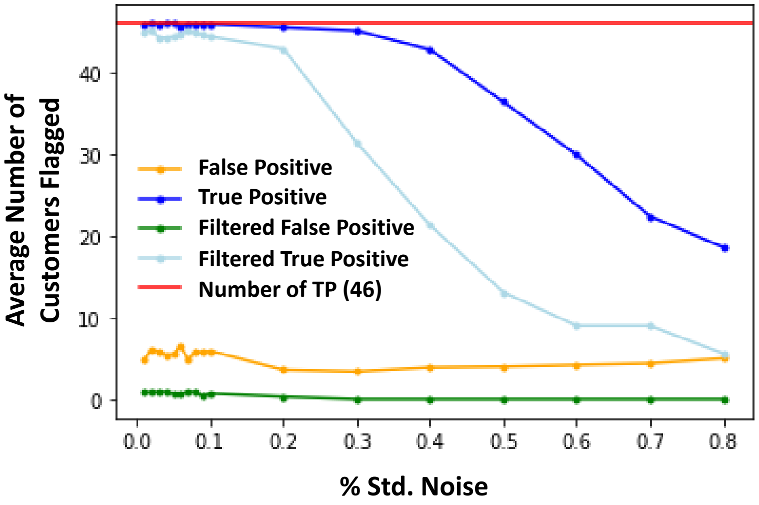


Figure 2-10. Number of flagged customers per noise level

Utility Data

The algorithm was applied to AMI voltage data from two different utilities using two feeders from each utility.

When applied to data from Utility #1, Feeder #1, and Feeder #2, the phase change detection algorithm did not flag any customers for any threshold. Given the information available for these feeders from the utility, we find this to be a reasonable result.

The algorithm was also applied to Utility #2, Feeder #1. The algorithm flagged 20 customers using an 80% threshold. These results could be further filtered by selecting customers with a k-vector agreement of 4, leaving 13 customers remaining. Although there is no way to know the ground truth without field verification, we can validate our algorithm's findings using the distribution system topology made available to us via publicly available from Google Earth and Google Street View Imagery. In total, we were able to partially validate 3 events affecting all 13 remaining customers.

The algorithm flags 10 customers that are detected to have changed from phase A to phase C at window 69. From the topology information on Google Earth we have available for this feeder, we found that these customers were on the same lateral. Figure 2-11 shows the flagged customers, and the lateral they belong to. Although the ground truth is not available, the fact that these customers are all on the same lateral and all flagged at the same moment does partially validate these results as a probable phase change event for this lateral.

In addition to the 10 customers flagged for an event at window 69, two customers were flagged for a change from phase A to phase B at window 69, and one customer was flagged for a change from phase B to phase A at window 39. Figure 2-12 shows the map of the two customers connected to the same transformer. Again, only the two customers labeled as serviced by this transformer were flagged as having this event. It is less confirmative than in the lateral case, but still provides some validation of the algorithm results.

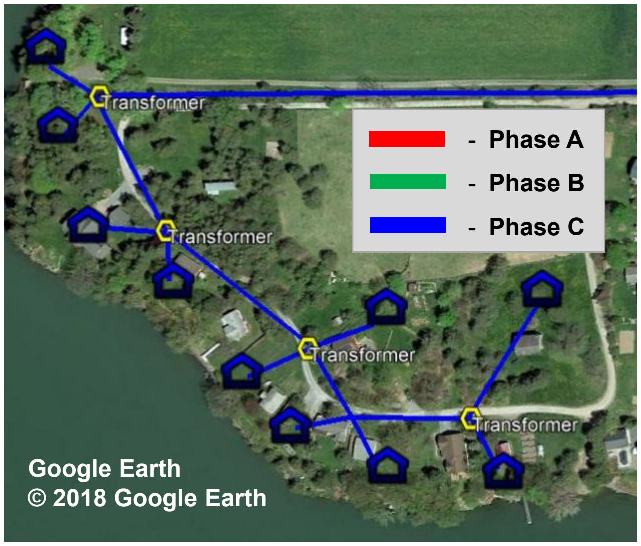


Figure 2-11. Ten customers flagged for change from phase A to phase B on Feeder #1 from Utility #2

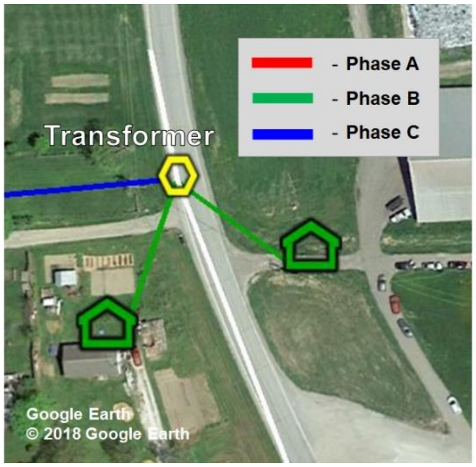


Figure 2-12. Two customers connected to the same transformer with a phase change event from phase A to phase B on Feeder #1 from Utility #2

2.3.4. Phase Change Detection Conclusion

This section presented an algorithm for detecting phase changes in voltage time series data, and a set of confidence metrics that can be used to evaluate the results of the algorithm. The proposed methodology leverages the ensemble step of the spectral clustering ensemble phase identification algorithm combined with a novel phase change detection algorithm to obtain predicted phases for windows of data, which can be used to determine whether changes of phase occur. This algorithm was applied to a synthetically generated voltage time series data set with phase change events, and to four utility datasets. Further work is needed on parameter tuning of the algorithm and use of the proposed confidence metrics. The results on the synthetic dataset demonstrate high accuracy and robustness to measurement noise. In addition, the algorithm identified three possible phase change events affecting 13 customers in one of the utility datasets, which we were able to verify using the grid topology made available through Google Earth and identified two other possible events in another utility dataset. We have demonstrated that the proposed algorithm demonstrates the ability to accurately identify true phase change events, with limited numbers of false positive events. This work was published as a conference paper [31].

2.4. AMI Data Requirements for Model Calibration

This section presents some general conclusions and recommendations for AMI data collection requirements as it relates to model calibration. Certainly, the requirements will vary somewhat by task, specific algorithm, and model calibration requirements. The following notes and conclusions should be taken in the context of the experiments, algorithm, and data used; however, they may serve as a foundation for utility decision making.

Table 2-2 shows a set of AMI data recommendations based on testing with a correlation-based phase identification algorithm described in Section 4.3.1.3. Table 2-3 shows a set of AMI data recommendations based on a meter-to-transformer pairing methodology based on the parameter estimation algorithm described in Section 5.1; a refined version of meter-to-transformer pairing algorithm is presented in Section 6.

Table 2-2. AMI data recommendations for correlation-based phase identification algorithms

Data Quality Manipulation	AMI Considerations Based on the Phase Identification Task
Measurement Interval	5 - 15-minute intervals are recommended
Meter Precision	At least 1 decimal on voltage measurements (240V) is required
Meter Bias	Bias does not impact phase identification results with this algorithm
Measurement Noise	< 0.25% maximum uniform random noise is recommended
Time Synchronization	> 1-min measurement intervals are required to account for the time synchronization errors
Missing Data	Sensitive to the *distribution* of missing data. Given uniformly distributed 4-hr missing data instances, with this algorithm, the percentage of missing data is required to be < ~4%
Data Availability	> 4 months of AMI voltage data are required

Table 2-3. AMI data recommendations for preliminary version of meter-to-transformer pairing algorithm

Parameter	**Without Base Noise**	**With Base Noise**
Measurement Interval	Any interval	30 and 60 min.
Average Vs Instantaneous	Any type for smaller intervals. Average for larger intervals (>30min)	Averaged data performs well especially in the presence of time synchronization issues.
Data Resolution	At least 0.1V and 0.1kW (1 decimal) required.	
Meter Bias	No Impact	
Meter Precision (Noise)	0.35-0.55 % maximum noise in Voltage: Corresponds to ANSI standards class 0.2	
Time Synchronization	Averaged and larger measurement intervals are more robust	
Missing Data	Low sensitivity to missing data	
Data Quantity	~ 1 month	Error decreases steadily with larger quantities of data.

The tables above represent two specific examples of defining AMI data collection recommendations for two specific algorithms. The following are some general AMI data recommendations that come from the work conducted across the project as a whole.

- Measurement Interval: 15-minute or 30-minute intervals for model calibration tasks, for load modeling at least 15-min for medium voltage and 5-min or better for low voltage applications.

- Average voltage generally performs better for these tasks than instantaneous voltage.

- Data Resolution: At least one decimal point for voltage magnitude, real power, and reactive power measurements.

- Data Availability: This varies significantly among algorithms, though all algorithms tested require at least one month of available data, and more data tends to improve algorithm performance.

- Missing Data: Requirements vary significantly depending on the algorithm in question
- Time Synchronization Issues: All algorithms tested had significant issues if the AMI timeseries were out of sync.

3. TEST CIRCUITS

This project leverages both synthetic datasets as well as a number of different utility datasets. The primary project goal is to reduce errors and unknowns in distribution system models. One key assumption from the beginning is that the exact quantity and form of the model errors is unknown. See Section 2.1 and [2] for more details. In addition, distribution system data itself (AMI, SCADA, or additional sensors) contains an unknown quantity of several different types of data issues. See Section 2.2 and [29], [32] for more details. Another consideration is that, in most cases, the utility datasets will not contain the ground-truth labels for the model characteristics under consideration. The project was fortunate to work with Utility #1 who provided some field-verification, but even in that case it was only for a small subset of the total feeder. These factors make the use of synthetic data essential for the algorithm development process. The synthetic data is fully controlled, with ground-truth labels, and can be injected with any amount of known data issues such as measurement noise, and any quantity of erroneous labels, such as incorrect phase labels. This allows for rigorous algorithm testing for robustness under a variety of conditions ranging from expected to extreme. Then in the second stage of research the algorithms can then be tested on the utility data with confidence in how the algorithms are likely to perform under real conditions. This combination of using both synthetic data and utility data provides the best environment for developing algorithms under known conditions and also testing them on real systems. The follow sections describe the synthetic dataset, Section 3.1, and the utility datasets, Section 3.2.

3.1. Synthetic Test Circuits

3.1.1. EPRI Ckt5 Feeder Model

Distribution load modeling research discussed in Section 10 along with several other research areas discussed in this report utilize the OpenDSS model of EPRI "Ckt5", which is included in the OpenDSS software distribution[1]. Ckt5 is a model of a real US 3.2-mile long 12.47 kV distribution feeder that supplies 1,379 residential loads with 591 service transformers. The original Ckt5 model has no load tap changer or line voltage regulators but includes four switched-capacitor banks. The key parameters of Ckt5 model are listed in Table 3-1.

Table 3-1. Key parameters of Ckt5

Parameter	Value
Medium voltage base kV (line-to-line)	12.47 kV
Number of devices	4,411
Number of buses	2,998
Number of nodes	3,437
Max. bus distance miles (km)	3.24 (5.22)
Number of transformers	592
Number of lines	2,429
Number of loads	1,379
Number of capacitors[2]	4 (600 kvar, 300 kvar, 450 kvar, 600 kvar)
Number of voltage regulators	0
Substation transformer type	10 MVA, 115/12.47 kVLL, D-Yg

[1] The OpenDSS Ckt5 feeder model can also be separately downloaded from: http://svn.code.sf.net/p/electricdss/code/trunk/Distrib/EPRITestCircuits/ckt5/

[2] The original Ckt5 feeder model has four switched-capacitors that were disabled to allow focusing on the impact of different load modeling aspects, as opposed to the impact of mismatching controller states between simulations.

Parameter	Value
LTC	No
Number of feeders supplied by the substation transformer bank	1
Load voltage sensitivity model	Constant P and Q[3]

Ckt5 feeder layout contoured with the distance from the substation and the baseline[4] voltage profile is shown in Figure 3-1 on the left and right, respectively.

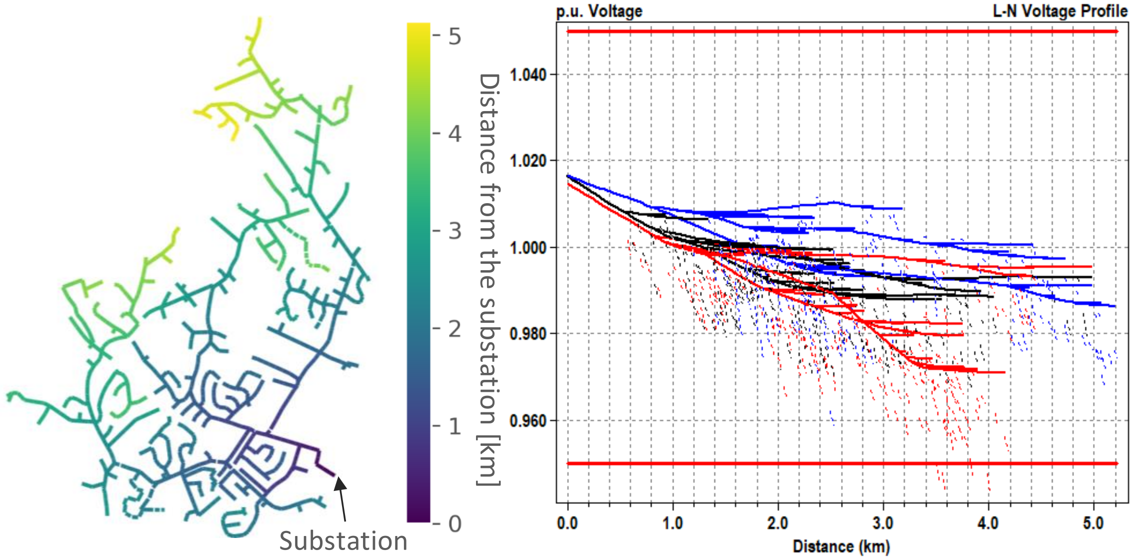


Figure 3-1. Ckt5 feeder layout contoured with the distance from the substation (left) and the baseline voltage profile (right)

Ckt5 feeder supplies primarily residential customers. In the original Ckt5 model included in OpenDSS distribution, the secondary circuits are modeled as follows. Each secondary circuit consists of a service transformer rated from 5 to 75 kVA supplying from one to six customers and each customer is served with a dedicated 100-ft 2/0[5] triplex service line. To better represent the diversity in real-world secondary circuit construction (and the resulting voltage drops/rises), the service line lengths were randomly modified to range from 40 feet to 180 feet. For details, see [33].

Various modified versions of the Ckt5 feeder model were implemented for the different research areas discussed in this report. As follows, selected versions are briefly introduced.

3.1.1.1. Ckt5 with 1-Minute AMI Load Profiles

Several research areas discussed in this report utilized a modified version of the Ckt5 model with 1-minute AMI load models established as follows, for details see [33]. For this Ckt5 version, each load was assigned a real power profile created based on real residential customer load data that was obtained from PecanStreet Inc. The load data consists of 1-minute time-averaged kW

[3] By default, all loads in Ckt5 are modeled with OpenDSS load model=4 (nominal linear P and quadratic Q) with CVRwatts=0.8 CVRvars=3. However, for the load modeling research discussed in Chapter 10, the voltage-sensitivity models of all loads were replaced to constant active and reactive power consumption.

[4] The "baseline" voltage profile refers to the voltage profile obtained without load multipliers and before applying load profiles.

[5] The original service line is not mentioned in Ck5 documentation but its impedance corresponds roughly to a triplex service line with gauge 2/0.

measurements recorded in Austin, Texas. The reactive powers of loads were represented through random inductive power factors between 0.79-0.99.

3.1.1.2. Ckt5 with Hourly AMI Load Profiles

Several research areas discussed in this report utilized a modified version of the Ckt5 model with hourly AMI load models that were established as follows, for details see [33].

This modified version of Ckt5 leveraged an AMI data set obtained from a US utility that included residential customer active and reactive power measurements for a full calendar year at 15-min time granularity (35,040 samples per profile). The data set included net metering for active and reactive power for 1,309 residential customers. The data set also included PV active power generation measurements for 691 of the 1,309 customers, who had a PV system. The native load was calculated for each customer from the customer's net metering and PV generation data (native load = net load + PV generation). As this utility AMI dataset was unrelated to the Ckt5 feeder, the data had to be associated to the model in such a way that was representative of an actual utility feeder. To create the reference case load model for Ckt5 using the AMI data set, a single (active and reactive power) native load profile from the AMI data set was assigned to each of the Ckt5 loads.

3.1.1.3. Ckt5 with AMI Load and PV Models

This Ckt5 model version was based on the version discussed in Section 3.1.1.1 but also included PV system models with annual AMI generation profiles, for details see [33], [34]. The PV systems were added as follows. Each feeder load, which was assigned AMI load profiles from a real customer with a PV system, was also assigned a PV system with the corresponding DC and AC ratings, and PV AMI real power measurement profile. The random AMI load and PV profile assignment process resulted in random PV distribution on the feeder shown in Figure 3-2 on the left. The simulated feeder net real and reactive power over the year is shown in Figure 3-2 on the right.

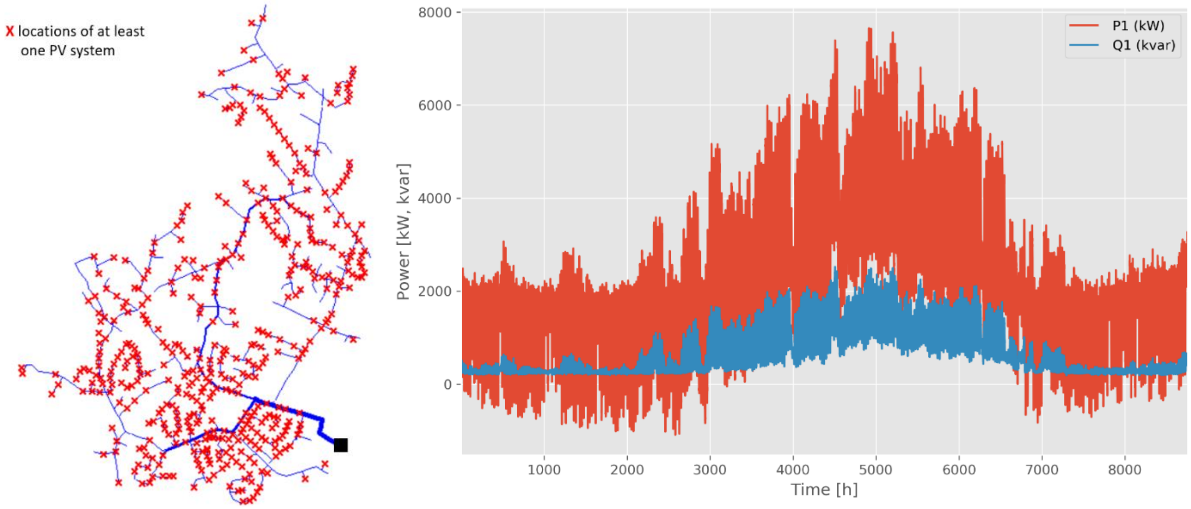


Figure 3-2. The topology of the modified version of EPRI's Ckt 5 feeder model highlighting service transformers with at least one downstream PV system (left) and simulated feederhead real and reactive power (right)

3.2. Utility Systems

There are three utilities that provided the test circuits leveraged in this project, with a total of nine feeders across the three utilities. The utilities are varied in location and characteristics, and Table 3-2 provides a broad overview of the feeder characteristics. The following sections provide further details on the utilities and feeders.

Table 3-2 - Utility Dataset Overview

Datasets	Number of Customers	Voltage (L-L kV)	Peak Load (MW)	Length (km)	Line Voltage Regulators	Capacitors	Additional Sensors
Utility #1, Feeder #1	847	12.47	3.6	5.0	1	1 fixed, 2 switched	10
Utility #1, Feeder #2	1285	11.85	4.7	10.1	3	2 switched	14
Utility #1, Feeder #3	546	12.47	2.1	3.8	0	1 fixed, 1 switched	4
Utility #1, Feeder #4	1053	11.85	3.3	11.2	3	2 fixed, 1 switched	7
Utility #2, Feeder #1	1104	12.47	2.8	22.3	3	2 fixed, 2 switched	0
Utility #2, Feeder #2	1410	12.47	2.8	22.0	5	1 fixed	0
Utility #2, Feeder #3	1153	12.47	2.0	5.4	0	0	0
Utility #2, Feeder #4	2385	12.47	3.8	8.2	1	2 fixed, 1 switched	0
Utility #2, Feeder #5	1309	12.47	1.8	2.5	1	1 fixed	0
Utility #2, Feeder #6	1188	12.47	1.4	3.2	0	1 fixed, 1 switched	0
Utility #3, Feeder #1	766	8.32	8.9	12.0	7	2 fixed, 1 switched	0
Synthetic	1379	12.47	7.3	5.2	0	4 fixed	10

3.2.1. Utility #1

Utility #1 used in this project is located in the southern United States and data from two feeders are used. The first feeder serves ~800 customers, includes one voltage regulator, and has 9 IntelliRupter® devices. The second feeder servers 1,255 customers and has 13 IntelliRupter® devices. The third feeder has 546 customers and 4 IntelliRupters®; likewise the fourth feeder serves 1053 customers and has 7 IntelliRupters®. Please see Table 3-2 (Utility #1 rows) for further details on the feeder characteristics. The AMI meters and IntelliRupters® provide timeseries data at 15-min intervals. The AMI data has individual measurements for real power and voltage, and the additional sensors have individual measurements for real power, reactive power, and voltage on each phase. Note that we are not referring to phasor measurement unit (PMU) devices; the additional sensors in this work are not high-resolution sensors but sensors recording data at a similar resolution to AMI meters.

Figure 3-3 shows a feeder diagram for Feeder #1 including markers for the voltage regulator (dark red diamond), the substation (black star), and the IntelliRupter® devices (gold pentagons). The additional sensors are distributed throughout the feeder, giving good location diversity. The customers are grouped by transformer, and the transformer locations are shown in circles colored by the original phase labels in the distribution system model because coordinates are available for transformer locations but not individual AMI meters.

The IntelliRupter® devices have several decimal points of precision in their measurements for voltage. However, the AMI voltage data does not include any decimal points in the recorded measurement. The smart meters measure the voltage much more accurately, but no decimal points were retained for data storage purposes. A full year of AMI and sensor data is available, with roughly 0.6% missing data during that year.

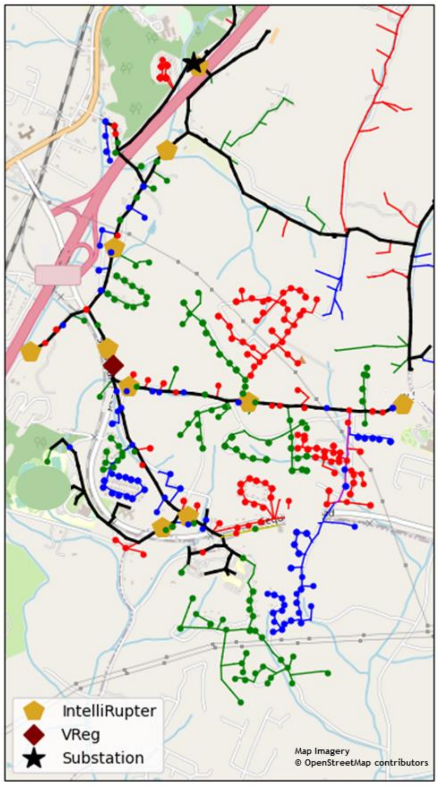


Figure 3-3. Full feeder diagram for Utility #1, Feeder #1 with customers colored by their original phase labels with red as Phase A, green as Phase B, and blue as Phase C. Additional sensors are shown in gold markers, the voltage regular as a red diamond, and the substation as a black star.

3.2.2. Utility #2

The AMI data used in this research covers an approximately 15-month period for all customers on six feeders. Table 3-2 (Utility #2 rows) shows a comparison of key characteristics of the six feeders. The data comes from a utility in the north-eastern United States. The utility has installed AMI smart meters for each customer in these feeders. The dataset contains individual AMI data for each customer. Each individual profile contains 15-minute average measurements for real power, reactive power, and voltage, as well as power generation type and a phase label (possibly incorrect) from the utility company. The power and voltage measurements are taken to an accuracy of four decimal places. Figure 3-4 shows a diagram of Feeder #3 as an example. The dataset for that feeder contains ~8% missing data spread throughout the customers, and this can be a challenge for certain algorithms.

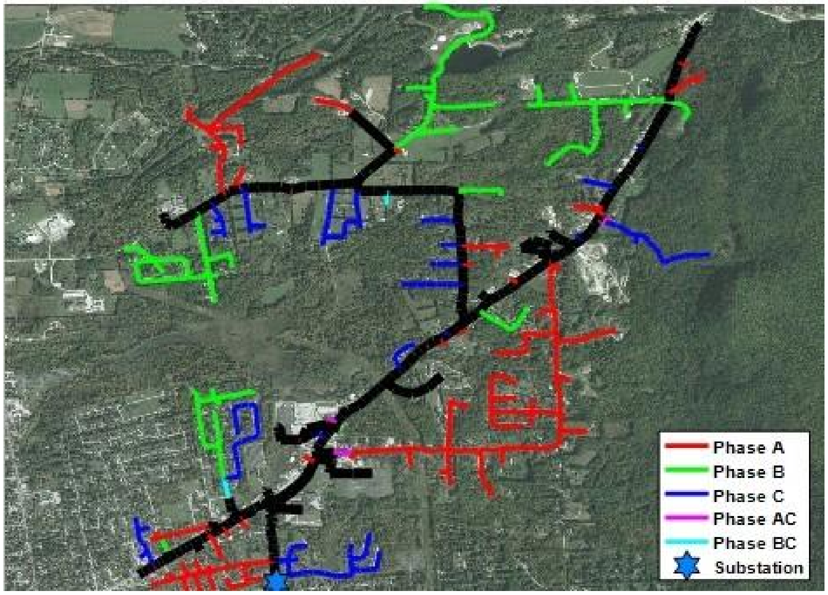
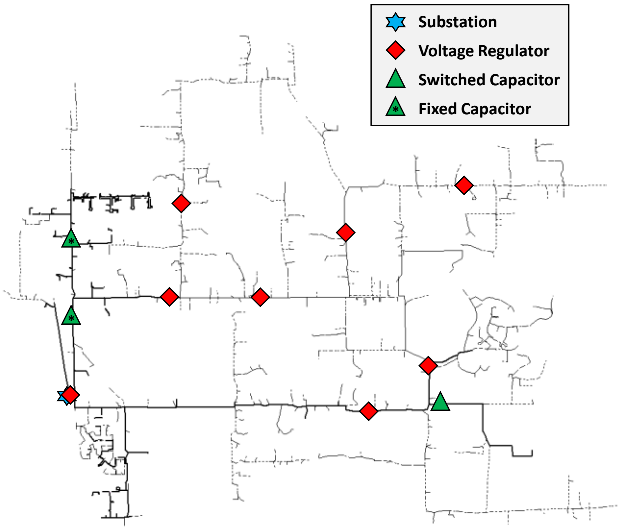


Figure 3-4. Diagram of Feeder #3 from Utility #2 with lines colored by phase label

3.2.3. Utility #3

The distribution management system (DMS) load modeling assessment discussed in Section 10.8 was conducted on a single utility feeder. The feeder topology is illustrated in Figure 3-5 on the left and the feeder key equipment are listed in the table in Figure 3-5 on the right. The utility feeder planning model was converted into OpenDSS.



Feeder Characteristic	Characteristic Value
Voltage (kV line-line)	8.32
Peak load (MW)	~8.9
Further bus distance (miles)	~7.5
Number of nodes	2,498
Number of sections	2,563
Number of loads	766
Types of loads	Mainly residential
Capacitors	2 fixed, 1 switched
Voltage regulators	8 voltage-controlled
DER	Limited BTM DER
Secondary circuit models	No*

Figure 3-5. The case study utility feeder model

4. PHASE IDENTIFICATION

4.1. Task Overview

Phase identification is the task of grouping the customers in a feeder by their phase. Looking at Figure 4-1, we want to know for each customer which phase that customer is connected to. We frame this problem by looking at the voltage time series data from customer AMI meters and data from additional sensors from the medium voltage if that is available. It is a well-established fact that utilities have an unknown quantity of error in the phase labels for their customers, and some utilities did not record phase labels at all. Accurate phase labels are important for load balancing of distribution systems, hosting capacity analysis, and other grid planning tasks. This project developed three phase identification algorithm, one of which leverages only voltage data from AMI meters (Section 4.3), one which leverages both voltage data from AMI meters as well as voltage data from additional sensors placed on the medium voltage (Section 4.2), and one phase identification method which is based on a state estimation approach (Section 4.4).

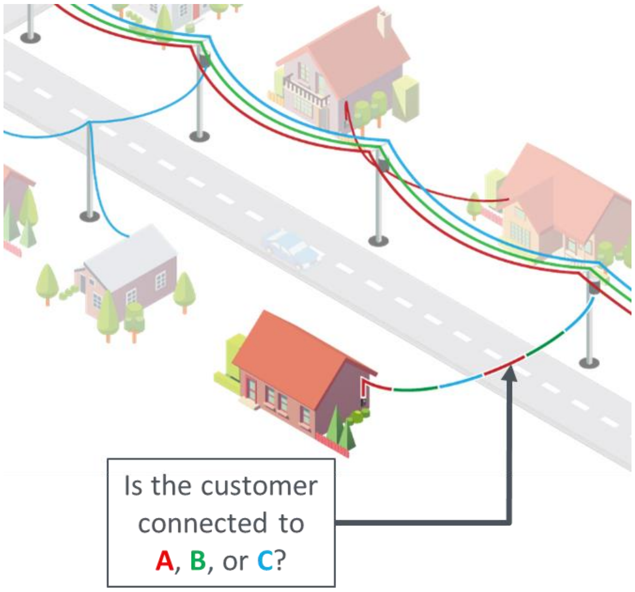


Figure 4-1. Phase Identification illustration

4.2. Sensor-based Phase Identification

The first phase identification method developed over the course of this project uses voltage timeseries data from customer AMI in addition to time series data from additional sensors placed on the medium voltage portion of the distribution system. Any type of sensor that records timeseries data at a similar resolution to the customer AMI data can be used. An example of this type of sensor are the IntelliRupter© relays used by Utility #1 described in Section 3.2.2. This type of sensor was only employed due to the utility data available for this work, and this does not constitute an endorsement of any particular type of sensor device. This work resulted in a conference paper publication [35].

4.2.1. Methodology

The proposed methodology leverages data from the AMI meters and the additional sensors and uses Pearson correlation coefficients between data streams of voltage timeseries to perform the phase identification task. This methodology is designed to work on a single feeder at a time and the data mentioned in the remainder of the paper is assumed to be from a single feeder. Figure 4-2 shows a flowchart of the overall methodology.

There are two pre-processing steps prior to the methodology shown in the flowchart. First, the voltage data is converted into a per-unit representation. Second, the difference between adjacent timesteps is taken, converting the original timeseries into a change-in-voltage timeseries. These steps are taken to enhance the utility of the Pearson correlation coefficients. Foundational work demonstrating this concept can be found in [13], [36], [37].

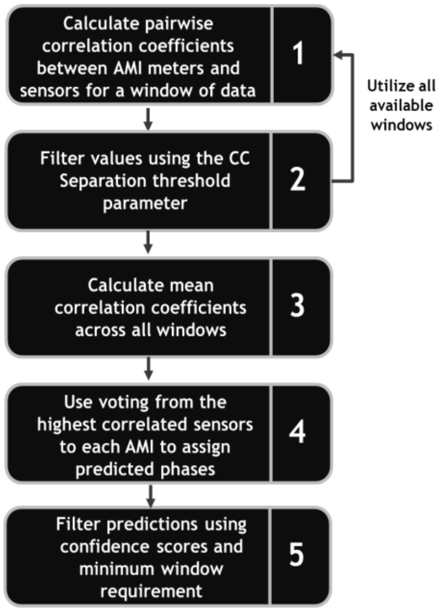

Calculate pairwise correlation coefficients between AMI meters and sensors for a window of data
1
Utilize all available windows
Filter values using the CC Separation threshold parameter
2
Calculate mean correlation coefficients across all windows
3
Use voting from the highest correlated sensors to each AMI to assign predicted phases
4
Filter predictions using confidence scores and minimum window requirement
5

Figure 4-2. Methodology flowchart performing voltage correlation coefficient analysis between all customer AMI voltages and sensor voltages for multiple windows of historical timeseries data

An ensemble approach is used to process the entire available dataset. In Step 1 of Figure 4-2, a period, or window, of data is selected for processing; a window of eight days was used in this work. Any customers with missing data during this time period are removed and correlation coefficients are calculated between the remaining customers and the available sensor data streams (i.e., Step 2). Figure 4-3 shows an example of this for one customer in one window. The correlation coefficients are shown between one customer and each available sensor data stream. All subsequent windows are processed in the same way, collecting the calculated correlation coefficients, until all available data has been used. The ensemble is used, both to deal with having missing data throughout the dataset, as well as to leverage the power of an ensemble approach, [13], [37]. In Step 3, the mean correlation coefficient is calculated using the correlation coefficients across the entire set of windows. This is shown in Figure 4-4 for the same customer shown in Figure 4-3. In Figure 4-4 the average correlation coefficient was taken across all windows (the time dimension). As you can see the correlation coefficients have shifted slightly in the averaged case.

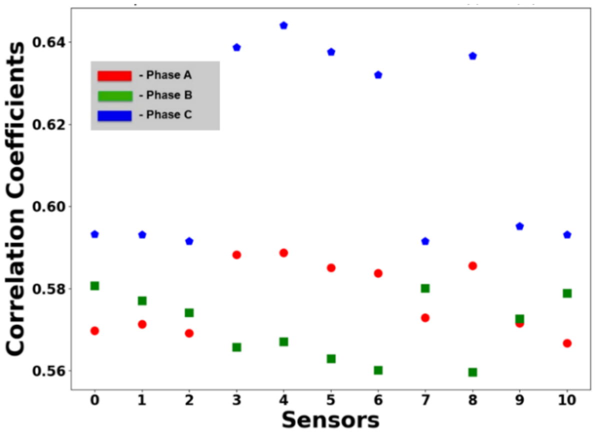


Figure 4-3. Correlation coefficients between one customer and all sensors for one eight-day window

In Step 4, a predicted phase for each customer is assigned based on votes from the highest correlated sensors. Our work used the top 5 most correlated sensors for the voting, but this number would depend on the number of sensors available in the system. Using a voting approach rather than taking only the highest correlated sensor provides additional algorithm robustness under a variety of feeder conditions. Finally, in Step 5, the customer predictions are filtered based on a minimum window requirement per customer as well as filtering based on confidence scores. Any customers with low confidence scores or too few windows available are considered to have low confidence in the predicted phase and should be either re-evaluated using more data or flagged for further post-processing investigation.

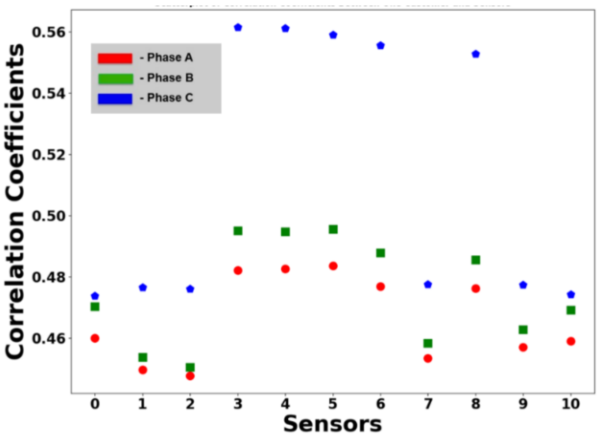


Figure 4-4. Correlation coefficients between one customer and all sensors averaged over all the windows (averaged over time)

4.2.2. Validation of Sensor Phase Labels

Prior to considering the task of predicting the customer phase labels, the accuracy of the phase labels on the additional grid sensing devices must be considered. Any errors in the sensor phase labeling would be transferred to the customer phase identification and render that analysis incorrect. Therefore, an analysis was conducted to validate the phase labeling of each sensor data stream by comparing the data streams between sensors and also with the substation. This was done by calculating correlation coefficients between all sensor data streams using the window method described above. The highest correlation coefficients were used as "votes" for the phase of the sensor data stream in question. If the highest correlated sensor data streams agree with the labeled phase of the data stream in question, then the phase label for that sensor data stream is considered validated.

Figure 4-5 demonstrates this approach. One sensor data stream was chosen, representing the center of the radar plot–each additional sensor is a point around the circle, and points are plotted for the correlation coefficients between the chosen sensor data stream and all other sensor data streams. Each radial axis has three points, one for the correlation coefficients for Phase A (red), Phase B (green), and Phase C (blue) of the other sensors respectively. The correlation coefficient magnitude increases as the point approaches the edge of the circle. It is clear from this plot that all the other sensors agree that this sensor data stream is measuring Phase A as it was originally labeled. This is shown by the outer red circle on the radar plot. All the sensor data stream phase labels were validated in this manner and no discrepancies were found with the original phase labels. This allows for the use of the sensor phase labels in the customer phase identification methodology described in Section 4.2.1. Had there been discrepancies in the sensor phase labeling, this could have been addressed in advance. Using this type of verification method does require that there be multiple sensors on the feeder and that the majority of the original sensor phase labels are accurate. We believe this is likely to be true in general as they are medium-voltage sensing devices. However, a utility could make this determination based on their own knowledge of the feeder. If this type of validation of the sensor phase labels is infeasible, then the sensor phase labels must be manually verified in advance of using the phase identification method proposed in this work.

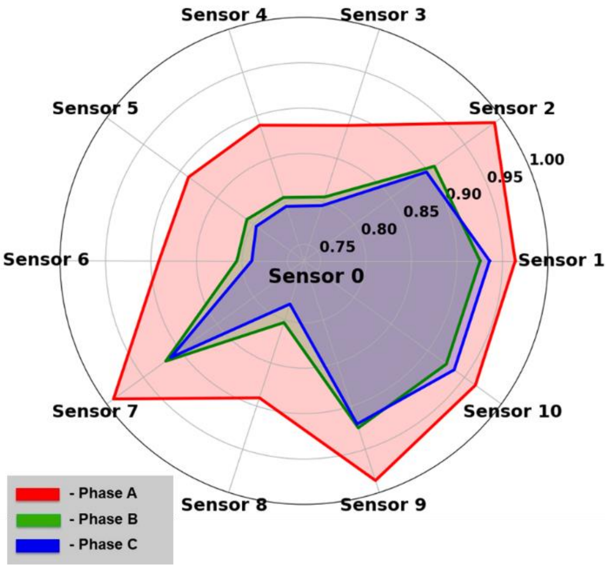


Figure 4-5. Example of phase validation for a sensor data stream. This sensor data stream (middle point) was labeled on Phase A and the highest correlated data streams from all the other sensors is from Phase A (red). The units of the radial axis are correlated.

4.2.3. Comparison of Sensor Method and Substation Method

This section shows the results of comparing the method based on leveraging additional grid sensors described in Section 4.2.1 with a similar method from the literature that utilizes correlations only with the substation. The method using only the substation data is similar to the methods in [38], [39]. However, the pre-processing and window ensemble steps described in Section 4.2.1 are also used to keep the comparison shown here as similar as possible. Thus, the only difference between the Sensor Method results and the Substation Method results is the use of the additional grid sensors.

4.2.3.1. Correlation Coefficient Spread

This section directly compares the correlation coefficients produced by the two methods. Figure 4-6 shows a boxplot for the correlation coefficient spread using the Sensor Method. Each subplot represents data from customers labeled in the utility model on Phase A, Phase B, and Phase C, respectively. Within each subplot, there is a box for each sensor phase. Thus, according to their labeled phase, each customer is included in one of the subplots and all boxes in that subplot, and correlation coefficients with the most correlated sensor are plotted for that customer. In each subplot, it is clear that the box for the labeled phase has higher correlation coefficients on average than the other two phases. Keep in mind that we expect there to be some degree of phase labeling errors in the distribution system model, so we would not expect the complete separation of the boxes. Contrast Figure 4-6 with Figure 4-7, which shows the same type of figure for the Substation Method. In Figure 4-7, there is no separation between the phases; each box contains approximately the same spread of correlation coefficients for each phase, implying that the correct phase labels cannot be accurately determined in general.

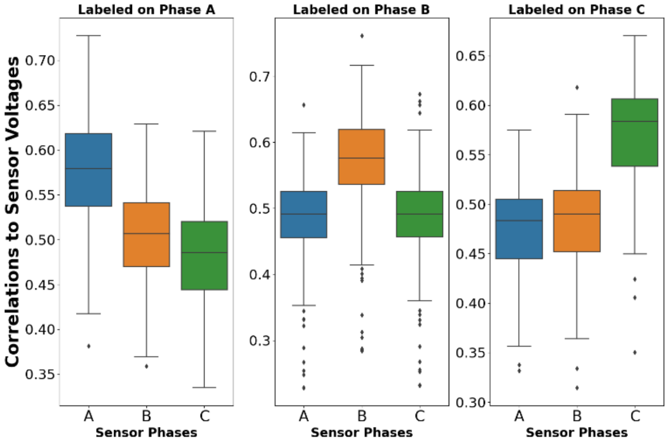


Figure 4-6. Sensor correlation coefficient boxplot showing that customers on a particular phase are significantly more correlated with the sensor voltage measurements on that same phase

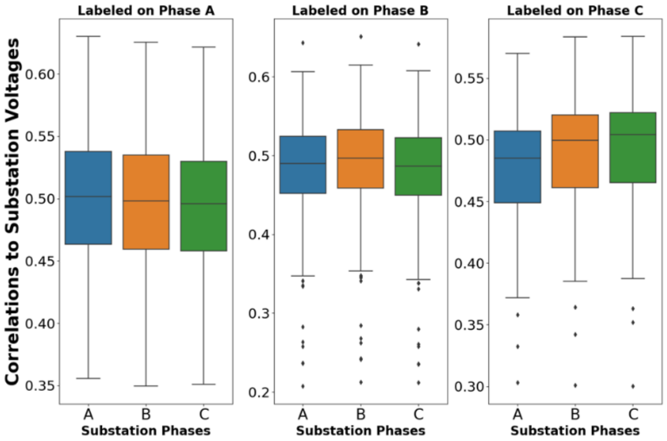


Figure 4-7. Substation correlation coefficient boxplot showing that you cannot easily distinguish customer phase using substation voltage measurements that are separated by a voltage regulator

For this work, only the customers downstream of the regulator, ~600 customers, are considered, as the customers separated from the substation by a voltage regulator are the most likely to have issues correlating correctly with the substation voltages. In this case, the AMI data, as well as the IntelliRupter® data was averaged to 30-min intervals to provide better correlation coefficients.

4.2.3.2. Confidence Scores

For a utility company to expend man-hours and financial resources based on algorithm predictions, there must be a methodology to judge confidence in the algorithm results overall and confidence in individual phase predictions. There are many feeder and customer characteristics that may affect the results. Figure 4-6 demonstrates that the phase identification task should be possible, but how confident are the individual phase predictions for each customer?

We propose four different metrics of confidence in customer predictions. Correlation Coefficient Separation is based on an analysis of the correlation coefficients for a particular customer, the Window Voting Confidence Score is an analysis of the window ensemble shown in Figure 4-3, the Sensor Agreement Score leverages the ensemble of all sensors on the feeder, and finally, a Combined Confidence Score combines the Window Voting and Sensor Agreement scores. For the first two scores, the results for the substation method are also shown. Note that all these confidence scores are data-driven metrics for prediction accuracy, and they can be derived directly from the results of the correlation coefficient calculations for each customer. Separate confidence metrics can be derived based on the system topology, such as if all customers on a transformer agree about the phase connection, but the objective of the work is to provide entirely data-driven without any reliance on access to topology information or relying on the accuracy of the system topology data.

The Correlation Coefficient (CC) Separation is based on looking at the highest correlated sensor for a particular customer. That sensor will have a correlation coefficient for each phase that it measures, thus three correlation coefficients to each customer. The CC Separation finds the sensor with the highest correlation coefficient to that customer and subtracts the next highest correlation coefficient on that sensor. That difference is defined as the CC Separation. The larger this difference is, the better separation there is between the correlation coefficient for the predicted phase of the customer and any other correlation coefficient. This can be seen in Figure 4-4 for one customer. For that customer, Sensor 3 has the highest correlation coefficient (blue circle); the CC Separation is calculated as the difference between the correlation coefficients marked by the blue circle and green square for Sensor 3. If the difference is small, then the confidence in the predicted phase will be low. This metric is used for filtering in Step 2 of Figure 4-2. If the CC Separation between a customer and a particular sensor is too low that means that multiple phases from a sensor were similarly correlated to that customer. Clearly that is not a desirable situation, and the correlation coefficients from that sensor are excluded, in that window, for that customer. This situation was seen to occur for sensors that were separated from the customer by the voltage regulator. Filtering these coefficients helps to ensure that only high-quality coefficients are included in the mean in Step 3.

The Window Voting Score leverages the ensemble nature of steps 1 and 2 shown in Figure 4-2. For each window, correlation coefficients are calculated between each customer (that is not missing data during that period) and each sensor. In the full methodology, the mean correlation coefficient is taken, however the windows can be considered individually and used to create this metric. In each window, the results of the sensor voting is considered a "vote," and the window voting metric is the ratio of windows that agree versus the total number of windows. For example, consider the case that there are 10 windows available. If, in 9 of the windows, the sensor voting resulted in Phase A, and in 1 of the windows the sensor voting resulted in Phase B, then the Window Voting Score would be 0.9. This can be interpreted as "90% of the windows throughout the year agreed on the predicted phase for this customer."

The Sensor Agreement Score is defined as the ratio of the sensors that agree on the predicted phase of a customer versus the total number of sensors in the system. For example, if there are 10 sensors used in the voting and 8 of them agree that the predicted phase for a customer is Phase A and two of the sensors predicted Phase B, then the Sensor Agreement Score would be 0.8, signifying that 80% of the sensors in the system agree on the predicted phase.

In practice, using three different confidence scores is onerous for standard operation in the utility setting. We propose to use the CC Separation Score as a first-pass filter to filter out customers with low CC Separation Scores for different treatment as this method cannot successfully perform the phase identification task on those customers. We also propose combining the Window Voting Score and the Sensor Agreement Score to form a single confidence score to be used as a second-stage filter. Both the Window Voting Score and the Sensor Agreement score will be between 0 and 1 and represent the same type of ratio. Thus, they can be multiplied together and retain their basic meaning and interpretation. The Combined Confidence Score can be used to further quantify the confidence in individual customer phase predictions after the CC Separation Score filter has been applied. A value of 1 for the Combined Confidence Score means that all sensors and all windows agree on the predicted phase of that customer. Figure 4-8 shows a histogram of Combined Confidence Scores for two feeders from Utility 1. Notice the couple of customers in the blue histogram on the left side of the figure. These are low confidence customers who would be excluded from the final results and flagged for potential further investigation.

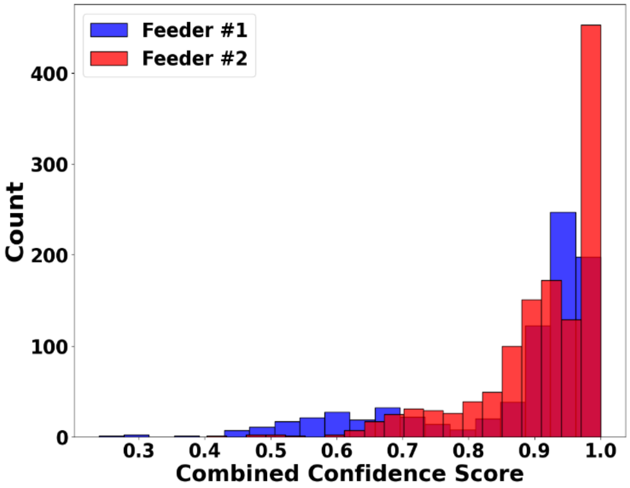


Figure 4-8. Combined Confidence Scores for Utility 1

4.2.4. Synthetic Data Results

The first simulation conducted provides base case results on the synthetic data for the sensor-based method. Normally distributed noise was injected into both the AMI data and the sensor data. For the AMI data, normally distributed noise with a mean of 1 (per-unit voltage) and standard deviation of 0.07% is injected into each customer in the dataset. The standard deviation came from a meter testing report provided by one of our utility partners. For the sensors, the standard deviation was 0.04%. This came from analysis of the noise present in the IntelliRupter® measurements from Utility 1. As the devices record on both sides of the relay, closed relays should have the same measurements on both sides allowing for an estimation of the noise present in the measurements.

Figure 4-9 shows the results of a missing data sweep and a measurement noise sweep, showing the CC confidence scores on the y-axis. Figure 4-10 shows results from the same sweeps with the Combined Confidence Scores on the y-axis. Notice that the algorithm is completely robust to missing data as low as the minimum requirements are met. One month of available data per customer, not necessarily contiguous, was determined to be the minimum requirement. Step 5 in Figure 4-2 shows the filtering step based on that requirement.

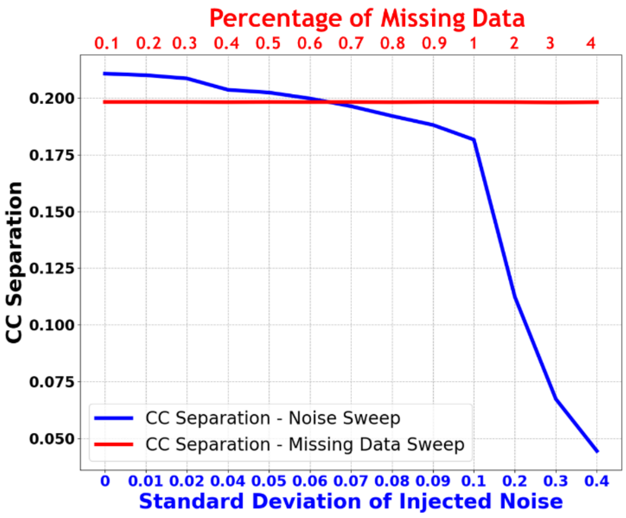


Figure 4-9. Synthetic data results showing a missing data sweep and noise sweep for the CC Separation metric

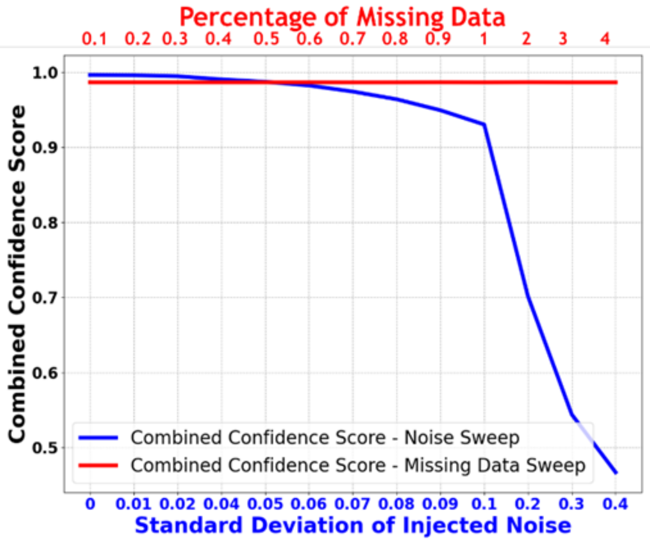


Figure 4-10. Synthetic data results showing a missing data sweep and noise sweep for the Combined Confidence Score metric.

4.2.5. Utility Results

The proposed Sensor Method identified 6 customers on Feeder #1 for Utility #1 who are predicted to have a different phase label than in the original utility model, while the substation method predicted 14 customers to have a different label. The set of 14 customers identified by the substation method includes 5 of the 6 customers identified by the proposed method. The confidence scores for the additional customers identified using the substation method are significantly lower than the confidence scores of the 6 customers identified using the sensor method. This indicates that those additional customers are likely to be false positive results and demonstrates the effectiveness of the sensor-based method.

The utility company responsible for this feeder performed field verification on the six customers predicted to be incorrect by the sensor phase identification method. In all six cases, the algorithm predictions were shown to be correct. Table 4-1 shows those customers with their confidence scores. There was also one low confidence customer, shown in the last row, which was excluded from the final results. Figure 4-11 shows satellite imagery of one of the customers, marked with the yellow pin. In the utility model, this customer was serviced by the transformer on the right and connected to Phase A. The predicted label was Phase C, and this was confirmed by field personnel. This customer is actually serviced by the Phase C transformer on the left side of the image. A field visit by utility personnel to visually inspect the secondary voltage connections confirmed the prediction. In Figure 4-12, the other five customers are shown on two laterals. The original utility labeling (top) showed that the four customers on the lateral on the left side were connected to Phase B and the remaining customer on the right side was connected to Phase C. The algorithm predictions are shown in the bottom panel and were verified by field personnel. The four customers on the lateral on the left side are connected to Phase C and the customer on the right side is connected to Phase B.

Table 4-1. Utility #1, Feeder #1 results

	Confidence Scores			
	CC Separation	Window Voting	Combined	Comments
1	0.089	0.9	0.9	Field-verified
2	0.13	0.98	0.98	Field-verified
3	0.12	0.97	0.97	Field-verified
4	0.13	0.94	0.94	Field-verified
5	0.15	0.97	0.97	Field-verified
6	0.066	0.94	0.94	Field-verified
7	0.088	0.48	0.24	Low Confidence

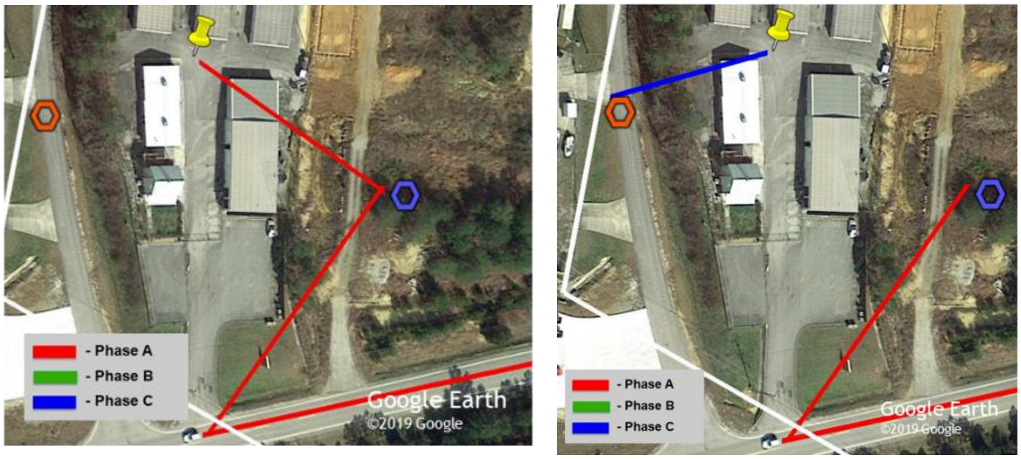


Figure 4-11. Satellite image with the left showing the original utility label of Phase A for the customer marked with a yellow pin and the right showing the true service transformer and phase C labels verified via field verification

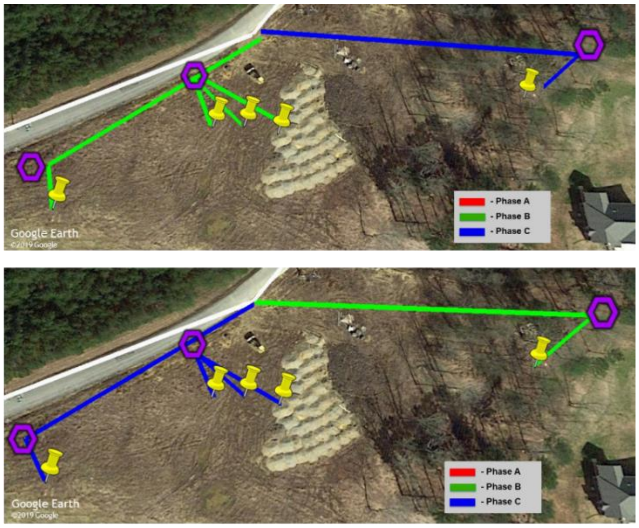


Figure 4-12. Satellite image showing five customers on two laterals predicted to have incorrect phase labels. The original utility labels (top) show four Phase B customers and one Phase C customer and the predicted labels (bottom), verified with field verification show four Phase C customers and one Phase B customer.

Ten customers were flagged as being incorrect on Utility #1 Feeder #2. Two of those were found by the utility to be on a different feeder. Table 4-2 shows the results for Feeder #2 with their confidence scores, excluding the two customers located on an adjacent feeder. Five customers were field-verified, and 3 customers were excluded due to low confidence scores. Thus, for all customers on the feeder, the algorithm achieved 100% accuracy on the high confidence flagged customers. The low confidence customers are flagged for further investigation. Figure 4-8 above shows the histogram of the Combined Confidence Scores for both feeders.

Table 4-2. Utility #1, Feeder #2 results

	Confidence Scores			
	CC Separation	Window Voting	Combined	Comments
1	0.143	1	1	Field-verified
2	0.121	0.929	0.868	Field-verified
3	0.105	0.951	0.951	Field-verified
4	0.116	0.965	0.965	Field-verified
5	0.3	1	1	Field-verified
6	0.039	0.578	0.463	Low Confidence
7	0.029	0.5	0.5	Low Confidence
8	0.002	0.415	0.667	Low Confidence

Eight customers were flagged on Utility #1 Feeder #3; those results are shown in Table 4-3. All three high confidence customers were field-verified to be accurately predicted by the algorithm. In Utility #1 Feeder #4, 84 customers were identified as potential incorrect phase labels. Sixty-nine of those customers were on a single lateral, which is all of the customers serviced by that lateral, see Figure 4-13. The green lateral shown in Figure 4-13 was identified (and field-verified) to be a Phase C lateral, as predicted by the algorithm. Again, all high-confidence customers, including the lateral, were field-verified by the utility as being accurately predicted by the algorithm. In each of the four feeders investigated, the algorithm was shown to be 100% accurate in the field-verification.

Table 4-3 – Utility #1, Feeder #3 Results

	Confidence Scores			
	CC Separation	Window Voting	Combined	Comments
1	0.1	1.0	1.0	Field-verified
2	0.067	0.92	0.92	Field-verified
3	0.063	0.92	0.92	Field-verified
4	0.099	0.6	0.4	Low Confidence
5	0.064	0.9	0.45	Low Confidence
6	0.018	0.56	0.56	Low Confidence
7	0.011	0.5	0.5	Low Confidence
8	0.054	0.75	0.5	Low Confidence

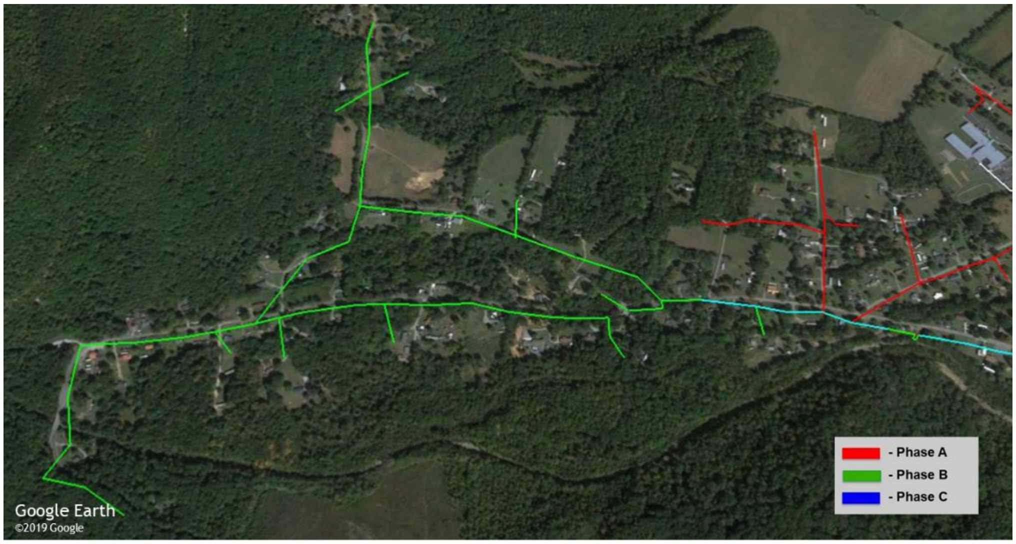


Figure 4-13 - Google Earth image of the lateral identified in Utility #1, Feeder #4

A parameter tuning analysis was also conducted using the Utility #1 datasets. The parameters of window size and filter criteria were examined, and a wide range of those parameters are acceptable and produce the same results shown here. Thus, the algorithm is not significantly sensitive to those parameters.

4.2.6. Summary of Sensor-based Phase Identification

This section presents a novel methodology for the phase identification task that leverages voltage magnitude data from AMI meters and additional, medium-voltage sensors providing 15-minute voltage measurements per phase. Although the feeder in this work uses IntelliRupter® sensors, this method generalizes to other, similar grid sensors. Correlation coefficient analysis is leveraged to provide a predicted phase for each customer along with easily interpretable confidence metrics for each prediction. The proposed method is shown to achieve significantly improved performance over the algorithm only using substation data. This is particularly important in the case where there are voltage regulators on the feeder. The separation between AMI meters and the substation data by the voltage regulator renders correlations with the substation unusable. Using the additional grid sensors, combined with customer AMI data and correlation coefficient analysis, provides a straightforward, interpretable solution to the phase identification task. The utility field-verified all 6 customers predicted to be incorrect on the feeder in question, providing excellent evidence of the efficacy of this methodology. This utility is also in the process of implementing this method into full-scale usage in their service area. This work resulted in this publication [35].

4.3. Ensemble Spectral Clustering Phase Identification

The ensemble spectral clustering phase identification algorithm is designed to use solely the voltage timeseries data from customer AMI meters; optionally it can also leverage the existing phase labels. This algorithm went through several stages of development; the following section presents the final version of the algorithm and the results using that final algorithm version. This work produced the following publications [13], [36], [40], [41].

4.3.1. Methodology

Figure 4-14 shows a flowchart representation of the proposed algorithm. The proposed methodology uses only voltage time series measurements from AMI meters, i.e. a single stream of data. That data is transformed in two ways. First, the voltages are converted into a per-unit representation using the ideal mean for the time series. Second, the time series are converted into a "voltage difference" representation by taking the difference of adjacent measurements of the time series. This results in a time series where the values represent the change in voltage at each time step. The remainder of this section is structured as follows. First, there are brief descriptions of the spectral clustering algorithm and clustering ensemble, which have a significant role in the phase identification method. Second, the phase identification algorithm itself is described in detail.

4.3.1.1. Spectral Clustering Process

Clustering, as a group of methodologies, is considered an unsupervised machine learning technique. This means that labelled data points are not required to use the method. K-means and hierarchical clustering are two other examples of unsupervised machine learning methodologies. [42] provides an in-depth description of the spectral clustering methodology. The implementation used in this research is the Python Scikit Learn implementation [43].

In general, spectral clustering calculates (or accepts as input) a pairwise affinity matrix between samples, computes the eigenvectors, and then clusters the data into a user-defined number of clusters using the eigenvectors. A Laplacian matrix is computed from the affinity matrix; this matrix will be approximately block-diagonal and from there the eigenvectors are computed. A subset of the eigenvectors is then used for the clustering step. In the Scikit Learn implementation there are two choices for the clustering step, either "k-means" or "discretize". Spectral clustering provides a non-linear dimensionality reduction of the input data, which differentiates it from other conventional clustering algorithms.

The proposed phase identification algorithm uses slightly different versions of spectral clustering in two locations. As shown in Figure 4-14, spectral clustering is used in Step 2. In that instance of the spectral clustering algorithm, the voltage time series data is used as input and the spectral clustering algorithm uses a radial basis function kernel to calculate the pairwise affinity matrix. The k-means spectral clustering option is used for the clustering step. More details on k-means can be found in [44]. Please see Section 4.3.2 for a discussion of the number of clusters.

In Figure 4-14 Step 4, spectral clustering is used again. The requirements for the clustering in this step are different from the requirements in Step 2, thus a different parameterization of spectral clustering is used. The primary differences are that there is a precomputed affinity matrix available and a different number of clusters is used. The co-association matrix generated by previous steps in the phase identification algorithm is used as input as a "precomputed affinity matrix" and the spectral clustering algorithm proceeds directly to calculating the Laplacian matrix and the eigenvectors. The "discretize" option is used for the clustering step. This option produced better results during testing than the k-means version of spectral clustering. K-means is known to be sensitive to initialization and we believe this to be the key factor in the "discretize" option achieving better performance in this final clustering step. This method turns the clustering into an optimal discretization problem; more details can be found in [45].

Spectral clustering is applied in this paper using a cluster ensemble technique. A cluster ensemble is the aggregation of a number of distinct clustering instances for use in a final "consensus clustering" algorithm. In particular, this work uses a co-association matrix method for combining the results of individual clustering instances. For an overview of cluster ensemble techniques please see [46]. The work in [47] demonstrates a cluster ensemble methodology using spectral clustering in the image segmentation domain. That work leverages similar principles to the work demonstrated here; however, the algorithm and implementation differ significantly. The cluster ensemble technique corresponds to Steps 3 and 4 in the following section.

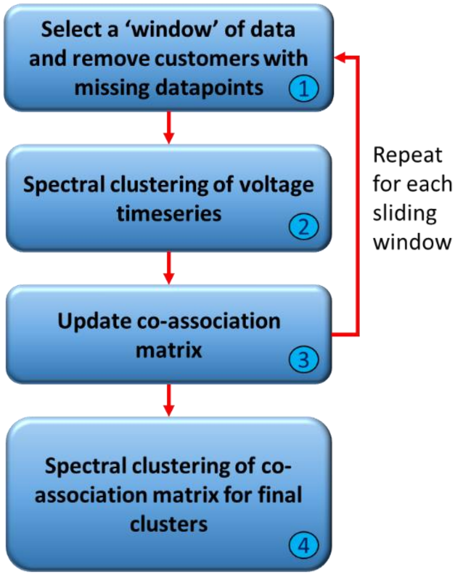

Select a 'window' of data and remove customers with missing datapoints
1
Spectral clustering of voltage timeseries
2
Update co-association matrix
3
Spectral clustering of co-association matrix for final clusters
4
Repeat for each sliding window

Figure 4-14. Spectral clustering phase identification methodology flowchart

4.3.1.2. Phase Identification Algorithm

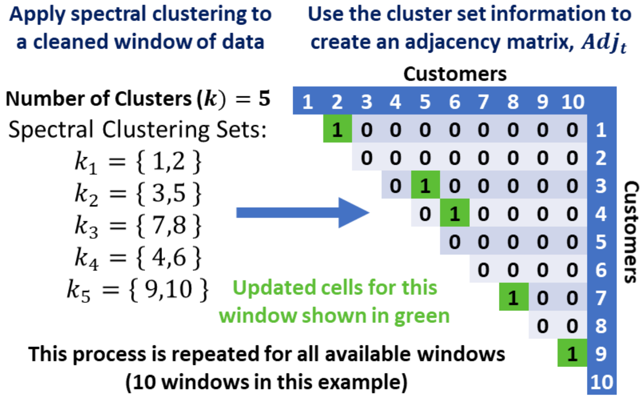
This section describes the phase identification algorithm in detail, and the numbered sections follow the steps shown in the Figure 4-14 flowchart.

First, the available historical data (voltage timeseries for each customer) are divided into 4-day periods or "windows" and all customers with missing data during that period are removed from that window. This process accomplishes several goals. It allows the use of all available historical data, only requiring a subset of the data for computation at one time. The window approach gives a method for dealing with missing data within the dataset. The ensemble nature of the window approach leverages the power of ensemble methodologies and helps account for variations in the dataset due to seasonal variability, customer load variability, or other factors. This step was based on research in [13], [37]. In each of steps 2 and 3 of the algorithm, only a single window is considered. See Section 4.3.2 for a more detailed discussion of the window size parameter.

Second, the remaining customers within that window, the ones with complete data during that window, are clustered using the spectral clustering algorithm. Six, twelve, fifteen, and thirty clusters were chosen for use during this step. This step returns a set of cluster labels from each clustering instance. The results from all four clustering instances are used for Step 3. See Section 4.3.2 for a discussion of the number of clusters used in this step. Foundational research for this step can be found in [13], [48]. However, both of those works use only a single value for the number of clusters.

Third, the cluster label information from Step 2 is then used to populate a co-association matrix. A symmetric $n \times n$ matrix A was created and initialized to zeros as a pre-processing step, where n is the total number of customers and $A_{ij} = A_{ji}$ is the weight between customers i and j. This weight represents the affinity between those customers; customers with larger affinities will have been clustered together in Step 2 more often than customers with low affinities. The matrix is updated with the spectral clustering results from each window. Each cluster produced by the spectral clustering algorithm is represented as an adjacency matrix, which is used to increment the corresponding field in the co-association matrix. This step is shown in Figure. 4-15 and the update step is shown in Figure. 4-15. For example, if customers 1 and 2 were clustered together, then the field in the matrix, row 1 and column 2 (as well as row 2 column 1) would be incremented to a larger weight. In practice, each update is done in two cells of the co-association matrix due to the symmetry of the matrix. Thus, in each window, customers which are clustered together are updated to have stronger weights in the co-association matrix. At this stage the algorithm repeats steps 1-3 for the subsequent window of data, as shown in the Figure 4-14, and the windows on the left side of Figure. 4-15. After all data has been processed in this way, meaning all available windows have been used, the result is a co-association matrix that contains the pairwise clustering information from all the instances of spectral clustering. An alternative way of viewing this matrix is as a histogram, which counts the number of instances where a customer i was clustered with a customer j. This step represents a novel contribution of this algorithm.

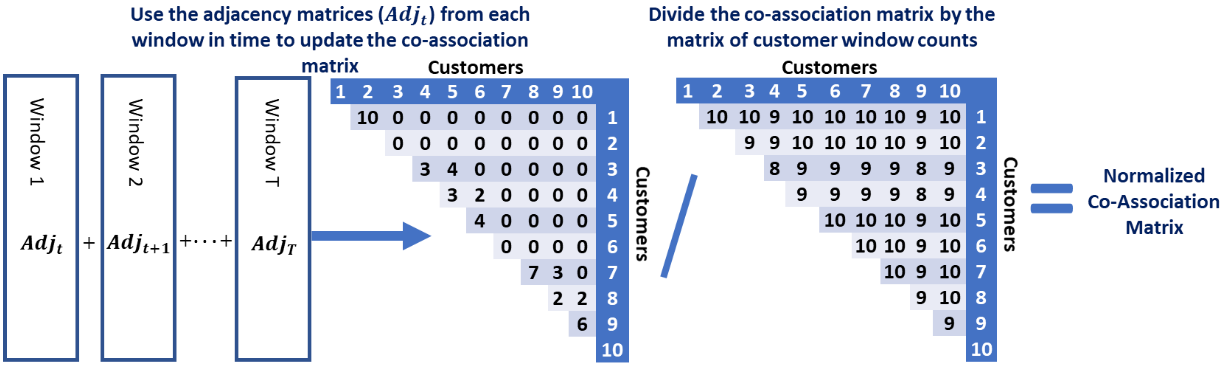
Finally, once all available data have been used, the co-association matrix is normalized to account for the influence of missing data on the matrix. This is necessary because each cell in the co-association matrix may have received a different number of increments due to customers being removed in windows because of missing data. A count is maintained through all available windows, counting the number of windows where each pair of customers was present in the window, meaning those customers could have been clustered together. This count is accumulated in a symmetric matrix of the same size as the co-association matrix. Each cell in the co-association matrix is divided by the corresponding cell in the count matrix to form a normalized version of the co-association matrix; this is shown on the right side of Figure 4-16. The final step uses the normalized co-association matrix as input to another spectral clustering algorithm for partitioning of the customers into final clusters representing the phases; this is shown in Figure 4-17. After the final clusters have been obtained, there are two choices in terms of finishing the phase identification task. If utility labels exist, and the majority are deemed accurate, the final clusters could be labelled using a majority vote of the utility labels contained within those clusters. However, this does require most of the labels to be accurate. Another possibility is to end the method at the determination of those clusters and leave the final determination of phase labels for the utility. This could either be done by comparing to known substation phases or via manual verification. The actual final number of clusters used in this step may depend on the characteristics of the feeder in question. Please see Section 4.3.2 for a more detailed discussion of the number of final clusters. Step 4 also represents a novel contribution of this algorithm. The use of the cluster ensemble with the co-association matrix is one of the key differentiators between the proposed method and the work in [13]. This significantly increases the robustness of the algorithm and removes the reliance on existing utility phase labels.



Apply spectral clustering to a cleaned window of data
Use the cluster set information to create an adjacency matrix, Adj_t
Customers
Number of Clusters (k) = 5
Spectral Clustering Sets:
$k_1 = \{1,2\}$
$k_2 = \{3,5\}$
$k_3 = \{7,8\}$
$k_4 = \{4,6\}$
$k_5 = \{9,10\}$
Updated cells for this window shown in green
This process is repeated for all available windows (10 windows in this example)
Customers

	1	2	3	4	5	6	7	8	9	10
1		1	0	0	0	0	0	0	0	0
2			0	0	0	0	0	0	0	0
3				0	1	0	0	0	0	0
4					0	1	0	0	0	0
5						0	0	0	0	0
6							0	0	0	0
7								1	0	0
8									0	0
9										1
10										

Figure. 4-15. Create adjacency matrix based on the results of a window clustering using 5 clusters

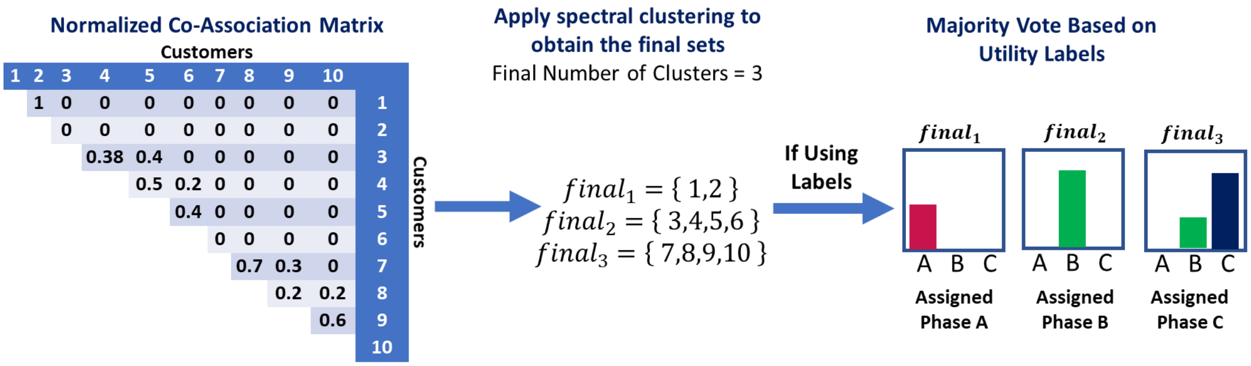


Use the adjacency matrices (Adj_t) from each window in time to update the co-association matrix
Divide the co-association matrix by the matrix of customer window counts
Window 1
Window 2
Window T
Adj_t
+
Adj_{t+1}
+···+
Adj_T
Customers
Customers
Customers
Customers
=
Normalized Co-Association Matrix

	1	2	3	4	5	6	7	8	9	10
1		10	0	0	0	0	0	0	0	0
2			0	0	0	0	0	0	0	0
3				3	4	0	0	0	0	0
4					3	2	0	0	0	0
5						4	0	0	0	0
6							0	0	0	0
7								7	3	0
8									2	2
9										6
10										

	1	2	3	4	5	6	7	8	9	10
1		10	10	9	10	10	10	10	9	10
2			9	9	10	10	10	10	9	10
3				8	9	9	9	9	8	9
4					9	9	9	9	8	9
5						10	10	10	9	10
6							10	10	9	10
7								10	9	10
8									9	10
9										9
10										

Figure 4-16. Updating the co-association matrix with the results of each individual window clustering and then normalizing the co-association matrix by dividing by the matrix tracking the customer presence in each window


Normalized Co-Association Matrix
Customers
Customers
Apply spectral clustering to obtain the final sets
Final Number of Clusters = 3
$final_1 = \{ 1,2 \}$
$final_2 = \{ 3,4,5,6 \}$
$final_3 = \{ 7,8,9,10 \}$
If Using Labels
Majority Vote Based on Utility Labels
$final_1$
$final_2$
$final_3$
A B C
A B C
A B C
Assigned Phase A
Assigned Phase B
Assigned Phase C

	1	2	3	4	5	6	7	8	9	10
1		1	0	0	0	0	0	0	0	0
2			0	0	0	0	0	0	0	0
3				0.38	0.4	0	0	0	0	0
4					0.5	0.2	0	0	0	0
5						0.4	0	0	0	0
6							0	0	0	0
7								0.7	0.3	0
8									0.2	0.2
9										0.6
10										

Figure 4-17. Final cluster assignment of which customers are on the same phase as each other. This is followed by the optional step of using the utility phase labels for the final phase assignment

4.3.1.3. Previous Version of the Algorithm

This algorithm went through one major developmental revision over the course of the project. The version described above is the final, and most robust, version of the algorithm. The final version was shown to produce more accurate phase identification results, see Figure 4-29 in Section 4.3.4 for an example of this. In the previous iteration, the same window ensemble is used, however instead of building a co-association matrix, each cluster (in each window) provides a "vote" for the predicted phase of the customers in that cluster using the majority vote of the original utility labels for the customers in the cluster. This provides a set of votes for each customer from each window. The final predicted phase is the majority vote from all the predicted phases from all windows. This version of the algorithm resulted in a conference publication [13]. This version is also the version used by CYME in their comparison work in Section 12.2.1.

4.3.2. Parameter Tuning Discussion

In the course of conducting the parameter tuning research for the spectral clustering phase identification algorithm, a general methodology for parameter tuning was developed. This methodology was demonstrated on the spectral clustering algorithm but could be applied in other situations where clustered algorithms have been used. The algorithm is based on an analysis of the silhouette score metric.

4.3.2.1. Silhouette Score Methodology

The silhouette score is used for the interpretation and validation of clusters of data. The index provides a measure of how well objects lie within their cluster. The silhouette coefficient is calculated using Equation (4-1),

$$\boldsymbol{s(i)} = \frac{\boldsymbol{b(i)-a(i)}}{\mathbf{max}\ (\boldsymbol{a(i)},\boldsymbol{b(i)})} \qquad (4\text{-}1)$$

Where *b* is the single linkage distance and *a* is the mean intra-cluster distance. The silhouette calculation produces a coefficient between -1 and +1, with values closer to one indicating that a sample *I* is well clustered, and negative values indicating that a sample *i* could reasonably be placed in a different cluster [49]

The silhouette score is a common metric used in silhouette analysis. It is calculated by taking the average of all silhouette coefficients. The silhouette score acts to quantify the overall behavior of the clustering. Thus, the silhouette score has higher values for clusterings that contain dense, widely separated clusters.

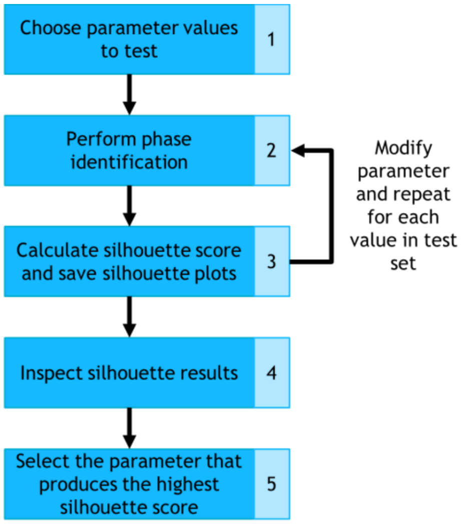

Choose parameter values to test
1
Perform phase identification
2
Calculate silhouette score and save silhouette plots
3
Inspect silhouette results
4
Select the parameter that produces the highest silhouette score
5
Modify parameter and repeat for each value in test set

Figure 4-18. Parameter selection methodology flowchart

The silhouette coefficient plot provides a visual tool for analyzing clustering and is an important part of silhouette analysis. In the silhouette coefficient plot, the silhouette coefficients are plotted by cluster. The x-axis of the cluster in the silhouette plot refers to the magnitude of its silhouette coefficient. Clusters with greater widths have samples with larger silhouette coefficients. The y-axis of a cluster in the silhouette plot indicates the number of samples it contains.

The silhouette score is one way of summarizing the silhouette coefficient information that characterizes a clustering. The accompanying silhouette coefficient plot provides meaningful information about the quality of each cluster. Step 4 in Figure 4-18 involves inspecting the behavior and shape of the silhouette plots produced in the parameter sweep. Clusters with below-average silhouette scores, wide fluctuations in the size, and a large proportion of negative silhouette coefficients usually indicate a poor clustering.

4.3.2.1.1. Selecting Parameters Using the Silhouette Score

The proposed methodology can be used for parameter selection in contexts where a pairwise distance matrix is calculated, and some type of clustering can be defined. This method is demonstrated here to set the number of clusters in the spectral clustering phase identification algorithm. This method is first demonstrated using five utility datasets.

In Steps 2-3 in Figure 4-18 phase identification is performed, and the silhouette score and silhouette plots are generated for each iteration of the different algorithm parameters in phase identification. As a final step, the best phase identification algorithm parameter is selected using the silhouette score. Further visual analysis can also be conducted using the silhouette coefficient plots. Wide fluctuation in cluster widths, and significant proportions of negative values, and other abnormal behavior can be easily seen by visual inspection.

4.3.2.1.2. Choosing the Window Size

To accommodate missing data, the spectral clustering ensemble approach performs phase identification using a sliding window of data for each customer. Customers with missing data in each window are discarded. Selecting the window size is also an important parameter in optimizing the performance of the phase identification algorithm. Selecting the window size involves a trade-off between an increased silhouette score/better clustering and the number of customers to be discarded due to missing data. Silhouette score analysis also provides insight into setting this parameter.

4.3.2.2. Parameter Tuning Results

The parameter-tuning methodology was applied to two datasets from the Utility #1 described in Section 3.2.1, three datasets from Utility #2 described in Section 3.2.2 in addition to the synthetic dataset described in Section 3.1.

Table 4-4 contains the silhouette scores calculated for all datasets during the parameter sweep. In each case the silhouette score peaks and then begins decreasing, allowing for a straightforward interpretation of the best parameter.

Only the downstream customers were used in the Utility #1, Feeder #1. Selecting four clusters was shown to be the best choice based on the proposed silhouette score parameter tuning method. Four clusters had the highest overall silhouette score, and this choice can be confirmed by visual inspection of the silhouette coefficient plots.

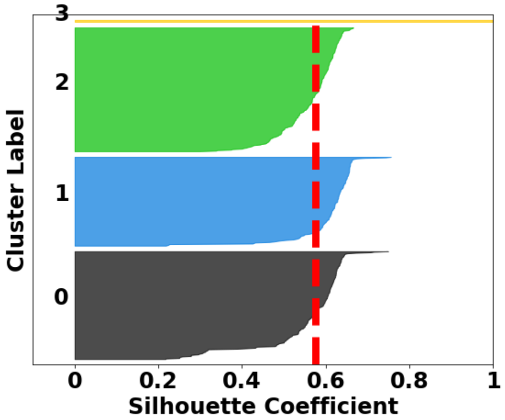


Figure 4-19. Plot of silhouette coefficients using 4 clusters in for Feeder #1 on Utility 1. The red line indicates the overall silhouette score.

The proposed method, when applied to the Feeder #1 from Utility #2, showed the optimal number of clusters to be 12. The “goodness” of the clustering can be visually confirmed by the silhouette coefficient plot shown in Figure 4-20. This conclusion makes sense when considering the three voltage regulators on this feeder.

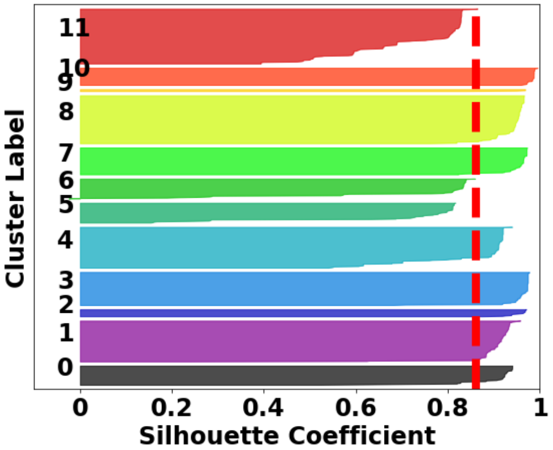


Figure 4-20. Plot of silhouette coefficients for the chosen number of clusters in Utility 2, Feeder #1. The red line indicates the overall silhouette score

Table 4-4. Silhouette scores from cluster sweep of each dataset

	Utility dataset					
Number of clusters	Utility 1 Feeder #1	Utility 1 Feeder #2	Utility 2 Feeder #3	Utility 2 Feeder #1	Utility 2 Feeder #4	Synth
3	0.376	0.33	0.687	0.354	0.354	0.560
4	**0.576**	0.402	0.688	0.443	0.444	0.630
5	0.497	0.498	0.699	0.515	0.513	0.668
6	0.390	0.562	0.706	0.596	0.597	0.692
7	0.387	0.620	0.708	0.665	0.674	0.**704**
8	0.208	0.616	**0.721**	0.727	0.735	0.579
9	-	0.657	0.712	0.793	0.784	-
10	-	0.684	0.714	0.833	0.772	-
11	-	0.692	-	0.853	0.776	-
12	-	**0.700**	-	**0.860**	0.776	-
13	-	0.663	-	0.823	**0.790**	-
14	-	0.627	-	0.829	0.772	-
15	-	-	-	0.817	0.652	-

In addition, the methodology used to select the window size parameter to be used in the phase identification algorithm. Selecting the parameter involves a trade-off between the number of customers lost due to missing data and a higher silhouette score. Figure 4-21 shows the silhouette score (solid lines) and number of customers not predicted due to missing data (dashed lines) as the window size changes for each utility dataset. We hypothesize that the two different curve shapes are related to the characteristics of the missing data in each dataset.

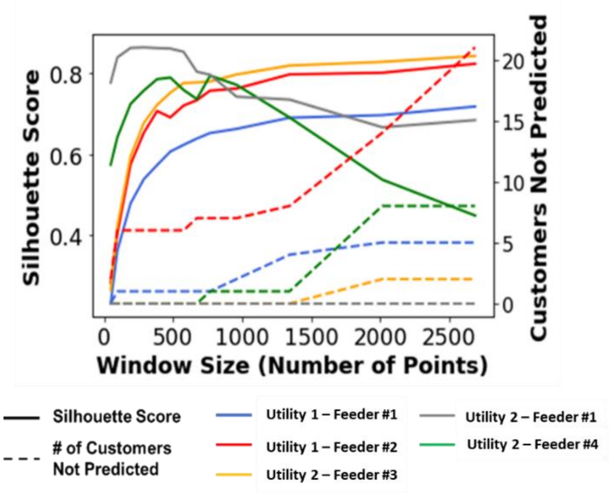


Figure 4-21. Silhouette score and the number of customers not predicted vs. the window size for the utility datasets

A parameter sweep of the window size was performed on the synthetic data with two different missing data injections, and varying percentages of missing data injected. In the first type of data injection, single measurement points were removed uniformly at random up to the percentage of data missing. In the second type of data injection, a missing data interval from 1 missing point to 48 contiguous missing data points was chosen uniformly at random and then injected into a dataset location chosen uniformly at random until each customer had the specified percentage of missing data. These two types of missing data injection simulate different distributions of missing data. For example, single missing values might be indicative of missing data due to communication failure, whereas longer missing periods might be indicative of short outages.

Figure 4-22 shows a plot comparing the silhouette score and the number of customers not predicted as the window size changes. The synthetic results mirror the results obtained using the utility data.

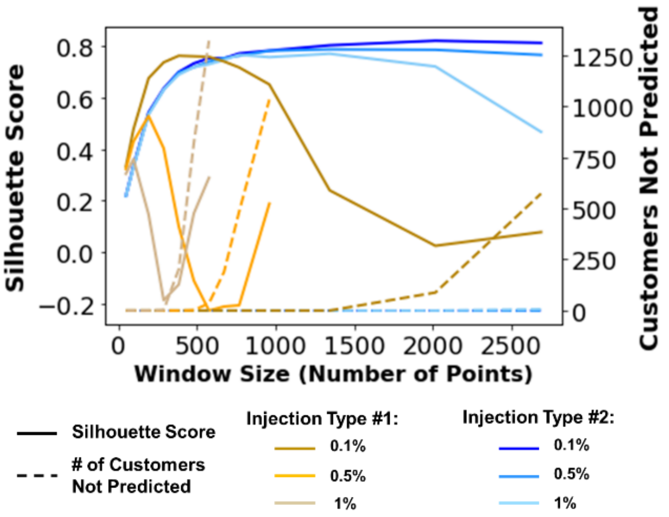


Figure 4-22. Silhouette score and the number of customers not predicted vs. the window size for the synthetic dataset.

4.3.2.3. Parameter Tuning Summary

This section presents a methodology for selecting parameters for distribution system model calibration algorithms utilizing distance matrices and demonstrated using a spectral cluster ensemble phase identification algorithm. The proposed methodology leverages the silhouette metrics to select parameters to optimize the clustering algorithm. This method was used to select the number of clusters, and window size in the spectral clustering phase identification algorithm. This work resulted in the following publication [40].

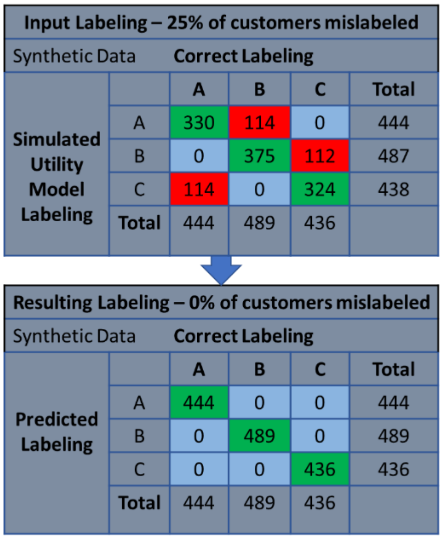
4.3.3. Ensemble Spectral Clustering Synthetic Data Results

This section contains the results for experiments conducted on a synthetic dataset #1, discussed in Section 3.1.1 where the ground truth labels are known. This allows for rigorous testing and manipulation of the dataset to create a more realistic test environment. The synthetic dataset allowed simulation of baseline results with ground truth phase labels for all customers and voltage timeseries free of measurement noise or other data issues. Realistic data issues were later added to the dataset for testing purposes.

Two types of data manipulations were used to further test the algorithm and simulate more realistic data conditions. The first type of data manipulation concerns AMI data collection methods such as the measurement interval, and meter resolution. Other data manipulations were related to possible data issues such as meter bias, measurement noise, and missing data. A combination of these factors was combined to create a test case by injecting uniformly distributed measurement noise up to 0.02% of the mean, injecting uniformly distributed meter bias of up to 0.02% of the mean, and by removing 0.2% of the measurements to simulate missing data. The AMI data collection methods used were 15-min average measurement interval, 0.1V resolution measurements, 6 months of available data, and AMI meter penetration of 100%. Furthermore, a set of customers was chosen to intentionally force their phase labels to be incorrect to simulate model errors. Various levels of mislabelling were chosen to test the effects of the quantity of mislabelled customers on the algorithm performance.

Figure 4-23 shows confusion matrices corresponding to testing the algorithm with 25% of the phases mislabeled. The top half of the figure shows the distribution of phases that were intentionally mislabeled for this simulation, and the bottom half of the figure shows that the algorithm was able to correctly identify all customer phases with 100% accuracy.

Table 4-5 shows the results of testing to determine the robustness to measurement noise of the proposed algorithm. Measurement noise is simulated as uniformly distributed noise up to a maximum value (positive and negative) of the percentage shown in the left column. For reference, American National Standards Institute (ANSI) standards define accuracy classes of 0.1, 0.2, and 0.5, [50]. The ANSI values reference error in real power measurements, however the voltage noise injected here does directly contribute to error in the real power measurements. These results use the dataset characteristics described in 4.1, testing varying levels of measurement noise. The values shown are the averaged results of five independent simulations. We can see that the algorithm is relatively robust to measurement noise in the dataset. There are no errors in phase predictions until 0.65% maximum noise and, at that level of noise, there are not consistent errors. That is driven by the randomness associated with the location and quantity of the noise within each simulation.



Input Labeling – 25% of customers mislabeled

Synthetic Data		Correct Labeling			
		A	B	C	Total
Simulated Utility Model Labeling	A	330	114	0	444
	B	0	375	112	487
	C	114	0	324	438
	Total	444	489	436	

Resulting Labeling – 0% of customers mislabeled

Synthetic Data		Correct Labeling			
		A	B	C	Total
Predicted Labeling	A	444	0	0	444
	B	0	489	0	489
	C	0	0	436	436
	Total	444	489	436	

Figure 4-23. 25% of customers mislabeled confusion matrices. Red represents mislabeled customers and green represents correctly labelled customers

Table 4-5. Averaged results of five simulations with increasing levels of noise added to the data

Max Noise Percentage	Number of Incorrectly predicted customers	Percent Accuracy
Original	0	100%
0.05%	0	100%
0.15%	0	100%
0.25%	0	100%
0.35%	0	100%
0.45%	0	100%
0.55%	0	100%
0.65%	0.2	99.99%
0.75%	0.4	99.97%
0.85%	1.0	99.92%
0.95%	3.6	99.69%
1.0%	7.6	99.46%

The proposed algorithm is compared to a phase identification algorithm with a similar approach developed in the beginning of this project. For algorithm details see [13].

The same experiment shown in Table 4-5 was conducted for the comparison algorithm in [29]. Those results suggest that the measurement noise should be <0.25% (approximately corresponding to the 0.2-meter class). The results shown in Figure 4-24 demonstrate considerable improvement, up to ~0.65% maximum noise before errors begin to occur.

Figure 4-24 and Figure 4-25 show the results of the testing on the synthetic dataset for both the algorithm in [13], listed as "Comparison Method" as well as the proposed algorithm. In both figures, the results shown are from a Monte Carlo simulation consisting of 500 individual simulations, where the only difference between each simulation is which customers were chosen to be mislabelled. The minimum, maximum, and mean percentages from the 500 simulations are shown for the comparison algorithm. For the proposed algorithm, the results for all three metrics (min, max, mean) are identical so only one line is plotted. The x-axis of Figure 4-24 shows the percentage of customers that have been injected with phase label errors for the purposes of testing, and the y-axis shows the percentage of those mislabelled customers that have been corrected. For example, for the minimum accuracy of the comparison method at 35% of customers mislabelled the method was able to correctly identify ~90% of customer phases; 35% of 1,369 customers is 479 customers. Therefore, 48 of those 479 customers remained incorrect after the algorithm completed. In contrast, the proposed algorithm retains 100% accuracy for all 500 simulations. Note the unevenness of the line corresponding to the minimum percentage of customers corrected (solid blue line) in Figure 4-24. This demonstrates the sensitivity of the comparison method to the configuration of customers who are mislabelled within the model. The proposed algorithm removes that sensitivity. Figure 4-25 shows the overall accuracy for all customers for the same Monte Carlo simulation.

Using the proposed algorithm, the dependence on the utility phase labelling has been removed and this algorithm achieves excellent performance under the test conditions. At some point, with enough phase label errors in the utility model, the ability to assign predicted phases in the last step of the model will degrade. However, the clusters themselves will still represent correct phase clustering. It will simply be a matter of correctly assigning those clusters to the appropriate phase, and that could be done as post-processing by the utility.

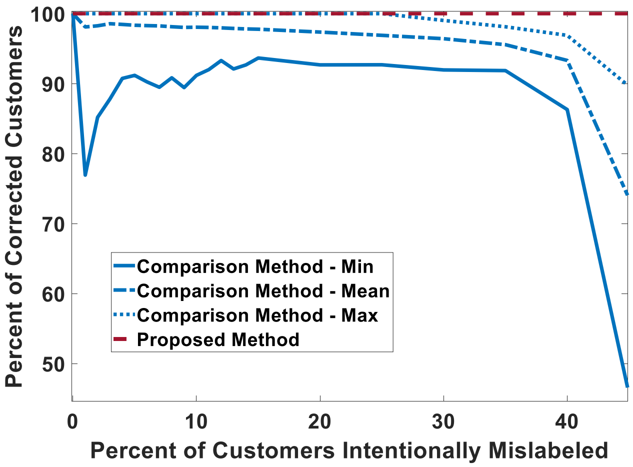


Figure 4-24. Percent of mislabeled customers corrected

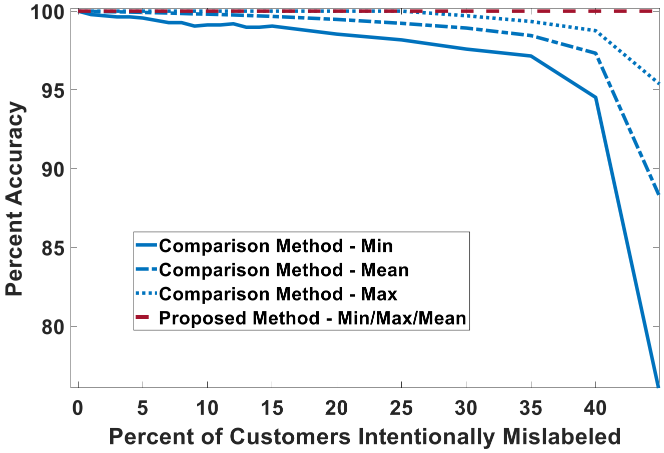


Figure 4-25. Overall percent accuracy proposed method versus comparison method

4.3.4. Ensemble Spectral Clustering Utility Data Results

These results use Feeder #3 from Utility #2 described in Section 3.2.2. Previous work suggests that the quantity of incorrect phase labels for this feeder is ~9%. Therefore, the utility phase labels are used to make the final phase assignment. The following results reflect using that approach.

Validation of the results on the utility dataset is challenging as there are not ground truth labels for this feeder. [13] presents a two-stage validation methodology that combines validation using topology information and validation using publicly available Google Earth and Google Street View imagery. There are extensive details and images in [13] documenting the validation for that method. Although only a subset of customer phase labels can be verified due to issues such as availability of imagery, tree cover, underground cabling, etc., there does exist a subset of customers for which the Google Earth imagery is reasonably conclusive for phase connectivity. Three novel examples using Google Earth satellite views and Street View are shown below as instances of the proposed algorithm making correct predictions.

The proposed algorithm predicted that 143 customers were incorrectly labelled in the utility model of this feeder, out of a total of 1,096 customers. This represents ~13% of the customers. Those figures do include 41 customers who were labelled as being present on the feeder, but their phase was labelled as "unknown." Excluding those customers gives a prediction that 112 customers were incorrectly labelled out of 1,055. This represents ~10.6% of the customers. The inclusion or exclusion of those customers changes the results slightly for the other customers on the feeder. Following are two examples where the results of the proposed algorithm can be validated using Google Earth imagery.

Looking at Figure 4-26, the left-hand image shows a transformer that was labelled in the utility model as Phase A but was predicted by the algorithm to be on Phase C. We can see the connection from the transformer connecting to the left-hand medium voltage wire at the top of the image. Continuing up the street, the next image shows the next transformer labelled as Phase C; this transformer is both labelled in the utility model and predicted by the algorithm as Phase C. However, this transformer is connected to the same left-hand medium voltage wire. Although not shown here for space reasons, continuing south on the street from the transformer in question, the next transformer, which serves 5 customers is labelled (and predicted) as Phase C. This transformer is also connected to the same medium voltage line. This strongly indicates that this transformer represents an error in the original utility model and a corrected phase label predicted by the proposed algorithm.

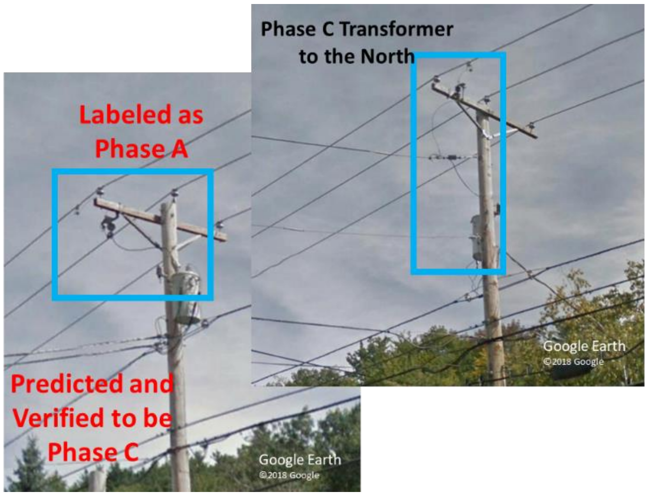


Figure 4-26. Example of an incorrect phase label that was corrected on Utility 2, Feeder #3

Figure 4-27 and Figure 4-28 show another example of the proposed algorithm correcting errors within the original model. Figure 4-27 shows the original labelling. All the customers shown are labelled to be on Phase A, which is served by the Phase A medium voltage line coming from the right-hand side of the image. All these customers were predicted to be on Phase B by the algorithm. Figure 4-28 shows the version of the labelling predicted by the algorithm. This result can be fully verified within Street View. The Phase A medium voltage line stops at the intersection on the right-hand side and does not continue down the street. In fact, the Phase B line coming from the bottom right of the image turns the corner and serves all six of the customers shown.

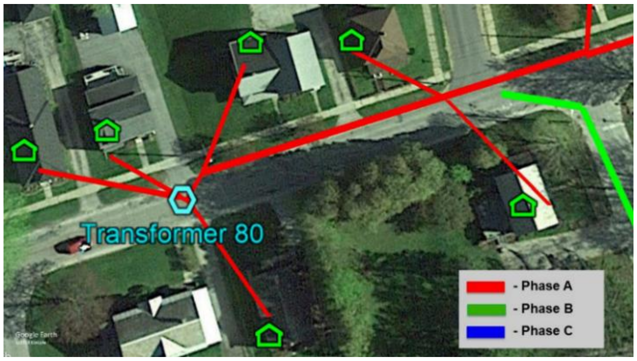


Figure 4-27. Original utility phase labels for a transformer on Utility #2, Feeder #3

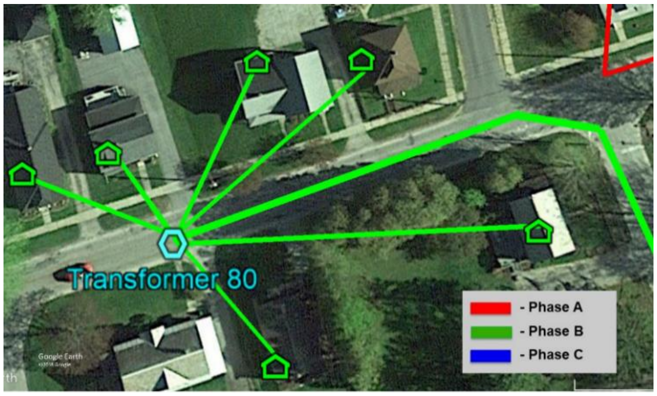


Figure 4-28. Phase labels as predicted by the phase identification algorithm and verified in Google Street View for a transformer on Utility #2, Feeder #3

Figure 4-29 shows an example of a case where the phase predictions from the method presented here show an improvement on the earlier version of the algorithm [13]. The earlier algorithm version is described in Section 4.3.1.3. The Google Earth imagery clearly shows that the final method located and corrected a set of customers that remained as errors in the utility model. This method also maintained the correct label for a customer that was labelled correctly in the model predicted incorrectly in the comparison method. Figure 4-29 shows an image with subfigures in each of the four quadrants. Home icons represent customers, light-blue hexagons represent transformers, red lines show Phase A lines, and blue lines show Phase C lines. The original labelling in the utility model is shown in the upper-left quadrant. In the upper-right image, showing the comparison algorithm predictions, two homes have changed phase. The one marked in yellow changed from C to A as did one marked in purple. The purple home that changed was a correct change. However, the customer marked in yellow was a correct label by the utility that was incorrectly predicted to be on a different phase. The yellow customer is actually on Phase C. The bottom quadrants are identical. In these quadrants, the proposed algorithm predictions and the actual labelling verified in Street View are shown. In addition to correctly labelling the customer marked in yellow, two customers to the right of this customer (orange) are also on Phase C, which is a change from A to C. Those two incorrectly labelled customers in the original model caused the previous version of the algorithm to misclassify the customer marked in yellow. Conversely, the final algorithm was able to determine the correct configuration.

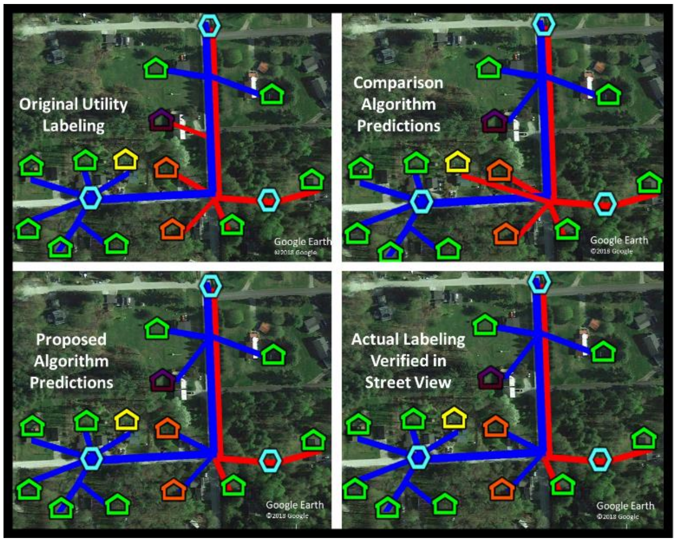

Original Utility Labeling
Comparison Algorithm Predictions
Proposed Algorithm Predictions
Actual Labeling Verified in Street View
Google Earth ©2018 Google
Google Earth ©2018 Google
Google Earth ©2018 Google
Google Earth ©2018 Google

Figure 4-29. The purple house was correctly predicted by both algorithms. The two orange houses were predicted incorrectly by the earlier version of the algorithm but correctly by the final algorithm. The yellow house was correct in the utility model, but it was predicted incorrectly by the earlier version and correctly by the final algorithm version. Customers from Utility #2, Feeder #3

There are a number of other examples on the utility feeder that were verified using the Google Earth imagery method not included in this report for space considerations.

4.3.5. Confidence Score Development for Commercial Implementation

This section details several refinements to the spectral clustering phase identification algorithm that were necessary to implement the algorithm in a commercial/utility setting. That implementation is detailed in Section 12.4.

Two key issues for commercial implementation of the spectral clustering phase identification algorithm are the issues of quantifying confidence in individual predictions and providing guidelines for setting algorithm parameters. Differences in the algorithms are such that different confidence metrics are required for the spectral clustering ensemble versus the confidence metrics used for the sensor-based method.

For the confidence metric, we are using a modified version of the Silhouette Coefficient. The Silhouette Coefficient is a well-established clustering analysis method that we have modified to be 'phase-aware' in this domain. The classic Silhouette Coefficient is defined as follows:

$$s = \frac{(b-a)}{max(a,b)}$$

Where a is the mean distance between a sample and all other points in its same cluster and b is the mean distance between a sample and all other points in the next nearest cluster.

Our modification is to require that the 'next nearest cluster' for the b value be a cluster predicted to be a *different* phase from the current cluster's phase. This effectively makes the Modified Silhouette Coefficient a measure of how likely a particular customer was to be assigned a different phase for its final prediction. The lower the Modified Silhouette Coefficient, the less confident the final phase prediction.

Figure 4-30 shows a histogram of the Modified Silhouette Coefficients for our synthetic dataset. The red arrow indicates a customer that is located on a different feeder that was added to the synthetic dataset for testing. Note that the customer has a much lower Modified Silhouette Coefficient, which demonstrates the metrics efficacy as a confidence score. As a guideline, any customer phase prediction with a Modified Silhouette Coefficient less than 0.2 should be considered low confidence.

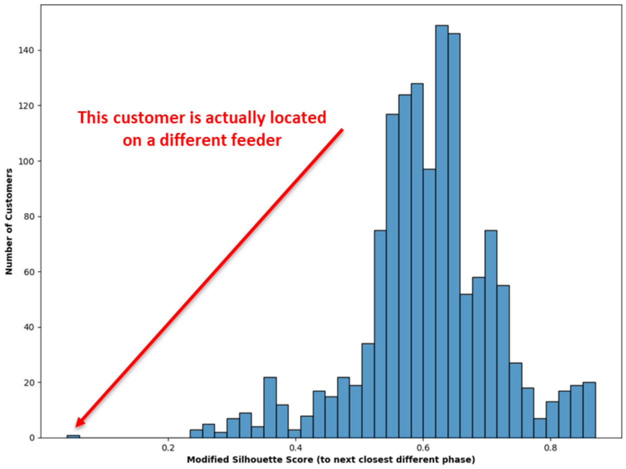


Figure 4-30 - Histogram of Silhouette Coefficients - red arrow marks a customer on a different feeder

Additionally, for commercial implementation, it is important to be able to provide reasonable guidelines for algorithm parameters (such as the window size in this case) and the amount of data required for the algorithm to perform well. We performed a parameter sweep for a large number of possible values from the window size ranging from 48 samples to 500 samples per window. Figure 4-31 shows the histogram for the Modified Silhouette Coefficients for the 48-sample case and the 384-sample case. Note that the Modified Silhouette Coefficients are much lower overall for the 48-sample case. The overall results of the parameter sweep indicate that the window size should be fixed at 384 samples, regardless of the total number of datapoints contained in the dataset. This test also demonstrates the usage of the Modified Silhouette Coefficient metric, not just as a confidence score for individual predictions, but also as an indication of the algorithm's overall effectiveness given a set of parameters or a particular dataset.

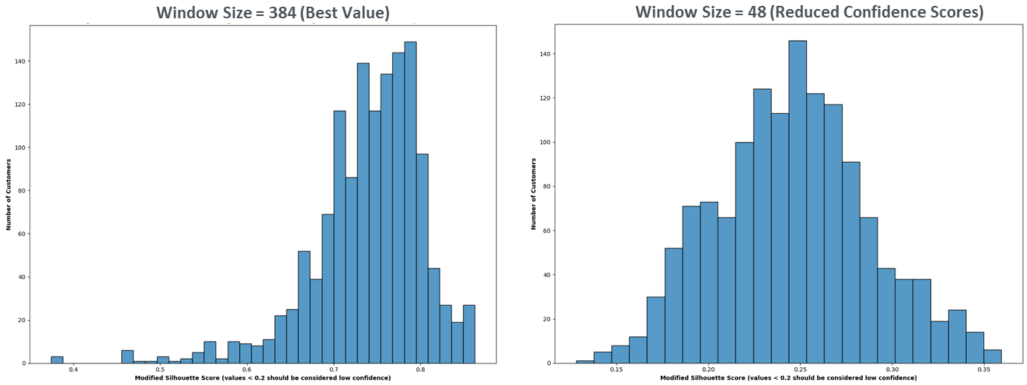


Figure 4-31 - Histogram of Modified Silhouette Coefficients for two window size values, 384 (left) and 48 (right)

The second key question for a commercial implementation of the algorithm is, 'How much data is required for the algorithm to perform the phase identification task?' For this question, we leveraged the Adjusted Rand Index metric. The Adjusted Rand Index quantifies the similarity between two instances of cluster assignment by comparing the number of pairs of customers clustered together in both clustering instances. This allows us to measure the consistency between two distinct runs of the phase identification algorithm. We would expect identical (or nearly so) results from two runs of phase identification on the same data under the same parameters. However, we know that there are minimum data requirements for the algorithm to function correctly and consistently. Using a dataset with too few datapoints would result in the phase identification algorithm potentially producing different results with each run due to the co-association matrix not being adequately populated. The Adjusted Rand Index allows us to quantify these effects. A sweep was conducted changing the number of datapoints available in the dataset in the range 1920 points (20 days with 15-min interval data) to 35040 points (1 year with 15-min interval data) using a 20 run Monte Carlo simulation for each value for the number of datapoints. An average Adjusted Rand Index was calculated over the Monte Carlo to quantify the effects on the phase identification results. Figure 4-32 shows the results of this test. On the blue y-axis is plotted the average Adjusted Rand Index, and on the red y-axis is plotted the average number of incorrectly identified customers. If the number of available datapoints is too small the algorithm performance begins to degrade. The navy dashed line at 11520 shows the recommended minimum number of datapoints for the algorithm. Note that the algorithm may continue to produce acceptable results for datasets with fewer datapoints, but we do expect some level of degraded performance.

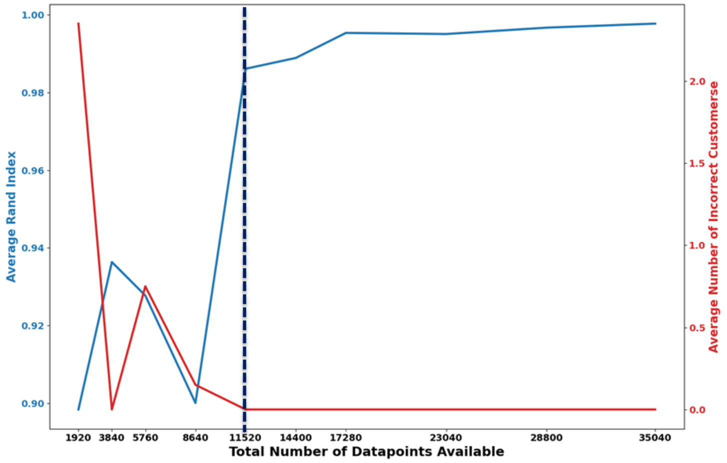


Figure 4-32 - Sweep of the number of available data points plotted with the average Adjusted Rand Index and the average number of incorrect customers

4.4. State Estimation-Based Distribution Transformer Phase Identification

Accurate monitoring of transmission systems has been achieved through power systems state estimation [51]. However, distribution system state estimation (DSSE), however, is not a mature technology, still facing many technical challenges, such as lack of visibility due to low level of sensing, insufficient communications infrastructure to collect data from smart meters, high system complexity, uncertainty in parameters and system topology, and unknown load behavior exacerbated by DERs [52]. With the increasing level of telemetered data from AMI in association with data streams from supervisory control and data acquisition (SCADA) systems, several power utilities might soon have enough metered data to monitor their systems with DSSE. In this scenario, however, uncertainty in parameters and topology can still harm the precision of state estimates in spite of measurement redundancy and system observability [53].

We have investigated the robustness of static DSSE to errors in topology. Namely, we have investigated how well the DSSE performs under errors in phase identification of single-phase distribution transformers. The results of this research were originally published in [54]. Phase identification of single-phase transformers is a challenging task and errors in phase labels of such transformers are common in practice [36]. Topology errors can lead to significant discrepancies in the estimates of power flows, which in its turn can lead to significant errors in state estimates. Hosting capacity analyses leverage those models and they are important for utilities to determine the amount and location of solar PV a given distribution feeder can support before updates are necessary. These methods usually rely on power flow-based (PF) steady-state or quasi-steady state simulations, which are very sensitive to errors in parameters.

4.4.1. Distribution System State Estimation

The foundation of the DSSE problem is described by the measurement Equation (4-2), which relate the system states with measurements and errors in measurements.

$$\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{e} \qquad (4–2)$$

where $\mathbf{z} \in \mathbb{R}^m$ is the vector of measurements, $\mathbf{x} \in \mathbb{R}^N$ is the vector of state variables, $\mathbf{h}: \mathbb{R}^N \to \mathbb{R}^m (m > N)$ is the measurement function that maps the states to the measurements, and $\mathbf{e} \in \mathbb{R}^m$ is vector of independent, zero-mean Gaussian measurement errors with σ_i as the standard deviation of the *i*-th measurement.

In our formulation, the vector of states contains real and imaginary parts of the nodal voltage phasors plus capacitor bank statuses (on or off) and voltage regulator tap positions. The measurements are composed of voltage and real and reactive power injection measurements from smart meters connected in low-voltage secondary distribution, SCADA measurements of substation voltage and power injections into the distribution grid, capacitor bank statuses, and position of load tap changers (LTCs) in voltage regulators. It is assumed that all are sampled every 15 minutes.

A weighted least-squares static state estimator (WLS-SE)[6] is used to solve this problem [51]. The resulting nonlinear program has the goal of finding the state estimate, $\hat{\mathbf{x}}$, that minimizes the weighted sum of the squares of the residuals, defined as the difference between the measurements and the measurement function, weighted by the standard deviation, as shown in Equation (4-3).

$$\min_{\hat{\mathbf{x}}} J(\hat{\mathbf{x}}) = \sum_{i=1}^{m} \left(\frac{z_i - h_i(\hat{\mathbf{x}})}{\sigma_i}\right)^2 \qquad (4–3)$$

where $J(\hat{\mathbf{x}})$ is the objective function of the WLS-SE. If the assumption of additive Gaussian noise and perfectly known parameters and topology holds, $J(\hat{\mathbf{x}})$ should follow a χ^2 (chi-squared) distribution with $m - N$ degrees of freedom. This problem can be solved iteratively using the Gauss-Newton algorithm by assuming the statuses of capacitor banks and tap positions are

[6] Due to the large time steps between measurement scans and unknown state dynamics of the distribution system, a dynamic state estimator was not used.

continuous variables. After the estimator converges these variables are discretized again to their closest discrete value.

4.4.2. Detecting Phase Label Errors

To detect errors in phase labels we have proposed a branch χ^2 test. This is a modified version of the goodness of fit χ^2 (or $J(\hat{\mathbf{x}})$) widely used in power system state estimation for bad data detection of measurements [51]. Traditionally, this goodness of fit test is applied to all measurements or pseudomeasurements considered in the state estimation process, given an expected false-alarm rate defined by a level of significance α. If the $J(\hat{\mathbf{x}})$ score is under a threshold, the state estimate is accepted as a correct result. Otherwise, the alternative hypothesis is accepted, and it is understood that the model poorly fits the data (measurements), therefore an error must exist in measurements or model (e.g., topology, parameters).

In this modified version of the χ^2 test, we apply Equation (4-4) to all measurements and pseudomeasurements (or virtual measurements) downstream of single-phase service transformers. These pseudomeasurements are zero-injection real and reactive power injections on buses where no loads nor generators exist used to provide additional topological information to the DSSE and improve observability and redundancy of measurements.

$$J_b(\hat{\mathbf{x}}) = \sum_{i\in\Omega_b}\left(\frac{z_i-h_i(\hat{\mathbf{x}})}{\sigma_i}\right)^2, \forall b \in \Omega_T \qquad (4–4)$$

where Ω_T is the set of all single-phase transformers and Ω_b is the set of all measurements and pseudomeasurements downstream of transformer b. Because under normal circumstances $J_b(\hat{\mathbf{x}})$ should follow a χ^2 distribution with ν degrees of freedom, the hypothesis test is designed with a null hypothesis H_0 of $J_b(\hat{\mathbf{x}})$ following a χ^2_ν distribution and the alternative hypothesis H_1 stating otherwise, as shown in Equation (4-5).

$$\begin{aligned} H_0 :\ & J_b(\hat{\mathbf{x}}) \le \chi^2_{\nu,1-\alpha} \quad (J_b(\hat{\mathbf{x}}) \sim \chi^2_\nu) \\ H_1 :\ & J_b(\hat{\mathbf{x}}) > \chi^2_{\nu,1-\alpha}\ \neg(J_b(\hat{\mathbf{x}}) \sim \chi^2_\nu) \end{aligned} \qquad (4–5)$$

Branches whose $J_b(\hat{\mathbf{x}})$ score fails a χ^2 test, i.e., H_1 is accepted, are flagged as incorrectly labeled. In a real-life application, those flagged branches could be subject to further investigation to determine their correct phase connection or a method like [53] could be used to try to infer the correct phase connection. Another alternative for correcting the phase label is to re-run DSSE with alternative phases and selecting the one that has the lowest $J_b(\hat{\mathbf{x}})$ and $J(\hat{\mathbf{x}})$ scores.DSSE converged in all 200 tests. It required between 7 and 8 iterations to converge.

4.4.3. Numerical Validation

To investigate how well the DSSE performs under errors in phase labeling, we have designed a Monte Carlo (MC) experiment where the performance of the DSSE to assess voltage magnitude of phases was compared to the performance of a power flow (PF) method under phase label errors. These simulations were performed on OpenDSS using a fixed system condition, so the load level and status of voltage control devices (status of capacitor banks and LTC position) are the same for all tests run in the paper. We have limited the scope of the study to errors in labels of *single-phase service transformers connected to three-phase buses only*. Because the measurements used in the DSSE from the load are in the low-voltage secondary system of the service transformer, any phase errors will directly translate into errors in the DSSE phases.

Two benchmarks were used to evaluate the precision of the results obtained by the DSSE. The first is simply the use of additive noise in the AMI and substation voltage measurements. The second is the use of a conventional PF method to estimate voltage in the load buses based on noisy power measurements obtained from AMI. This second benchmark tries to emulate the common practice of using PF methods in distribution studies to evaluate the performance of a given system in a predefined scenario. The PF method used is an unbalanced three-phase load flow obtained by adding phase label errors to the source code of the OpenDSS circuit file. Note that the voltage measurements are not used in the unbalanced three-phase load flow, only the noisy power measurements are inputs.

The OpenDSS model of EPRI Circuit 5 test system has 595 transformers, including 3 single-phase voltage regulators, and 94 single-phase service transformers connected to three-phase buses. We have selected 5 levels of errors in phase identification of single-phase transformers: 0% (no errors), 1%, 5%, 10%, and 33% of incorrect labels of phases of single-phase transformers. For each scenario, we have run 20 OpenDSS simulations where single-phase transformers connected to three-phase buses were chosen at random and reassigned to an incorrect phase until the percent of incorrect labels was reached. We have assumed that all load and generator (solar PV) nodes are metered, and they report voltage magnitude, real and reactive power to the DSSE. Furthermore, we assumed that the state estimator has also access to voltage magnitude and real and reactive power flow measurements from the substation.

All meters that monitor the substation, loads, and solar PV nodes are assumed to be class 0.5. Meter class defines the maximum percent error of each measurement, and this meter class error drives the ability to accurately model the distribution system. We have modeled measurement errors as an additive Gaussian random variable with a standard deviation equal to one-third of the meter class. Under this model, the noise will rarely be larger than the meter class. This type of model provides a compromise between the Gaussian noise model considered in the WLS-SE formulation and the meter model.

4.4.4. AMI Voltage magnitude prediction

The metric we have chosen to evaluate the results is the root means square of the $n_{|V|} = 1385$ true voltage magnitudes obtained from the simulation ($|V|_i^{true}$, without errors) versus estimated measured or estimated ($|V|_i^{est}$, state estimator and power flow) voltage magnitude: $|V_{rmse}^{err}| = \sqrt{\frac{1}{n_{|V|}}\sum_{i=1}^{n_{|V|}}(|V|_i^{true} - |V|_i^{est})^2}$. All voltage magnitudes are in per unit.

The results summarized in Figure 4-33 show that the results of the DSSE are much more consistent (lower variance) than the results of the power flow method for all cases. The meter error boxplots are shown to provide a baseline on errors due to noise. It is reasonable to assume that estimation errors inferior to noise level of voltage meters are accurate enough. We can see that DSSE root-mean-square error (RMSE) of voltage magnitude error estimates are consistently smaller than noise level for of up to 10% of incorrectly labeled single phase transformers. Furthermore, the DSSE's median RMSE is lower than power flow for all cases, except the first where there is no phase label error. From these results it is possible to confirm that the DSSE is a more robust method for voltage magnitude estimation than the power flow-based method.

4.4.5. DSSE versus PF-based voltage estimation: 10% case

The analysis of voltage magnitude estimation error can also be performed on an *individual* AMI basis. Instead of using the RMSE of all errors, which is a summary metric for characterizing the results of a method for an instance of a MC run, we have compared voltage estimation errors at the individual meter level. We have selected the 20 MC tests performed in the case where 10% of single-phase transformers connected to three-phase buses have incorrect labels and we have compared the result of each voltage estimation using DSSE and PF methods.

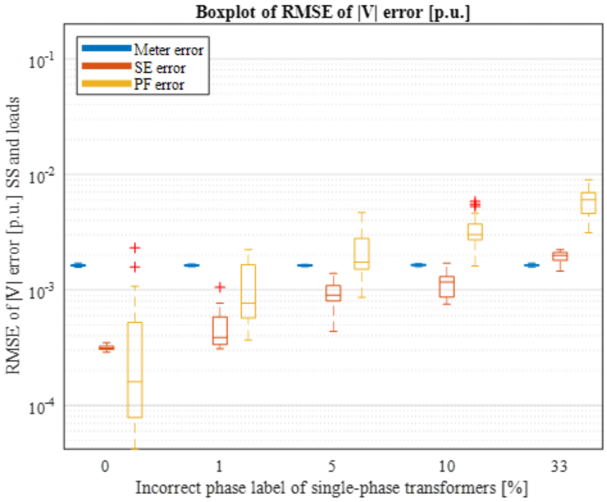


Figure 4-33. Boxplots of voltage magnitude errors under different single-phase transformer phase errors

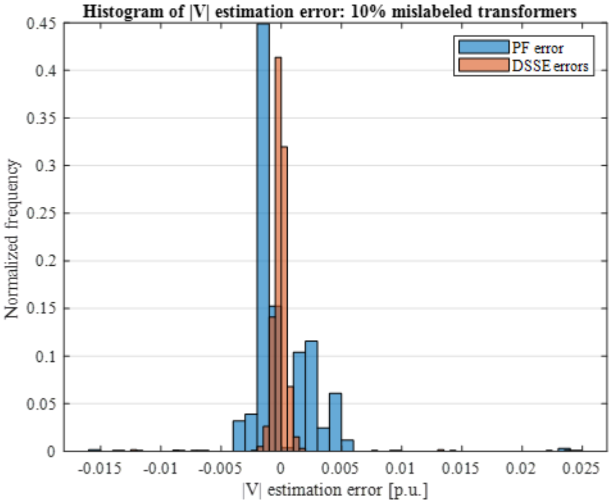


Figure 4-34. Histogram of voltage magnitude errors of estimates obtained with PF (blue) and DSSE (red) for 10% of mislabeled transformers

The normalized histograms in Figure 4-34 were obtained from the voltage magnitude estimation errors of each metered node using the two methods compared to the actual (non-noisy) voltage measurement. The results show that DSSE produces voltage estimates with more errors concentrated close to zero than the PF-based method, which translates into lower level of error. Further, it is possible to see that there are many instances where the PF-based voltage presents significantly large voltage estimation errors, which does not happen as often for the results generated by the DSSE.

4.4.6. Phase error identification using DSSE

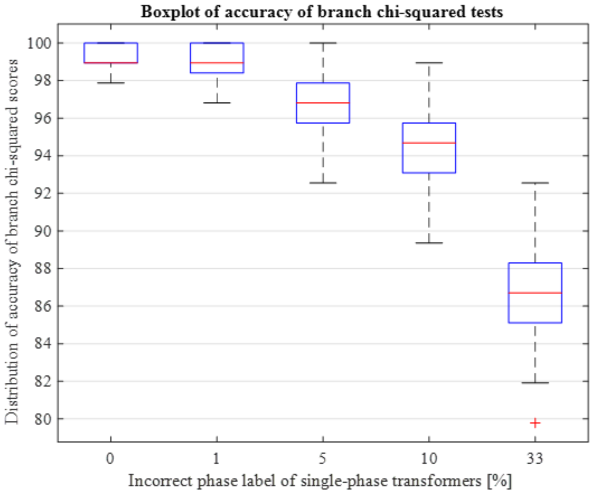


Figure 4-35. Boxplot of the accuracy of the $J_b(\hat{\mathbf{x}})$ phase error detector for each scenario.

While *detection* of errors is trivial[7] using a bad data detection method associated with DSSE, *identification* of the sources of these errors is much more challenging, because it can involve, topology, parameters or measurements. The effectiveness of the $J_b(\hat{\mathbf{x}})$ test for identifying errors in phase labeling of single-phase distribution was also tested in the MC simulations.

In all cases, the number of average detector errors are far smaller than the number of correct predictions, which demonstrates the efficacy of the method for detecting errors in transformer phase labels. Further, the small number of high $J_b(\hat{\mathbf{x}})$ scores in correctly labeled branches indicates that the mismatch between models used for DSSE and the measurements is quite small, which means that we should expect small errors in state estimates in those nodes where we have loads and solar PV generation.

Boxplots for accuracy (ratio between correctly labeled branches over number of all branches) of phase error identification is shown in Figure 4-35. The phase error detection method presents a very high accuracy in cases where the number of phase errors is low, and the score decreases when the number of errors increases. This result highlights the decrease in the performance of the DSSE method when subject to larger amounts of topology errors, which in turn reduces its ability to correctly predict if a given branch has had its phase correctly labeled.

4.5. Phase Identification Summary

This project developed three phase identification algorithms, one that uses only customer voltage timeseries data from AMI meters, one that can also leverage additional sensors placed on the medium voltage if they are available as well as the AMI voltage timeseries, and one phase identification method based on a state estimation approach.

The results show that the DSSE method is a promising tool not only for distribution monitoring but also for the calibration of distribution parameters to improve visibility into DER

[7] Given redundancy of 1.23, we have seen consistent detection of errors using Chi-squared method (χ^2).

operations and planning. Further, even under errors in topology, the voltage estimates by DSSE are highly accurate. These results show that DSSE can be a much more reliable tool than PF for distribution system studies. Also, identification of these phase errors can be achieved through DSSE error processing. Even though the results for the identification of phase mislabeling are satisfactory, they still leave room for improvement.

For the AMI-based methods, there were five publications produced related to the first two phase identification methods, 3 conference papers [13], [35], [40] and 2 journal articles [36], [41]. The algorithms were tested using synthetic datasets to thoroughly test robustness under a variety of real-world data challenges as well as being tested on several utility datasets. In the case of the sensor-based phase identification algorithm, field verification of the results on two utility feeders from Utility #1 demonstrates the effectiveness of the algorithm on utility data and provides high-confidence in using the algorithm. Utility #1 is also in the process of implementing the sensor-based method for widespread usage in their system.

5. SECONDARY SYSTEM TOPOLOGY AND PARAMETER ESTIMATION

5.1. Parameter Estimation

The parameter estimation task is defined as estimating the parameters of the secondary network between customers and transformers. Looking at Figure 5-1, there are two possible secondary networks, shown in blue and red, for the customer at the bottom of the figure. This task seeks to estimate the wire types and wire lengths between each customer and their service transformer. Accurate secondary networks are critical for grid planning task and DER integration tasks such as hosting capacity analysis.

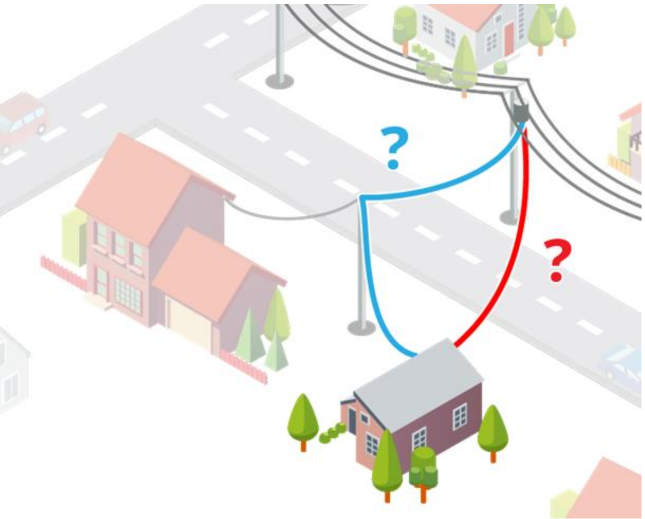


Figure 5-1. Parameter Estimation illustration

5.1.1. Methodology

The overall objective of distribution system secondary circuit topology and parameter estimation (DSPE) is to determine the topology, resistance (R), and reactance (X) parameters of a secondary circuit (shown in red in Figure 5-2) by leveraging AMI measurements of voltage (V), real power (P), and reactive power (Q) (shown in blue in Figure 5-2). This paper focuses on typical North American split-phase secondary circuits that can be perfectly represented (under appropriate assumptions) with single-phase equivalent circuits. More details on split-phase secondary circuit modeling can be found in [55].

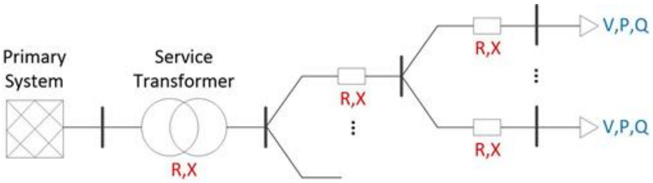


Figure 5-2. Secondary circuit topology and parameter estimation problem [56]

We follow the linear regression parameter estimation (LRPE) method for secondary circuits described in [57] and [28], [58], [59]. The LRPE method utilizes the linear approximation of voltage difference between two customers connected in parallel:

$$|V_1| - |V_2| \approx I_{R2}R_2 + I_{X2}X_2 - I_{R1}R_1 - I_{X1}X_1 \qquad (5\text{-}1)$$

Where V_1 and V_2 are the voltages measured at customers 1 and 2, I_{R1} and I_{R2} are the real currents ($I_R = P/V$) and I_{X1} and I_{X2} are the reactive currents ($I_X = Q/V$) flowing to each customer P is the measured real power and Q is the measured reactive power. This linear approximation typically has negligible error [60]. By convention, we assign customers 1 and 2 such that the voltage is, on average, higher at customer 1 than customer 2 (i.e., mean $(V_1) \geq$ mean (V_2)). Many utilities do not know the secondary circuit topologies.

In [56], we apply linear regression topology and parameter estimation (LRTE) algorithm to generate the entire secondary circuit models, including topology, using only the measurements. The algorithm processes one secondary circuit at a time (e.g., the circuit containing all customer on a single transformer). For each meter pair on that circuit, the algorithm solves the linear regression problem based on a slightly modified version of Equation. (5-1):

$$V_1 - V_2 = I_{R2}R_2 + I_{X2}X_2 - I_{R1}R_1 - I_{X1}X_1 + E \qquad (5\text{-}2)$$

where E has been added to account for a possible offset (bias error) in the meter voltage measurements. Note this equation applies to both the situation where the wires between the transformer (or some other upstream node) and the two customers are parallel and when they are in series: for series connections, I_{R1} and I_{X1} will be found to be zero by LRTE. The values of R and X are recorded as the parameters for that meter pair, but the value E is not recorded.

The meter pair with the best fit (as determined by the highest Pearson correlation coefficient: R_2value) when using linear regression is assumed to be the true topology. The method then proceeds to estimate the remaining secondary circuit parameters by proceeding from the tree leaf nodes towards the tree root node. That is, the two customers found to be the best pair create a virtual node at their meeting point. This virtual node is then run through LRTE with the remaining meters on the circuit until the full circuit topology is found.

5.1.1.1. Step 1: Transformers with Multiple Customers

To compute the parameters and topology of all customers on a feeder, we use the three-step process that follows. First, we determine the topology and parameters of the secondary circuits of transformers with more than one customer by pairing the customers on the same transformer with one another. Next, we pair customers who are the only customers on a transformer with other such customers to derive the parameters for these single customers. Finally, we pair transformers with multiple customers with one another, using the virtual nodes found in the first step, to determine if there is additional impedance between the final virtual nodes and the transformers. Combined, these three steps derive the secondary circuit topology and parameters for all customers connected to the feeder. The three steps are described in detail in the following sections.

For transformers with multiple customers connected, all possible pairs of customers on the same transformer are evaluated. For example, for transformers with 3 customers, all 3 possible combinations of customers pairs (1 and 2, 2 and 3, 3 and 1) are evaluated. AMI data for voltage and real and reactive power, are used to solve for R_1, X_1, R_2, and X_2 using Equation 5-2.

To understand how well the linear regression fits the data, we computed the Pearson correlation coefficient (R^2) of the fit.

Once R^2 values were computed for all customer pairs, the pair with the highest R^2 value was assigned as the actual topology. If this pair was found to be connected in parallel, then a new virtual node, representing the point where these two parallel lines connect, was created. The voltage at the virtual node was found by adding the voltage drop to the measured voltage:

$$V_{virtualnode} = V_1 + I_{R1}R_1 + I_{X1}X_1 \qquad (5\text{-}3)$$

Based on the convention previously mentioned in Section 5.1.1, mean(V_1) ≥ mean(V_2). Thus, Equation 5-3 ensures that the volt- age at the virtual node is, on average, higher than the voltage at each of the customers. Other methods for computing $V_{virtualnode}$, such as averaging the voltage drop found across each customers 1 and 2, does not guarantee that the virtual node voltage is higher than the customer voltages, which could lead to problems in later parameter calculations involving the virtual node. Real and reactive power consumption at the virtual nodes were found by summing the real and reactive power of the two customers.

As a second iteration, all remaining customers (those not contributing to the virtual node) were again paired with one another and also were paired with the virtual node. Again, the pair with the highest R_2 value was assigned as the actual topology, and a virtual node was created. This method was repeated until all customers were included in the topology. We note that this linear regression is very fast, taking less than 0.1 s on a typical laptop circa 2016. Because of the iterative nature, the number of linear regressions that must be calculated increases with the number of customers. However, total runtime remains modest even for transformers with many customers; the linear regressions for all pairs of a 20-customer transformer, for example, can be run in about 2 minutes. Additionally, this process only needs to be run once: DSPE derived parameters will continue to be valid unless there is a system configuration change.

For most customers, the voltage fluctuations depended heavily on the real current draw and only weakly on the reactive current. That is, the reactive current was typically only responsible for explaining a small fraction of the variance in the voltage drop, in part because the real and reactive power draws were often well-correlated. Therefore, we have more confidence in the derived resistance values than reactance values. In some cases, including when negative reactance is found, it may be best to use the found resistance and then assume a reasonable X/R ratio to determine the reactance.

5.1.1.2. Step 2: Transformers with Only One Customer

On transformers with a single customer, Step 1 will not work: we only have one voltage measurement (at the one customer), and so cannot solve for R and X between two voltage measurements using Eq. 5-2. Instead, we find a nearby transformer that also has a single customer by finding the shortest latitude/longitude distance to another single-customer transformer on the same phase. Because of the high voltage level on the primary system, the per unit resistance on the primary voltage system between two nearby transformers is much smaller than the per unit resistance on the low voltage secondary system from transformer to customer, as per resistance is inversely proportional to voltage squared.

For example, on a 12-kV system, 900 feet of wire has the same per unit resistance as approximately 1 foot of wire on the low-voltage system. Thus, the primary voltage side of two nearby service transformers can be assumed to be identical with little impact to the estimated resistances. Thus, two customers on nearby transformers (who are each the only customer connected to their transformers) can be compared in the same fashion as described in Step 1 as though they are on the same transformer.

However, following this method means that we have included the transformer resistance and reactance in the parameter estimation. Utilities typically have access to the kVA rating of transformers, and hence the transformer resistance can be fairly accurately estimated using the values based on National Electrification Association specifications. There are no United States standards for transformer impedances, but we used a typical value of 2% impedance for all transformer sizes. The transformer resistance and reactance are subtracted out from the estimated parameters to find the customer resistance and reactance. An additional benefit of this method is that including the transformers in parameter estimation can validate or identify errors in the transformer sizes in the utility models. Specifically, the one noted time when a negative value resulted for customer resistance, inspection found this was clearly a typographical error in transformer kVA, which was labeled as 1 kVA but was in fact 10 kVA.

5.1.1.3. Step 3: Pair Multi-Customer Transformers With One Another

The impedances found in Step 1 are not guaranteed to be the complete impedances of the secondary circuit because the final virtual node found may not be located at the transformer and there may be additional wiring between the virtual node and the transformer leading to additional impedance. To account for this possibility, we paired transformers with multiple customers with one another. This is similar to Step 2, though instead of using voltage and power measurements at single customers, we used the calculated voltage and power at the final virtual nodes found in Step 1.

In most cases, the virtual node from Step 1 was at the transformer, such that after subtracting out the transformer impedance, essentially zero remaining impedance was found. In some cases, slightly non-zero impedances were found (e.g., a resistance of 0.002**Ω**), which are likely due to slight variation of transformer impedance from the assumed values.

In some cases, though, the virtual node from Step 1 was away from the transformer. For these cases, the additional impedance of the wiring between the transformer and the virtual node must be accounted for and added to the secondary topology found in Step 1.

5.1.2. Results

5.1.2.1. Results on Synthetic Data

The experiment design and results for different cases have been summarized below. Corresponding trends can be observed in Figure 5-3 for error in pairing transformers and Mean Absolute Error (MAE) of all resistance estimations. The pairing transformers algorithm results shown are for an initial meter-to-transformer pairing algorithm based on the parameter estimation algorithm. The final version of the meter-to-transformer pairing algorithm is described in detail in Section 6.

1) **Sampling Time Interval**: The 1-minute granularity time series measurements are grouped by the new interval size and then averaged (or the last reading is taken for instantaneous voltage measurements) to obtain the new timeseries measurement for 5, 15, 30 and 60 minutes.
Observations: Figure 5-3a shows that when the data is perfect, i.e., no noise is added, the time interval does not impact accuracy. Averaged voltage data performs better than its instantaneous counterparts for larger intervals.

Data Resolution: Each measurement for the required time interval data set is rounded to the desired resolution. The maximum resolution provided by the simulation is 10mV and 0.1W. This corresponds to 2 and 4 decimal points each.

Observations: At least 0.1V and 0.1kW resolution (1 decimal point each) was required to maintain reasonable levels of accuracy as shown in Figure 5-3b. Finer resolution beyond that do not result in significant improvement.

2) **Time Synchronization:** A maximum offset from 0-4 minutes was added to each meter at the 1-minute interval resolution, and its impact was studied for the different time intervals.

   Observations: As the reporting time interval increases, the algorithm becomes more robust to time displacement errors (Figure 5-3c). This assumes the quantity of data is the same.

3) **Quantity of Data:** The impact of quantity of data was studied by utilizing data from different numbers of months.

   Observations: A small and steady increase in the accuracy was observed when more data is utilized. Figure 5-3d shows a 2-3% improvement in performance was observed between using 1 month and 1 year of perfect (noiseless) data.

4) **Meter Bias:** For each meter a bias error is chosen at random and added to each measurement of that meter. The bias ranged from 0-2% of the mean at intervals of 0.5% and sampled uniformly between the negative and positive of the maximum bias level.

   Observations: Figure 5-3e reveals that bias does not impact linear regression or correlation methods. For example, in the former it is absorbed as constant noise and only impacts the intercept and not the slope of the line.

5) **Uniform Noise:** Noise is injected into each meter reading at random within the range [-Max, +Max]. The maximum noise level is varied from 0 to 1% of the nominal at steps of 0.1% and the noise is distributed uniformly between the negative and positive of the maximum level.

   Observations: Noise noticeably affects the accuracy of the algorithm as seen in Figure 5-3f. Averaged and larger time intervals (15 minutes) tend to be more robust to noise; 0.35-0.55% noise in voltage data is a reasonable threshold. For smaller time intervals (5min) a significant increase in error is seen beyond the 0.35% threshold. For larger time intervals, while the increase in error is more linear the accuracy drops below 95% beyond the recommended noise range.

6) **Missing Data:** Random data points from the data sets were dropped after averaging to the required time intervals. This can be caused due to failure in the communication network, packet drops, outages etc.

   Observations: Missing data does not impact the algorithm performance in both cases if sufficient data is used (here, 4-6 months). This is possibly because linear regression and correlation can perform well in the presence of missing data.

7) **Unknown Power Factor:** Often, smart meters do not record kVAr, and there is no power factor information available. Many algorithms based on AMI data, require kVAr readings. To get around this, random power factors were employed. Additionally, the value of knowing the type of cable (here, X/R ratio= 0.304) was explored.

Observations: Knowledge of the cable type used by the utility is extremely powerful. Figure 5-3h compares the results of DSPE when kVArh was measured (indicated by 1) vs when kVArh was not measured but the cable's X/R ratio was utilized (indicated by 0). Using a fixed X/R ratio, i.e. constant line slope in the linear regression formulation, the algorithm performs as well as the original case.

While Figure 5-3 compares the impacts of testing a range of individual error cases, these scenarios might not provide sufficient insight into optimal AMI capabilities as many of these errors occur simultaneously. Thus, to mimic realistic data with reasonable error levels, the following errors were injected into the 1-minute interval base data set. The noise levels conform with meter standards such as ANSI C12.

- Data Resolution – 0.1 V
- Meter Bias – 0.2%
- Meter Precision (Noise) – 0.2%
- Time Synchronization Issues — 1 min displacement for 20% of meters

The results from the simulations with multiple errors simultaneously injected show similar overall trends to the simulations of the errors in isolation from Figure 5-3. Table 2-3 (in Section 2.4) shows combined recommendations for topology estimation algorithms using smart meter data based on the results from testing the AMI data parameters and errors in isolation as well as comparing with the results from injecting errors simultaneously.

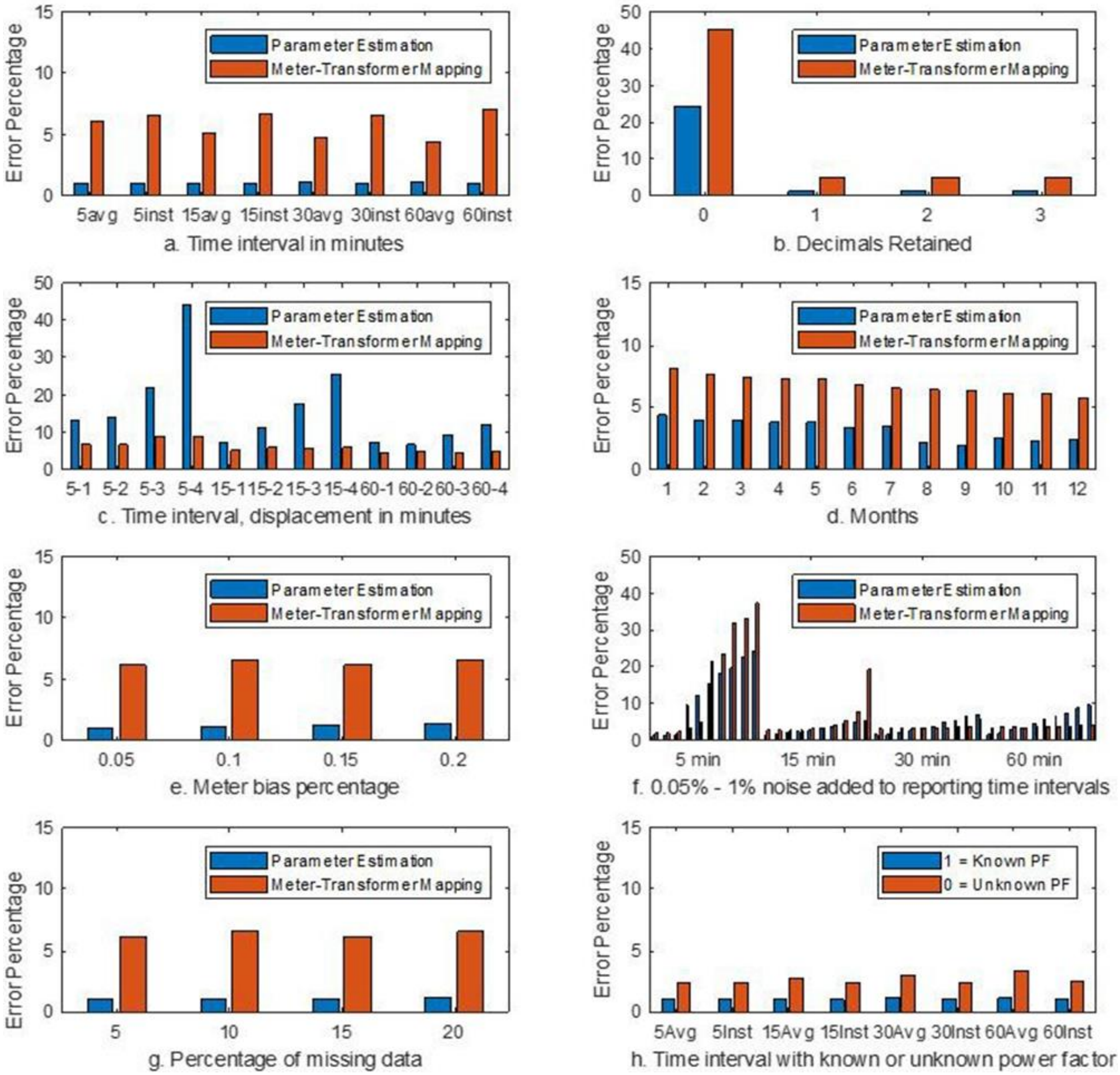


Figure 5-3. Algorithm Accuracy Results when varying: 1) Reporting Type and Measurement Interval, b) Data Resolution, c) Time Synchronization, d) Quantity of Data, e) Meter Bias, f) Uniform Noise, g) Missing Data, and h) Unknown Power Factor

5.1.2.2. Results on Utility #2

In this section, we present results of DSPE applied to data from Utility #2. DSPE-found topology and parameters are compared to the utility secondary model. To compare parameters, DSPE resistance is compared to model distance (following conductors in the model) using typical resistances per distance for different wire types (Appendix A in [61]). For example, aluminum #2 American Wire Gauge (AWG) triplex has a resistance of 0.058Ω per 100ft (round-trip resistance for a 240 V load).

Figure 5-4 shows results for a transformer for Utility 2, Feeder #1 with two customers. DSPE results (Figure 5-4a) show a parallel configuration, with similar resistances for the two customers. The utility secondary model (Figure 5-4b) confirms a parallel configuration and similar distances (and hence expected resistances) from the transformer to each of the customers. This relationship between DSPE resistance and utility secondary model distance from transformer to customer is plotted in Figure 5-4c, and shows that DSPE results are consistent with the secondary model distances, assuming #2 AWG wiring, which is indeed commonly used by this utility for secondary systems.

One value of DSPE: resistances from transformer to customer cannot be calculated from secondary topology alone (even if that is available from a utility); accurate wiring type is also required. DSPE, however, resolves the topology and parameters without any need for the wire type as input, and should be accurate for all possible wire types.

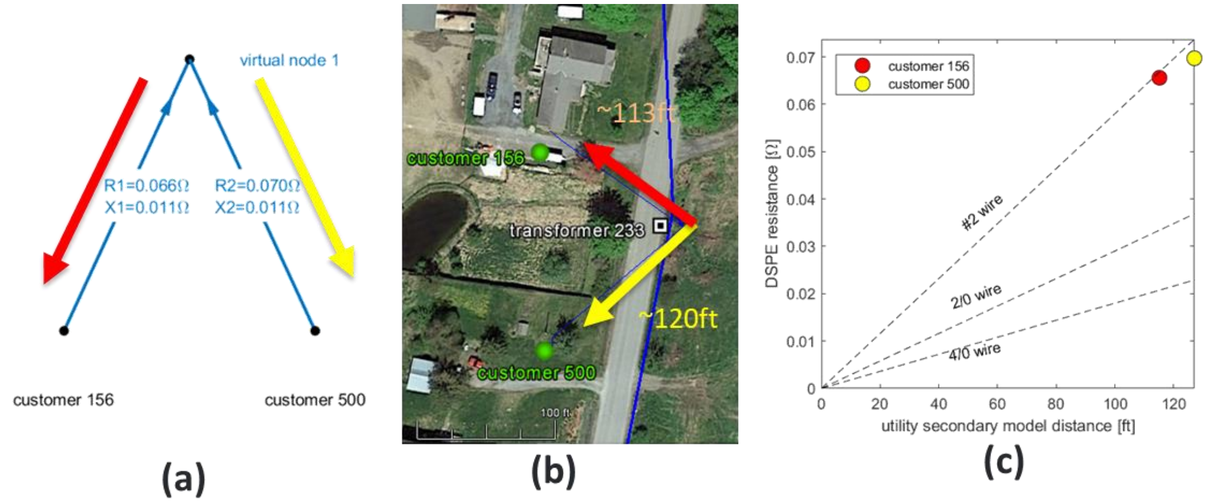


Figure 5-4. (a) DSPE results for a transformer with two customers on Utility #2, Feeder #1. (b) Satellite map showing utility secondary model for connections of customers to this transformer. (c) Scatterplot of the DSPE resistance (y-axis) versus the utility secondary model distance (x-axis). Dashed lines show the resistance of different wire types.

DSPE was run for all customers on Utility 2 Feeder #s 1, 2, and 3. Summary results, showing the distribution of DSPE resistances found, are shown in Figure 5-5. On all three feeders, it is most common to have customer resistances around 0.05Ω, which is consistent with about 100 ft of #2 wire. However, even though this is most common, there are several customers varying significantly from this value: resistances of 0.15Ω, three times the 100 ft assumption, were still common. This shows the value of DSPE on a feeder-wide basis, as it can be used to determine customer-specific parameters rather than relying on assumptions, which may, on average be accurate but which on a customer-by-customer basis may significantly under (or over) estimate the resistance. The impact of errors in the resistance and reactance parameters, such as from the errors that result from assuming 100 ft of #2 wire, is partially dependent on the power draw of each customer.

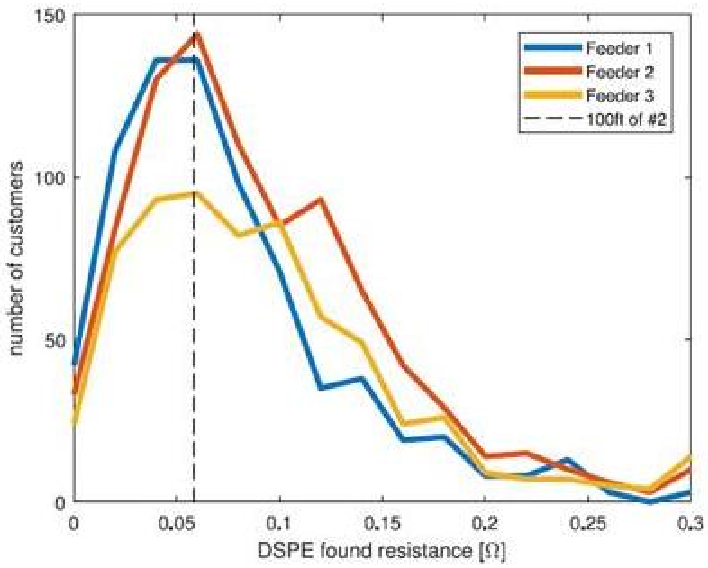


Figure 5-5. Distribution of customer resistances (x-axis) across each of the three Utility 2 feeders

Customers with high power draw and significant differences between actual resistance and modeled resistance will have large discrepancies between measured and modeled voltage. An extreme example can be seen in Figure 5-6. Customer 541 on Feeder #1 is a dairy barn with high power draw (up to 40 kW) and low DSPE resistance between the transformer and customer meter. The 100ft of #2 wire assumption overestimates the voltage drop, resulting in a large underestimation of the customer voltage.

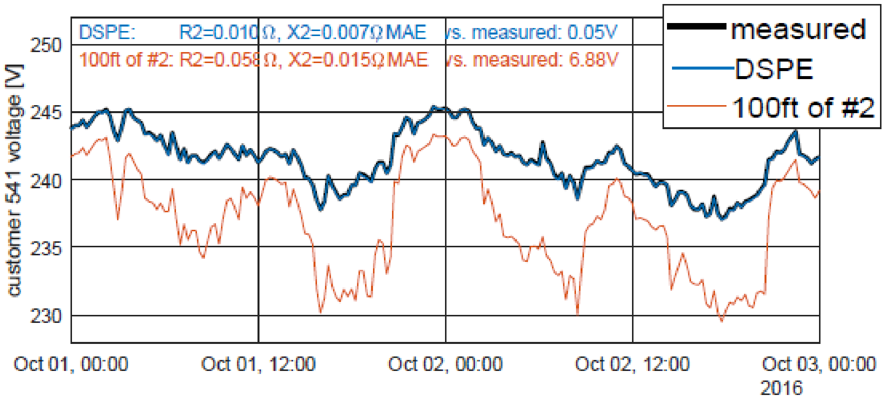


Figure 5-6. Measured voltage and calculated voltage based on DPSE parameters and a 100 ft #2 wire assumption for customer 541 on Feeder #1. The resistance values for each method and the MAE compared to the measured voltage are shown in the

5.1.2.3. Results of the Sensitivity Analysis

We applied filters to the input data to test the sensitivity of DSPE results to input data quality of voltage, real power, and reactive power AMI measurements. We first compared different amounts of input data to DSPE. For example, for transformer 233, the DSPE resistances of both customers when considering only 2,000 data points was within 5% of the value found when considering all 17,023 data points. 2,000 data points corresponds to approximately 21 days of 15-minute resolution data. That is, DPSE could have been accurately applied to transformer 233 after less than a month of data collection. Other transformers had similar results: of 25 transformers with two customers that were evaluated for sensitivity to the amount of data, all but three had converged to less than 5% difference from full data when 8,000 data points were considered (less than 3 months of 15-minute resolution data). When this difference threshold was relaxed to 10%, all 25 test transformers had converged by 6,200 data points (about 2 months of 15-minute data).

Next, we examined the sensitivity of DSPE results to varying resolutions of voltage, real power, and reactive power measurements. The actual resolution of the measurements was 0.1 V, 0.04 kW, and 0.04 kvar; worse resolutions were simulated by rounding the data to different intervals. The DPSE results with these worse resolutions were compared to the results with full resolution to determine the impact of the worse resolution.

The sensitivity to voltage resolution was small. For transformer 233, results within 5% of the full resolution case were obtained for voltage resolutions as poor as 1 V. 20 of the 25 two-customer transformers evaluated for sensitivities needed voltage resolution of only 1 V to maintain DSPE results with 5% of full resolution data; and all 25 were within 10% of full resolution data when using only 1 V resolution voltage.

Real power resolutions of 0.5 kW or less resulted in greater than 5% deviations in DPSE resistance values versus full resolution. Over the 25 test transformers, 18 of 25 resulted in less than 5% deviation from full resolution data when using 0.5 kW resolution power and 23 of 25 resulted in less than 10% deviation. Reactive power resolutions of worse than 0.5 kvar similarly led to greater than 5% deviations from full resolution results. However, accurate DSPE results did not depend on having reactive power measurements at all. W hen no reactive power measurements are used, DSPE results are very similar (less than 3% difference) to results when using full resolution reactive power measurements. Similarly, over the 25 test feeders, the difference between using no reactive power measurements at all and using full resolution reactive power measurements was always less than 10%. This is likely caused by the high correlation between real and reactive power and is a significant result as it demonstrates that DSPE can be accurate even when no reactive power measurements are available. Additional sensitivity analyses were performed to understand the impact of measurement errors to DSPE results. Adding stochastic errors to the AMI measurements introduced stochastic differences in DSPE results. However, for added stochastic measurement errors ranging from 0% to 10% of average measured voltage, real power, or reactive power, DSPE results were still within 5% of the results with no stochastic errors added. Constant bias errors in measurements (i.e., if voltage measurements are 2V too high over all timesteps) have no impact on DSPE results.

5.2. Summary of Parameter Estimation

Distribution system parameter estimation (DSPE) results agreed well with imagery and the utility secondary model for the customers examined. When applied over entire feeders, significant variations were seen from the simple assumption that all customers have 100 ft of #2 wire, demonstrating the value of DSPE. Sensitivity analysis showed that the DSPE method is robust: modest amounts of data (around 3 months of 15-minute resolution data) and measurement resolutions (1 V, 0.5 kW) were sufficient to derive accurate parameters. For this set of mostly residential customers, reactive power measurements were not found to be necessary, but this may change for other customers with less correlated real and reactive power consumptions. The DSPE method can be leveraged to identify and fix errors in the utility geographic information system (GIS) data. It can also be leveraged to automatically create accurate and detailed secondary system models based on AMI measurements while minimizing the need for costly manual labor and field inspections. As a result, the presented practical DSPE method can help utilities to greatly increase the accuracy and detail of their distribution grid models to enable high DER penetrations. This work resulted in several publications [32], [62], [63].

6. METER-TO-TRANSFORMER PAIRING

6.1. Task Overview

The meter-to-transformer pairing task is defined as grouping customers by their service transformer using customer AMI data. For example, the goal is to determine if the customer in Figure 6-1 is connected to the transformer and customers on the left, or the transformer and customer on the right. In this case, a two-stage methodology was developed that leverages customer voltage, real power, reactive power timeseries, and the existing service transformer labels. In the first stage, errors in transformer labeling are flagged using pairwise correlation coefficients calculated from the voltage timeseries of each customer. In the second stage, the results from pairwise linear regression, i.e., the mean-squared-error (MSE) fit value and the reactance distance between customers, are used to assign new customer groupings to the customers on the flagged transformers. Figure 6-2 shows a flowchart of the methodology with the two stages marked with brackets.

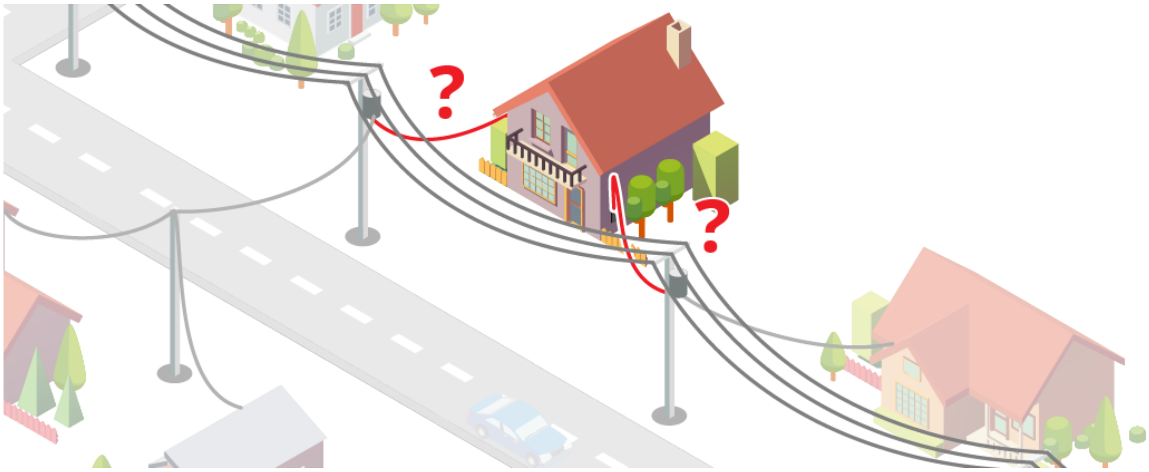


Figure 6-1. Meter-to-transformer pairing illustration

6.2. Methodology

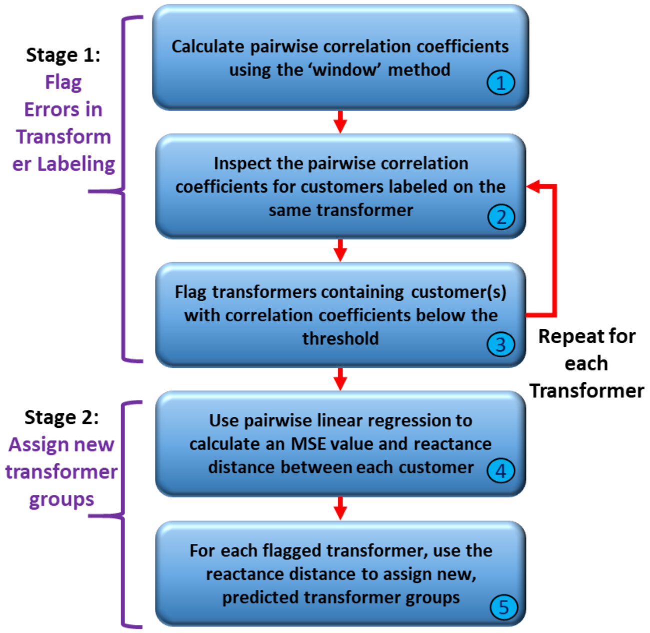

Stage 1:
Flag Errors in Transformer Labeling
Calculate pairwise correlation coefficients using the 'window' method
1
Inspect the pairwise correlation coefficients for customers labeled on the same transformer
2
Flag transformers containing customer(s) with correlation coefficients below the threshold
3
Repeat for each Transformer
Stage 2:
Assign new transformer groups
Use pairwise linear regression to calculate an MSE value and reactance distance between each customer
4
For each flagged transformer, use the reactance distance to assign new, predicted transformer groups
5

Figure 6-2. Meter-to-transformer pairing methodology flowchart

6.2.1. Methodology Overview

The first stage of the methodology is to flag errors on a per-transformer basis. In Step 1, Pearson correlation coefficients are calculated between each pair of customers. The total number of correlation coefficients would be the number of customers squared divided by two (as order does not matter). This number could be further reduced if computation time became a problem by only calculating the pairwise correlation coefficients for customers labeled on the same transformer. As a pre-processing step, the voltage timeseries are converted into per-unit representation and the difference taken between adjacent measurements in time. This results in a timeseries that represents the change in voltage at each timestep, measured in per-unit. These pre-processing steps are based on the literature and prior work in the phase identification task. To account for missing values in the data, individual 4-day "windows" of data are considered and any customers with missing data during that time period are discarded. Once the entire dataset has been processed in this way, each pair of customers will have several correlation coefficients, one from each window where they were both present. The mean of all the correlation coefficients is taken to be the final value. In steps 2 and 3, each transformer is inspected independently. The correlation coefficients for all customers labeled on that transformer are inspected to verify if all those customers are well correlated with each other; if any customers are poorly correlated with other customers, then that transformer is flagged. This operation is repeated for all transformers. Setting the threshold for classifying "poor" correlations is a critical question for this method. This threshold is referred to as β in Algorithm 6-1. We propose a ranking system for the flagged transformers. By using multiple values for this threshold, a ranking of flagged transformers is created, with transformers containing extremely poorly correlated customers flagged first. This prioritizes the transformers that are more likely to contain errors in labeling.

The second stage of the methodology is to re-assign the customers in each flagged transformer into their correct transformer groupings. This is one of the key contributions of this paper because most algorithms are not able to identify the correct transformer when errors are identified. Identifying the correct grouping is achieved using the results of linear regression, Step 4. Equation 6-1 shows the formulation of the regression problem, and Figure 6-3 shows a visual representation of the formulation. The regression is done for each pair of available customers. This is ordinary least squares (OLS) regression formulation, using the Scikit-learn implementation in Python [43]. On the left side is the voltage difference between the two customers (v), and on the right side is the real power (p), reactive power (q), and the regression coefficients (r) and (x). In this case, the regression coefficients directly correspond to the electrical resistance, in the case of (r) and the electrical reactance, in the case of (x). This can be seen for the two example customers shown in Figure 6-3 where the r represents the electrical resistance between each customer and the closest point of electrical connection (purple box) and x represents the reactance between each customer and the closest point of electrical connection. Further details on this formulation can be found in [27]. This formulation produces a pairwise measure of the regression fit, MSE was chosen in this case, as well as coefficient values corresponding to the resistance and reactance "distances" for the pair of customers. The "reactance distance" is defined as the sum of the two x terms in Figure 6-3. Likewise, the "resistance distance" is defined as the sum of the two r terms. The MSE value is used as a "goodness of fit" metric for the pairwise regression. Although not a traditional "goodness of fit" metric, the MSE is a measure of the error on the regression fit between customers and can thus be used to quantify the "goodness" of the regression fit. The MSE formula is shown in Equation 2, where n is the number of timesteps available and v is the difference in voltage between customers, shown on the left side of Equation 6-1.

$$v_1 - v_2 = \frac{p_1}{v_1} * r_1 + \frac{q_1}{v1} * x_1 - \frac{p_2}{v2} r_2 - \frac{q_2}{v_2} * x_2 \qquad (6\text{-}1)$$

$$MSE\left(V, \hat{V}\right) = \frac{1}{n} \sum_{i=0}^{n} (v_i - \hat{v}_i)^2 \qquad (6\text{-}2)$$

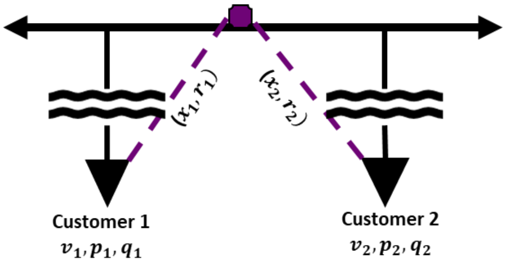


Figure 6-3. Linear regression model to calculate the measure of fit (MSE), reactance distances (x), and resistance distances (r). The box represents the closest point of electrical connection between the two customers.

There are two situations where the regression fit will be good, and Figure 6-4 illustrates those cases. On the top, the pairwise regression was between two customers serviced by the same transformer, and the point of electrical connections (purple box) is on the low-voltage side of the transformer. On the bottom, the two customers are each serviced by their own transformer and the closest point of electrical connection is on the high side of the transformers. These two cases can be distinguished from one another using the reactance distance between the pair of customers. Because transformers have higher X/R ratios than lines, crossing over two transformers will result in significantly higher calculated reactance values than in the case shown on the top. The reactance threshold to differentiate between these two cases is thus set at 0.046 Ohms based on the known reactance of two transformers in this model. The reactance threshold parameter is defined as γ in Algorithm 6-1.

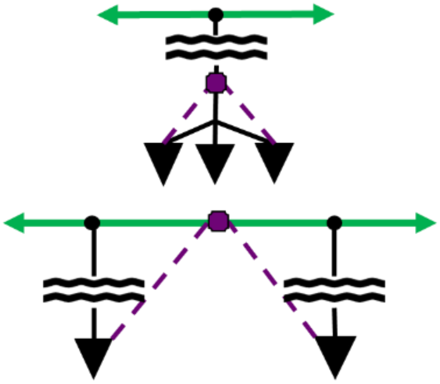


Figure 6-4. The regression fit is good in two cases. The top case shows the regression between two customers on the same transformer and the bottom case shows the regression between two customers each being the only customer served by a transformer.

In the case where the regression fit is poor (high MSE values), there are again two situations. These situations are demonstrated in Figure 6-5. On the top, the pair of customers are connected to different phases. On the bottom, the pair of customers are serviced by different transformers that also serve other customers. The regression fit is poor in this case due to the influence of the other customers on those transformers not considered in the physics-based regression model used. Both these cases can be identified by the poor fit, and they are eliminated from consideration as possible pairs of customers on the same transformer. Thus, any pair of customers with a high MSE value is not considered a candidate to be labeled on the same transformer. The MSE threshold to differentiate between the cases shown in Figure 6-4 and Figure 6-5 is defined as μ.

In Step 5 from Figure 6-2, each of the customers on the flagged transformers is considered for re-assignment to a new transformer grouping. If a customer has a low correlation coefficient (β) relative to other customers in its original transformer grouping, then it is re-assigned. Otherwise, the original transformer label is retained. If it is to be re-assigned, then the remaining pairwise reactance distances (pairs with MSE values lower than the threshold μ) are examined. The customer is then assigned a new transformer grouping with customers having reactance distances less than the 0.046 Ohms threshold, γ. If no such customers exist, then that transformer is considered to have only that one customer.

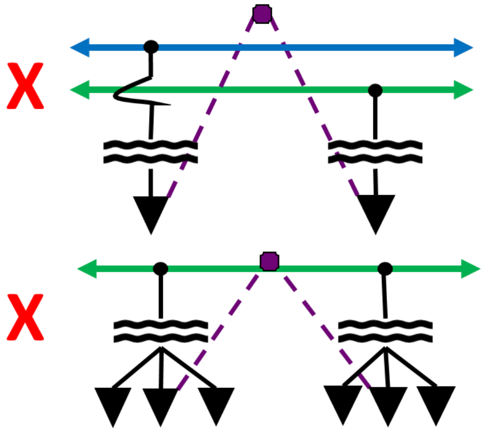


Figure 6-5. The regression fit is poor in two cases. On top, the regression is between two customers connected to different phases, and on the bottom, the regression is between two customers served by different transformers that also serve other customers.

Algorithm 6-1. Algorithm description where β is the correlation coefficient (CC) threshold, μ is the MSE threshold, and γ is the reactance distance threshold

```
Stage 1:
For each original transformer group
  If any CC < $\beta$
    Flag transformer
  Else
    Transformer group is considered good and no action is taken
Sort flagged transformer list by lowest CC found

Stage 2:
For each flagged transformer
  For each customer labeled on the flagged transformer
    If the majority of CC with other customers in the transformer group < $\beta$
      Calculate pairwise regression with all other customers
      If any (MSE < $\mu$) and (($x_1 + x_2$) < $\gamma$)
        Assign new transformer grouping to that set of customers
      Else
        Customer is on a transformer by itself
    Else
      Customer retains original transformer label
```

6.2.2. *Setting the MSE Threshold Parameter*

A key parameter in this methodology is the MSE value which is considered 'high' (μ) to discard customers in the situations described in Figure 6-5. The MSE value produced by the pairwise linear regression is primarily determined by if the physical connection is correct for the regression fit between the two customers. However, the underlying characteristics of the dataset also contribute to the range of the MSE values. Testing has shown that measurement noise will shift the MSE values to be higher or lower overall, depending on the level of measurement noise. We propose to use the minimum MSE value to set the MSE threshold parameter correctly. The minimum MSE value is the lowest across all pairs of all customers, meaning the MSE for the "best fit" pair. This gives an indication of where the acceptable range of values occurs for a particular dataset. Figure 6-6 demonstrates this fact for three different levels of injected measurement noise. For this type of testing, measurement noise up to a +/- maximum value was injected uniformly at random to each measurement in the dataset. The three cases shown in Figure 6-6 are as follows: The clean dataset, no measurement noise injected, is shown in blue. The medium noise injection, shown in green, is 0.1% maximum of the nominal voltage value, which is 240V in this case.

Maximum noise values of 100W and 100Var were chosen for active and reactive power respectively. The largest noise injection, shown in purple, is 0.2% maximum for the voltage, 200W for real power and 200VAR for reactive power. The American National Standards Institute (ANSI) defines meter accuracy classes of 0.1, 0.2, and 0.5 [50]. Those meter standards were the drivers behind choosing those particular values for the noise injection. The stars mark the minimum pairwise MSE value for the dataset, and the lines indicate the acceptable range for the MSE value determined to be the "high" threshold. This means that for a given case, any MSE value on the line will correctly discard the pairwise scenarios shown in Figure 6-5.

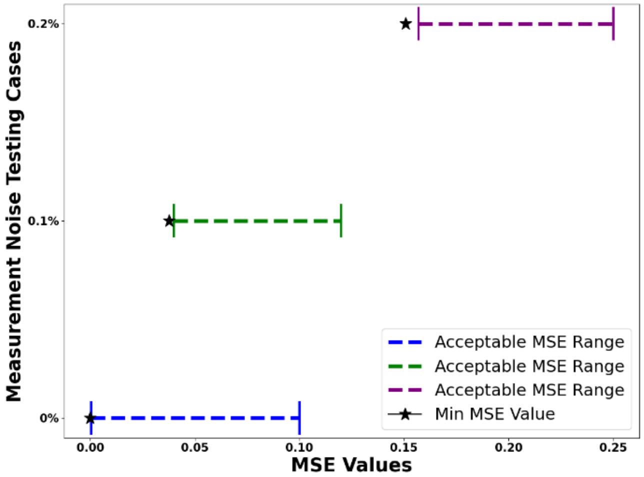


Figure 6-6. Example of the acceptable values for the MSE threshold (μ) based on three different injected measurement noise scenarios. Stars mark the minimum MSE value for each case and the dashed lines between the vertical markers mark the acceptable range to filter customer pairs based on the MSE threshold.

Figure 6-6 shows the shift to the right in MSE values as the measurement noise in the dataset increases, meaning that the acceptable MSE values are higher if the overall noise in the dataset is increased. This demonstrates that there is not a universal value of μ that could be found. Thus, a method of setting the value of μ based on the characteristics of the dataset must be used. Notice that in each case, the minimum MSE value (star marker) is just before the acceptable range. Even in the 0% and 0.1% there is slight separation between the minimum MSE value and the acceptable range. Also notice that the acceptable range is relatively large in each case, approximately 0.1. We propose using the knowledge of the minimum MSE as a starting place to set an MSE threshold within the acceptable range. For example, by taking the minimum MSE value and adding a small factor to it. Thus the "good" and "poor" regression fits described in Figure 6-4 and Figure 6-5 are defined in this manner. In the results shown in the following section, we used 0.01 as the additive factor to determine μ. This type of heuristic does need further verification using utility datasets. However, it is demonstrated to work well in the simulations presented in this work.

This methodology results in a set of transformer labels that contain both original transformer labels and new customer groupings. Due to the large number of possible configurations, it is not possible in most cases to map the new transformer groupings to original transformer designations. One example of this is the case where four customers were originally labeled on one transformer and the results show that two customers are serviced by one transformer and two customers by another transformer. In this case, there is no way to determine which pair should retain the original label using the given information. There are many such other examples.

However, the groupings of customers produced by the proposed methodology will match the customer groups served by a particular transformer. Therefore, the new transformer groupings must be assigned physical transformer names as a post-processing step.

6.3. Synthetic Results

The methodology proposed above was validated on the synthetic dataset described in Section 3.1.1. Using this type of data ensures that a detailed analysis of the results using ground truth transformer labeling can be accomplished.

To simulate customers with incorrect transformer labels in the results below, a specified percentage of customers had their transformer labels altered to an incorrect label. The incorrect labeling and the choice of customers to mislabel were done uniformly at random. Measurement noise and missing data issues were also injected into the dataset for a subset of the results shown below. The measurement noise and missing data injections were done uniformly at random.

The primary metric reported in this section is "transformer accuracy." In the context of this paper, transformer accuracy is defined as the percentage of transformers that have the correct group of customers assigned to them (i.e., the transformer groupings) divided by the total number of transformers in the system. This metric requires that, for a particular transformer grouping to be considered accurate, the group must contain all customers actually serviced by that transformer and also not contain any customers not serviced by that transformer.

The remainder of the results section is divided into two subsections. In the first subsection, results for the error flagging stage are discussed. Then in the second section, results for the end-to-end method are discussed.

6.3.1. Transformer Error Flagging

A key question for the error flagging stage of the method is the issue of false-positive results. One main goal of algorithms using AMI data for algorithmic validation of distribution system models is to ensure that there are few false-positive results, as false positive results would be introducing novel errors into the model. It is difficult to determine a broad heuristic for the correlation coefficient threshold in the error flagging step. Our testing has shown that threshold is highly dependent on the measurement noise in the dataset and that will vary according to particular feeders, similar to the way that the MSE threshold was affected by the measurement noise in Figure 6-6. However, by ranking the flagged results according to the correlation coefficient, utilities can be assured that they are working on the transformers that are the likeliest to contain errors. Figure 6-7 shows the number of transformers flagged for two different simulations, one a baseline without any measurement noise (blue) and one with 0.1% maximum measurement noise (green). In both cases, the same 10% of customers were injected with incorrect transformer labels. The red, dashed line represents the true number of transformers that contain label errors and should be flagged. Marked with arrows is the range of thresholds for which all transformers with errors were flagged and no transformers were flagged incorrectly. In the noise case, the threshold range is shifted to the left and is from 0.51 to 0.56 and for the baseline case it is 0.67 to 0.83. Those ranges will be different depending on the noise level as well as which customers are labeled incorrectly. This figure omits customers that were labeled as being the only customer serviced by a transformer because the flagging methodology will not consider those customers.

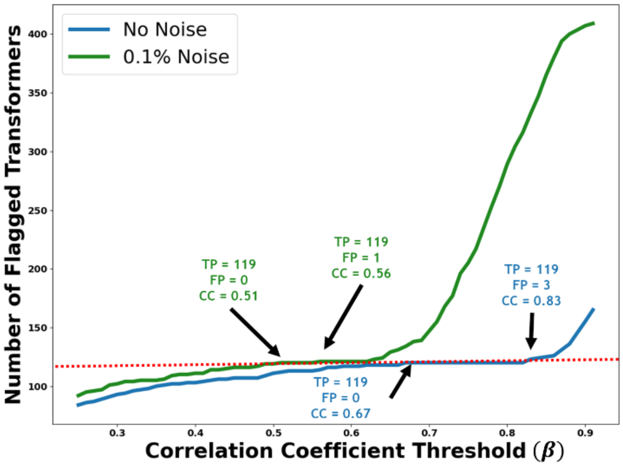


Figure 6-7. Ranked flagged transformers showing true positives (TP), false positives (FP), and correlation coefficients (CC) β for two different noise injections (green line and blue line). The dashed red line represents the actual number of transformers that should be flagged.

6.3.2. *Full Methodology Results*

The first set of simulations tested the algorithm's robustness by adding measurement noise and missing data to the dataset. Table 6-2 details these results. In both cases, 10% of customers were randomly given incorrect transformer labels. The correlation coefficient threshold, β, used was 0.7. In practice, the ranking methodology should be applied here. Which customers are mislabeled has an effect on the algorithm's accuracy. To better quantify the results, a 500 run Monte Carlo simulation was conducted, and the results were averaged. In the first simulation, measurement noise was added uniformly at random in the range +/- 0.1% for voltage, +/- 100W for active power, and +/- 100VAr for reactive power. In the second simulation, 0.1% missing data was added in addition to the measurement noise. The injected noise and missing data were kept consistent for each run. The customers with incorrect transformer labels were chosen randomly for each Monte Carlo simulation to isolate the effects of which customers were mislabeled. The noise case had an average accuracy of 99.5%. The second case with both noise and missing data shows a slight decrease in transformer accuracy over the noise-only case, about 2 additional incorrect transformers on average.

Table 6-2. Monte Carlo simulation testing different mislabeled customers in each run with a fixed measurement noise and missing data injection over 500 runs

Mislabeled Customers Changing with each Monte Carlo (500 runs)	Average Transformer Accuracy	Average Incorrect Transformers
Noise Added	99.5%	2.95
Noise & 0.1% Missing Data Added	99.1%	5.2

Next, a smaller Monte Carlo simulation composed of 10 runs was conducted where the measurement noise, missing data, and mislabeled customers were re-selected randomly for each run. This simulation tests the random effects induced by all three sources of randomness, and the results are shown in Table 6-3. The reason for decreasing the number of Monte Carlo simulations for this test case is due to the increased processing time necessary to calculate the pairwise regression for each new injection of noise and missing data. That resulted in an average accuracy of 99.4%. These simulations demonstrate the algorithm robustness against measurement noise and missing data.

Table 6-3. Monte Carlo simulation testing injected measurement noise, missing data, and mislabeled customers in each of the 10 runs

Noise/Missing Data/Mislabeled Customers Changing with each Monte Carlo (10 runs)	Average Transformer Accuracy	Average Incorrect Transformers
Noise & 0.1% Missing Data Added	99.4%	3.7

The next set of simulations demonstrates the accuracy of the algorithm as the percentage of mislabeled customers increases. This is shown in Figure 6-8. Even with very high levels of mislabeled customers, the accuracy remains high. At 50% of customers mislabeled, the average transformer accuracy is >97.5%. Recall that the transformer accuracy is the percentage of transformer groupings that are correct. The injection of incorrect labels is done as a percentage of customers rather than transformers, thus the reason the percentage of mislabeled customers on the x-axis does not precisely match the transformer accuracy shown by the red line.

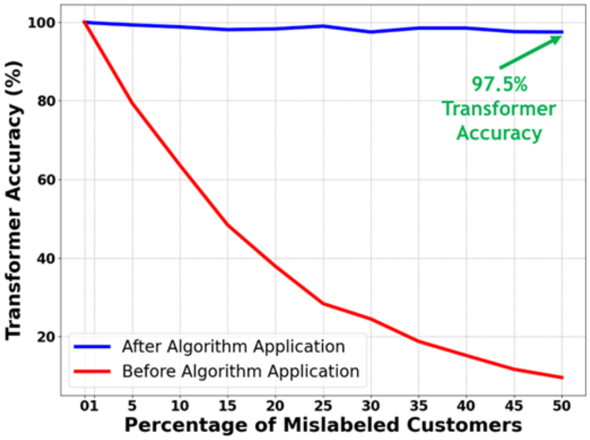


Figure 6-8. Transformer accuracy with different percentage of customers with incorrect transformer labels. The predicted transformer groupings are shown in blue and the original transformer groupings with injected errors are shown in red.

Note that as more customers are mislabeled, the algorithm results in larger numbers of transformer groupings that must be mapped back to a particular physical transformer. An interesting result is that there is little dependence on the number of incorrectly labeled customers on algorithm accuracy. The small number of incorrect customers are mainly due to the influence of customers labeled as being the only customer served by a transformer. Recall that the error flagging portion of the algorithm, Steps 1-3 in Figure 6-2, is unable to flag customers who are labeled as being the only customer serviced by a transformer. That method relies on the ability to compare correlation coefficients between customers labeled on the same transformers; thus, customers labeled by themselves are necessarily omitted. That type of error is passed through the algorithm without changes. Additionally, the majority of the errors shown in these results come from customers who were labeled as the only customer serviced by a transformer in the original labeling. As the overall percentage of errors increases, the number of that type of customer also increases. If a customer is erroneously labeled on a transformer with other customers but is in fact the only customer on a transformer, stage 2 of the algorithm can correct that error. It might appear that stage 2 of the algorithm could correctly account for customers served by a single customer transformer without omitting them, however in practice those customers have proven more difficult to deal with. Thus, in this work, they are omitted by the nature of the way the error flagging step works.

6.3.3. Comparison to Other Methods

These results were compared with two other algorithms for the same task. The linear regression methodology from [38] was implemented, as well as the correlation coefficient methodology from [32], [38].

The linear regression methodology from [38] is as follows. A pairwise OLS regression is calculated between all customers, similar to the regression described in Section 5.1.1. The pair of customers with the highest coefficient of determination (R^2) are considered to be serviced by the same transformer, and they are combined into a virtual metering point. Then the pairwise regression is done again between the new virtual metering point and all other customers. This process is repeated until all remaining R^2 values are below some threshold. Even with omniscient knowledge of the mislabeled customers, we were unable to find a threshold that produced good results on this dataset. For the noise case shown in

Table 6-2, this method had extremely low accuracy due to the virtual metering points not performing well in the regression. In many cases, the virtual metering points caused the predicted transformer labeling to become one large grouping.

The correlation coefficient method from [32] is similar to a second method proposed in [38]. In this method, the pairwise correlation coefficients are calculated as discussed in Section 6.2.1. The "window" method is used here as well to provide a better comparison with our proposed method, although that was not used in [32], [38]. The authors in [32] also add a distance-based filtering aspect to the method; that is also omitted here as that requires additional topology information that the proposed method assumes is unavailable or untrustworthy. Once the correlation coefficients have been calculated, any pairs of customers who have a higher correlation coefficient than a chosen threshold are considered to be on the same transformer. Again, choosing that threshold is quite difficult, and it is unclear how that threshold could be determined in practice. Using the omniscient knowledge of the ground-truth, we chose the best threshold to give a comparison to our proposed method. In the noise-added test case, the best results for the correlation coefficient only method were 97.6% transformer accuracy (14 incorrect transformers), compared to 99.5% transformer accuracy with the proposed method.

6.4. Utility Results for Utility #2

The utility dataset used in this work is described in Section 3.2.2. There are not ground truth labels for this dataset, thus the following example is shown as a proof-of-concept that the proposed method works given real data. The reactive power measurements for this dataset were determined to be unreliable, thus the results shown below rely on stage 1 of the methodology shown in Figure 6-2 and analysis of Google Earth imagery. In the absence of ground truth labels, publicly available Google Street View images can be used to validate certain algorithm predictions.

Figure 6-9 shows satellite imagery of two transformers and four customers, and the original model shows that all four customers are connected to the southern (bottom) transformer. However, this transformer was flagged by stage 1 of the algorithm, and inspection of Google Street View imagery confirms the configuration in Figure 6-10. Two customers are connected to the south transformer and two are connected to the north transformer. Table 6-4 shows the pairwise correlation coefficients for this set of four customers, and the two groupings of two can be clearly seen. Note that the correlation between Customer 1 and Customer 2 is only 0.77, demonstrating that real data can often contain factors that lower the correlation coefficients even between customers on the same transformer. The algorithm also correctly identified several other known transformer labeling issues on this feeder that had been previously identified in other work.

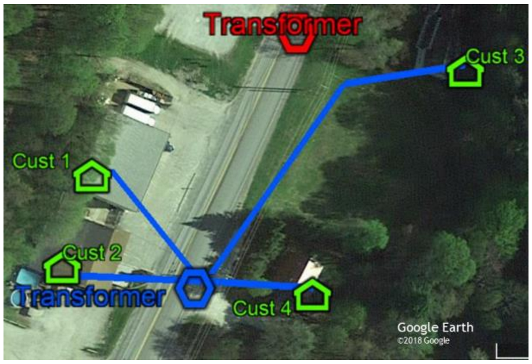


Figure 6-9. Original utility labeling for transformers on Utility #2, Feeder #3

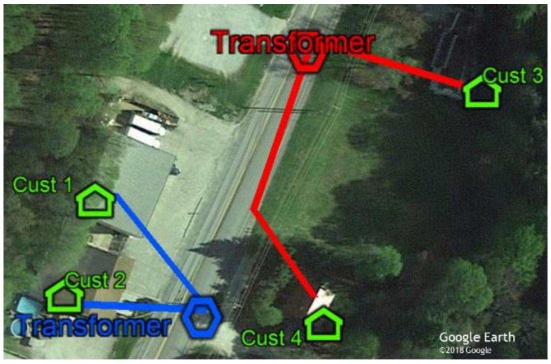


Figure 6-10. Actual labeling verified using Google Earth imagery for transformers on Utility #2, Feeder #3

Table 6-4. Correlation coefficients for the customers shown in Figure 6-9 and Figure 6-10

	Cust #1	Cust #2	Cust #3	Cust #4
Cust #1	1	0.777	0.434	0.575
Cust #2	0.777	1	0.344	0.446
Cust #3	0.434	0.344	1	0.958
Cust #4	0.575	0.446	0.957	1

6.5. Meter-to-transformer Pairing Summary

This work developed a two-stage algorithm based on correlation coefficient analysis and physics-based linear regression for the task of grouping customers by service transformer, and it leverages the increasing proliferation of AMI data. The two-stage method achieves >99% accuracy in the presence of measurement noise and missing data on a feeder with 1379 residential customers and 10% of those customers having incorrect transformer labels. The method also provides a clear improvement upon two other similar methods in literature. This work produced 2 conference paper publications [64], [65].

7. MEDIUM-VOLTAGE RECONFIGURATION DETECTION

7.1. Task Overview

Power Distribution Systems typically have a radial topology, but very often they are equipped with switching devices such as tie switches, sectionalizers, IntelliRupters, and circuit breakers that allow the system topology to be reconfigured or sections of the system to be connected to nearby feeders. For example, in Figure 7-1, we would like to know which direction the current is flowing on this street and what the states of the switches (triangles) are. This capability allows improvements in reliability and continuity of service through fault isolation and load transfer, in addition to allowing optimization of system losses, to name a few advantages. Knowing system topology is important for power systems operations and planning. For instance, errors in the knowledge of distribution system topology might lead to incorrect estimates of solar PV hosting capacity.

Due to frequent system reconfigurations and limited telemetering, the status of those switching devices may be unknown or uncertain. Furthermore, maintenance and outages might require operating a switch, and it is possible that reporting this action might suffer a delay or not happen at all. Errors in communication or telemetering temporary unavailability can lead to unaccounted system reconfiguration. In this chapter, we have proposed methods that leverage field data (e.g. SCADA, AMI) for detecting distribution system reconfiguration and identifying the new system topology by identifying the status of the switching devices.

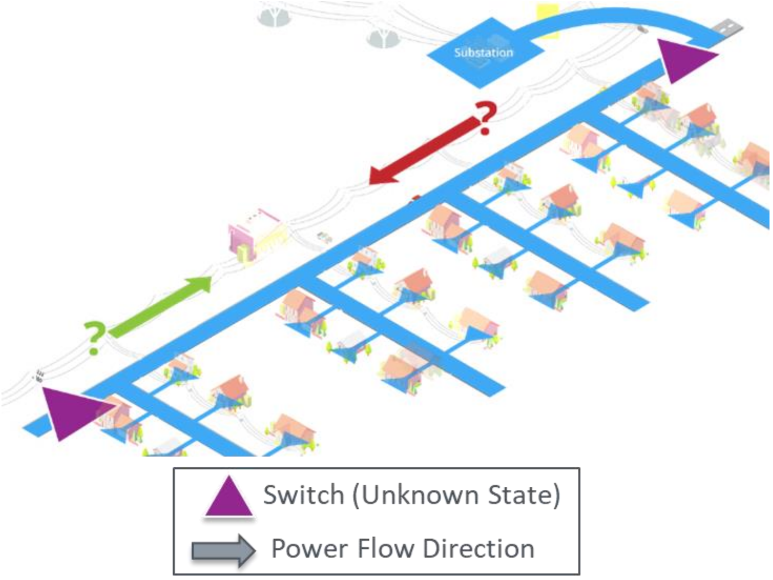


Figure 7-1. Topology Detection illustration

7.2. Medium-Voltage Topology Reconfiguration Detection and Identification

As a result of a collaboration with Texas Tech University, we have developed a method for Distribution System Topology Identification (DSTI) using time-series data from voltage measurements. The goal of this method is to determine which distribution grid switches are closed and which ones are open in real-time. We have compared two machine learning approaches to predict the topology of the distribution grid using two classifiers. The first is a one-versus-all Linear Discriminant Analysis (LDA) classifier and the second is a Regularized Diagonal Quadratic

Discriminant Analysis (RDQDA) classifier. Based on knowledge of the number of radial topology that a given system has and labeled voltage magnitude measurements ($\mathbf{V}_t$) from known time-stamps (t), the classifier is trained with trend vectors ($\boldsymbol{\delta}_t$), which are features extracted from the voltage measurement data streams that represent the change in voltage that occurred in between voltage measurement scans, so we have $\boldsymbol{\delta}_t = \mathbf{V}_t - \mathbf{V}_{t-1}$.

It is important to highlight that this method relies on some simplifying assumptions. First, we consider that the voltage time-series measurements are synchronized. We expect that small synchronization errors relative to the 15-minute sampling rate would not greatly harm the performance of the classifier as long as all voltage measurements are taken either before or after the topology transition occurred. We also assume that the topology of the system is always radial, and all customers are being served. Therefore, the scenario of interest in this paper is when a distribution grid reconfiguration has been successfully completed. We consider that if a power outage has been captured by a scan of voltage measurements (e.g., all meters downstream of a switch are not responding) then it becomes a power outage location problem, which is outside the scope.

To validate the performance of the DSTI algorithms, multiple simulations in OpenDSS using the IEEE 123-bus test feeder were run using the proposed approach. This system has 5 radial topologies, therefore there exist 20 possible topology transitions (see Figure 7-2). Additionally, there are 5 classes that represent unchanged topology between measurement scans. In total there are 25 possible switch transitions that represent the classes for the topology change classification problem. We consider that the classifiers are trained and run with data streams from 91 voltage measurement devices co-located with system loads.

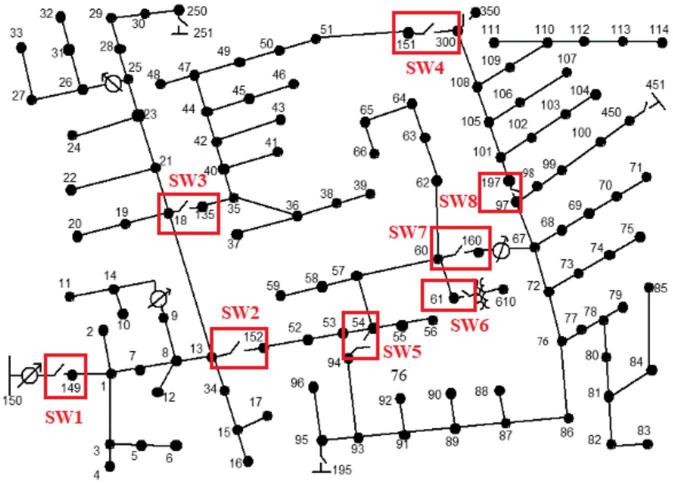


Figure 7-2. IEEE 123-bus test feeder showing its eight switches.

The sequential time simulation is yearlong to show the DSTI approaches robustness not only to load variations throughout a day but also seasonal load variations. For each time series, a random switching event is programmed to occur, and the voltage measurements collected are used to either train the algorithm or predict the current topology of the distribution system. The training of the one-versus-all LDA classifier was constructed using 3 months (January through March, 8,639 trend vectors) of voltage magnitude measurements collected at 15-minute intervals from the 91 nodes of the test system that contained a load. During the 3-month training, a total of 600 switching events occur randomly spanning the entire 25 possible switching events. The remaining 9 months of data were used for validation (26,399 trend vectors) during which 150 switching events occur randomly also spanning the 25 possible switching events.

The amount of noise added to the voltage measurements was determined based on the meter class. Two different classes of electric meters were established: no noise and class 3, which have an error of 0% and 1% of true value, respectively. To model this random error as a Gaussian, it was assumed that the standard deviation of the noise corresponds to a meter class of 3, so 99% of random noise realizations fall within the tolerance of the meters.

Three yearly trials were performed to average the effects of random noise. For each trial, 150 switch changes are randomly made throughout the 26,399 trend vectors over the 9 months validation period. In the case without noise, the classifier was able to correctly predict the topology transition for 100% of the time, which shows its robustness to load variations. When Gaussian noise is added, the classifiers can correctly predict the class of the topology transition more than 98% of the time in all three trials. The results of the classifiers with noisy data is summarized in Table 7-1.

Table 7-1. LDA and RDQDA simulation results with noise

Trial #	LDA – Noise modeled as error Class 3 Meter error				RDQDA – Noise modeled as error Class 3 Meter error			
	Correct	*Incorrect*	*% Correct*	*False Positive*	*Correct*	*Incorrect*	*% Correct*	*False Positive*
1	148	1	98.7	2	147	3	98.0	10
2	147	2	98.0	1	145	5	96.7	5
3	147	1	98.0	4	148	2	98.7	8

Another test was run to test the robustness of the classifiers to number of measurements. In this second test the number of meters was reduced from 91 to 30 and trained and tested following the same approach as before. In this case, the classification accuracy reduced to a minimum of 94%, as shown in Table 7-2. This DSTI method was summarized in the conference paper [66].

Table 7-2. LDA and RDQDA simulation results no noise and 30 voltage measurements

Trial #	LDA – No noise and 30 voltage measurements				RDQDA – No noise and 30 voltage measurements			
	Correct	*Incorrect*	*% Correct*	*False Positive*	*Correct*	*Incorrect*	*% Correct*	*False Positive*
1	144	6	96.0	8	147	3	98.0	10
2	143	7	95.3	15	145	5	96.7	5
3	141	9	94.0	9	148	2	98.7	8

8. REGULATOR AND CAPACITOR STATES AND SETTINGS

8.1. Task Overview

Voltage regulators and switching capacitors are widely used by utilities to regulate conditions on the grid. While the states and settings of these devices can often be queried and modified remotely, they are rarely recorded and stored over time. However, this information is valuable for a variety of grid modeling and analysis tasks (e.g., identifying misoperations or device failures, tracking total operations for maintenance concerns, calibrating models, etc.).

In this task, two different approaches were taken to recover the historical states and settings of these regulating devices. The first approach uses state estimation algorithms to determine the regulator and capacitor states; this approach was tested on two different open-source test circuits and different algorithms were tested based on the type and resolution of the available data. The second approach applies a variety of data-driven methods directly to the available measurements, leveraging a variety of techniques like optimization and machine learning; the effectiveness of this approach was quantified in several ways, including its utilization in a reactive power allocation algorithm.

Additionally, in this task, data-driven methods were developed to estimate the states and settings of voltage regulators and switching capacitors. Initial approaches were developed and applied within a reactive power allocation algorithm (Section 8.3.3) to improve customer reactive power modeling. The methods were then generalized for other applications, as detailed in Section 8.3.1 and Section 8.3.2.

8.2. Using State Estimation to Track Capacitor and Voltage Regulator States

One of the solutions for parameter error detection proposed in this report is the utilization of the generalized DSSE introduced in 4.4. This method includes variables such as capacitor bank statuses and voltage regulator tap positions as state variables in addition to traditional nodal voltage phasors. Bad data detection features χ^2 for error detection and the largest normalized error method for error identification. The states are considered static measurements that are updated once an error is detected and the bad data processing method identifies that one of the parameters might be incorrect. The DSSE then investigates if a transition has occurred and updates the discrete variable if a new suitable discrete value is found.

With this setup, it is possible to leverage OpenDSS Quasi-Static Time-Series (QSTS) capabilities and simulate a series of time-series data obtained from a distribution system that has voltage controls and time-varying loads. We have assumed that the error obtained by aggregating AMI real and reactive power flow measurements in the secondary sides of distribution system transformers is small. Therefore, it is possible to infer the voltages in the primary sides of those transformers given voltage magnitude measurements on their secondary windings from AMI data.

This approach was tested using a modified version of the IEEE 13-bus test system with load shapes and capacitor bank regulators, 110 measurements, and 74 state variables, which gives a global redundancy index of 1.487. Out of those 110 measurements, only 75 come from meters (20 real power injections, 20 reactive power injections, and 35 voltage magnitudes). The remaining are pseudo-measurements (15 real power injections, 15 reactive power injections, 2 capacitor bank statuses, and 3 transformer tap positions). No line power flows were considered, even though those might be available in real distribution systems from reclosers, sectionalizers, and other devices.

8.2.1. DSSE with capacitor and transformer tap correction

Detecting a change in transformer taps is very challenging when voltage measurements from AMI are available. Tap changers typically operate by increasing or decreasing one tap position, thus changing their transformation ratio by 0.625%. This change is close to the precision class of smart meters. Therefore, using any static state estimation method in this scenario will hardly ever detect a single tap change with confidence.

To test if this method would work with higher precision measurements, we have considered that the precision of voltage magnitude measurements is equal to 0.1%, while 1% was considered for other measurements (capacitor status, tap positions, and real and reactive power injections). These values were used both to express the size of Gaussian noise added to each measurement and to determine the weights of the WLS. By doing so, we have run a week-worth of state estimations (673) with new measurements every 15 minutes. We have observed that the state estimator was able to track tap position and capacitor bank statuses very accurately. Among the 8 tap changer transitions recorded, the DSSE was able to detect 7 of them in the first time step after they occurred and the remaining tap transition was detected and updated with a delay of 1 time step. Capacitor bank statuses changed 19 times in the simulation, and in all cases, the DSSE was able to capture the transitions when they happened. There was only one false positive for capacitor bank transition, which was then corrected in the following time step. See Figure 8-1 and Figure 8-2 for tap changes and capacitor banks switches respectively.

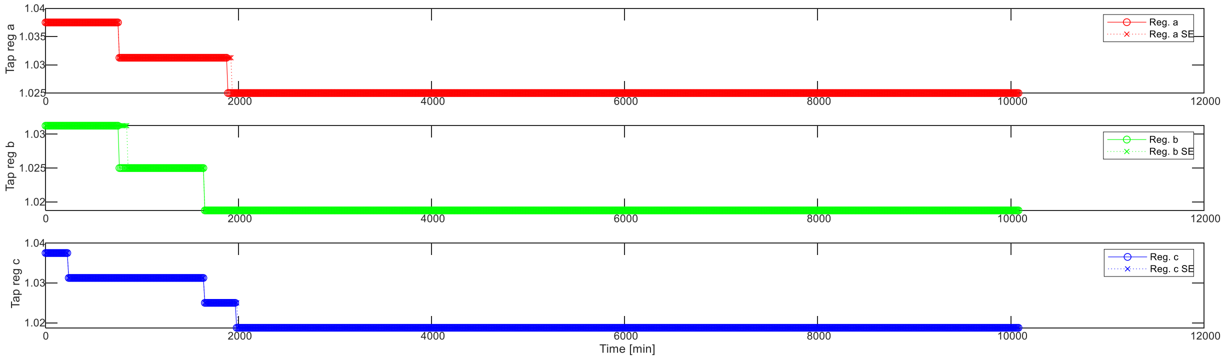


Figure 8-1. Tap changer tracking over 7 days

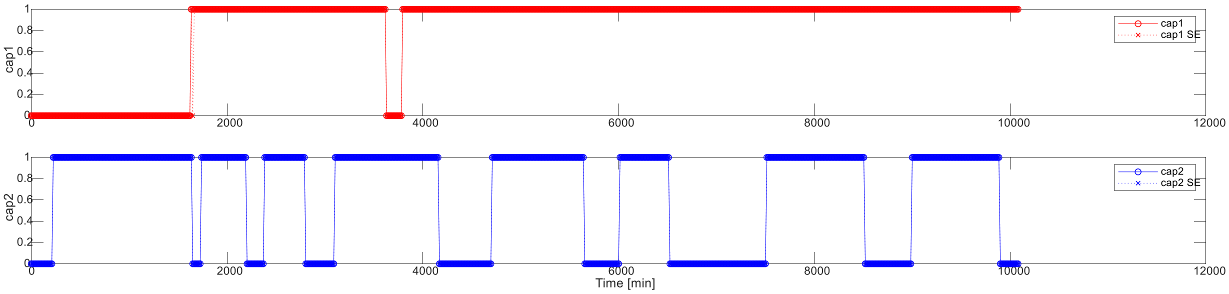


Figure 8-2. Capacitor bank status tracking over 7 days

Another criterion for determining if the DSSE has been successful was the error in voltage phasor norm when compared to OpenDSS's result. A successful state estimation is considered if the norm of the error of voltage phasors is less than 0.1%. The average L-2 norm of the voltage error was 0.0289%, the maximum error was 0.28% and the DSSE obtained precise results in 98.663% of the cases. The DSSE obtained a solution, i.e. converged, in 99.85% of cases.

Tests with AMI voltage magnitude errors of up to 0.5%show that the DSSE is still capable of detecting the correct tap positions after a few time-steps, but not immediately following the transition.

8.2.2. Validation in Larger Test Systems

The developed DSSE was applied for large systems such as EPRI's circuit 5 (3,440 nodes). The software can run in a few seconds with voltage estimation errors in the order of 0.1%, see Figure 8-3.

To approximate our simulation scenario from a real-life case, we have considered that the DSSE has very limited access to measurement data. In this initial analysis, we consider that only (1) substation voltage magnitude and power measurements are available, and (2) AMI voltage magnitude and power measurements are available for all loads.

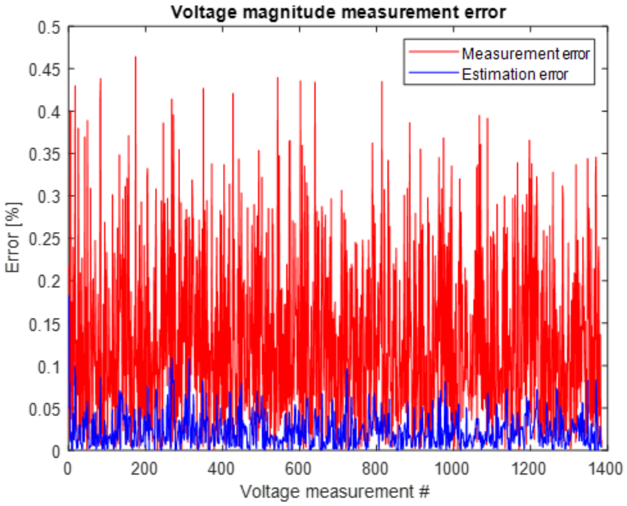


Figure 8-3. Reduction of voltage magnitude error with DSSE

After system reduction, we are left with 5,892 state variables (5,885 real and imaginary parts of complex nodal voltages, 4 capacitor statuses, 3 tap positions[8]) and 7,275 measurements (0 power flows, 1,385 real power injections, 1,558 pseudo-measurements[9], 1,385 reactive power injections, 1,385 voltage magnitudes, 4 capacitor statuses, 3 tap positions), which gives a global redundancy index (i.e., the ratio between the number of measurements and state variables) of 1.23. This means that there are 1,558 nodes that are not directly measured but whose *voltage magnitude and angle* can be estimated with the DSSE.

To assess the impact of different meter classes on the accuracy of DSSE results, we have performed a parametric study in which we have tested meter classes from 0.1 to 2. Because the tests

[8] We have added 3 single-phase voltage regulators in the substation and 4 capacitor controllers to our OpenDSS model of EPRI's Circuit 5 feeder so we could test our DSSE with parameter test.

[9] These are simply zero-injection real and reactive power equations added based on the knowledge we have from the system.

are very sensitive to random noise realizations, we have repeated the test 5 times for each case. The results are shown in Figure 8-4.

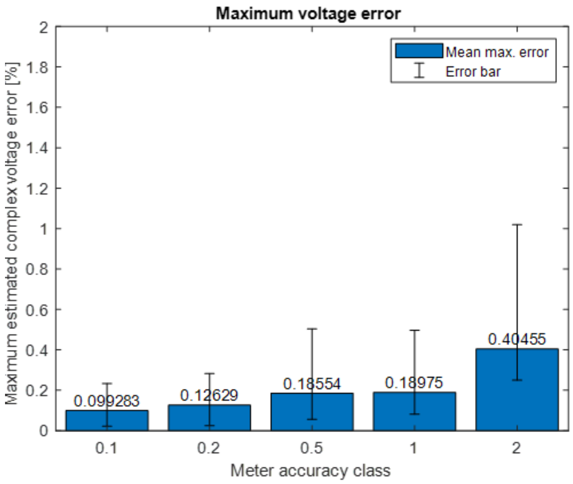


Figure 8-4. Maximum complex voltage estimation error obtained with DSSE for different meter classes

We can see that for lower accuracy meters (class 1 and 2), DSSE is capable of estimating very accurately the complex voltages for all 2,943 nodes of the reduced system, even though only 1,385 of them are covered by a meter. For higher accuracy meters (classes 0.1 and 0.2) we can see that the lack of redundancy of DSSE starts to become a limiting factor for improving complex voltage estimation accuracy.

To assess the impact of increased redundancy for the state estimator, we have run experiments where we would add more measurements in the distribution grid. Those were performed by assuming it would be possible to obtain information from automation equipment installed in the grid such as relays and IntelliRupters. Those would be capable of measuring voltage magnitudes and line real and reactive power flows where they were installed (3 to 9 additional measurements). In this experiment, we have tested what would be the effect of adding between 1 (GRL of 1.23) and 2,000 (GRL of 1.9) of those meters. All meters were assumed to be Class 0.5. The maximum complex voltage estimation error decreased from an average of 0.18% (only substation and AMI meters) to 0.1% (case with 2,000 sensors) while the RMSE of complex voltage error was reduced from 0.06% to 0.017%. Therefore, we can conclude that there is little advantage of adding many sensors in terms of improving voltage estimation accuracy. The effects of redundancy in bad data processing are likely to be significant but they have not yet been studied.

8.2.3. Leveraging Micro-PMUs for Capacitor and Tap Changer Status Detection

This project has also partially funded research on the utilization of micro-PMUs to detect changes in LTC positions and capacitor bank switching [67]. To simplify the problem of simultaneous real-time monitoring of measurements and parameters we have proposed a framework that employs high-granularity micro-PMU measurements to obtain temporal decoupling of error analysis of measurements and parameters.

When a recent measurement scan from SCADA and smart meters is available, gross error analysis of measurements is performed as a post-processing step of nonlinear DSSE using non-linear least square estimator (NLSE). In between scans of SCADA and AMI measurements, which are considered to take 15 minutes, a linear state estimator (LSE) using micro-PMU measurements and linearized SCADA and AMI measurements is used to detect parameter data changes caused by the operation of voltage control devices. The micro-PMU data was assumed to be obtained 120 times per second, but the data was down sampled to run the least square estimator (LSE) every 100 ms. We have implemented this simulation in the IEEE 13-bus test system to validate the efficacy of this method. We have assumed that the DSSE does not have access to measurements of LTC position nor capacitor bank statuses. In addition to that, the distribution system has two micro-PMUs: one is located in the substation and the other is in the middle of the main feeder. These micro-PMUs can measure both nodal voltages and power flows.

For every execution of the LSE, the variance of the unsynchronized measurements is updated according to the uncertainty introduced by load dynamics into the micro-PMU data, which are modeled as an Ornstein–Uhlenbeck (OU) random process, as shown in Figure 8-5.

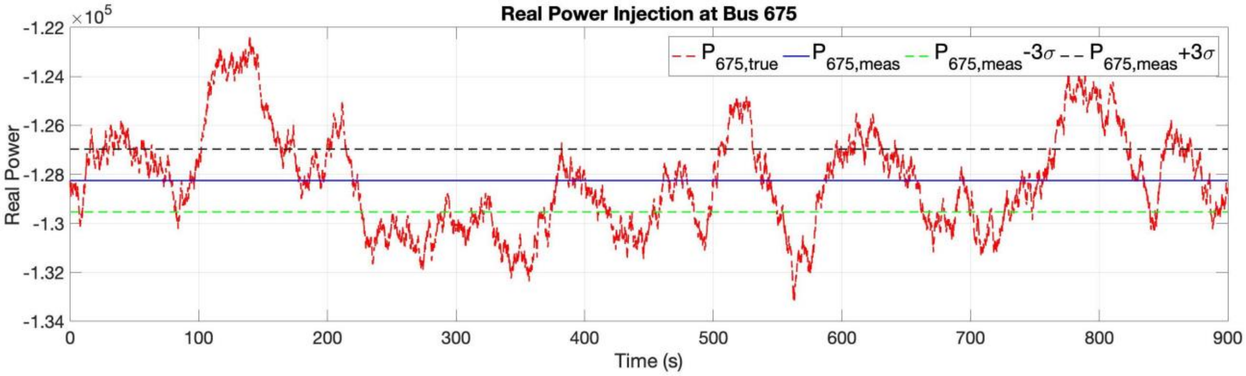


Figure 8-5.Time-series of a variable driven by an OU process versus its measurement at t=0.

Considering this model, we can see that SCADA and AMI data are unlikely to capture variations introduced by loads and the measurements become obsolete as time progresses. The proposed framework includes an uncertainty accumulation procedure that models the expected decrease in measurement trustworthiness as time passes and measurements become more and more outdated. This is implemented by increasing the variance of each SCADA/AMI measurement over time as a function of the estimated load variation parameters modeled as the OU process. The update of variance of unsynchronized measurements can avoid the wrong detection of errors and can model the trustworthiness of outdated or obsolete data. Test results highlight that the LSE and NLSE processes successfully work together to analyze bad data for both measurements and parameters.

With this load model and temporal uncertainty propagation, we can obtain a reliable model for the *detection* of parameter changes, but it is still necessary to obtain a method for parameter change *identification*. When an error is detected by the chi-squared test, it is assumed that any gross errors in measurements were already adequately processed by the NLSE, therefore, the causes of parameter errors are assumed to be capacitor banks and LTCs. Because of the discrete nature of LTC turns ratio and capacitor bank statuses, a hypothesis test-based approach was used to verify if changes in parameters provide a better fit to the data. If the reactive power flowing into the feeder has changed in a value that is close to the value of the capacitor bank, then the capacitor bank operation is likely to have caused the change in voltage. This leads to the capacitor bank parameter change hypothesis. Otherwise, if no large reactive power changes are measured by the micro-PMU in the substation, the micro-PMU downstream of the LTC is used to detect if the voltage has increased or decreased, which will create the hypothesis for tap change. A likelihood ratio test is then used to verify the correctness of the identification of the parameter error. To obtain the likelihood of the alternative hypothesis, i.e., that one parameter has changed, a new state estimation is performed with the alternative parameters of the system and then, with the result of this new DSSE, the likelihood function is calculated.

To test the framework's effectiveness, a sequence of volt/VAr control devices operations occurs over 15 minutes (900 seconds), as shown in the bottom plot of Figure 8-6. In the top plot of the same figure, we can see the chi-squared score of the DSSE with parameter correction (blue line) and without it (orange line). In all cases, when there is a change in the parameters of the system, the value of χ^2 becomes significantly larger than the detection threshold. When the parameter correction is performed, the goodness of fit scores is always smaller than the detection threshold.

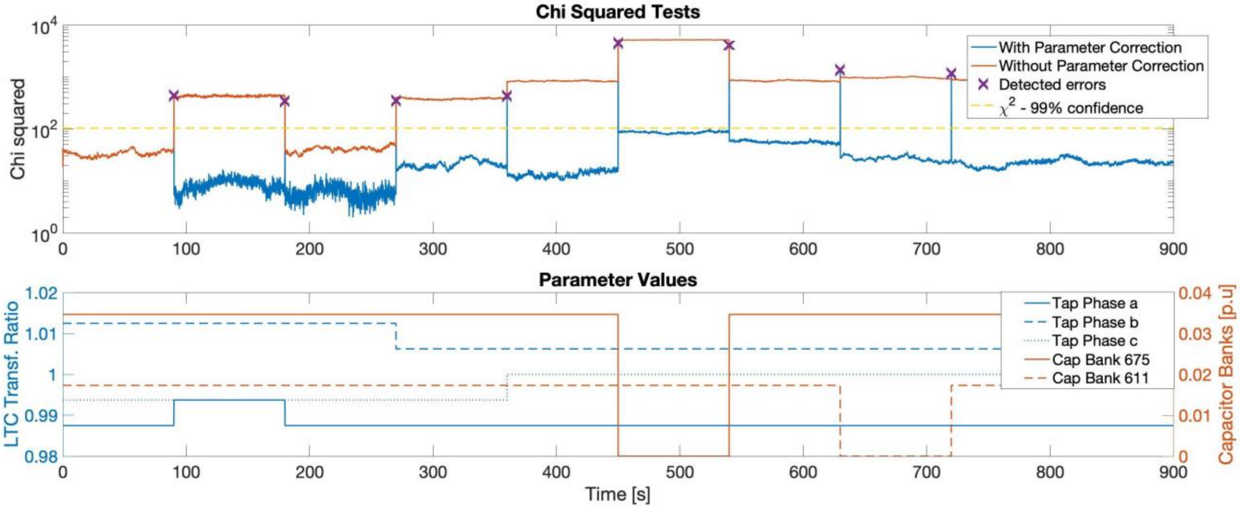


Figure 8-6. Time-series of DSSE during LSE phase tracking capacitor and LTC changes.

This method can allow the DSSE method to decouple the bad data processing of errors in measurements and parameters through the use of separate NLSE and LSE. With these innovations working together, the proposed DSSE can leverage micro-PMUs to enhance both measurement and parameter error detection on distribution grids.

8.2.4. Conclusion and Lessons Learned

The DSSE-based methods proposed in this project have shown potential for application in the detection and identification of errors in parameters of voltage control devices in distribution systems. The methods were validated using data from simulations in OpenDSS and their focus was to track changes of the tap position of voltage regulators as well as determine if a given capacitor bank was switched on or off at a given moment. The tests considered imperfect conditions like noise in measurements, errors in phase connection of transformers (see Section 4.4), and load variations. Numerical challenges were constantly found during the development of the state estimation algorithm due to details in some transformer models within OpenDSS and the poor matrix condition of the Jacobian and Gain matrices used in the solution of the WLS-SE. These numerical issues were rooted in system admittances with very distinct magnitudes that came from the distribution system models, which led to floating-point calculation errors. These were solved totally or partially with the use of system reduction algorithms [68] and the use of the matrix algebra algorithms better suited for the solution of sparse and ill-conditioned linear algebra problems. The solution of every problem was hard to diagnose and took a very long time to debug.

At the same time, many other potential sources of error were not considered, including issues in synchronization of smart meter and SCADA data, other sources of topology and parameter errors, errors in transformer models and parameters, delays in telemetering system, and very low measurement redundancy. When using field data made available to the project, the level of redundancy was found to be extremely low, barely allowing the system to be observable. Additionally, the combination of the aforementioned challenges with unknown system parameters in the secondary of some unusual transformer connections, lack of phase measurements in some three-phase loads, errors in the system model parameter database, lack of measurement availability and potential measurement errors, and potential parameter errors (e.g. transformer off-load tap changer positions) have resulted in an unsuccessful application of DSSE to the real data. This result is not surprising as the distribution system was not designed to run a DSSE and some potentially incorrect system models could not be re-verified. These factors highlight the importance of measurement system design and measurement redundancy in the success of the application of DSSE.

8.3. Using Data-Driven Methods to Track Capacitor and Voltage Regulator States

Estimating the historical tap position states has many practical applications, such as in distribution state estimation, identifying interactions with other grid-connected devices [69], [70] or misoperations of controllable devices, or tracking total operations that inform maintenance decisions [71]. Existing approaches include using downstream measurements to estimate tap positions of open-delta regulators [72] or using a voltage sensitivity-based approach to estimate excessive tap changes [71], but less work has been presented on generalized methods using low-resolution measurements, which we address in Section 8.3.1.

The estimation of historical switching capacitor states has practical applications similar to those of the regulator tap position estimation (e.g., distribution state estimation, identifying unintended interactions or misoperations of controllable devices, or tracking total operations that inform maintenance decisions). Existing approaches for characterizing capacitor banks from measurement data include applying a backward Kalman filter to substation measurements [73], identifying their location from power quality data [74], and using the slopes of current and voltage transient waveforms around the switching events [75], but less work has been published on methods to determine their states from low-resolution measurements. This knowledge gap is addressed by the proposed methods in Section 8.3.2.

8.3.1. Regulator Tap Position Estimation

To estimate the tap position of a voltage regulator directly using data-driven methods, measurements from the nearest upstream and downstream devices are required to assess the voltage difference across the regulator. For example, in the dataset from Utility #1 for Feeder #1, IntelliRupter measurements were available close to each side of the line voltage regulator. One of the interesting challenges using these measurements was that there were several measurement types available for each IntelliRupter, including instantaneous values and 15-minute averaged values. The plots below show the difference between the voltage measurements of the downstream and upstream IntelliRupters for each of these two measurement types.

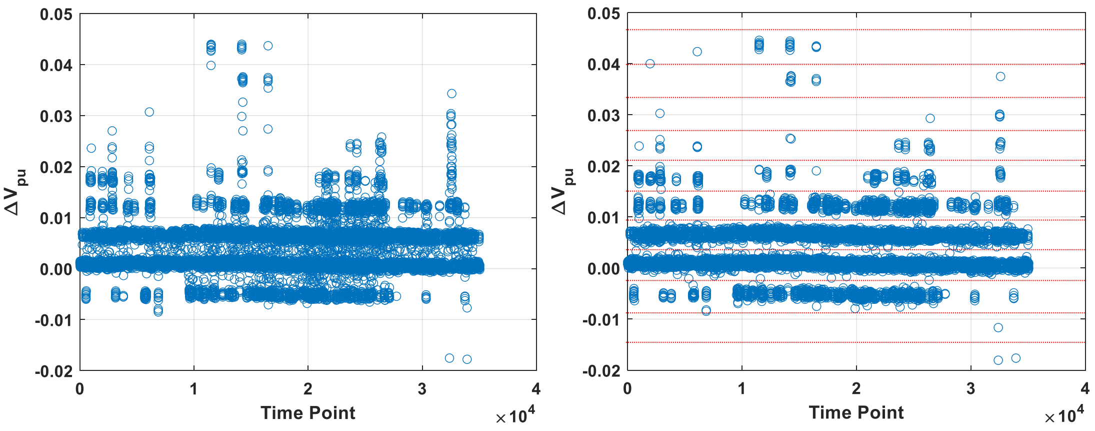


Figure 8-7. Voltage difference across a voltage regulator using averaged (left) and instantaneous (right) IntelliRupter measurements

In both plots, we can see that 1) the various tap positions manifest as horizontal planes, and 2) there appears to be less "noise" around these planes when using the instantaneous measurements. This result is in line with expectations—if the regulator changed its tap position in the middle of the 15-minute measurement window, the average value reported will appear to be between two adjacent tap positions. Because this phenomenon is less of a concern with the instantaneous measurements and the horizontal tap position planes were generally consistent between the two measurement types, threshold values were selected from the instantaneous measurement data for determining regulator tap positions (depicted as red dotted lines on the instantaneous data in Figure 8-7). These thresholds were then applied to categorize each data point by tap position.

However, instead of manually separating the data into different tap position bins, an optimization approach was proposed and implemented that would classify the tap position for each data point and return an array of all tap positions. Because the objective was to locate the centers of the data clusters, the proposed approach attempts to maximize the cosine similarity of the data clusters, subject to some constraints, or represented as:

Objective:

Maximize

$$\sum_{t=1}^{T} \max\left(cosine\ similarity(x_t)\right)$$

Constraints:

$$\textbf{\textit{For, }} \forall \boldsymbol{j} \in \boldsymbol{J}$$

$$w = c_j - c_{j-1} \quad \text{when } j > 1 \qquad (1)$$

$$0.25 \leq w \leq 2 \qquad (2)$$

$$-32 \leq c_j \leq 32 \qquad (3)$$

Where $x_t$is the voltage differences at each timestamp *t*, T is the set of all the time points, c denotes the set of tap positions, J is the maximum number of tap settings possible for a regulator, and *w* is the width between two consecutive tap positions. Usually, regulator tap positions are evenly spaced, so the width between each tap is assumed constant, as captured by constraint (1). Constraint (2) limits the value of the width, as the width between taps is usually around 0.75 V, and (3) provides the total number of positive and negative tap positions expected. While this optimization problem can be solved using many different approaches, a grid-search method was applied here, and the results are presented in Figure 8-8. A visual inspection of the results reveals that the method was able to successfully identify the centers of the data clusters (i.e., the regulator tap positions).

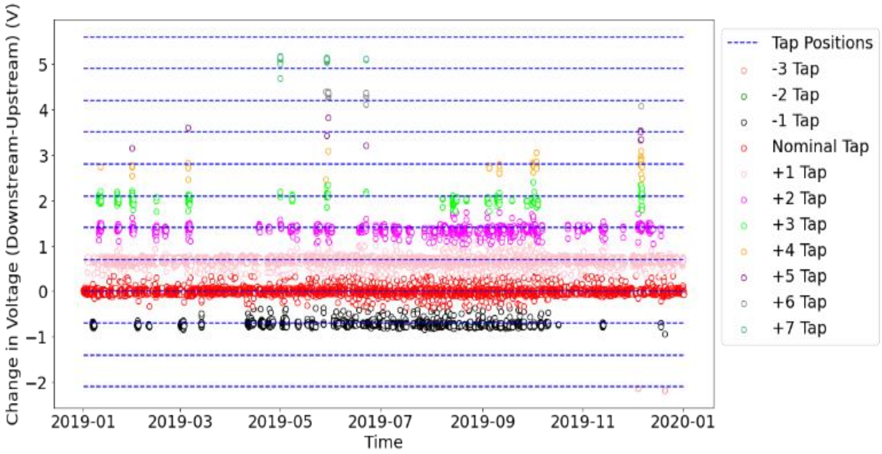


Figure 8-8. Tap position estimation results from optimization approach

8.3.2. *Switching Capacitor State Estimation*

A process similar to the one used in Figure 8-7 was applied to reactive power measurements to separate and categorize the data points by switching capacitor states. The dataset from Utility #1 for Feeder #1 had two switching capacitors, so we know that there are 4 different states available (both on, both off, and each one on by itself). The plots below in Figure 8-9 show the averaged and instantaneous reactive power measurements, and a visual inspection of these plots reveals that the data generally fall within 4 different horizontal clusters, representing the 4 available combinations of capacitor states. Therefore, we were able to manually select threshold values for each state combination from the instantaneous measurements (shown as red dotted lines in Figure 8-9), then apply those thresholds to generate the time-series profiles of states for each switching capacitor.

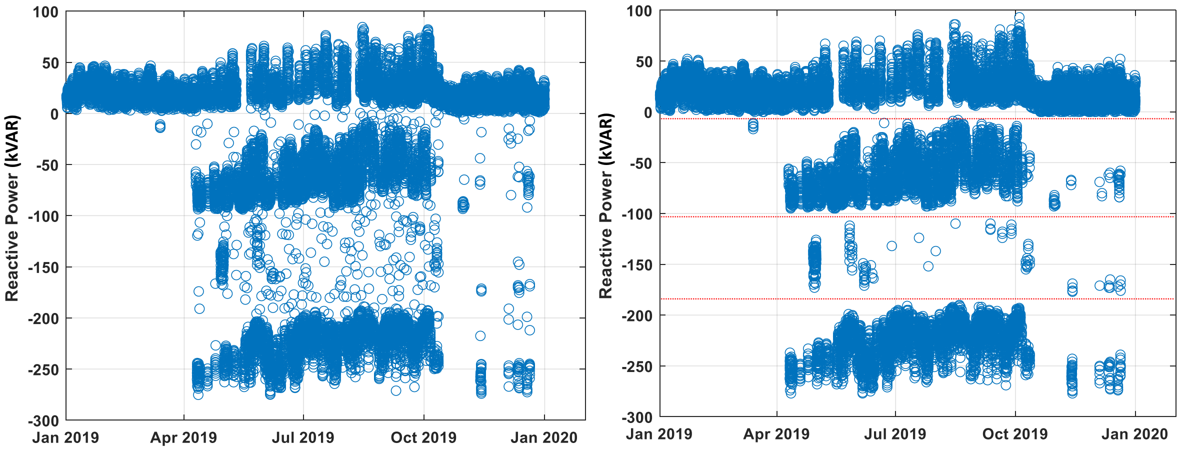


Figure 8-9. IntelliRupter reactive power measurements, averaged (left) and instantaneous (right)

The methodology for estimating capacitor states in Figure 8-9 utilized an empirical approach in which the reactive power measurement time series were visually clustered by separating the data with horizontal lines. This method worked well for that particular dataset, but there is no guarantee that horizontal lines could always be used, as this approach assumes that the discrete capacitor states do not result in any overlapping data points.

A more generalized approach was applied that incorporated real power measurements as well and represented the underlying data as a scatter plot of (P, Q) points, as proposed in [76]. This representation removes the time-series component but allows the capacitor states to show up as various planes of clustered data. The k-means clustering approach was utilized to identify the capacitor state clusters in Figure 8-10 after applying min-max scaling to the raw data and performing a principal component analysis (PCA) to reduce the dimensionality of the data. A visual inspection of the results reveals that the method was able to successfully identify the clusters, which can then be translated back to capacitor states.

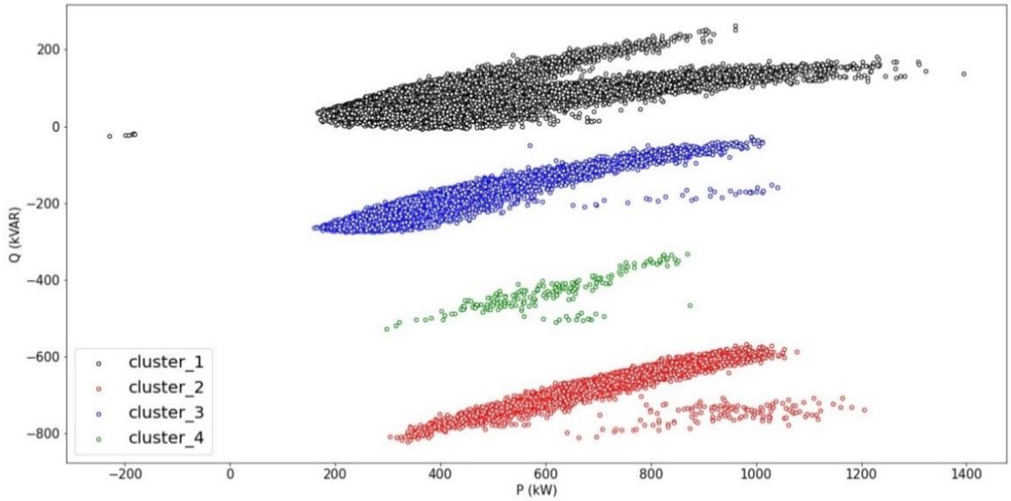


Figure 8-10. Switching capacitor state estimation results from the machine learning approach

8.3.3. *Reactive Power Allocation*

To test the effectiveness of the methods for estimating the regulation device states, the outputs were utilized in a reactive power allocation algorithm applied on Feeder #1 from Utility #1. Modeling customer reactive power consumption is often overlooked or oversimplified by assigning a constant power factor to customers based on feeder-head measurements, which results in an underrepresentation of daily and seasonal changes in customer loads. Because customer real and reactive power consumptions impact the voltages throughout a feeder [77], improving customer load modeling would also improve the spatial and temporal calibration of a variety of circuit metrics (e.g., evaluating losses in the network). While many energy meters being installed today can measure reactive power consumption, that feature is often not utilized. Historically, data from energy meters were only used for billing purposes and only considered real power consumption, meaning the existing systems may not be well-suited to handle the additional data streams or have the capacity to store enough data for modeling purposes. However, there are a variety of new devices being installed on distribution grids that can measure reactive power flows and can interface more easily with existing utility practices. These devices present an opportunity to apply existing power allocation methodologies to allocate reactive power measurements taken throughout a feeder to corresponding customer locations.

The approach developed in this project applies power flow analysis to generate unique reactive power profiles for each customer on a feeder using a combination of real power AMI data and reactive power measurements from multiple grid locations. First, the feeder is separated into various “zones” for which reactive power measurements are available. For each time point, starting with the zones furthest downstream from the substation and working backward, reactive power measurements are allocated to the customer locations in each zone according to their share of real power consumptions. Then, a power flow analysis is performed iteratively to adjust for the effects of network losses. After each zone converges on a solution, the algorithm advances to the next time point and starts the process over again until every customer location has been allocated a reactive power consumption for all time points.

This approach was tested on Feeder #1 from Utility #1 (shown in Figure 8-11), for which real power AMI measurements were available for all customers, and both voltage and complex power measurements were available from 10 IntelliRupters throughout the circuit. The objective was for the reactive power allocation algorithm to generate semi-synthetic reactive power profiles for all customers.

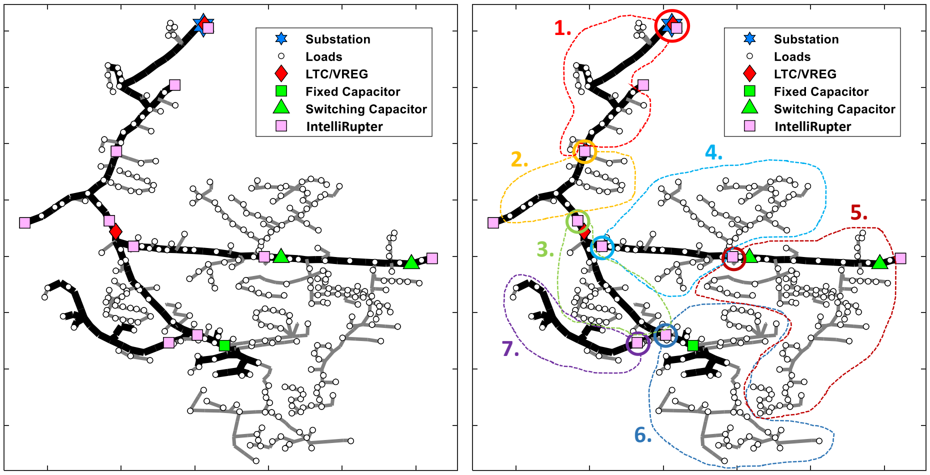


Figure 8-11. Circuit Plot of Utility #1, Feeder #1 with and without the customer zones identified

Because the reactive power allocation methodology was going to be applied at each time point of the year, we first had to ensure that the regulating devices in the circuit were calibrated. That is, the tap position of the voltage regulator and states of the two switching capacitors in the circuit had to be determined before the reactive power allocation steps because all these regulating devices also influence circuit voltages and power losses. Therefore, the results from the estimation methods shown in Figure 8-7 and Figure 8-9 were utilized here.

After estimating the regulator tap positions and switching capacitor states, all regulating devices could be controlled and set manually before the reactive power allocation for each time point. Once the regulating devices were set, the reactive power was allocated to the loads in each zone using an iterative process until the simulated reactive power values at the zone heads in OpenDSS matched the measured values from the IntelliRupters. This process was repeated for each zone, starting with the zones furthest downstream and working backward toward the substation. The algorithm steps can be summarized as:

1. Compile the circuit model in OpenDSS
2. Add monitors to the circuit corresponding with the IntelliRupter locations
3. Set the states of all regulating devices
4. Set the real power consumption of all the loads based on their AMI data
5. Solve the initial power flow
6. Calculate the per-phase PF of the IntelliRupter at the head of the zone

7. Calculate the per-phase kVAR required at the corresponding monitor location in the circuit model that matches the measured per-phase PF values
8. Allocate reactive power to any 3-phase loads in the zone based on their share of the total real power
9. Allocate the remaining reactive power to the 1-phase loads in the zone, subject to reasonable PF limits (0.75 lagging to 0.95 leading)
10. Solve the power flow and check for convergence
11. Repeat Steps 7-10 until the zone has converged
12. Move on to the next zone and repeat Steps 6-11 until all zones have converged
13. Move on to the next time point and repeat Steps 3-12 until reactive power has been allocated to all loads for all time points

Overall, the procedure outlined above resulted in unique reactive power time-series profiles for all loads and represented different power factor values for each phase of each zone and sometimes a different power factor for the 3-phase loads in each zone as well.

After applying the new load profiles to the customers in the feeder, the accuracy of the proposed methodology was then tested by running a yearlong QSTS simulation. Then, the results were analyzed using established methods to visualize the output time-series data [78]. Compared to the actual IntelliRupter power measurements, the results from the allocation procedure matched pretty well. For Zone 5, which is downstream of the voltage regulator and contains the two switching capacitor banks, the reactive power allocation results on all phases were within ±5 kVAR for nearly 95% of the time points, as shown in Figure 8-12.

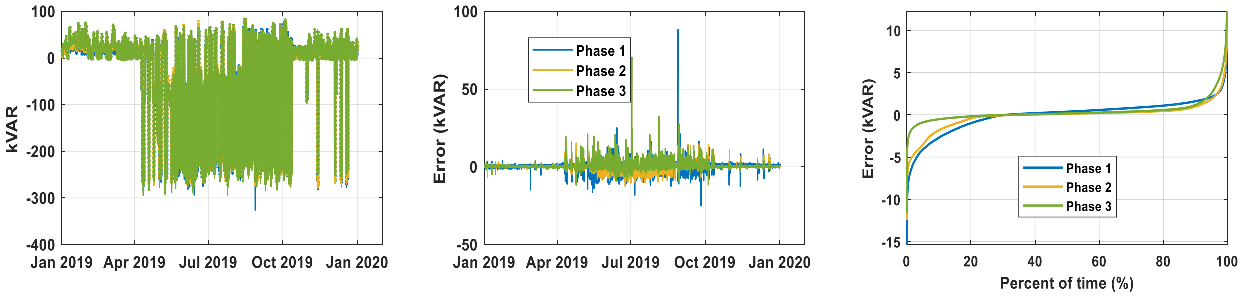


Figure 8-12. Comparison of simulation results and IntelliRupter measurements for Zone 5.

From the QSTS simulation results, the simulated customer voltages could also be compared to the actual measured voltages from the available AMI data. The results from this analysis are presented in Figure 8-13, which revealed that the average voltage errors for each customer were small (RMSE < 0.02 p.u.) but skewed positive, meaning that on average the simulated voltages were higher than the measured values. This positive skew can be explained, in part, by the lack of secondary modeling present in the OpenDSS circuit model. In other words, the voltage drops across the secondaries are not being captured in the circuit model, resulting in slightly higher voltages compared to the measured values.

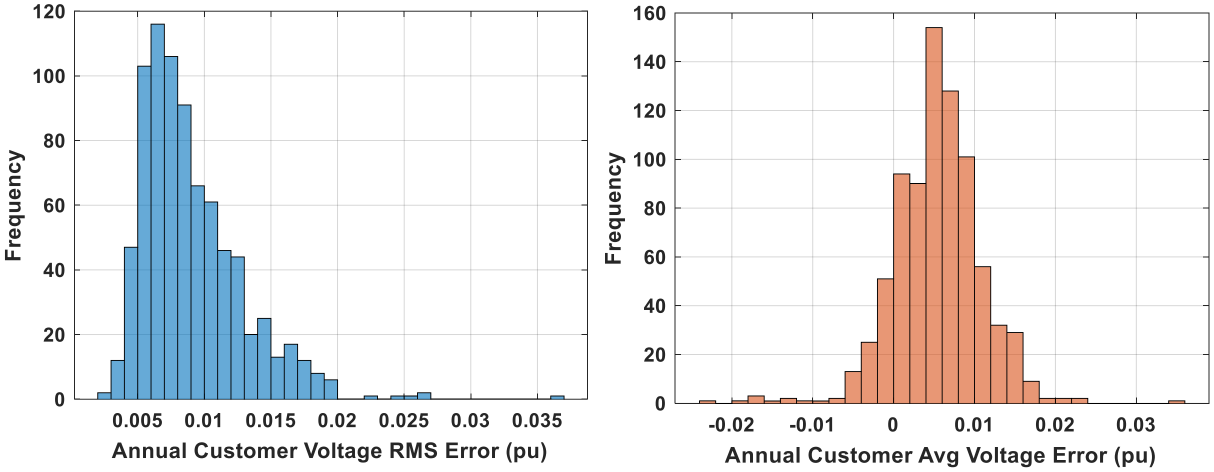


Figure 8-13. Comparison of simulated customer voltages and AMI measurements

8.4. Summary

Accurate estimations of regulation devices', like switching capacitors' and voltage regulators', settings have implications for a variety of grid modeling and analysis tasks. The work in this task provided several ways to recover this valuable information from various measurement sources available on modern distribution grids. The proposed methods included applying distribution system state estimation algorithms to extract regulating device states and data-driven methods that could be directly applied to measurement data with limited prior knowledge of the underlying system. The approaches were tested on validated open-source feeder models and also on an actual utility feeder model with corresponding real-world measurements. Overall, the work presented in this task provides a variety of options to extract actionable information on voltage regulators and switching capacitors. The proposed solutions are suitable to different measurement types and resolutions (both temporal and spatial), making them useful tools for distribution system analyses and improving distribution model fidelity.

9. PV SYSTEM DETECTION, PARAMETERS, AND SETTINGS

9.1. Task Overview

There is an ever-growing number of solar photovoltaic (PV) installations in the United States and worldwide. As the cost of these installations decrease [79], the scope of solar PV impacts on distribution grids is poised to continue to expand. Notably, the impact of behind-the-meter (BTM) solar PV on distribution grids is highly parametric in nature. For example, it is difficult for utilities to know the locations of privately interconnected BTM PV in distribution networks, which may change rapidly in regions with permissive net metering regulations. These interconnections may cause significant changes in the topology of a distribution network, posing a risk to distribution model fidelity. Furthermore, as shown in Figure 9-1, it is harder still to know the size, tilt, and azimuth of these PVs. These parameters of the PV installations can lead to significant implications for security constraint violations, and it is important for utilities to have knowledge of these parameters.

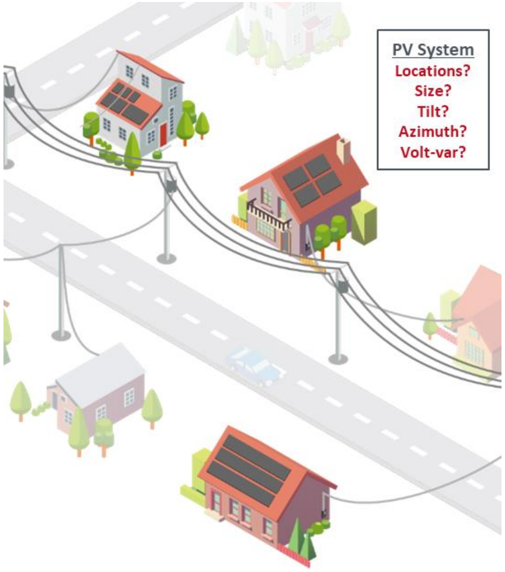


Figure 9-1. PV System Identification illustration

Moreover, the IEEE 1547-2018 standard on smart inverters has specified reactive power priority [80]. A wide variety of reactive power control modes for advanced inverters have been introduced for the purposes of counteracting the inherent volatility of active power injections of distributed PV [81]. The members of this family of control modes have unique impacts of distribution grids, and the fragmentation in their implementation further poses a risk to network model fidelity for utilities.

Critically, many utilities do not have complete or up-to-date information on these parameters and settings that characterize the impact of the PVs present within their grids. To solve this; in this chapter we introduce a number of data-driven modeling algorithms to detect and estimate unknown parameters for BTM PV. These algorithms fall into two general categories:

Model-Derived Methodology

The first category of PV detection and parameter estimation methods developed in this chapter is model-derived. By this, we mean that the estimation or detection method leverages physics-informed information from a distribution system model that is already known to the utility, to detect or estimate parameters for unknown PV systems within that model. These methods rely on simulated physical quantities from the feeder model, such as line flows of active or reactive power, or nodal voltages.

Furthermore, a common model-derived quantity explored in this chapter are methods based on model-derived voltage sensitivity matrices. These matrices capture the normalized changes in voltage magnitudes with respect to real and reactive power injections at a set of buses under study in the model.

Data-Driven Methodology

In contrast with the model-derived methods, several detection and estimation methods developed in this chapter are entirely data-driven. By this, we mean that no information from the distribution circuit model is included in the algorithm. In these algorithms, the parameters of the PV systems are characterized solely from advanced metering infrastructure (AMI) data.

In this chapter, we consider M net load time-series measurements from an AMI sensor for a customer with PV that are modeled as:

$$p_t^{net} = p_t^{nat} + p_t^{pv} + \epsilon \quad t = 1, \dots, M$$

$$q_t^{net} = q_t^{nat} + q_t^{pv} + \epsilon \quad t = 1, \dots, M$$

where the error of the sensor is assumed to be distributed according to a standard normal distribution, i.e., $\epsilon \sim N(0, \sigma^2)$. The variance of the error is typically on the order of 0.01 for modern AMI systems. Note that in this section of the paper we adopt the convention that injections from a node into the distribution system are positive and flows from the distribution system into the load through the AMI are negative, which simplifies the development of the algorithms greatly.

The data-driven methods presented in this paper are concerned with learning the parameters of the BTM PV systems from these data streams, without concern for the underlying system model. These methods have the advantage of circumventing the aforementioned issues in distribution circuit model fidelity faced by feeders with high penetrations of distributed PV.

Overview of Problems and Proposed Methods

Throughout this chapter, we will present data-driven and model-derived methods in parallel for the various detection and estimation problems that are posed. For some problems, we present only model-derived or data-driven methods, whereas other problems enjoy both types of solutions.

9.2. PV Size, Tilt, and Azimuth Estimation

In this section we present research on deep neural network approaches for estimating PV size, tilt, and azimuth using only behind-the-meter data. The methodologies presented are primarily derived from the results developed in [82].

9.2.1. PV Size Estimation

Data Generation

Two independent data sets were used to form a synthetic data set for training and testing: Load data from a Pecan Street data set and PV generation data from a UMass Trace Repository. The load and PV generation profiles from these data sets were combined to obtain the net load data using the following equation:

$$NetLoad_n(l,p,s) = Load_l - (PV_p \times ScalingFactor_s \times (1 + r))$$

where, $r \in [\ 0.5, 0.5]$ is a random constant. Figure 9-2 illustrates the data generation procedure when generating data for PV size estimation.

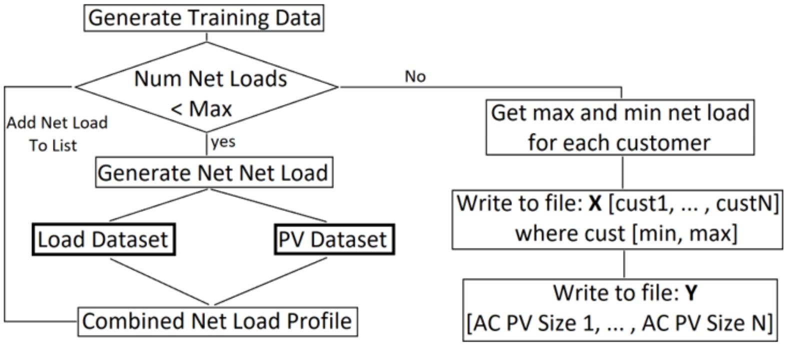

Generate Training Data
Num Net Loads < Max
No
yes
Add Net Load To List
Generate Net Net Load
Load Dataset
PV Dataset
Combined Net Load Profile
Get max and min net load for each customer
Write to file: **X** [cust1, ... , custN] where cust [min, max]
Write to file: **Y** [AC PV Size 1, ... , AC PV Size N]

Figure 9-2. Data generation flowchart

Classification

After the net load profiles are generated, each customer's profile is sliced into daily profiles and fed into the convolutional neural network (CNN). The output of the CNN is a binary value showing whether the CNN has detected a PV output pattern in that daily net load profile or not. This is done for all the daily profiles of a customer to obtain daily detected outputs. Once all the days are classified as "with PV" or "without PV," if the number of days for which outputs are detected as "with PV" is greater than a threshold (e.g., 200 days out of 365 days), we detect that customer as a "with PV" customer and assign a label.

Simulation

To have an accurate and computationally efficient PV detection method, it is necessary to know how the input data parameters impact the accuracy and computational burden. These parameters include the size of training data set (determined by the number of training customers and the temporal resolution of data), as well as the mislabeled ratio (the fraction of number of customers in training data that are mislabeled). Each of these parameters has a direct impact on the performance of the trained classifier. These impacts are simulated by developing parameter sweeps. For each parameter value, a 10-fold training and testing was performed. The results were averaged over 10 folds. At each fold, a new set of synthetic net load data was generated to train the classifier. Once the classifier was trained, a new set of synthetic net load test data was generated to evaluate the trained classifier on new unseen data for 1,000 customers. This process was repeated 10 times to get an average training and test accuracy.

Results

Linear regression (LR) provided a good performance when preprocessing the customer net loads and building a model from the minimum and maximum net load per customer. LR is considered as a benchmark for analysis. A similar approach was taken in this set of simulations by passing the maximum and minimum net loads per customer to the deep neural network (DNN) as input. The DNN then predicted the size of the installed PV from these inputs. The simulations were conducted using a dataset of 80 training customers, 20 validation customers and 1,000 test customers. The DNN was trained for 400 epochs with early stopping implemented. The input features were the minimum yearly day and night net load values. When evaluated on this new data set, the proposed DNN provides a mean absolute percentage error (MAPE) of 4.578% and mean absolute error (MAE) of 0.246 kW. In contrast, the benchmark LR approach provides a MAPE of 5.312% and MAE of 0.266 kW. See Table 9-1.

Table 9-1. AC PV size estimation error: 1,000 test customers, south facing PV panels, PV tilt equals latitude

	LR	DNN	DNN	DNN	DNN	DNN
	100 Train Cust	100 Train Cust	100 Train Cust	100 Train Cust Yearly	100 Train Cust Yearly	300 Train Cust Yearly
	Yearly	Yearly	Yearly	Min/Max	Min	Min
	Min/Max -	Min/Max -	Min/Max Night/Day	Night/Day + Uniform	Night & Day + Uniform	Night & Day + Uniform
Training Results						
Median APE (%	2.1487	1.8433	**1.2254**	1.4546	1.3786	1.3911
Mean APE (%)	2.5431	2.3752	**1.7969**	1.8818	1.8923	1.9342
Median AE (kW	0.1158	0.0996	**0.0671**	0.1671	0.1658	0.1781
Mean AE (kW)	0.1540	0.1497	**0.1204**	0.2933	0.2982	0.3097
Max AE (kW)	2.6666	3.0370	**2.1260**	2.2966	2.1925	2.4984
Test Results						
Median APE (%	2.1396	2.0041	1.4965	1.6930	**1.4443**	1.4636
Mean APE (%)	2.5229	2.6097	2.2113	2.2817	**2.0868**	2.0913
Median AE (kW	0.1180	0.1111	0.0841	0.0943	**0.0784**	0.0820
Mean AE (kW)	0.1652	0.1704	0.1482	0.1532	0.1457	0.**1448**
Max AE (kW)	3.1172	2.9579	3.2354	**2.9052**	3.1648	4.3770

9.2.2. Tilt and Azimuth Estimation

Data Generation

The PV data was generated using irradiance data sourced from NREL Measurement and Instrumentation Data Center. The irradiance data is at the coordinates Latitude: 35.41861° North and Longitude: 108.08828° West for the year 2012 with a resolution of 1-minute. The irradiance data at this location was passed into a simulated PV system using PVLib. The simulated PV panel was the CEC module named 'Yingli Energy China YL270C 30b'. The plane of array ground diffuse and direct normal irradiance (DNI) were calculated using the global horizontal irradiance (GHI) data for this location. The relative air mass was calculated based on the sun's zenith angle. The plane of array diffuse sky radiation was calculated using the diffuse irradiance, DNI, zenith angle and azimuth angle, and also the panels tilt and azimuth. The angle of incidence and plane of array irradiance was then calculated. The PV cell and module temperature were calculated using the plane of array irradiance, the average wind speed at 3 m and the dry bulb air temperature. The power output of the panel was then calculated for the CEC module using the plane of array irradiance and cell temperature. This process was repeated to generate PV profiles for PV systems with tilt and azimuths randomly distributed between [10,45] degrees and [90,270] degrees respectively. In previous quarters, the methodology for generation of net load data was developed and presented.

Feature Extraction

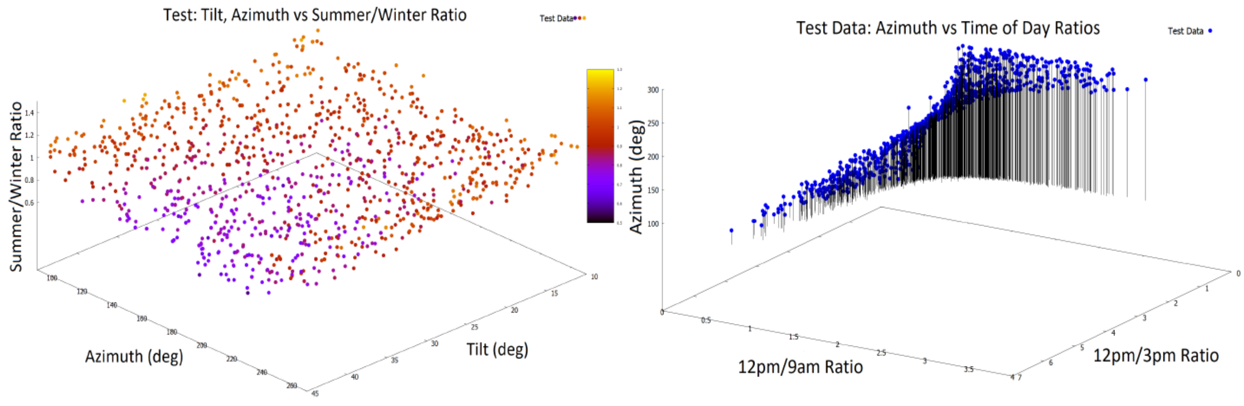
Features are extracted from the net load data that have a relationship with the tilt and azimuth. The ratio $R_1 = SummerNL_{Min}/Winter_{Min}$ was selected to estimate the tilt of the PV panel. The maximum daily power generated from a PV varies differently over the course of the year depending on the tilt of the panel. If the panel is flat, the maximum daily power output will be in mid-summer. Conversely, if the panel is tilted, the maximum daily power will be in the spring and fall months, or possibly even a maximum in the winter for significantly tilted systems. The ratios $R_2 = 12pmNL_{Min}/9amNL_{Min}$ and $R_3 = 12pmNL_{Min}/3pmNL_{Min}$ are used to estimate the azimuth of the installed PV system. These ratios make use of the information that the PV power generated throughout the day varies depending on the panel's azimuth.

Simulation Parameters

It was found that a data set of 900 training and 100 validation customers (1,000 in total) was necessary to achieve sufficient performance when estimating the tilt and azimuth of customer PV systems. All models were evaluated on a separate test data set of 1,000 customers. Each customer had a PV system that was configured with a random tilt and azimuth in the ranges [10; 45] and [90; 270] degrees respectively, where an azimuth of 180 corresponds to due south. A set of simulations were conducted that calculated the DNN input feature ratios, using net load data of varying resolution: Res = [1, 5, 15, 30, 60] minutes. The simulations investigate how robust is the proposed DNN method for tilt and azimuth estimation to imperfect training data. The DNN was trained with data sets consisting of [0; 10; 20; 30; 40; 50; 90] % mislabeled customers. Each mislabeled customer in this training set has their recorded tilt and azimuth reset to a random tilt and azimuth in the ranges [10; 45] and [90; 270] degrees respectively.

Results

To estimate the tilt and azimuth of installed PV systems, the three ratios corresponding to the minimum summer net load/minimum winter net load, minimum 12pm net load/minimum 9am net load, and finally minimum 12pm net load/minimum 3pm net load were calculated for each customer over the course of the year. These ratios are plotted in the figures below. Figure 9-3 illustrates the relationship between the calculated ratios and the tilt and azimuth.



(a) Summer/Winter Ratio vs Tilt and Azimuth

(b) Time of Day Ratios vs Azimuth

Figure 9-3. Input ratios for tilt and azimuth estimation

We summarize the research results as follows:

1. Deep neural networks can accurately estimate the size of installed PV systems with a MAPE of 2.09% using only features extracted from net load data, i.e., the minimum overall net load and the minimum nighttime net load over one year. This is a lower error than the benchmark linear regression approach. Over 85% of PVs have sizes estimated < 4% error using the proposed deep neural network. When the PV data is generated using a range of tilt and azimuth values, the AC PV size is estimated with a MAPE of 3.98%.
2. When estimating the tilt and azimuth of PV systems, the proposed deep neural network approach significantly improves upon the estimation error when using the benchmark linear regression model. The tilt and azimuth MAPE are 10.1% and 2.8% respectively when using the proposed deep neural network approach. This corresponds to an absolute error of 2.55° and 4.71° respectively. This is a 2.0% and 3.7% reduction in the tilt and azimuth percentage errors respectively when compared to linear regression.
3. A higher net load resolution of one minute improves the accuracy of the DNN when estimating AC PV size, however a higher resolution net load data does not lead to an improvement in accuracy when estimating tilt and azimuth. A net load data resolution of 1 minute provides a tilt estimation error that is statistically equal to that observed for a resolution of 60 minutes. A resolution of 60 minutes does provide a statistically lower azimuth error than 1-minute net load data resolution.

9.3. PV Location Estimation

This section describes a method that leverages model-derived voltage sensitivities in tandem with AMI data streams for estimation of solar PV location in distribution circuits. We first present a method to solve this problem for distribution circuits with fixed voltage regulation equipment, referencing the results of [83], and then augment the method to be robust against the impact of interacting voltage regulation equipment in the distribution circuit, referencing the results of [84].

9.3.1. Without Voltage Regulation Equipment

In this section, we use a voltage sensitivity matrix-based approach for estimating the location of solar PV. We follow the perturb and observe the methodology described at the beginning of this chapter to form this sensitivity matrix from a distribution circuit model. The injection of PV active power at a given location results in changes in the voltage magnitudes. These changes are fairly linear and consistent along time periods with solar irradiance variations. Let *i* be the node (the electric point) corresponding to a phase *p* of bus *b*. The set $\mathrm{N} = \{1, \dots, i, \dots N\}$ contains all such nodes in the system. We denote the change in voltage magnitude of node *i* with respect to a power injection change at a PV location ℓ as:

$$s_{i\ell} = \partial V_i / \partial P_\ell$$

The *N* nodal voltages in the system change due to this power injection. Thus, a vector of sensitivities with respect to injection at location ℓ can be written as $\mathbf{s}_\ell = \partial \mathbf{V} / \partial P_\ell$, where $\mathbf{V}$ is the vector containing all the node voltage magnitudes in the circuit. Using a distribution circuit model, one can obtain the values of the vector $\mathbf{s}_\ell$ for a given location ℓ by the following perturb and observe methodology [85]:

1. Solve the three-phase unbalanced power flow of the circuit for a baseline condition without the PV system,
2. Solve a second power flow with the PV system installed at location ℓ and all voltage regulating equipment (VRE) disabled, and
3. Record the voltage magnitude differences at each node by comparing the power flows with and without PV.

If the PV systems considered are known to be 3-phase, then the possible locations correspond to all the 3-phase buses in the circuit. If the PV systems considered are 1-phase, then the space of possible locations corresponds to all the nodes in the circuit. Let us denote by $\mathcal{L} = \{1, \dots, \ell, \dots L\}$ the set of all possible PV locations. By solving a power flow for the baseline and one power flow with the PV at each location ℓ, one can determine an $N \times L$ sensitivity matrix of node voltage changes with respect to PV injections at the L locations:

$$\mathbf{S} = \partial \mathbf{V} / \partial \boldsymbol{P}$$

We note that the voltages at the slack node do not change, and that the changes in voltages at nodes connected by switching devices are identical. The matrix $\mathbf{S}$ is full rank if the matrix column corresponding to nodes in the slack bus and one of each pair of nodes that are terminals of switches have been eliminated.

Estimation Method Using Measured Voltage

Using AMI data-streams in tandem with the sensitivity matrix, we will show that it is possible to use this sensitivity matrix to form a linear regression model to perform the estimation. Let us first consider an $N \times M'$ matrix **Z** that contains the data stream of measured node voltage magnitudes at *N* nodes for a given time horizon *H*. We are interested in the change of voltage magnitudes over time (as a function of the changes of PV injections). By taking the simple difference from one measurement scan to the next, we can obtain a matrix of voltage differences $\mathbf{D}'$ of size $N \times (M' - 1)$. During, the night the changes in voltage due to solar PV are zero. Thus, we select intervals of measurements during the day, where solar PV power magnitude as well as the variation in power is likely to be significant. This subset of voltage differences is denoted by a matrix $\mathbf{D}$ of size $N \times M$. In the trivial case of a single point, the matrix $\mathbf{D}$ corresponds to a vector **d** of size $N \times 1$. The vector **d** contains the measured changes in voltage magnitude due to the PV injection change at a given location ℓ. Thus, the vector **d**, down-scaled by the size of the PV system must be equal to one column of matrix **S**, the exact column of sensitivities corresponding to that location $\boldsymbol{\ell}$. In other words, it must be true that if the PV is installed at location ℓ, then:

$$\mathbf{s}_\ell = \partial \mathbf{V} / \partial P_\ell = \frac{1}{\alpha} \mathbf{d}$$

The measurements obtained from actual sensors will unavoidably contain errors due to sensor class, model inaccuracies, etc. Thus, the scaled vector **d** will be close to $\mathbf{s}_\ell$, but not exactly equal. Each column of matrix **S** represents the "direction" of the changes in voltage. The location of the PV system can then be determined by finding the column that is best aligned with the direction of the measured vector **d**. We want to estimate a vector **x** such that $\mathbf{Sx} = \mathbf{d}$. This problem is known to have a unique least-squares solution:

$$\hat{\mathbf{x}} = (\mathbf{S}^{\mathbf{T}}\mathbf{S})^{-1}\mathbf{S}^{\mathbf{T}}\mathbf{d}$$

Because the columns of $\mathbf{S}$ are linearly independent, $(\mathbf{S}^{\mathbf{T}}\mathbf{S})^{-1}$ is computable. The vector $\hat{\mathbf{x}}$ is the projection of $\mathbf{d}$ onto the subspace $\mathbf{S}$. It is the minimization of the components of $\mathbf{d} - \mathbf{Sx}$, such that $\|\mathbf{d} - \mathbf{S}\hat{\mathbf{x}}\| \leq \|\mathbf{d} - \mathbf{Sx}\|, \forall \mathbf{x} \in \mathbb{R}^L$. To incorporate not only one point, as in vector **d**, but more information available from the data stream, one can obtain a metric that captures the changes in voltage magnitude for a given period during the day. A suitable metric is the sum of the M_{pos} positive values of changes in voltage from matrix **D.** Thus instead of vector **d**, we use the following vector:

$$\bar{\mathbf{d}} = \frac{1}{M_{pos}} \sum_{t=1}^{T} pos(\boldsymbol{d}_t)$$

If we have a single vector of voltage magnitude deviations **d** obtained from the difference of voltage measurements at two points in time, resulting on an estimated value $\hat{\mathbf{x}}$. The vector of estimated voltage differences at each node is given by:

$$\hat{\mathbf{d}} = \mathbf{S}\hat{\mathbf{x}}$$

The normalized residuals of the voltage differences d_i are assumed to have a normal distribution $r_i \sim N(0,1)$, where: $r_i = (\hat{d}_i - d_i)/\sigma_i$. Voltage meters and smart meters usually have an error of less than 1%. In this paper, we assume that $\sigma_i = 0.01$. The least squares solution $\hat{\mathbf{x}}$ minimizes the sum of the squares of r_i:

$$\sum_{i=1}^{M} s_i^2(\mathbf{x}) = \chi^2 \geq \zeta = \sum_{i=1}^{M} s_i^2(\hat{\mathbf{x}})$$

We note that the value of ζ can alternatively be computed as:

$$\zeta(\hat{\mathbf{x}}) = [\mathbf{S}\hat{\mathbf{x}} - \mathbf{d}]^T \boldsymbol{\Omega}^{-1} [\mathbf{S}\hat{\mathbf{x}} - \mathbf{d}]$$

where $\boldsymbol{\Omega}^{-1}$ is a diagonal matrix with entries $1/\sigma_i$. The probability that the above event $\chi^2 \geq \zeta$, is given by the chi-square distribution:

$$\Pr[\chi^2 \geq \zeta] = 1.0 - \Pr[\zeta, \nu]$$

where $\nu = M - L$ is the number of degrees of freedom. $P = \Pr[\chi^2 \geq \zeta]$ represents the confidence level of the PV injection being at the estimated location. The smaller the value of ζ, the better the estimation will be. If a data stream is used instead of a single difference measurement, then the vector $\bar{\mathbf{d}}$ should be used in the above equations.

In summary, the goodness of fit is determined by the following process:

1. Compute the estimate $\hat{\mathbf{x}}$ using vector $\bar{\mathbf{d}}$
2. Compute the objective function ζ
3. Compute the probability $\Pr[\chi^2 \geq \zeta]$

Results

We utilized the IEEE 13 bus distribution system, which operates at 4.16 kV, has unbalanced loading, is relatively short and is highly loaded. This circuit also has a single voltage regulator at the substation, overhead and underground lines, shunt capacitors, and one in-line transformer. The circuit topology is presented in Figure 9-4. This IEEE 13 bus system contains buses that have 1, 2, and 3 phases. It has a total of 32 nodes (bus-phase combinations). It is assumed that meters that can measure the voltage magnitude are located at each one of the 32 circuit nodes. The loads are provided with the same load time profile.

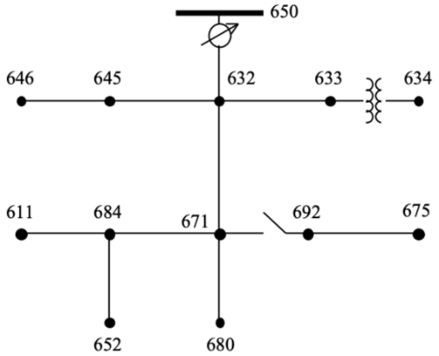


Figure 9-4. Diagram of the IEEE 13 bus system

The PV profile is based on irradiance data provided by NREL and represents the actual irradiation values observed on January 1, 2011 in Oahu, Hawaii. The PV profile is applied to the PV systems at one specific location. The PV profile selected has a 10-second resolution for one day, which represents a total of 8,640 data points. It is observed that highest values of PV irradiance as well as the highest changes in the PV profile occurred between 11 am and 1pm. For the simulations in the next two sections, all the voltage regulating equipment were fixed.

Estimation with 3-Phase PV System

The estimation process starts with determining the matrix **S** by connecting PV systems sequentially at each 3-phase bus and recording the changes in voltage magnitude in the 32 nodes in simulation. We assume that the injection of PV power has a power factor of 1.0. This matrix has a size of 5 locations times 32 nodes. Figure 9.4 (top) illustrates the columns of the sensitivity matrix **S**. Each one of the bar series for the five locations represents a unique signature on how the power injection at those locations changes the voltages in each one of the 32 circuit nodes. In order to be able to obtain solution for the estimates, **S** must be full rank.

We conducted the test by using only the voltage measurements in the time range from 11 am to 1 pm, because this is the time of the day when solar PV output is usually the highest. We assume a 5-minute (300 seconds) resolution of voltage measurements. To form vector $\bar{\mathbf{d}}$, we selected sub-intervals of 10 minutes. With this vector $\bar{\mathbf{d}}$, we computed the estimate of vector $\hat{\mathbf{x}}$ for each interval. The voltage variation caused by solar PV variation during an interval allows us to pinpoint the location of the PV.

As an example of the expected relation between the voltage sensitivities and measurements vector, let us consider the voltage measurements in the range from 12:50 pm to 1:00 pm. Figure 9-4 (top) represents the expected changes in voltage magnitude (the matrix **S**), while Figure 9-5 (bottom) presents the actual vector $\bar{\mathbf{d}}$ obtained using the equations above. We observe that the shape of the vector $\bar{\mathbf{d}}$ is closely resembles the shape of the column of matrix **S** corresponding to location number 3. Thus, the PV must be located at location 3.

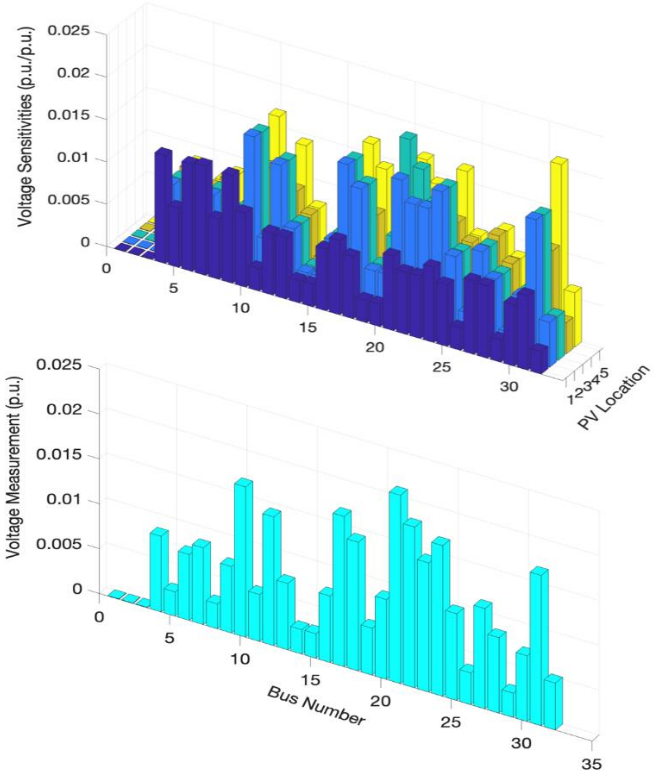


Figure 9-5. Representation of the sensitivity matrix S (top) and vector $\bar{\mathrm{d}}$ (bottom)

The size of the PV system modelled to generate the matrix **S** is 1000 kW. Table 9-2 presents the values of the estimated vector $\hat{\mathbf{x}}$ for the estimated locations versus actual locations during the time period from 11:30 to 11:40am. The high values in the diagonal indicate that the estimation is correct. Specifically, a high value close to 1.0 means that the PV system is highly likely to be located at that bus, while a value closer to zero means that the PV is highly unlikely to be located at that bus.

Table 9-2. Values of $\hat{\mathrm{x}}$ for estimated and actual locations from 11:30 a.m. to 11:40 a.m.

Estimated Location	Actual Location				
	633	671	675	670	680
633	1.138	0.0205	0.0205	0.0205	0.0205
671	0.0401	1.158	0.0401	0.0401	0.0401
675	0.0173	0.0173	1.135	0.0173	0.0173
670	0.0069	0.0069	0.0069	1.125	0.0069
680	-0.0316	-0.0316	-0.0316	-0.0316	1.086

The goodness of fit for the results of both estimations are presented on Table 9-3.

Table 9-3. Statistical results for 3-phase PV estimations

PV Location	633	671	675	670	680
Simulation 1: from 11:30 am to 11:40 am					
ζ	0.0517	0.0517	0.0517	0.0517	0.0517
$\Pr[\chi^2 \geq \zeta]$	0.9997	0.9997	0.9997	0.999	0.999
Simulation 1: from 12:20 pm to 12:30 pm					
ζ	0.0047	0.0047	0.0047	0.0047	0.0047
$\Pr[\chi^2 \geq \zeta]$	1.0	1.0	1.0	1.0	1.0

We repeat similar estimations for thirty different ranges between 11 am and 1 pm. We obtain the values of the estimated voltage differences $\hat{\mathbf{d}} = \mathbf{S}\hat{\mathbf{x}}$. We then compute the objective function of the estimation, i.e., the sum of the normalized residuals. Figure 9-6 presents the value of the objective function for the thirty estimations developed in this manner. This value of the objective function $\zeta(\hat{\mathbf{x}})$ is compared with the value of the χ^2 function at 95% confidence for M-N degrees of freedom, which is equal to 40.11. In all the estimations, the objective function is significantly smaller, indicating a good fit. This shows that the algorithm works seamlessly over a range of different time points in the day.

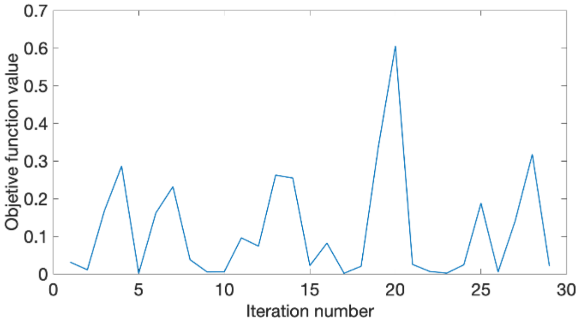


Figure 9-6. Objective function for 30 trials in different intervals

Estimation with 1-Phase PV System

For the simulation of 1-phase PV systems, we assume a granularity of 150 s, and measurement vectors are computed using 5-minute time intervals. We used a PV size of 2,000 kW for these simulations. These new values have been selected to test the method's robustness for varying values of timestep and PV size. There are 32 1-phase nodes in the system. As with the 3-phase analysis, we avoid the slack bus, one of the two buses at the ends of the switch, and the regulated buses to have a total of 20 possible PV locations to be analyzed. For the 1-phase analysis, we perform the same analysis for each PV location. We conduct the test for each phase separately for phases A, B and C to reduce error in the calculations. It is assumed that the phase information available to the utility is reasonably accurate.

The goodness of fit values for the 1-phase PV location estimation are presented in Table 9-4, which shows the strong performance of the method.

Table 9-4. Statistical results for 1-phase PV estimations

Phase	A		B		C	
Loc	ζ	$\Pr[\chi^2 \geq \zeta]$	ζ	$\Pr[\chi^2 \geq \zeta]$	ζ	$\Pr[\chi^2 \geq \zeta]$
611					1E-05	1.00
632	3E-04	1.00				
632			5E-04	1.00		
633	3E-04	1.00	5E-04	1.00	1E-05	1.00
645					1E-05	1.00
646			5E-04	1.00	1E-05	1.00
652	3E-04	1.00				
670	3E-04	1.00	5E-04	1.00	1E-05	1.00
671	3E-04	1.00			1E-05	1.00
675	3E-04	1.00	5E-04	1.00	1E-05	1.00
680	3E-04	1.00	5E-04	1.00	1E-05	1.00

Conclusion

A method has been described for the estimation of the location of 3-phase and 1-phase PV systems in distribution circuits. The method is based on voltage measurements and their differences across time and assumes an accurate distribution circuit model and fixed controls of voltage regulating equipment. The numerical results indicate that the method is highly accurate for both 1-phase and 3-phase PV systems location estimation. For the case of a 1-phase system, the estimation must be conducted for each phase separately.

9.3.2. With Voltage Regulation Equipment

Impact of VRE on Sensitivities

In the presence of VRE, the impact of VRE actions and the PV power injection both produce changes in voltage magnitude in the circuit nodes and in some change the sensitivity columns may not be independent vectors. To illustrate this, let us consider Figure 9-7, where the sensitivities of voltage with respect to VRE action and PV injections are independent from each other, and hence their impact can be added to the S matrix. However, situations are found in which $\boldsymbol{s_{taps}}$ or $\boldsymbol{s_{Cap}}$ are linearly dependent on the columns of matrix S. This situation corresponds to Figure 9-8, where a voltage change that is due to a VRE action can be expressed in terms of the PV sensitivity column vectors. In such case, the Gramian of S becomes singular and $(\mathbf{S^T S})^{-1}$ cannot be computed. Therefore, an approach must be developed that can estimate PV injections, tap changes, and switching capacitor actions regardless of the structure of VRE sensitivities.

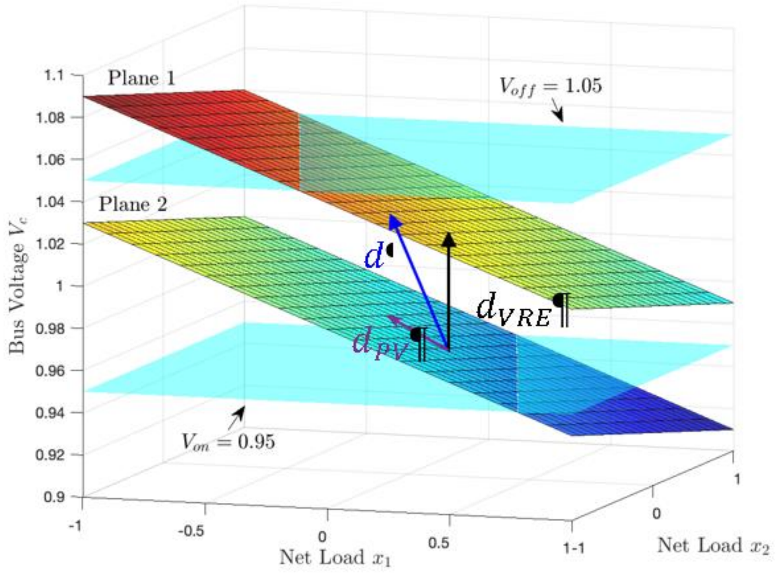


Figure 9-7. Illustration of VRE impact independent from sensitivity factors

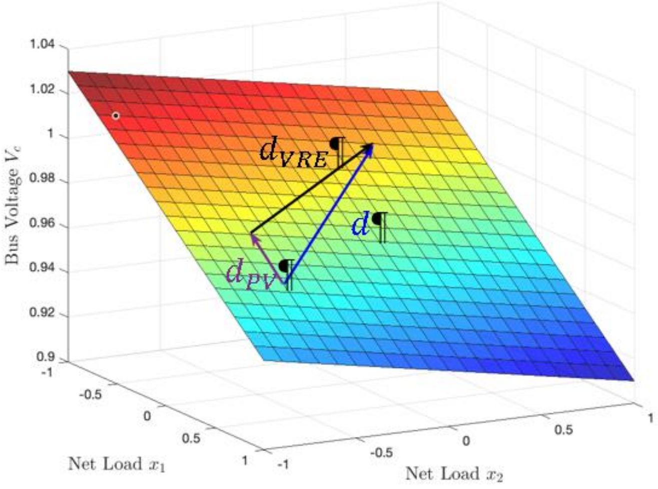


Figure 9-8. Illustration of VRE impact when it depends on PV sensitivity vectors

Proposed Method

The method extends the principles used so far for PV location and injection estimation by first determining whether a VRE actions is present in the voltage magnitude measurement data stream. We reference [83, p.] throughout this section for describing this method. Let us define $\hat{d}_{VRE}$ as the estimation of the changes in voltage due to VRE actions in a distribution circuit. The correct estimation of the PV injections would discount the effect due to the VRE in changes in voltages d:

$$\hat{x} = (\boldsymbol{S}^T\boldsymbol{S})^{-1}\boldsymbol{S}^{\boldsymbol{T}}(d - \hat{d}_{VRE})$$

To estimate $\hat{d}_{VRE}$, it is necessary to know how VRE actions impact the estimation. The change in voltage due to a tap change will have components that are usually related to those PV location sensitivities in conflict with the VRE. The changes in voltage occur in fixed amounts, proportional to the number of taps changed. It is possible to obtain an estimation of a voltage changes due to VRE, d_{VRE} by simulating VRE actions in the distribution circuit and performing an estimation for the resulting voltage changes. Let us denote the result of this estimation by $\hat{x}_{VRE}$. We have that:

$$\hat{x}_{VRE} = (\boldsymbol{S}^T\boldsymbol{S})^{-1}\boldsymbol{S}^{\boldsymbol{T}} d_{VRE}$$

This particular estimation vector corresponds to a single VRE device. A matrix $\boldsymbol{X}_{VRE}$ can be formed when considering all the VRE devices in the circuit:

$$\boldsymbol{X}_{VRE} = (\boldsymbol{S}^T\boldsymbol{S})^{-1}\boldsymbol{S}^{\boldsymbol{T}} D_{VRE}$$

where $D_{VRE} = \left[d_{VRE_1}, \ \dots, d_{VRE_K}\right]$ so that $\boldsymbol{X}_{VRE} = \left[\hat{x}_{VRE_1}, \ \dots, \hat{x}_{VRE_K}\right]$. Each vector $\hat{x}_{VRE}$ of $\boldsymbol{X}_{VRE}$ is the expected footprint that a tap change will leave in the estimation. Once $\boldsymbol{X}_{VRE}$ has been determined using simulation, it can be used to determine the presence of tap changer actions, if the resulting estimation vector is similar to $\hat{x}_{VRE}$.

When a change in voltage contains the impact of VRE and PV for a given point in time, the resulting estimation vector will contain components associated with both the PV location and the expected estimation $\hat{x}_{VRE}$:

$$x_{PV+VRE} = (\boldsymbol{S}^T\boldsymbol{S})^{-1}\boldsymbol{S}^{\boldsymbol{T}} d$$

The matrix $\boldsymbol{X}_{VRE}$ can now be used to determine whether any VRE action took place by performing a second estimation on the resulting estimation vector:

$$v = ((\boldsymbol{X}_{VRE}^T\boldsymbol{X}_{VRE})^{-1}\boldsymbol{X}_{VRE}^T) \cdot x_{PV+VRE}$$

The resulting vector v will provide non-zero values for those components associated with the VRE devices that operated at that specific point in time. For example, if 2 step changes occurred for a certain voltage regulator, the resulting v component may be 2.03. For devices that didn't take action at that point in time, a value close to 0 will appear. A non-linear filter $\boldsymbol{\phi}$ needs to be applied to remove the values close to zero and to obtain integer components from $\boldsymbol{v}$. Once a VRE action is detected, the impact on the voltage is computed by multiplying by $\boldsymbol{X}_{VRE}$ – which leads to the estimation components of tap change – and finally by $\boldsymbol{S}$, which leads to the estimated voltage increase due to that VRE action:

$$\hat{d}_{VRE} = \boldsymbol{S} \cdot \boldsymbol{X}_{VRE} \cdot \phi(v)$$

With the subtraction of the impact of VRE on voltage increments it is not only possible to predict the location of PV systems, but also the action taken by controlling devices. Putting together these formulas in $\hat{x} = (\boldsymbol{S}^T\boldsymbol{S})^{-1}\boldsymbol{S}^{\boldsymbol{T}}(d - \hat{d}_{VRE})$, the formulation of the method now becomes:

$$\hat{x} = (\boldsymbol{S}^T\boldsymbol{S})^{-1}\boldsymbol{S}^{\boldsymbol{T}} \left(d - \boldsymbol{S} \cdot \boldsymbol{X}_{VRE} \cdot \phi\Big(((\boldsymbol{X}_{VRE}^T\boldsymbol{X}_{VRE})^{-1}\boldsymbol{X}_{VRE}^T)(\boldsymbol{S}^T\boldsymbol{S})^{-1}\boldsymbol{S}^{\boldsymbol{T}} d\Big)\right)$$

For clarity purposes, the following nomenclature is used. The number of nodes being monitored is called N, which is also the dimension of the measurement vector; L is the total number of possible PV Locations (number of columns of S matrix) and K is the total number of VRE devices in the circuit. The matrices and vectors in the formula above are defined as follows:

- $\boldsymbol{S}$ is the sensitivity matrix. It has a size of N x L.
- d is the measurement vector: $[d_1, \dots, d_N]^T$.
- $\boldsymbol{X}_{VRE}$ includes all the estimation vectors due to VRE devices. It has a size of L x K.
- v is the vector of estimated VRE actions. Hence its dimension is equal to the number of total VRE devices in the circuit: $[v_1, \dots, v_K]^T$.

This is the general formulation used to estimate PV injections for a given point in time considering VRE actions. The number of tap changes can also be predicted by looking at the term

$$\phi\Big(((\boldsymbol{X}_{VRE}^T \boldsymbol{X}_{VRE})^{-1} \boldsymbol{X}_{VRE}^T)(\boldsymbol{S}^T \boldsymbol{S})^{-1} \boldsymbol{S}^{\boldsymbol{T}} d\Big).$$

Numerical Results

In this section the resulting estimations from the IEEE123 test feeder are analyzed. Consider the estimation output for 16 different simulations within the time window 11:35 to 11:40 pm, taking the VRE sensitivity vectors as dependent from the vectors in S. The simulation with PV systems at nodes 36.1, 40.1 and 37.1 included a tap change at this point in time. The time resolution is 300s (5 min). In each simulation, a 1-phase 100 kW PV system was placed.

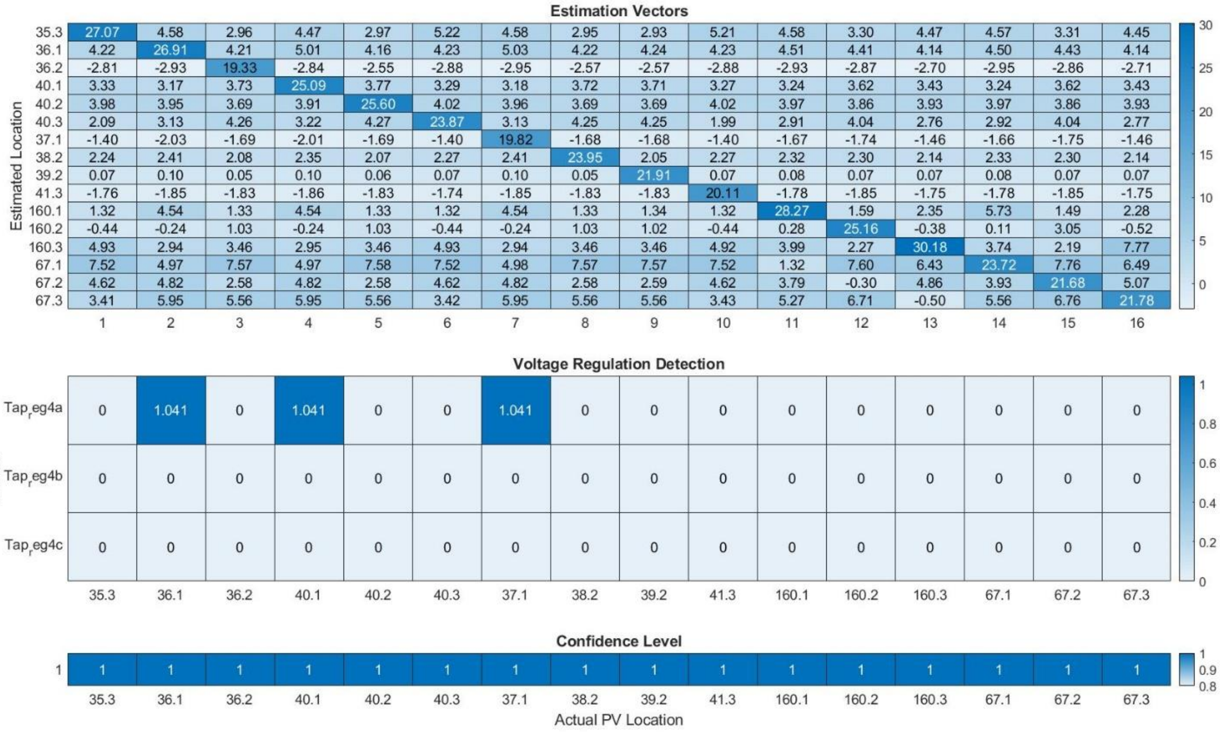


Figure 9-9. Heatmap plot of results of PV location estimation for various PV case scenarios. The PV injection estimations are accurate and have a high confidence level, and consider the action of discrete voltage regulation equipment.

Conclusion

A method has been developed and tested for estimating the correct PV system location for changes in voltages under the action of VREs. The accurate PV injections and high confidence levels corroborates the results of the analysis.

9.4. BTM PV Control Setting Estimation Methods

Advanced solar PV inverter control settings may be out of date or completely unavailable in distribution circuit models, and the manual entry process for the collection and maintenance of these data is highly prone to error [2]. This is of particular importance as the deployment of BTM PV systems expands, as these systems are operated under a large variety of control settings in accordance with the IEEE 1547-2018 standard [80]. The wide variety of possible control settings of a BTM PV advanced inverter results in a scenario where the control settings may be unknown or uncertain to a distribution engineer. The algorithms developed in this section seek to address this issue by allowing an engineer or utility to reconstruct these settings from the net load data without any input from the customer or the solar installer. The methods presented are primarily data-driven, with varying degrees of reliance on a distribution system model.

9.4.1. Voltage Sensitivities

A distribution circuit's voltage sensitivity matrices can be used to estimate BTM PV control settings, in a similar manner to how they were used to estimate the location of unknown BTM PV in the previous sections of this chapter. In this chapter of the report, for the sake of brevity, we will assume that voltage regulating equipment (VRE) settings are held fixed. However, the sensitivity-based methods for estimating the control parameters of BTM PV can be utilized in contexts where VRE state changes occur by using results from Section 9.3.2, and by referencing related works on sensitivity-based estimation including VRE, such as [83], [86].

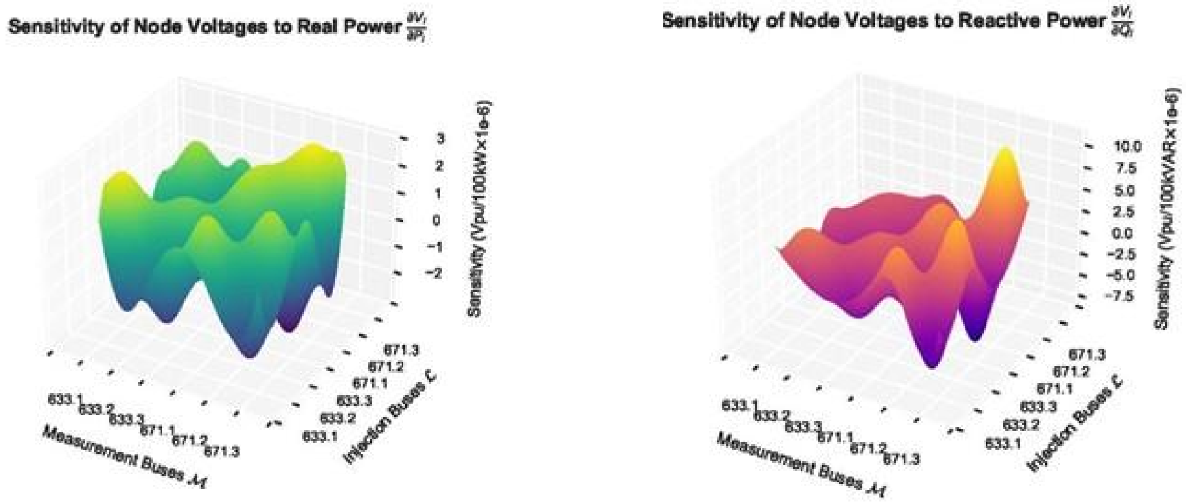


Figure 9-10. Voltage sensitivity matrices (interpolated to highlight independence) corresponding to real (left) or reactive (right) power. each matrix captures the normalized changes in node voltage at each observable bus to changes in the power injections

As described in Section 9.3., the voltage sensitivity matrices for a distribution feeder can be understood as a static, model-based quantity that is fixed inter-temporally for the distribution system under analysis. Constructing this matrix is straightforward, provided a distribution circuit model is available. The procedure for constructing these matrices is outlined in Algorithm 1 in [87]. Notably, we will also utilize the voltage sensitivities to reactive power injections; which, assuming circuit parameters are held fixed and injection magnitudes are normalized, can be used to construct independently estimated real and reactive power injection states. This enables the estimation of the power factor control setting of a BTM PV system with solely voltage magnitude measurements. The distinction has been noted in the literature [87], [88], and we leverage this distinction later in this chapter.

9.4.2. Control Curve Regression

To reconstruct the control settings of the BTM PV, an additional solution, which is intuitive and easily interpretable, is to filter the historical AMI data to expose the control setting. This can be achieved by empirically locating "extreme" historical observations that are likely to well-approximate the control curve response.

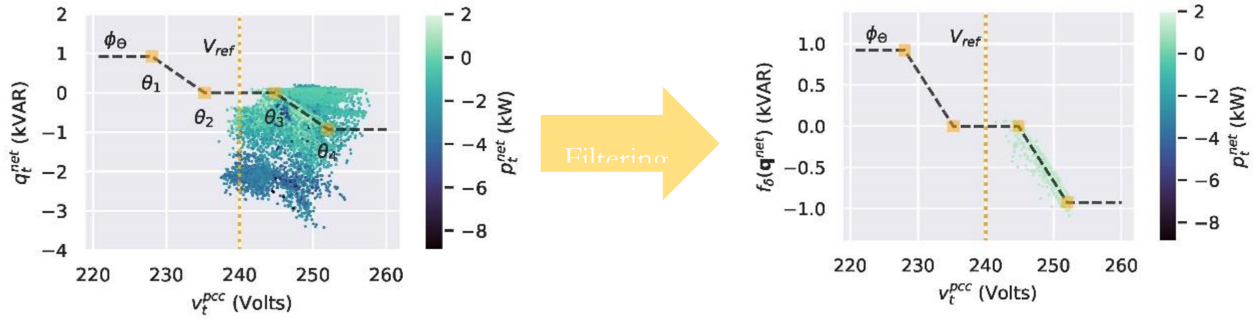


Figure 9-11. Example of filtering historical AMI dataset (left) to expose an unknown BTM Volt-VAR characteristics curve (right). True PV reactive power response shown as dashed black/orange. All samples are reactive power as a function of PCC voltage.

Depending on the control setting under study, the choice of what makes an AMI measurement "extreme" differs. In systems with Volt-VAR control, selecting the subset of observations with smallest net real power demand yields a subset of the observations that are a good approximation for the control curve, shown in Figure 9-9. Note that with Volt-VAR control, the goal is to estimate a vector of parameters $\Theta = [\theta_1, \dots, \theta_4] = [(V_1, Q_1), \dots (V_4, Q_4)]$.

For estimating fixed power factor control, filtering the historical AMI measurements with respect to the highest historical point of common coupling (PCC) voltage measurements yields a subset of the observations that are a good approximation for the control curve, shown in Figure 9-12. The precise details of this filtering procedure are available in Algorithm 1 of [87].

To perform the filtering, as a first step, we can simply filter out all the nighttime data points, and solely focus on the daytime observations for performing our estimation. However, the customer load will likely still produce reactive power during the daytime. In this case, the data should be further preprocessed to remove the influence of the customer's load reactive power. Ideally, the final process will approximate the response of the PV inverter. This technique is often known as "solar disaggregation," and we will describe this in the subsequent section.

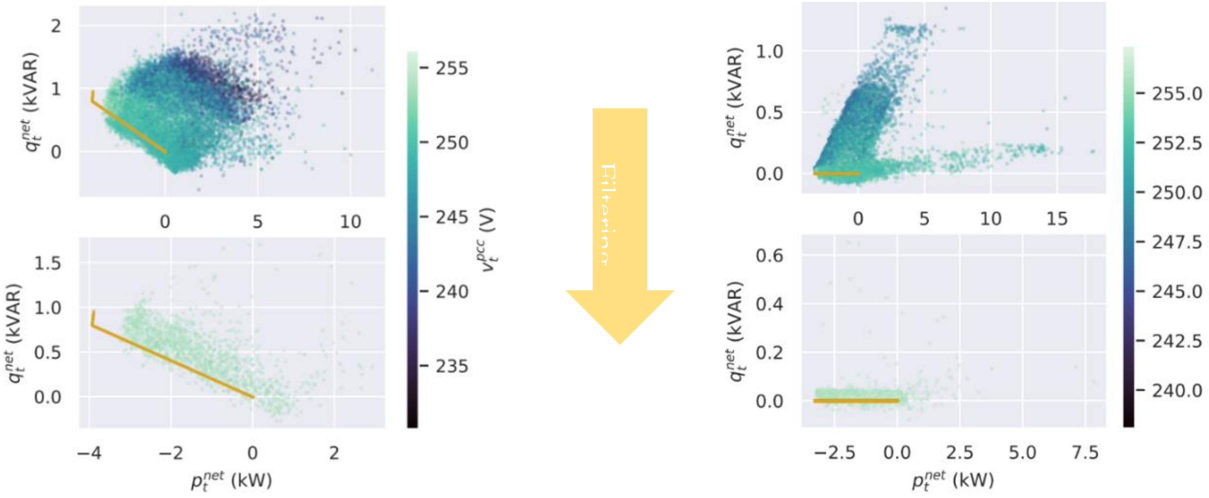


Figure 9-12. Example of filtering historical AMI datasets to expose fixed non-unity power factor control (left) and unity power factor control (right) curves. True PV reactive power response shown as orange. All samples are reactive power as a function of real power

9.4.3. Reactive Power Disaggregation

A fundamental problem to be solved to achieve estimate unknown solar PV inverter control settings from AMI data is the so-called "solar disaggregation" problem. In summary, this problem is concerned with separating the additive time-series contributions of the "native" power demands of the customer p_t^{nat}, q_t^{nat} and the PV real and reactive power contributions p_t^{pv}, q_t^{pv} from net load time-series real and reactive power measurements from the AMI p_t^{net}, q_t^{net} at each timestep t.

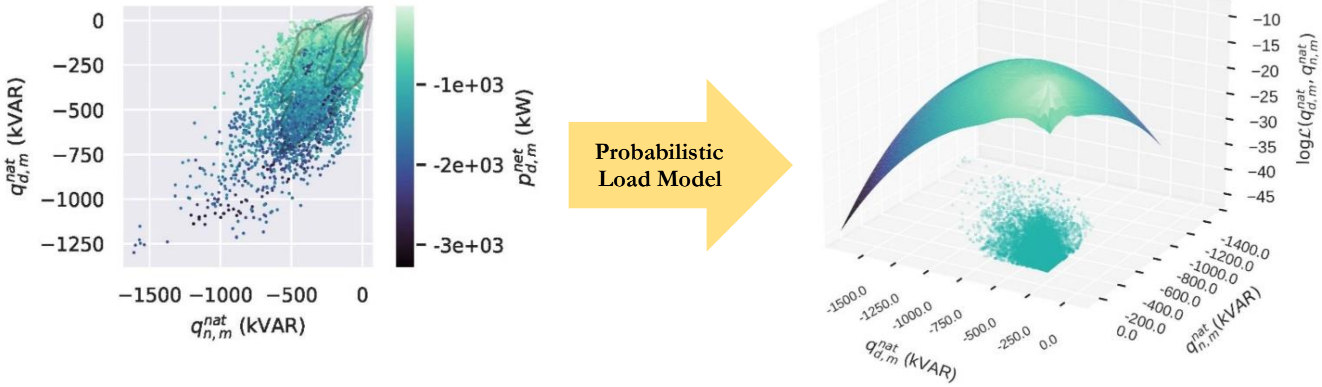


Figure 9-13. Left: Scatterplot of 1 year of daytime vs. nighttime native reactive power measurements at 1 week granularity for 678 loads in the feeder without PV. Right: Probabilistic reactive load model

Various solutions have been proposed for the problem of active power disaggregation of solar and native customer demand from net load measurements in the literature [89]. To achieve the reconstruction of inverter control settings, it is necessary to generalize these methods to the problem of reactive power disaggregation, by forming a probabilistic load model for the native reactive power consumption of loads without BTM PV in the distribution system, as illustrated in Figure 9-13.

By disaggregating the native and PV reactive power signals from the net load AMI signal, it is possible to reconstruct the setting given constraints on the structure of the control curve. Imposing the constraints allows the reactive power disaggregation problem to simultaneously provide a maximum likelihood estimate for the BTM PV's unknown control curve. Precise details of this method are available in[90]. At a high level, we seek the solution $\widehat{\Theta}$ to the maximum likelihood estimation (MLE) problem of the form:

$$\max_{\Theta} p(\boldsymbol{q}_n^{net}, \boldsymbol{q}_d^{net} | \Theta)$$

where $\boldsymbol{q}_n^{net}, \boldsymbol{q}_d^{net}$ are M-dimensional vectors of *net* nighttime and daytime reactive power measurements from the AMI at the bus of interest, p is the approximate joint probability distribution of nighttime and daytime *native* reactive power observations at loads in the feeder *without* BTM PV, and Θ is a parameter vector for the control curve. The estimated joint probability distribution p is often called a generative distribution. The use of this methodology for disaggregation is described further in [89], [90].

9.4.4. Sparse Time-series Sampling

The control setting estimation methods in the previous sections suffer from several common pitfalls of data-driven methods, such as significant data input requirements, or the need for manual parameter tuning. The voltage sensitivity-based method requires a distribution system model, the disaggregation method requires a large amount of historical data from other customers without PV, and the control curve regression method requires the tuning of a filtering parameter.

To resolve this issue, an additional contribution of this chapter leverages recent results in sparse sampling [91], [92], which allows an engineer to extract the control setting of a BTM PV system from AMI data with a greatly reduced number of measurements. In addition to the benefits of this method outlined in the previous paragraph, this method is valuable in scenarios where computational cost and AMI data integrity are of concern.

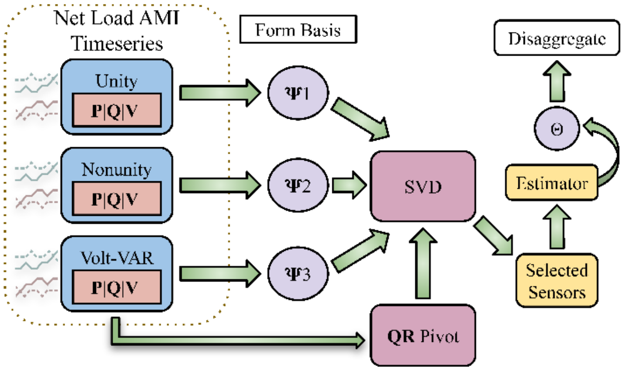


Figure 9-14. Diagram of the Sparse Time Series Sampling Control Setting Estimator using the singular value decomposition

The sparse time-series sampling method for reconstructing DER control settings relies on the sparse sensor placement for reconstruction algorithm implemented in [91]. The method hinges upon constructing a low-rank matrix of candidate control curves using the singular value decomposition (SVD). The candidate curves are constructed such that they meet the respective standard of the control curve type under study, such as the IEEE 1547-2018 standard for the case of the Volt-VAR curve or fixed power factor control curves. The mathematical formulation of this algorithm is available in [93]. The low-rank basis matrix is then used to extract the BTM settings via the QR decomposition. The general pipeline of the algorithm is illustrated in Figure 9-14.

9.5. BTM PV Power Factor Control Setting Estimation

9.5.1. Unity and Nonunity Power Factor Control Characteristics

Control settings for BTM PV can be understood in terms of a control curve, also known as a characteristics curve. For fixed power factor control, this control curve is a simple line in the complex power plane fixed at the origin. Therefore, for both unity and nonunity power factor control, the reactive power response of the PV system can be entirely characterized by the slope of this line, i.e., $q_t^{pv} = \frac{\Delta q}{\Delta p} p_t^{pv}$ at each timestep t. Therefore, the goal of this section of the report is to estimate $\frac{\Delta q}{\Delta p}$ from AMI data.

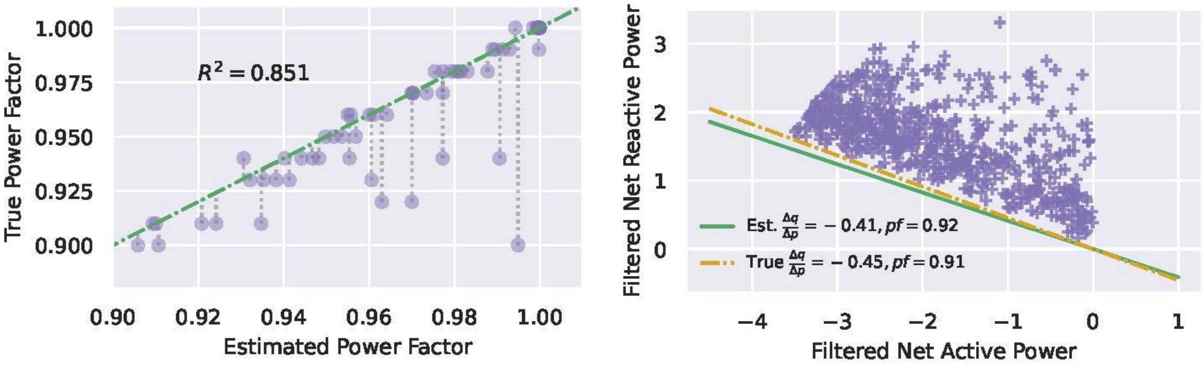


Figure 9-15. True power factor control settings vs. estimated setting via control curve regression (left), and an example of an estimated nonunity power factor control curve (right).

9.5.2. Estimation Using Control Curve Regression

Provided that full access to the AMI data for a customer with BTM PV is available, a regression approach combined with the filtering methodology described in Section 9.5.2 can be used to recover the power factor control setting of BTM PV, as depicted in Figure 9-15. Using ordinary least-squares regression or a robust regression method such as Huber regression or L1 norm approximation, an engineer can reconstruct unity or nonunity power factor control curves with a high level of accuracy. The precise details of the filtering methodology and the various regression methods available for fitting the control curve are available in [87].

9.5.3. Estimation Using Voltage Sensitivities

If an engineer has access to a distribution system model that can be reasonably assumed to be accurate, this model can be used to construct an "interleaved" sensitivity matrix composed of alternating columns of voltage magnitude sensitivities to real and reactive power as described in Section 9.5.1. Access to this sensitivity matrix is valuable because it significantly reduces the data input requirements to estimate the power factor control settings of BTM PV. This interleaved sensitivity matrix is defined as an $N \times 2L$ matrix of real numbers, where N is the number of measuremed nodes in the distribution circuit and L is the number of buses under study that may potentially have BTM PV. The structure of this matrix is:

$$\boldsymbol{S}_{PQ} = \begin{pmatrix} \frac{\partial V_1}{\partial P_1} & \frac{\partial V_1}{\partial Q_1} & \cdots & \frac{\partial V_1}{\partial P_L} & \frac{\partial V_1}{\partial Q_L} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{\partial V_N}{\partial P_1} & \frac{\partial V_N}{\partial Q_1} & \cdots & \frac{\partial V_N}{\partial P_L} & \frac{\partial V_N}{\partial Q_L} \end{pmatrix}$$

Using voltage meters in the system that give the engineer access to voltage profile data $v_i(t)$ at any bus at the buses $i = 1, \ldots, N$, we define a $N \times 1$ column vector $\overline{\boldsymbol{d}}$ as a containing samples of voltage differences Δv_i at each node during timepoints with high voltage difference due to volatile BTM PV injections, as described in [83], [83], [87]. The $\boldsymbol{S}_{PQ}$ matrix and the $\overline{\boldsymbol{d}}$ vector form a least-squares regression problem, which allows the engineer to estimate the contributions of real and reactive power injections from the candidate buses in the power system.

The estimation of the injection state vector can be found through any choice of linear regression model. In this least-squares case, a closed form solution is available as:

$$\hat{\boldsymbol{x}} = \left(\boldsymbol{S}_{\boldsymbol{PQ}}^{\boldsymbol{T}} \boldsymbol{S}_{\boldsymbol{PQ}}\right)^{-1} \boldsymbol{S}_{\boldsymbol{PQ}}^{\boldsymbol{T}} \overline{\boldsymbol{d}}$$

this solution is described in [83], [87]. Furthermore, the underdetermined nature of the linear system formed results in many scenarios where regularization can be used to improve the estimation, such as the well-known least absolute shrinkage and selection operator (LASSO) or ridge regression, which places tunable penalties on the ℓ_1 and ℓ_2 norms of the estimation vector $\hat{\boldsymbol{x}}$, respectively. Further details of these alternative, regularized methods for the problem of power factor estimation using voltage sensitivities are available in [87], and a classic overview of regularized linear models is available in [94].

The estimation vector can be used to form an estimate for the power factor control setting of the inverter-based DER(s), using the definition of the power factor and the estimated injections $\hat{P}_l, \hat{Q}_l$ at the L under study. The power factor estimate at bus l is therefore:

$$\widehat{\mathrm{pf}}_l = \frac{\hat{P}_l}{\sqrt{\hat{P}_l^2 + \hat{Q}_l^2}}$$

An example of the results of this method for the IEEE 13 bus feeder are shown in Figure 9-15, where the true 3-phase injection state is $S_{633} = 150 + j100\ kVA$ and $S_{671} = 25 + j45\ kVA$. In addition, we show a bootstrap confidence interval for the estimation formed by resampling the sensitivities entries at random, to characterize the precision of the estimator.

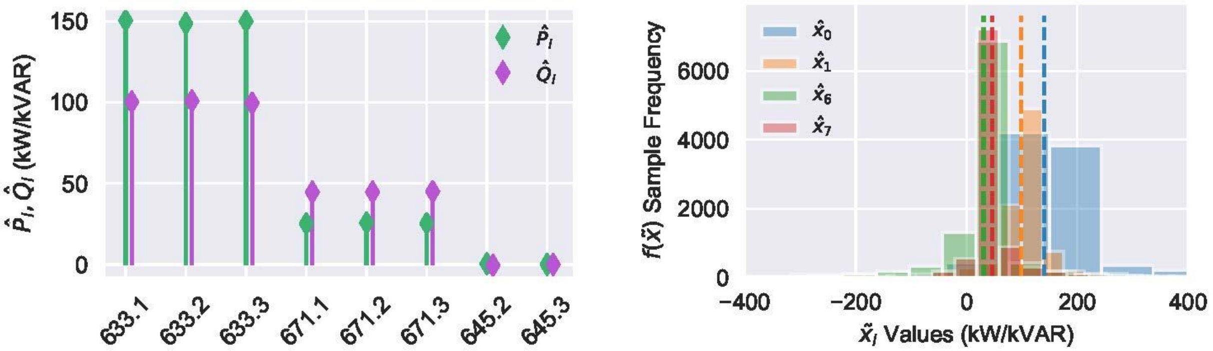


Figure 9-16. Sensitivity-based estimation for a BTM PV system with fixed power factor control (left) and bootstrap confidence interval plot for entries 0,1,6 and 7 of the vector (right)

9.6. BTM PV Volt-VAR Control Setting Estimation

9.6.1. Volt-VAR Control Characteristics

A BTM PV system with an advanced inverter operating under Volt-VAR control schedules the reactive power injection or absorption of the inverter according to one of several control states. The active state is determined according to a "control curve" or "characteristics curve" as defined by the IEEE 1547-2018 standard. This curve can be described through a vector of parameters or "knots," which are pairs of reactive power and PCC voltage values. We can write this vector of parameters for the control setting as $\theta = [(V_1, Q_1), \ldots, (V_4, Q_4)]$, and find the best-fit control curve $\phi_\theta(v_t^{pcc})$ which describes the PV reactive power response q_t^{pv}.

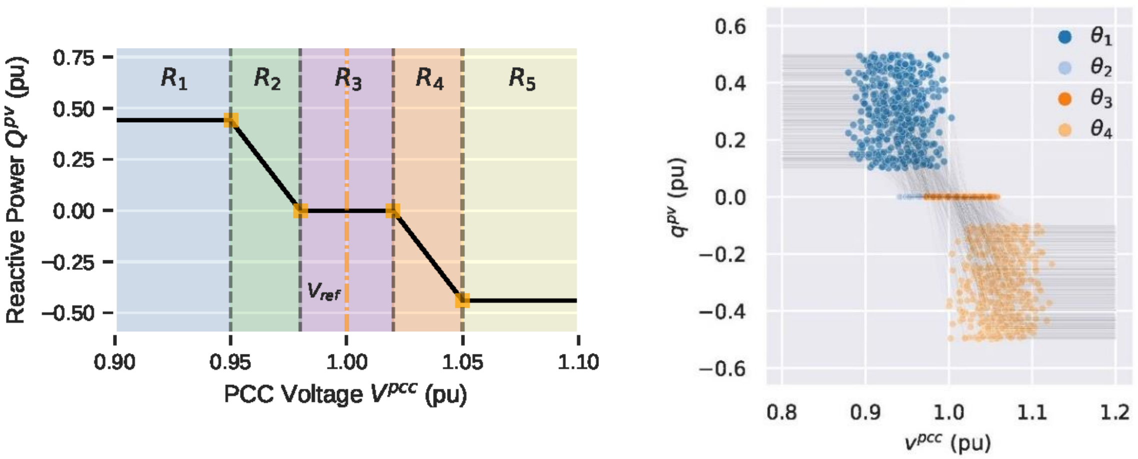


Figure 9-17. An example Volt-VAR characteristics curve (left) and the set of IEEE 1547-2018 compliant parameters that are estimated by the algorithms in this section (right)

$$q_t^{pv} = \phi_\theta\left(v_t^{pcc}\right) = \begin{cases} Q_1 & v_t^{pcc} \le V_1 \\ \dfrac{Q_2 - Q_1}{V_2 - V_1}\left(v_t^{pcc} - V_1\right) + Q_1 & V_1 < v_t^{pcc} < V_2 \\ 0 & V_2 \le v_t^{pcc} \le V_3 \\ \dfrac{Q_4 - Q_3}{V_4 - V_3}\left(v_t^{pcc} - V_3\right) + Q_3 & V_3 \le v_t^{pcc} \le V_4 \\ Q_4 & v_t^{pcc} > V_4 \end{cases}$$

IEEE 1547-2018 defines four of these knots, and the overall structure of the control curve, as well as the range of possible values simulated in this report, is shown in Figure 9-17. The goal of this section of the report is to introduce and describe various methods to estimate these control curve states or knots solely from net load AMI data. As with power factor control setting estimation, a challenge with this problem is how to separate the BTM PV and native demand contributions to accurately recover the control curve. In the subsequent sections, we will present several methods for solving this problem and accurately recovering the BTM control setting with a high degree of accuracy.

9.6.2. *Estimation Using Control Curve Regression*

Volt-VAR control settings of BTM PV can be directly estimated with non-linear least-squares regression by directly solving the following optimization problem numerically.

$$\min_{\Theta} \sum_t \left(f(\boldsymbol{q}^{net})_t - \phi_\Theta\left(v_t^{pcc}\right)\right)^2$$

Here, $f(\boldsymbol{q}^{net})_t$ represents the output of the AMI data filtering method at timestep t described in the previous sections. Note that this optimization problem can be solved with the additional constraint that $\widehat{\Theta}$ satisfies the IEEE 1547-2018 standard or can be solved entirely unconstrained. We can then compute the mean absolute percentage error and the residuals of the estimated parameters $\widehat{\Theta}$ against the true parameters of the BTM system. The results of this method are summarized below in Table 9-5. Additional details on the implementation of this algorithm and its performance are available in [90].

Table 9-5. Summary of estimation performance for activated Volt-VAR control parameters from using constrained and unconstrained regression for 701 customers with BTM PV.

Algorithm Type	Volt-VAR Parameter	MAPE (701 Customers)
Unconstrained Regression	V_3	5.331%
	V_4	0.308%
	Q_4	9.701%
IEEE 1547-Constrained Regression	V_3	0.425%
	V_4	0.206%
	Q_4	2.870%

9.6.3. Estimation Using Reactive Power Disaggregation

The disaggregation method described in the introduction section of this chapter, which is achieved via a machine learning algorithm to probabilistically model the behavior of the loads on the feeder without PV, allows the user to simultaneously disaggregate the native reactive power time-series q_t^{nat} and PV reactive power time-series q_t^{pv} from the measured net load AMI reactive power time-series q_t^{net} at each timestep t, as shown in the left-hand side of Figure 9-18.

Simultaneously, the data-driven disaggregation model allows the engineer to find a maximum likelihood estimate for the control curve, as shown in the righthand side of Figure 9.17. These two problems are complementary in nature, and their precise formulation is available in [90].

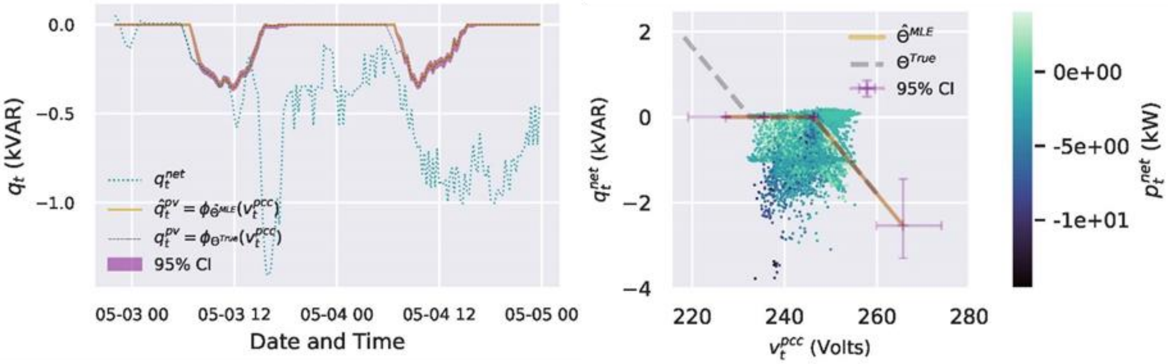


Figure 9-18. Example reactive power disaggregation results from the time-series perspective (left) and the curve fitting perspective (right). 95% confidence interval bands shown in right.

9.6.4. Estimation Using Sparse Time-series Sampling

To use the sparse time-series sampling method to reconstruct the Volt-VAR control curve, an engineer must specify the number of permitted time-series samples P that are then optimally selected using the cost-constrained QR decomposition described in [91]. This is achieved by forming a physics-based design matrix of candidate control curves evaluated over the data stream of the observed control variable. For N candidate Volt-VAR control curves with M time-series measurements, this matrix is structured as below.

$$\boldsymbol{\Psi} = \begin{bmatrix} \phi_{\Theta_1}(v_1^{pcc}) & \cdots & \phi_{\Theta_N}(v_1^{pcc}) \\ \vdots & \ddots & \vdots \\ \phi_{\Theta_1}(v_M^{pcc}) & \cdots & \phi_{\Theta_N}(v_M^{pcc}) \end{bmatrix}$$

The optimal low-rank design matrix $\boldsymbol{\Psi_r} \in \mathbb{R}^{M\times R}$ to extract the temporal patterns is then found through the truncated singular value decomposition (SVD), as described in [93]. The algorithm then selects an optimal linear measurement matrix $\boldsymbol{C} \in \mathbb{R}^{P\times M}$ that allows us to extract p timesteps where the estimation problem is well-conditioned. The injection state estimation can then be related to the design matrix and measurement matrix via the following linear system.

$$\boldsymbol{\hat{x}} = \boldsymbol{\Psi_r}\boldsymbol{\hat{a}} = \begin{cases} (\boldsymbol{C}\boldsymbol{\Psi_r})^{-1}\boldsymbol{q}^{net} & if\ P = R \\ (\boldsymbol{C}\boldsymbol{\Psi_r})^{\dagger}\boldsymbol{q}^{net} & if\ P > R \end{cases}$$

Where † is the Moore-Penrose pseudoinverse and $\boldsymbol{q}^{net}$ is the M-dimensional *net* reactive power time-series seen by the AMI.

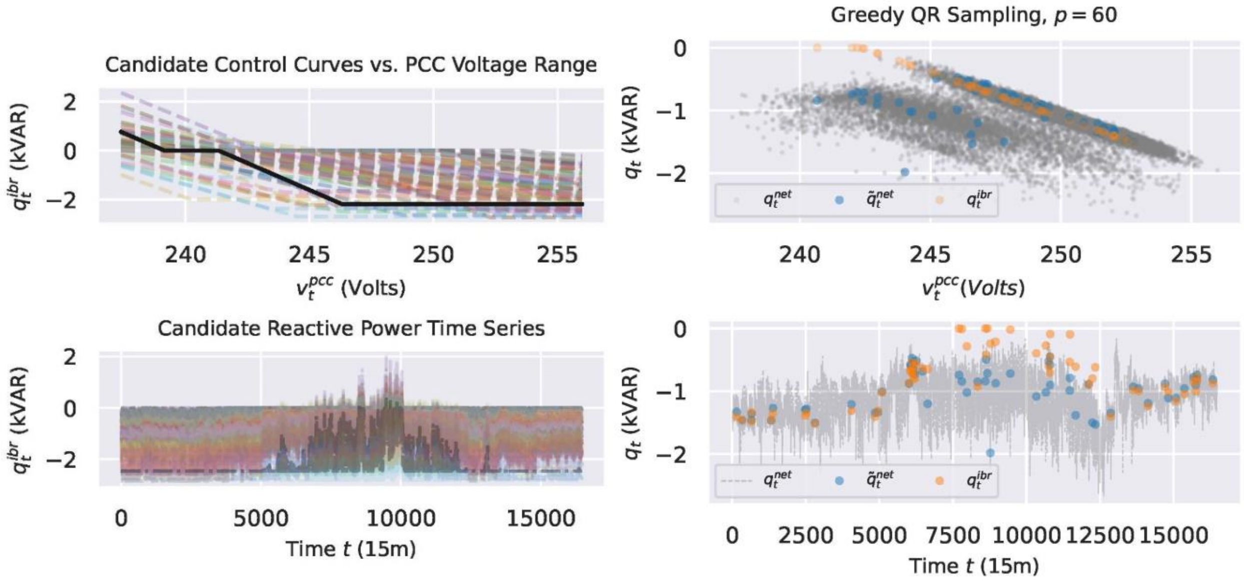


Figure 9-19. A set of candidate Volt-VAR curves as a function of PCC voltage and time (left) and the results of the sparse time-series sampling method of reconstructing the control curve response (right).

Referencing [91], a relationship between the low-rank design matrix $\boldsymbol{\Psi_r}$ sampled by the measurement matrix $\boldsymbol{C}$ can be formed with the QR decomposition. This relationship has one of the following forms, depending if the number of measurements $P = R$ or if $P > R$:

$$\begin{cases} \boldsymbol{\Psi_r^T}\boldsymbol{C}^T = \boldsymbol{QR} & if\ P = R \\ (\boldsymbol{\Psi_r}\boldsymbol{\Psi_r^T})\,\boldsymbol{C}^T = \boldsymbol{QR} & if\ P > R \end{cases}$$

In the case of Volt-VAR, the candidate control curves are evaluated over the measured PCC voltage time-series to provide a range of possible reactive power responses that could be responsible for the unknown contribution of q_t^{pv} to q_t^{net}. These columns of the data matrix are then reordered into a chronological time-series, and the preferred filtering variable within the AMI data is used a cost constraint. In the case of Volt-VAR, this was found to be the real power time-series, as discussed in the previous control curve regression section. The optimal samples selected are therefore the p timesteps measured by the AMI so that $|q_t^{pv}| > |q_t^{nat}|$. An example of this process is illustrated in Figure 9-19. The selected samples can then be input into the control curve regression procedure.

9.7. Summary of Completed Work

This chapter has presented a set of physics-based, data-driven methods for the detection and estimation of the many parameters that characterize the impacts of BTM PV on distribution grids. The presented algorithms provide a variety of methods for predicting unknown BTM PV characteristics under varying data input requirements, and with varying levels of dependence on feeder models. The diversity and scope of the algorithms presented allows for engineers and utilities to achieve the characterization of BTM PV in many analysis scenarios faced by industry.

The use cases of the algorithms can be broadly grouped into "model-derived." or "data-driven." Model-derived algorithms in all cases incorporate information from both the distribution feeder model; specifically, they use model-derived sensitivity matrices. Therefore, the algorithms that fall into this category require access to a distribution feeder model that is assumed to be accurate. In contrast, "data-driven" algorithms denote algorithms that only require AMI data inputs to accomplish the designated task. These data input categories, the algorithms, and their use cases can be summarized as follows:

1. Model-derived algorithms
    a. Location estimation of BTM PV in distribution feeders
        i. Section 9.3.1 – Location estimation without VRE impacts
        ii. Section 9.3.2 – Location estimation with VRE impacts
    b. Power factor estimation of BTM PV in distribution feeders
        i. Section 9.5.3 – Power factor estimation from voltage magnitudes
2. Data-driven algorithms
    a. Estimation of BTM PV size, tilt, and azimuth
        i. Section 9.1 – Estimation of PV size
        ii. Section 9.2 – Estimation of BTM PV tilt and azimuth
    b. Estimation of BTM PV control characteristics
        i. Section 9.4 – Summary of methods for varying AMI data access scenarios
        ii. Section 9.5 – Power factor estimation
        iii. Section 9.6. – Volt-VAR control setting estimation

The presented algorithms provide a suite of methods for characterizing the impacts of PV on distribution feeders. The varying scenarios and data input requirements considered allow the methods to be robust to a number of practical scenarios that may be faced by utilities and engineers, including cases where accurate distribution feeder models may not be available. This work resulted in several publications [82], [83], [83], [84], [87], [90], [93].

10. LOAD MODELING

10.1. Task Overview

Accurate load modeling within distribution planning tools is critical for effective system planning and equally important as the accurate representation of lines, transformers, voltage regulation equipment, and other utility assets. Growing penetration of distributed energy resources (DER) require distribution planners to assess a broader range of system conditions beyond peak load using increasingly detailed assessment methods such as quasi-static time-series (QSTS) power flow simulations and hosting capacity (HC) analyses. These assessments require more granular and accurate load models. Advanced Metering Infrastructure (AMI), and other modern utility measurement data streams, provide unprecedented visibility to distribution system loads and other conditions. However, it is not clear how conventional utility distribution load modeling practices, which are focused on representing peak load conditions based on limited measurement data, should evolve to address the emerging needs while effectively leveraging the increased visibility to loads provided by AMI and other emerging data streams.

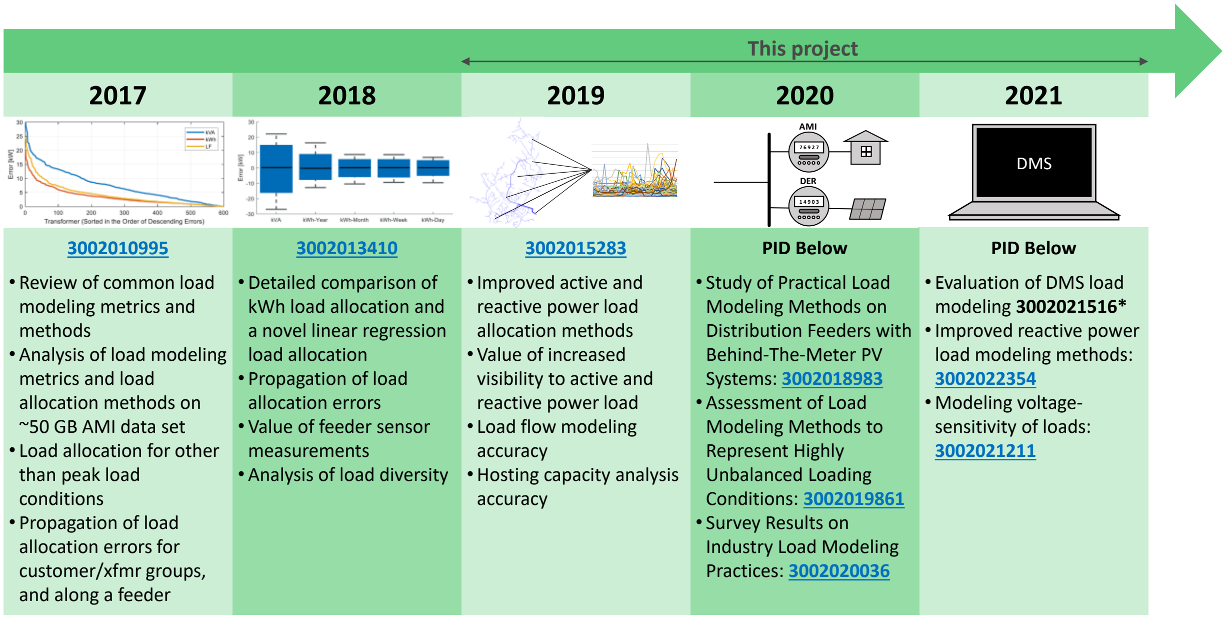
The objective of this task was to develop and evaluate improved, yet practical, distribution load modeling methods through leveraging AMI and other emerging data streams as illustrated in Figure 10-1. To this end, this task involved seven (7) research areas focusing on different aspects of distribution load modeling. The scopes of the seven research areas are summarized in Table 10-1 and discussed in more detail in the following sections.

Figure 10-1. Emerging data streams for load modeling

Table 10-1. Distribution load modeling research areas addressed in this task

Report Section	Load Modeling Research Area	Scope of the Research Area	Publication
10.2	Industry Survey	This research involved conducting an industry survey of the state-of-the-art utility load modeling practices to: 1) Identify what data utilities have available and use for distribution load modeling, 2) Recognize the ongoing methods and practices commonly applied in the industry, and 3) Detect research and development needs to support future load modeling practices. Report ID: 3002020036	[95]
10.3	Spatial and Temporal Load Modeling	In this research area, active and reactive power load modeling methods were developed and evaluated for improved spatial (across the feeder at a given time) and temporal (diverse feeder load conditions over time) representation. Report ID: 3002015283	[33]
10.4	Load Modeling with BTM PV	This research area investigated practical methods to model loads and PV generation on distribution feeders with high penetration of net metered behind-the-meter (BTM) PV systems that mask the utility load measurements. Report ID: 3002018983	[96]
10.5	Phase-Specific Load Modeling	The objective of this research area was to assess how well conventional utility load modeling methods represent highly unbalanced loading conditions, and how the methods can be improved to better represent such conditions. Report ID: 3002019861	[97]
10.6	Voltage Sensitive Load Modeling	In this research area, a new method was developed for estimating system-wide parameters of voltage-sensitive load models. In this research area, the impacts of commonly used voltage sensitive load models and model parameters were also analyzed. Report ID: 3002021211	[34]
10.7	Reactive Power Load Modeling	The objective of this research area was to 1) develop a method to disaggregate capacitor bank reactive power generation from feederhead reactive power measurements, and 2) investigate the accuracy of different reactive power load modeling methods and propose approaches to improve them. Report ID: 3002022354	[76]
10.8	Load Modeling in DMS	This research area assessed the load modeling and power flow simulation of a commercial DMS and compared them to those of a distribution planning software. Report ID: 3002021516	[98]

This task was performed by EPRI as a part of EPRI's on-going Enhanced Distribution Load Modeling effort, which has the objective of developing improved spatial and temporal distribution load modeling methods leveraging AMI and other emerging data streams. The roadmap of the effort is illustrated in Figure 10-2.



2017	2018	2019	2020	2021
3002010995	3002013410	3002015283	PID Below	PID Below
• Review of common load modeling metrics and methods • Analysis of load modeling metrics and load allocation methods on ~50 GB AMI data set • Load allocation for other than peak load conditions • Propagation of load allocation errors for customer/xfmr groups, and along a feeder	• Detailed comparison of kWh load allocation and a novel linear regression load allocation • Propagation of load allocation errors • Value of feeder sensor measurements • Analysis of load diversity	• Improved active and reactive power load allocation methods • Value of increased visibility to active and reactive power load • Load flow modeling accuracy • Hosting capacity analysis accuracy	• Study of Practical Load Modeling Methods on Distribution Feeders with Behind-The-Meter PV Systems: 3002018983 • Assessment of Load Modeling Methods to Represent Highly Unbalanced Loading Conditions: 3002019861 • Survey Results on Industry Load Modeling Practices: 3002020036	• Evaluation of DMS load modeling **3002021516*** • Improved reactive power load modeling methods: 3002022354 • Modeling voltage-sensitivity of loads: 3002021211

Figure 10-2. Roadmap of EPRI's on-going distribution load modeling research effort

10.2. State-of-the-Art Industry Survey

This research area is discussed in detail in an EPRI Technical Report [95].

In this research area, EPRI conducted a survey on distribution load modeling with utilities that participate in EPRI's Distribution Operations and Planning program. The survey, which consisted of 25 carefully selected questions, had three key objectives:

1. Identify what data utilities have available and use for distribution load modeling
2. Recognize the ongoing methods and practices commonly applied in the industry, and
3. Detect research and development needs to support future load modeling practices.

Figure 10-3 summarizes the key areas of the survey. Responses were collected from 42 distribution system planners in North America and internationally (the geographic spread is illustrated in Figure 10-4). The participants represent 26 companies across 31 states in the United States, in addition to 5 companies from Canada, South Africa, and South Korea. In this way, the survey results illustrate the diversity of conditions that are currently experienced by distribution engineers to represent load and generation conditions in their systems.

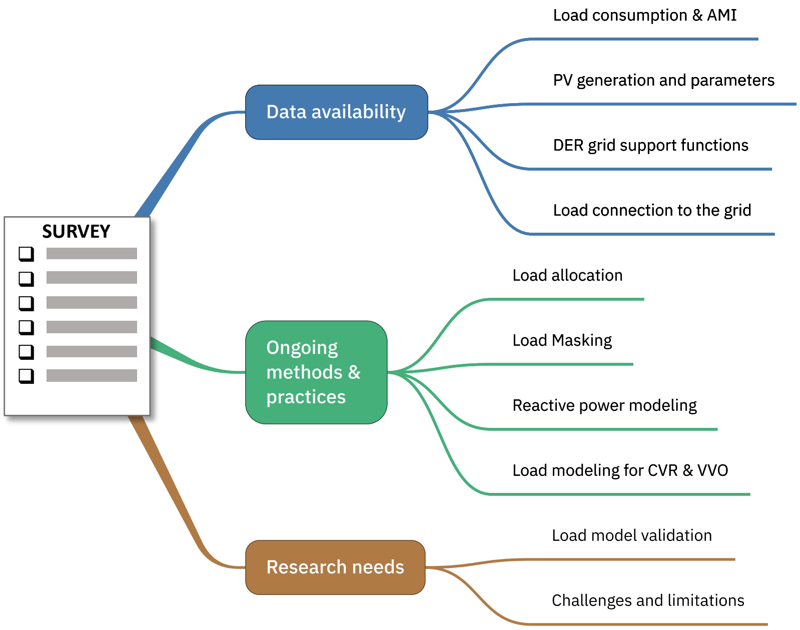

SURVEY
Data availability
Load consumption & AMI
PV generation and parameters
DER grid support functions
Load connection to the grid
Ongoing methods & practices
Load allocation
Load Masking
Reactive power modeling
Load modeling for CVR & VVO
Research needs
Load model validation
Challenges and limitations

Figure 10-3. The key areas of the load modeling survey

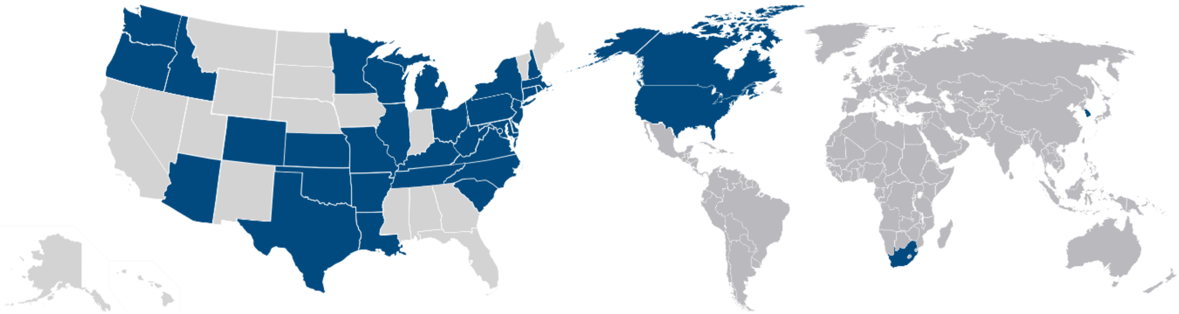


Figure 10-4. The geographic spread of the responses

The key findings from the survey were as follows:

- Data availability and usage varies based on respondents and data types
  - AMI penetration levels range from 0% to 100% of the customers with 15-minutes being the most common AMI time resolution.
  - Some load data is widely available and used. For example, load energy consumption (kWh) is available for all respondents and is used in planning studies by 30% of the respondents.
  - AMI data is more widely available than it is used in planning. For example, AMI active power (kW) measurements are available for 74% of the respondents, but only 7% use the data for planning studies. Similarly, 64% of the respondents have access to AMI voltage measurements but only 4% have the data integrated in their planning models.
  - PV system AC (kVA) rating is the most commonly available and used PV information with 78% of the respondents having it available and 22% using it in planning. PV system DC (kW) rating is also available for 70% of respondents but used in planning only by 12% of them. For most of the respondents, customer net metering measurements are available but PV system kW, kvar, or V measurements are not available.

  - Some emerging load modeling data is commonly neither available nor in use. For example, DMS state estimated load data is available to only 30% of the respondents and is used in planning by no one.
  - The survey results illustrate a potential in distribution planning to increase the utilization of data that is readily available but currently not used.
- Current load modeling practices are commonly based on load allocation
  - Loads are allocated to the peak load conditions by 74% of the respondents.
  - Load allocation is performed based on transformer kVA rating and customer kWh rating by 43% and 36% of the respondents, respectively.
  - While feeder head sensor measurements are used for load allocation by 98% of the respondents, 50% of respondents also use recloser measurements for load allocation.
- It is common to use simple or no voltage sensitivity models for loads
  - Constant power load models (load models that are not voltage sensitive) are frequently used by 82% of respondents.
  - Most (52%) respondents do not vary voltage sensitivity models of loads based on customer type.
- While planning is largely based on assessing peak load conditions, time-series assessments are emerging
  - All respondents assess peak load conditions in planning.
  - Yearly load profiles are available for 22% of the respondents and 20% of the respondents use them for planning studies.
  - Only 7% of the respondents perform 576-hour (peak and minimum load days for each month).
- Several potential improvement areas are identified to address the challenges and limitations related to current load modeling practices
  - Load masking and time-series load modeling are the most common challenges/limitations of current load modeling practices indicated by respondents.
  - Increased availability and integration of AMI data are the most common improvement needs for load modeling indicated by respondents.

10.3. Spatial and Temporal Load Modeling

This research area is discussed in detail in an EPRI Technical Report [33].

With growing penetrations of PV and other DER, there is an increasing need for more detailed representation of both spatial and temporal detail of loads. Spatial load modeling detail refers to the level of granularity to feeder load conditions for a given time instance. Spatial load modeling detail is important to accurately reflect the impact that distributed PV has on the feeder. Temporal load modeling detail refers to the level of granularity to diverse feeder load conditions over time. As the distribution impacts from PV and other DER vary greatly over time, temporal load modeling detail is important as it allows to more accurately capture DER distribution impacts.

This research area analyzed improved spatial and temporal *active and reactive power* load modeling methods. This research area was divided into two parts:

1. **Time-Granularity Analysis:** This first part focused on analyzing the impact that AMI load data time granularity has on quasi-static time-series load flow accuracy.

2. **Assessment of Spatial and Temporal Load Modeling Methods**: This second part focused on analyzing how different load modeling methods and levels of visibility to loads and DER influence load flow and hosting capacity analysis accuracy.

These two parts are contrasted in Table 10-2 and discussed in more detail in the following subsections.

Table 10-2. Two parts of spatial temporal load modeling research

Part 1: Time-Granularity Analysis	Part 2: Assessment of Spatial and Temporal Load Modeling Methods
Objective: Analyze how time granularity of load data influences load flow accuracy	**Objective**: Analyze how different load modeling methods and levels of visibility to loads and DER influence load flow and hosting capacity analysis accuracy
Compare different time granularities (1-min, 5-min, 15-min, 60-min)	Perform all analysis at 15-min time granularity
Perform all load modeling through 100% AMI penetration without DER	Compare different load modeling methods and visibility levels
Analyze scenarios without existing PV	Analyze scenarios without and with existing PV
Use Pecan Street data set (1-min)	Use real utility AMI and PV data (15-min)

10.3.1. Time-Granularity Analysis

Utilities frequently record AMI data at 1-hour or 15-min resolution. Few, if any, utilities currently record AMI data at higher time granularity. There is limited understanding in how time granularity of AMI data influences load flow modeling accuracy and what granularity of AMI data would be needed for a given desired level of accuracy. To shed light into this, this first part of the spatial-temporal load modeling research area analyzed the impact that load data time granularity has on QSTS simulation accuracy.

This assessment was performed on a modified version of the EPRI Ckt5 feeder model with 1-minute AMI load profiles introduced in Section 3.1.1.1. Annual QSTS simulations were run at different time granularities (resolutions) to assess how the time granularity impacts the accuracy nodal voltages or element currents simulated with QSTS. The higher time granularities resulted in very large analysis input data and result data sets as illustrated in Table 10-3. In total, more than 4.6 billion rows of result data were processed and analyzed as a part of the time granularity analysis. The quantity of results data clearly illustrates the challenges introduced by the emerging distribution system data sets and increasingly granular assessment methods.

Table 10-3. The size of the time granularity analysis results data sets

	hourly	15-min	5-min	1-min
Annual Time-steps	8,760	35,040	105,120	525,600
Monitored Buses	3,437			
Monitored Sections	2866			
Monitored Transformers	594			
Resolution-based Data Points	6.04e7	2.42e8	7.25e8	3.63e9
Total Data Points	**4,652,164,440**			

The QSTS simulation accuracies were analyzed separately for nodes/transformers/lines in the medium voltage primary and low-voltage secondary networks. Table 10-4 provides a very high-level view of the time granularity results data set by showing the average (over all nodes/elements in a given category) errors in representing the worst-case (over time) conditions for the different time granularities. Clearly, coarser time granularities lead into higher QSTS simulation errors. Curiously, the errors roughly doubled as time granularity was decreased from 5-min to 15-min and from 15-min to 60-min. Moreover, the accuracy at the edge of the system were more affected by the granularity of the AMI data. A very high-level guideline to distribution utilities would be to aim to have 15-min time granularity or better for medium voltage related applications and 5-min time granularity or better for low voltage related applications. However, the recommended/necessary time granularity depends on the accuracy requirements of a given application.

Table 10-4. Average errors in representing the worst-case conditions for the analyzed time granularities

Location	Quantity Analyzed	5-min	15-min	60-min
Medium Voltage Nodes	Voltage (pu)	+0.0006 (1.0038)	+0.0013 (1.0038)	+0.0027 (1.0038)
Service Transformer Low-Voltage Nodes		+0.0015 (0.9928)	+0.0033 (0.9928)	+0.0059 (0.9928)
Low-Voltage Customer Nodes		+0.0019 (0.9879)	+0.0041 (0.9879)	+0.0070 (0.9879)
Medium Voltage Lines	Element Loading (% of Rated)	-0.23 (8.66)	-0.48 (8.66)	-0.84 (8.66)
Low-Voltage Service Lines		-2.99 (23.46)	-5.62 (23.46)	-9.25 (23.46)
Low-Voltage Service Transformers		-7.01 (65.71)	-13.28 (65.71)	-21.58 (65.71)

The key take-away of the time-granularity analysis were as follows:

- Time granularity of AMI data can considerably influence QSTS simulation accuracy.
- The recommended time granularity depends on the application of interest and the desired level of accuracy.
  - For example, if the focus of the analysis is on the medium voltage, sufficient accuracy may be achieved with 15-min data.
  - On the other hand, if the focus is on the low voltage elements, data with 5-min or finer time granularity may be necessary to capture individual spikes in the demand that could create extreme conditions.
- The errors are higher at high loading conditions compared to low loading conditions because of the higher magnitude in element loading being smoothed out by the coarser resolution data.
- Load modeling with AMI data provides fairly good approximation of the behavior of the feeder no matter which resolution, particularly as compared to conventional utility load modeling practices.
- The spatial-temporal load modeling analysis attempted to inform what is missed when AMI data is not available or utilized.

10.3.2. Assessment of Spatial-Temporal Load Modeling Methods

The second part of the spatial-temporal load modeling assessed how different load modeling methods and levels of visibility to loads and DER influence load flow and hosting capacity analysis accuracy. The assessment involved the following four steps:

- Step 1: Simulate the physical system utilizing an AMI active and reactive power profile for each load.
- Step 2: Prepare synthetic "field measurement" data necessary for the given load modeling method. The synthetic field measurement data was obtained from the simulation performed using the utility AMI active and reactive power profiles.
- Step 3: Model loads through load allocation or other method.
- Step 4: Perform QSTS and hosting capacity simulations with the utilized load modeling method and quantify the accuracy of the given method.

Table 10-5 summarizes the spatial and temporal load modeling cases that were evaluated.

Table 10-5. Spatial and temporal load modeling cases analyzed shaded by level of detail (green: high detail, red: low detail)

Objective	Case Abbreviation	Details	Load Active Power Measurements		Load Reactive Power Measurements		SCADA Visibility (kW & Amps)	Load Allocation Frequency
			Temporal Visibility	Spatial Visibility	Temporal Visibility	Spatial Visibility		
Ref.	Ref (BaseCase)	Loads modeled directly with their AMI active and reactive power data	15-min kW	Load-specific kW	15-min kvar	Load-specific kvar	N/A	N/A
Allocation Frequency	BAU (ALBAU)	Business-as-Usual: Load allocation for the annual peak load	Peak month kWh	Load-specific kWh	Yearly average PF	System-wide average PF	Feeder head	Peak month
	Monthly (ALMonthly)	Load allocation for the peak load of each month	Monthly kWh	Load-specific kWh	Yearly average PF	System-wide average PF	Feeder head	Month
	Time-wise (AL15Min)	Load allocation for each 15-min instant	Monthly kWh	Load-specific kWh	Yearly average PF	System-wide average PF	Feeder head	15-min
Sensors	BAU_S (ALBAU_S)	BAU case with 3 additional feeder sensors	Peak month kWh	Load-specific kWh	Yearly average PF	System-wide average PF	Feeder head & 3 feeder sensors	Peak month
	BAU_TS (ALBAU_TS)	BAU case with AMI data aggregated to each service transformer	Peak month kWh	Load-specific kWh	Yearly average PF	System-wide average PF	All service transformers	N/A
	Ref_PAMI (BaseCase_PAMI)	AMI kW data only (no AMI kvar data)	15-min kW	Load-specific kW	Yearly average PF	System-wide average PF	N/A	N/A
Reactive Power Modeling	BAU_FPF (ALBAU_FPF)	BAU case with feeder head reactive power profile	Peak month kWh	Load-specific kWh	15-min kvar	Feeder head kvar	Feeder head	Peak month
	BAU_CYPF (ALBAU_CYPF)	BAU case with customer-specific yearly average power factors	Peak month kWh	Load-specific kWh	Yearly average PF	Load-specific PF	Feeder head	Peak month
	Monthly_CYPF (ALMonthly_CYPF)	Monthly case with yearly average power factor of each customer	Monthly kWh	Load-specific kWh	Yearly average PF	Load-specific PF	Feeder head	Month
	Monthly_CMPF (ALBAU_CMPF)	Monthly case with monthly average power factor of each customer	Monthly kWh	Load-specific kWh	Monthly average PF	Load-specific PF	Feeder head	Month

The assessment was conducted on a modified version of the EPRI Ckt5 feeder model with 1-Hour AMI load profiles introduced in Section 3.1.1.2. The QSTS simulation results (voltage and thermal loading at different levels on the feeder) from the analyzed spatial temporal load modeling cases are summarized in Table 10-6.

Table 10-6. Summary of spatial and temporal load modeling QSTS simulation results

Objective	Case Name (Case Abbreviation)	Details	Lowest Node Voltage Magnitudes [p.u.]			Highest Thermal Loading [% of Rating]		
			MV Nodes (MV)	LV Nodes of Xfmrs (LV1)	Customer Nodes (LV2)	MV Lines	LV Lines	Xfmrs
Ref.	Ref (BaseCase)	Loads modeled directly with their AMI active and reactive power data	0.9888	0.9779	0.9720	12.40	26.81	70.04
Allocation Frequency	BAU (ALBAU)	Business-as-Usual: Load allocation for the annual peak load	+0.0025	+0.0048	+0.0062	-0.73	-14.28	-26.98
	Monthly (ALMonthly)	Load allocation for the peak load of each month	+0.0025	+0.0047	+0.0060	-0.70	-13.62	-25.80
	Time-wise (AL15Min)	Load allocation for each 15-min instant	+0.0019	+0.0040	+0.0053	-0.61	-13.50	-25.42
Sensors	BAU_S (ALBAU_S)	BAU case with 3 additional feeder sensors	+0.0024	+0.0047	+0.0059	-0.62	-14.16	-26.56
	BAU_TS (ALBAU_TS)	BAU case with AMI data aggregated to each service transformer	+0.0022	+0.0023	+0.0026	-0.09	-6.98	-0.06
	Ref_PAMI (BaseCase_PAMI)	AMI kW data only (no AMI kvar data)	+0.0022	+0.0023	+0.0024	-0.08	+0.25	+0.19
Reactive Power Modeling	BAU_FPF (ALBAU_FPF)	BAU case with feeder head reactive power profile	-0.0040	-0.0025	-0.0013	-0.46	-13.99	-25.98
	BAU_CYPF (ALBAU_CYPF)	BAU case with customer-specific yearly average power factors	+0.0036	+0.0060	+0.0073	-0.73	-14.23	-26.92
	Monthly_CYPF (ALMonthly_CYPF)	Monthly case with yearly average power factor of each customer	+0.0036	+0.0058	+0.0071	-0.70	-13.57	-25.72
	Monthly_CMPF (ALBAU_CMPF)	Monthly case with monthly average power factor of each customer	+0.0001	+0.0021	+0.0033	-0.70	-13.56	-25.70

The key take-away from the QSTS analysis were:

- Load modeling using real power measurement and a fixed power factor can provide relatively adequate results at the medium voltage level.
- Having a fixed power factor will impact the way individual loads are modeled and thus the results from secondaries (low voltage).
- The frequency of the real power (with fixed PF) load allocation show only minor improvements.
- Additional sensors with real power measurement can improve load modeling accuracy. In particular, sensors at the service transformer or customer level noticeably improve the accuracy.

In addition to QSTS analysis, the accuracy of the 12 considered spatial temporal load modeling cases were quantified with hosting capacity analysis. Five feeder locations were chosen for evaluating hosting capacity:

- Location A: Single-phase MV bus associated with the LV bus with the highest voltage in the QSTS simulation base results.
- Location B: Single-phase MV bus associated with the LV bus with high voltage magnitudes in the QSTS simulation.
- Location C: Single-phase MV bus at the feeder end.
- Location D: Three-phase MV bus at the mid feeder.
- Location E: Single-phase LV bus with the highest voltage magnitude.

The five hosting capacity locations of interest represent a range of feeder hosting capacity characteristics that were used to evaluate and compare alternative load modeling methods. The five locations are illustrated in Figure 10-5. For each location, hosting capacity was analyzed against steady-state voltage limits defined in ANSI C84.1 (+/- 5%) and the thermal constraints of each line and transformer as defined in the feeder model.

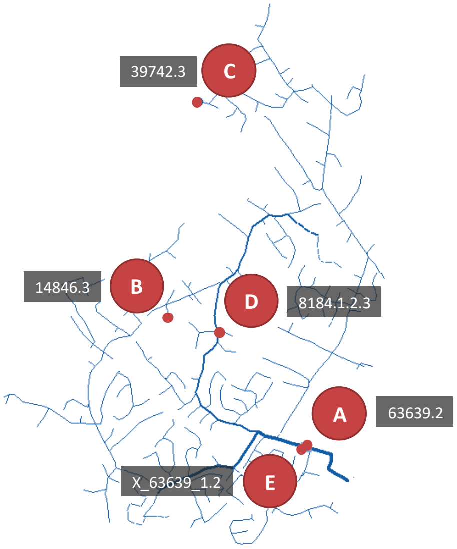


Figure 10-5. The five locations of interest evaluated in the HC analysis

Figure 10-6 illustrates the nature of hosting capacity results by showing the percentiles of the daily hosting capacity errors in colors for the ALBAU load modeling case at location D (base case hosting capacity results are shown in grey). Here, ALBAU method overestimates the hosting capacity as it does not properly capture the diversity of loads. Figure 10-7 illustrates the annual hosting capacity duration curves for selected methods at location D.

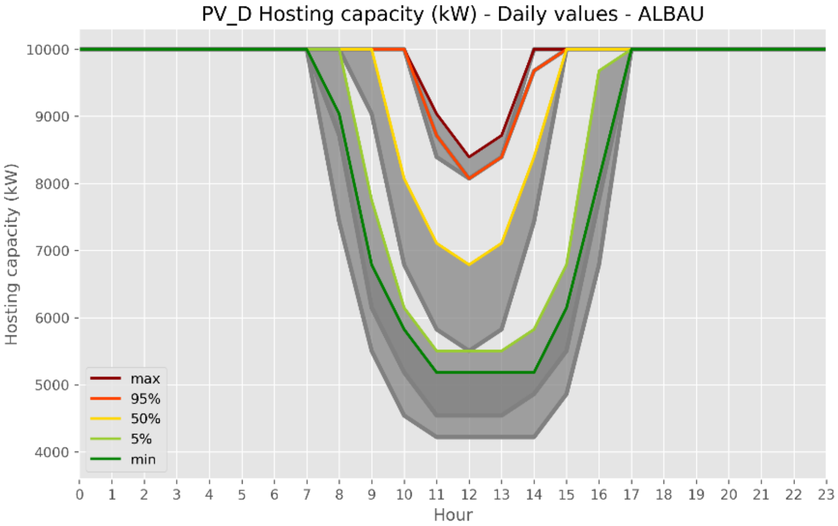


Figure 10-6. Example hosting capacity results: percentiles of the daily hosting capacity errors for the ALBAU load modeling case at location D

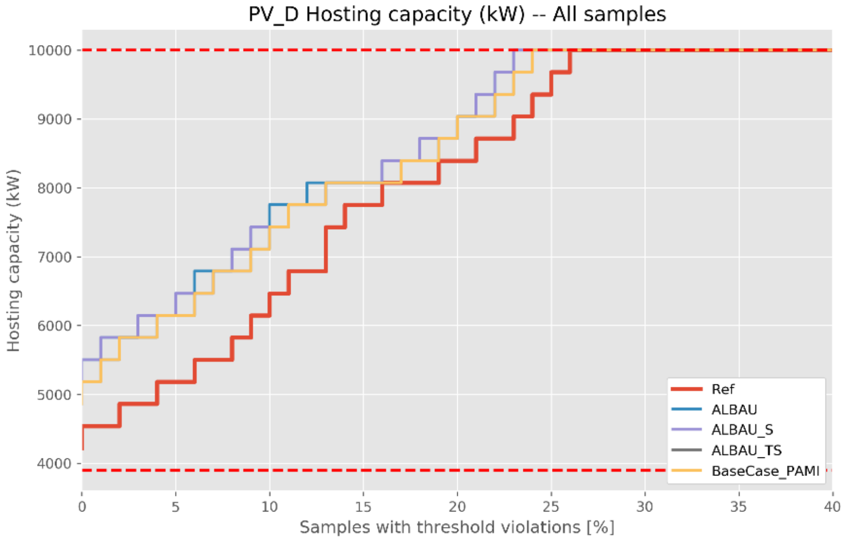


Figure 10-7. Example hosting capacity results: annual hosting capacity duration curve for selected cases at location D

Table 10-7 summarizes the hosting capacity results of the spatial temporal load modeling analysis. While the results have many details, a general conclusion of the results follows:

- Hosting capacity accuracy depends on the load modeling method and the feeder location.
- Spatial visibility to loads (AMI kW or transformer sensors) and more granular reactive power modeling (e.g. feeder power factor) tend to improve the accuracy of the hosting capacity analysis.

Table 10-7. Summary of spatial and temporal load modeling hosting capacity results

Objective	Case Name (Case Abbreviation)	Difference in Minimum Hosting Capacity [% with respect to reference HC]				
		PV Location A	PV Location B	PV Location C	PV Location D	PV Location E
Allocation frequency	Business-As-Usual (ALBAU)	31.6%	40.0%	25.0%	22.8%	25.0%
	Monthly Load Allocation (ALMonthly)	26.3%	40.0%	25.0%	22.8%	25.0%
	Step-wise Load Allocation (AL15Min)	31.6%	20.0%	25.0%	15.2%	25.0%
Sensors	Branch sensors (ALBAU_S)	31.6%	40.0%	25.0%	22.8%	25.0%
	Transformer sensors (ALBAU_TS)	10.5%	20.0%	25.0%	15.2%	25.0%
	Real Power AMI (BaseCase_PAMI)	10.5%	20.0%	25.0%	15.2%	25.0%
Reactive power modeling	BAU: Feeder head step-wise PF (ALBAU_FPF)	31.6%	20.0%	25.0%	7.6%	25.0%
	BAU: Customer-specific yearly avg PF (ALBAU_CYPF)	31.6%	20.0%	25.0%	22.8%	25.0%
	Monthly: Cust-specific yearly avg PF (ALMonthly_CYPF)	26.3%	40.0%	25.0%	15.2%	25.0%
	Monthly: Cust-specific monthly avg PF (ALMonthly_CMPF)	31.6%	20.0%	25.0%	15.2%	25.0%

Finally, Table 10-8 shows the computational performance and the data requirements for the 12 spatial temporal load modeling cases. The computational time of each method is in the same order of magnitude although there are some differences between the methods. Some methods also result in many more load allocation factors but this may or may not be a concern depending on how the load allocation method would be implemented in simulation software. The key difference between the analyzed methods is in the data requirements. Some methods require several orders of magnitude more input data that must be accessed, stored, and prepared for the analysis. This is not straightforward and distribution utilities and their data sets may not be prepared for such a daunting task. Hence, it is important to consider where the added data provides value that justifies the effort required.

Table 10-8. Computational performance and data requirements for the spatial temporal load modeling cases

Objective	Case Name (Case Abbreviation)	QSTS Time [s]	HC Time [h]	Load Data Points	Allocation Factors
Reference	Reference (BaseCase)	3,587	8.24	96,640,320	-
Allocation Frequency	Business-As-Usual (ALBAU)	3,573	7.48	35,040	1,379
	Monthly Load Allocation (ALMonthly)	3,910	8.49	35,040	16,548
	Step-wise Load Allocation (AL15Min)	3,928	9.00	35,040	48,320,160
Sensors	Branch sensors (ALBAU_S)	3,574	7.51	140,160	1,379
	Transformer sensors (ALBAU_TS)	4,039	9.51	20,463,360	1,379
	Real Power AMI (BaseCase_PAMI)	3,573	6.12	48,320,160	-
Reactive power modeling	BAU: Feederhead step-wise PF (ALBAU_FPF)	4,188	16.65	70,080	1,379
	BAU: Customer-specific yearly avg PF (ALBAU_CYPF)	3,647	7.73	35,040	1,379
	Monthly: Cust-specific yearly avg PF (ALMonthly_CYPF)	3,565	8.12	35,040	16,548
	Monthly: Cust-specific monthly avg PF (ALMonthly_CMPF)	3,559	8.01	35,040	16,548

10.4. Load Modeling with BTM PV

This research area is discussed in detail in an EPRI Technical Report [96].

Many distribution utilities have limited to no visibility to the operation of behind the meter (BTM) PV. Most utilities have only one meter per customer, metering the customer net load and providing limited visibility to the customer native load masked by PV generation. This limited visibility makes accurate load modeling difficult. High penetration of single-phase BTM PV also has the potential to increase magnitude and variability of the load unbalance, which makes accurate load modeling particularly important [99].

This research area investigated practical methods to model loads and PV generation on distribution feeders with high penetration of *net metered* BTM PV systems that mask the utility load measurements. In particular, this research area assessed five cases with different load modeling approaches that are summarized in Table 10-9 and motivated below.

- **Case 1**: A separate meter was assumed for loads and BTM PV systems. In this case, loads and PV generation were directly modeled through AMI active and reactive power measurements. Case 1 was used as a reference case to quantify the accuracy of Cases 2-5.
- **Case 2**: The BTM PV systems are not modeled explicitly, and the loads are allocated based on customer net-metered kWh and scaled based on feeder net active power profile.
- **Case 3**: The loads are modeled identically to Case 2, but the BTM PV systems are modeled explicitly based on the PV system AC and DC ratings, and a generic DC-power generation profile.

- **Case 4**: The BTM PV systems are modeled identically to Case 3, but the loads are scaled based on feeder estimated native active power profile, as opposed to the feeder net active power profile. In Case 4, loads are still allocated based on customer net load kWh.
- **Case 5**: The BTM PV is modeled as in Cases 3 and 4, and the loads are allocated based on estimated customer native load kWh. In Case 5, loads are scaled identically to Case 4. Case 5 investigated the value of allocating loads based on estimated native load kWh as opposed to net load kWh.

Table 10-9. The five analyzed approaches to model loads on feeders with high penetration of BTM PV shaded with the color of detail (green: high, red: low)

Case #	Customer Measurement Used for Load Allocation (kWh)	Feeder Head Measurement Used for Load Allocation	Approach for Load Scaling in QSTS Simulation and TSHC Analysis	Approach for Reactive Power Load Modeling	Approach for PV System Modeling
Case 1	-	-	AMI data	AMI data	AMI data
Case 2	Net metering kWh	Net phase currents at feeder head	Net active power at feeder head	Computed from net FH measurements	No model
Case 3	Net metering kWh	Net phase currents at feeder head	Net active power at feeder head	Computed from net FH measurements	PV systems following a time-series profile
Case 4	Net metering kWh	Net phase currents at feeder head	Approx. native load at feeder head	Computed from approx. native load at feeder head	PV systems following a time-series profile
Case 5	Approx. native load kWh	Approx. native load FH power	Approx. native load at feeder head	Computed from approx. native load at feeder head	PV systems following a time-series profile

The analysis was performed on a modified version of the EPRI Ckt5 distribution feeder with a high penetration of BTM PV systems introduced in Section 3.1.1.3. For each Case 1-5, quasi-static time-series (QSTS) power flow simulations and time-series hosting capacity (TSHC) analysis was first performed and then, the accuracy of Cases 2-5 was quantified using Case 1 as a reference. For QSTS simulations, the accuracy of the maximum and minimum voltage magnitudes, and the maximum thermal loadings caused by forward and reverse power flow were analyzed. For TSHC, both the minimum and percentiles of the annual hosting capacity were analyzed.

The key take-aways from this analysis were as follows:

- Scaling the allocated loads for time-series assessments based on feeder native active power estimated based on SCADA measurements and estimated PV generation profile (Case 4) resulted in considerably better accuracy as compared to scaling the loads based on the net load profile (Cases 2 and 3).
- Cases 2 and 3 cannot be easily extended to evaluating future scenarios (e.g., load growth) or what-if studies (e.g., smart inverter functions) given that these methods do not accurately model native load and PV generation.
- Modeling loads with net-metering data *while also* modeling PV systems (Case 3) leads to double counting the feeder PV generation and thus, considerably over-estimating PV impacts.
- Modeling loads with net-metering data without modeling PV systems (Case 2) can reasonably well represent the *current* (historical) feeder conditions. However, this approach under-estimates PV impacts particularly at the low-voltage secondary circuits.

- Allocating loads based on net-metering kWh for each customer (Case 4) resulted in the best approximation of the HC across the five feeder hosting capacity locations of interest by capturing not only the minimum value but also other representative percentiles in the year-long hosting capacity analysis.
- Allocating the loads based on customer estimated native load kWh (Case 5) resulted in higher load values for customers with PV systems, as compared to allocating the loads based on customer net load kWh (Case 4). Conversely, customers without a PV system were allocated smaller load values.
- Allocating loads based on estimated native load monthly kWh for each customer (Case 5) did not considerably improve the analyzed metrics as compared to allocating the loads based on net-metering kWh for each customer (Case 4). This could be partially explained by the feeder peak load (for which loads were allocated) occurring during evening hours when PV generation was small. Larger differences between Cases 4 and 5 may be obtained for feeders with peak load occurring during higher PV generation times. The performance of these two approaches can also differ more for other metrics that were not analyzed in this research area.

10.5. Phase-Specific Load Modeling

This research area is discussed in detail in an EPRI Technical Report [97].

The objective of this research was to assess how well conventional utility load modeling methods represent highly unbalanced loading conditions, and how the methods can be improved to better represent such conditions. To this end, eight approaches to represent highly unbalanced feeder loading conditions, which are summarized in Table 10-10, were analyzed as a part of this research area.

Table 10-10. Eight analyzed approaches to model loads on feeders with highly unbalanced loading shaded with the color of detail (green: high, red: low)

Case #	Case Name	Feeder Model	Feederhead Amps for Load Allocation	Feederhead kW for Scaling Loads	Reactive Power Modeling	Motivation
1	Ref	Unbalanced	N/A	AMI P	AMI Q	Reference case
2	Ref Bal	Balanced	N/A	AMI P	AMI Q	Accuracy of balanced circuit modeling
3	BAU	Unbalanced	Phase-Avg	Total kW	Total kvar (ALBAU_FPF)	Accuracy of BAU
4	BAU Bal	Balanced	Phase-Avg	Total kW	Total kvar (ALBAU_FPF)	Accuracy of BAU with balanced circuit modeling
5	BAU 3ph-I	Unbalanced	Phase-Specific	Total kW	Total kvar (ALBAU_FPF)	Value of phase currents in load allocation
6	BAU 3ph-P	Unbalanced	Phase-Avg	Phase kWs	Total kvar (ALBAU_FPF)	Value of phase-specific kW profiles
7	BAU 3ph-I-P	Unbalanced	Phase-Specific	Phase kWs	Total kvar (ALBAU_FPF)	Combined value of phase currents in load allocation & phase-specific kW profiles
8	BAU 3ph-I-P-PF	Unbalanced	Phase-Specific	Phase kWs	Phase-specific power factors	Combined value of phase currents in load allocation & phase-specific kW & kvar profiles

For this research area, a modified version of the EPRI Ckt5 feeder model was implemented with a significant, yet realistic, level of unbalance. Figure 10-8 shows the voltage profiles of the original and modified EPRI Ckt5 feeder models during a high voltage unbalance time instance. As is evident from Figure 10-8, the modified feeder model has considerably higher voltage unbalance, as desired.

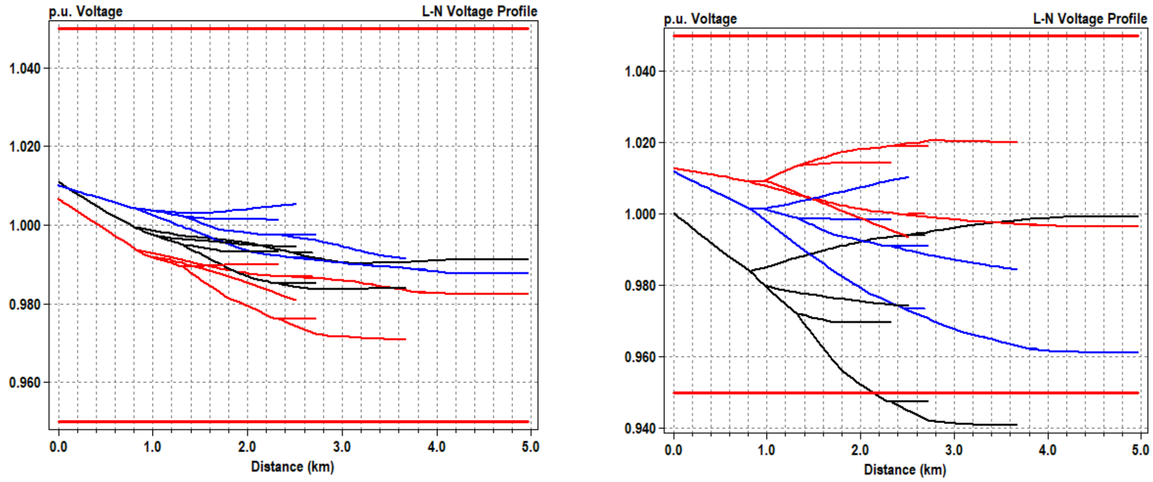


Figure 10-8. The voltage profiles of the original EPRI Ckt5 feeder model (left) and the modified version with high unbalance (right) during a high voltage unbalance time instance. Each color represents voltage magnitudes for a given phase in the model.

The modified version of EPRI Ckt5 OpenDSS feeder model was used to analyze the accuracy of load modeling Cases 1, 3, and 5-8. However, to analyze the accuracy of balanced circuit modeling in Cases 2 and 4, a balanced version of the feeder model was created in OpenDSS. In the process of creating the balanced feeder model, a bug in OpenDSS was detected and fixed in its source code. Prior to fixing this bug, the balanced models created with the software produced incorrect impedance values for some types of conductors and thus, resulted in incorrect voltage profiles.

Table 10-11 lists the high-level summary of the QSTS simulation results for the eight analyzed load modeling cases. While the results are discussed in detail in the published EPRI report, key findings can be summarized as follows.

- The load modeling approaches based on load allocation (Cases 3-8)
  - Captured feederhead active power fairly accurately.
  - Overestimated feeder reactive power consumption (due to double counting the losses).
  - Underestimated the node-average time-minimum 3ph-bus L-L voltages.
  - All methods (except those based on balanced circuit model) had comparable accuracy in capturing the time-maximum element loadings.
  - Did not accurately capture service transformer and service line loadings.

  These findings match with load modeling analysis presented in Section 10.3.

- Balanced circuit modeling (Cases 2 and 4)
  - Did not capture any current or voltage unbalance and hence, significantly underestimated the voltage drops over time and across the feeder nodes, but particularly at the low-voltage level. The time-minimum voltages were underestimated on average by 0.01 – 0.026 p.u.
  - Represented less accurately feederhead phase active and reactive powers.
  - Captured less accurately the time-maximum MV line loadings but was comparably accurate to unbalanced modeling in capturing LV line and transformer loadings.

  - For these reasons, balanced circuit modeling is not recommended particularly on feeders with high load unbalance.
- Using phase-specific currents in load allocation (Cases 5, 7 and 8) as compared to using phase-average currents (Case 3 and 6)
  - Represented better the range of L-N and L-L voltage magnitudes (more accurate voltage profile and/or more accurate range between phases and phase-phase pairs).
  - Represented better the feederhead L-N and L-L voltage unbalance.
- Using phase-specific load profiles for scaling loads in QSTS simulation (Cases 6, 7 and 8) as compared to using phase-average load profiles (Cases 3 and 5)
  - Using phase-specific load profiles in QSTS simulation without phase-specific currents in load allocation (Case 6) captured feederhead phase powers, and feeder current and voltage unbalance was worse than using phase-specific currents in load allocation without phase-specific load profiles in QSTS simulation (Case 5).
  - If the loads were allocated based on phase-specific currents, scaling loads in QSTS simulation based on phase-specific load profiles (Case 7 vs. Case 5) resulted in minimal difference/improvement in the accuracy of simulated time-minimum node voltages. This may be explained by the load allocation time instance (feeder peak load) capturing the node time-minimum voltages fairly accurately. Phase-specific load profiles may have a higher benefit on others' metrics.
- Best overall accuracy was obtained with using phase-specific currents in load allocation and phase-specific load profiles in QSTS simulation (Case 7 and 8). Using phase-specific power factor profiles (Case 8 vs. Case 7) did not provide clear improvement for the metrics analyzed. More noticeable differences may be experienced for metrics not analyzed here.

Table 10-11. Summary results for the eight analyzed approaches to model loads on feeders with highly unbalanced loading

Case Name	Bus-Average of Time-Minimum Voltages - Difference to Ref Case [p.u.]			Average of 3ph Buses L-L Time-Min Voltages - Difference to Ref Case [p.u.]	Element-Average of Time-Max Loadings - Difference to Ref Case [% of NormAmps]		
	MV Buses	Xfmr LV Buses	Customer LV Buses		MV Lines	LV Lines	Xfmrs
Ref	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Ref Bal	0.009	0.021	0.026		4.542	0.154	0.784
BAU	0.005	0.003	0.002	0.006	0.245	13.934	25.471
BAU Bal	0.003	0.015	0.020		4.412	13.962	25.384
BAU 3ph-I	0.007	0.005	0.004	0.006	0.322	13.925	25.515
BAU 3ph-P	0.006	0.004	0.002	0.006	0.187	13.865	25.237
BAU 3ph-I-P	0.007	0.006	0.005	0.006	0.272	13.865	25.310
BAU 3ph-I-P-PF	0.007	0.006	0.005	0.006	0.271	13.860	25.296

Time-series hosting capacity was performed with QSTS simulations quantifying the maximum rated size of a single PV system that can be accommodated at one of five feeder locations of interest without exceeding defined thresholds at any time during the year. Figure 10-9 illustrates the aggregated magnitudes of the hosting capacity errors for Cases 3, 5, 6, 7 and 8[10]. As can be seen in the figure, total values for feeder head (FH) active power and power factor (Case 5) resulted in the same percent error magnitude as using phase-specific values for FH active power and power

[10] HC was not analyzed for Cases 2 and 4 for which the QSTS simulation results demonstrated significant error levels.

factor (Case 8). Cases with phase-specific FH active power and total FH power factor (Cases 6 and 7) resulted in the biggest aggregated error for hosting capacity calculation across the three-phase and single-phase locations.

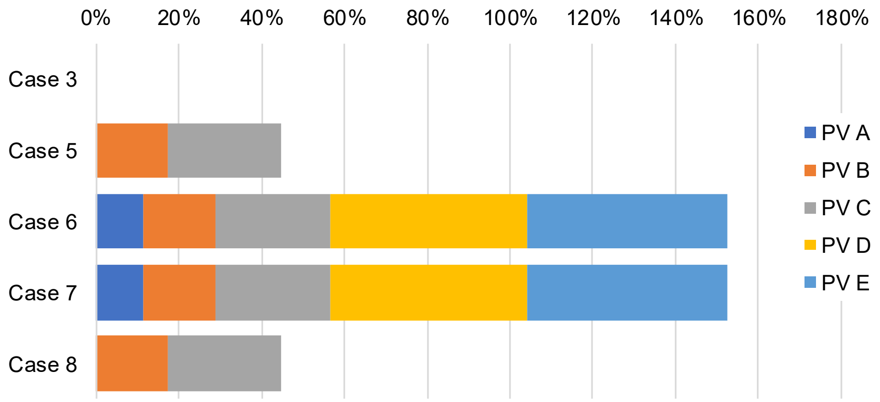


Figure 10-9. Aggregated magnitude of HC errors

Separate from the analysis above, this research area also analyzed how accurately the 10 load modeling methods discussed in Section 10.3 represent the current and voltage unbalance. The key findings of this analysis were as follows.

- Modelling loads with the AMI active power or with transformer-level active power measurements were most accurate in capturing time-maximum current and voltage unbalances.
- Methods based on allocating the loads for annual peak or monthly peak were least accurate in capturing time-maximum current and voltage unbalances.
- As compared to allocating loads to annual/monthly peak, performing load allocation for each time instance yielded slightly more accurate time-maximum current and voltage unbalances.
- As compared to using feederhead sensor only, leveraging feeder sensors yielded more accurate time-maximum current unbalances but negligible improvement in time-maximum voltage unbalances.

10.6. Voltage Sensitive Load Modeling

This research area is discussed in detail in an EPRI Technical Report [34], and had two parts:

1. Develop a new method for estimating system-wide parameters of voltage-sensitive load models, and
2. Analyze the impacts of commonly used voltage sensitive load models and model parameters.

These two parts are discussed in the following two subsections.

10.6.1. Estimating the Parameters of Voltage Sensitive Load Models

A method to estimate voltage-sensitive model parameters, [100], based on AMI data was developed. The method, which is illustrated in Figure 10-10, starts by processing customer by customer AMI active power, reactive power, and voltage measurement profiles, filtering voltage change events from them, and sampling a measurement point before and after each voltage change event. Then, exponential model parameters n_p and n_q are estimated with ordinary least squares (OLS) linear regression for each voltage change event for each customer. Next, the estimated parameters are combined to the feeder level as a weighted average based on customer energy consumption. Last, statistical representative values for the parameters are selected for the entire feeder.

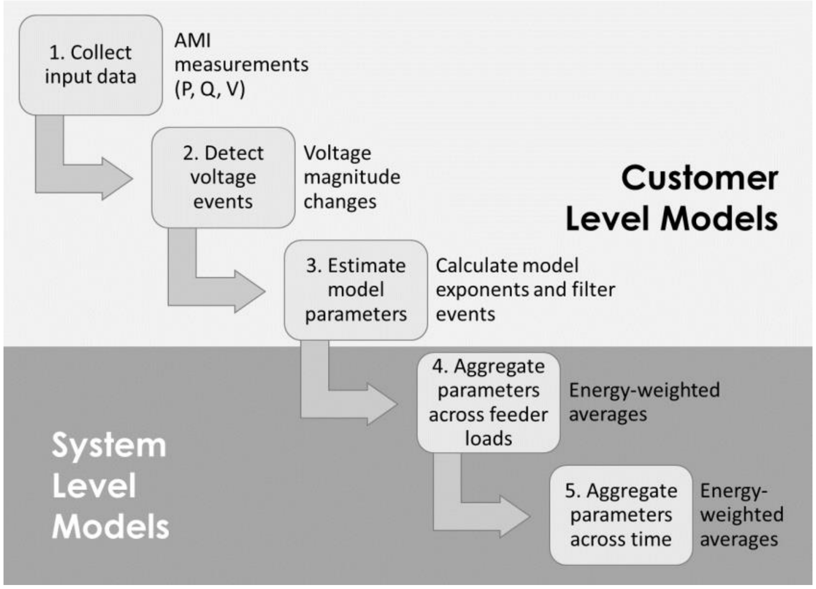

1. Collect input data
AMI measurements (P, Q, V)
2. Detect voltage events
Voltage magnitude changes
Customer Level Models
3. Estimate model parameters
Calculate model exponents and filter events
4. Aggregate parameters across feeder loads
Energy-weighted averages
System Level Models
5. Aggregate parameters across time
Energy-weighted averages

Figure 10-10. Method to estimate the parameters of voltage-sensitive load models

The developed parameter estimation method was tested on a utility AMI data set resulting in the following feeder-level parameter estimates: $n_p = 3.32$ and $n_q = 4.29$. These parameters were somewhat high compared to values reported in the literature but still within the expected ranges. While it was not possible to validate the accuracy of the estimated parameters, their impact on QSTS simulation was analyzed as a part of the analysis discussed in the following subsection.

10.6.2. Analysis of Typical Voltage-Sensitive Load Models

Selected voltage-sensitive load models and model parameters commonly used in the industry were analyzed. The models and model parameters were selected based on a review of literature and distribution modeling software. The following five voltage-sensitive load model-parameter configurations were analyzed:

- Case 1: ZIP model with constant power
- Case 2: ZIP model with constant current
- Case 3: ZIP model with constant impedance
- Case 4: Exponential model with sensitive values ($n_p = 1.7$, $n_q = 4$)
- Case 5: Exponential model with estimated parameters ($n_p = 3.3$, $n_q = 4.3$)

Cases 1-3 were selected to capture the corner case impacts of the ZIP model, Case 4 to capture the voltage sensitive exponential model, and Case 5 to capture the impact of the exponential model with the parameters estimated with the developed method. For each case, an annual QSTS simulation was performed on the EPRI Ckt5 feeder model. To focus the analysis on the impacts of the voltage-sensitive load models and parameters, the loads were modeled in the QSTS simulations with their AMI active and reactive power recordings. For each case, the resulting minimum voltage magnitudes and maximum element loadings were captured and compared to the other cases. Table

10-12 lists the summary results of the voltage sensitive load modeling comparison. The key findings can be summarized as follows.

- Node-average time-min voltages were very similar for all node types (differences ≤0.007 p.u.).
- Element-average time-max element loadings were very similar for all element types (differences ≤4.673%).
- Differences in the feeder energy consumption were ≤1.612%, which can be important for CVR and similar assessments.
- The differences were negligible for node voltage and element loading related metrics, but could be significant for energy consumption related metrics, which could be important for CVR or similar assessments.

Table 10-12. Summary of the voltage sensitive load modeling comparison

Case Name	Bus-Average of Time-Minimum Voltages - Difference to ZIP_P Case [p.u.]			Element-Average of Time-Max Loadings - Difference to ZIP_P Case [% of Rated]			Feeder Energy Consumption - Absolute Percentage Difference to ZIP_P Case [%]	
	MV Buses	Xfmr LV Buses	Customer LV Buses	MV Lines	LV Lines	Xfmrs	Active Power	Reactive Power
ZIP_P	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
ZIP_Z	0.003	0.004	0.005	0.789	0.753	3.174	0.822	0.349
ZIP_I	0.002	0.002	0.002	0.424	0.421	1.725	1.612	0.650
ExpSens	0.004	0.005	0.006	0.682	0.669	2.826	0.245	0.723
ExpEst	0.006	0.007	0.007	1.159	1.073	4.673	1.108	0.889

10.7. Reactive Power Load Modeling

This research area is discussed in detail in an EPRI Technical Report [76].

Reactive power demand of loads have commonly been modeled through power factor that some utilities adjust based on customer class, feeder voltage level, and other characteristics. The conventional reactive power load modeling approaches (fixed-in-time and feeder-wide power factor) provide limited spatial-temporal accuracy and detail as power factors are known to vary across customers and time. However, it is challenging to improve the accuracy and detail of reactive power load allocation given that the commonly available feeder reactive power measurements are typically masked by switched unmonitored capacitor banks.

This research area investigated the accuracy of different reactive power load modeling methods and proposed approaches to improve them. This research area was divided into the following two parts.

1. **Feederhead Native Reactive Power Estimation:** In this first part, a practical method was proposed to estimate the reactive power produced by capacitor banks to allow separating feeder net reactive power measurements to native reactive power and reactive power produced by capacitors.
2. **Improved Reactive Power Load Modeling:** In this second part, different reactive power load modeling approaches were assessed, and a load modeling method was proposed to improve the spatial and temporal diversity of the reactive power load model.

These two parts are discussed in the following subsections.

10.7.1. *Feederhead Native Reactive Power Estimation*

In this part, a practical method was developed for estimating the feeder native reactive power demand based on SCADA and AMI measurements. As a side product, the developed method also estimates the status of switched capacitor banks and feeder reactive power losses. The developed algorithm is based on identifying clusters in correlations of feederhead active power and reactive power, as illustrated in Figure 10-11 on the right. The algorithm was initially developed and tested on various modified versions of EPRI Ckt5 by changing the location, size, number of capacitor banks on the system, one of which is shown in Figure 10-11 on the left. The performance of the developed algorithm was further tested on several utility feederhead measurement data sets, different time-granularity of the feederhead measurements, etc.

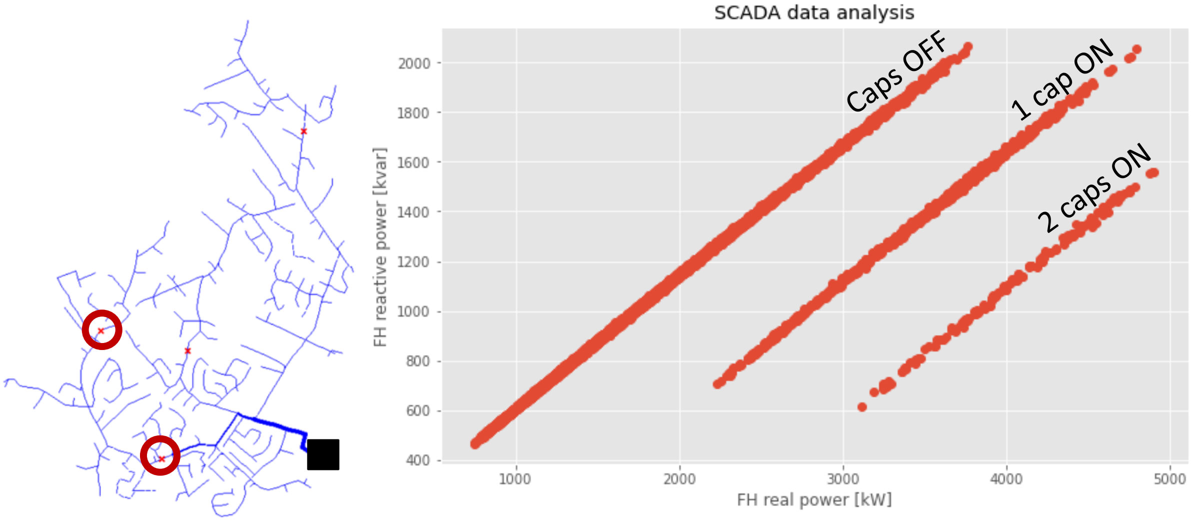


Figure 10-11. A modified version of the EPRI Ckt5 feeder with the capacitor locations shown with the red circles (left) and the principle of capacitor state identification algorithm (right)

The key findings from this research related to estimating the feeder native reactive power can be summarized as follows.

- Capacitor state changes can be identified by plotting the feederhead real power versus the feederhead reactive power. Distinct hyperplanes can be observed in many cases.
- In some cases, feeder seasonal power factor characteristics can make it challenging to identify the distinct hyperplanes. This challenge can potentially be overcome by creating separate feederhead real power vs. reactive power scatter plots for each month of the year and/or hour of the day.
- Some points on the scatter plot may fall in-between the hyperplanes due to a capacitor state change occurring in the middle of the time-step (1 hr or 15 min depending on SCADA measurement frequency).
- A methodology was proposed to identify capacitor states by incrementally clustering datapoints by hyperplanes so that the native reactive power flow at the feederhead can be determined.

10.7.2. *Improved Reactive Power Load Modeling*

Methods were also developed for modeling the reactive power of individual customers based on SCADA measurements, AMI real power measurements, and the feeder native reactive power estimated with the developed method. The methods developed in this research area expanded the spatial-temporal load modeling methods discussed in Section 10.3 that largely focused on active power modeling.

In this research area, six different load modeling methods listed in Table 10-13 were developed to investigate the value of four aspects to reactive power load models:

- Benefits of scaling loads with separate active power and reactive power profiles,
- Benefits of power-based load allocation method (as opposed to current-based),
- Benefits of phase-specific load allocation method, and
- Benefits of step-wise allocation.

Table 10-13. Summary of the load modeling methods analyzed in this research area

Description	Load P model		Load Q model		Visibility level	Load allocation method	Load allocation frequency	Load scaling for QSTS Simulation	Motivation
	Temporal	Spatial	Temporal	Spatial					
Business-As-Usual (ALBAU)	Peak month kWh	Customer-specific	Yearly average PF (feederhead)	1 PF for all loads	Current mag. – Customer kWh	Current-based allocation	Once for Peak month	1 profile (feederhead kW)	Current utility practice
Business-As-Usual (ALBAU_pqProf)	Peak month kWh	Customer-specific	Allocated on yearly avg PF – Time-varying due to Q prof.	1 PF for all loads	Current mag. – customer kWh	Current-based allocation	Once for Peak month	2 profiles (feederhead kW/kVAR)	BAU allocation but scaling load with real and reactive power profiles (vs 1 profile)
Business-As-Usual – Improved PF (ALBAU_improvedPF)	Peak month kWh	Customer-specific	Iterated load PF for feederhead – Time-varying due to Q prof.	1 PF for all loads	Feederhead kW/kvar – customer kWh	Current-based allocation	Once for Peak month	2 profiles (feederhead kW/kVAR)	Improving the load's power factor to better reflect feederhead power factor
Power-based LA – Yearly LA (ALBAU_PQ_total)	Peak month kWh	Customer-specific	Allocated for feederhead – Time-varying due to Q prof.	Following customer kWh so no diversity	Feederhead kW/kvar – customer kWh	Power-based allocation (P&Q) – phase AVG	One for peak month	2 profiles (feederhead kW/kVAR)	Load allocation to reflect feederhead real and reactive powers using phase average P&Q values (2 Cfactors)
Power-based LA – Yearly LA (ALBAU_PQ_phase)	Peak month kWh	Customer-specific	Allocated for feederhead – Time-varying due to Q prof.	Following customer kWh so no diversity	Feederhead kW/kvar – customer kWh	Power-based allocation (P&Q) – phase-specific	One for peak month	2 profiles (feederhead kW/kVAR)	Load allocation to reflect feederhead real and reactive powers using phase specific P&Q values (6 Cfactors)
Power-based LA – step-wise LA (AL15min_PQ_phase)	Monthly kWh	Customer-specific	Allocated using Monthly kWh	Following customer kWh so no diversity	Feederhead kW/kvar – customer kWh	Power-based allocation (P&Q) – phase-specific	15 Min	N/A	Increased load allocation to match time-varying real and reactive power at feederhead.

The methods listed in Table 10-13 were tested on the EPRI Ckt5 feeder model introduced Section 3.1.1.2. An annual QSTS simulation was performed for each load modeling method and the accuracy of each method was quantified against a BaseCase scenario, which was identical to Section 10.2 and modeled all loads with their AMI active and reactive power profiles. The accuracy of the six developed load modeling methods was evaluated with respect to simulated node voltages and element loadings. The key findings from this assessment included the following.

- For a case study analyzed in this research area, the improved reactive power load modeling methods noticeably increased the accuracy of simulated feeder voltages but resulted in only a minor improvement on the accuracy of simulated element thermal loadings. This indicates that accurate reactive power load modeling may be more beneficial for applications that require accurately simulated voltages than for applications that require accurately simulated distribution equipment thermal loadings.

- Power-based load allocation or a load allocation with an improved power factor improves the reactive power load modeling accuracy by better representing the real and reactive power at the feederhead (instead of current magnitude) with more noticeable differences for peak load and other high loading conditions.
- The benefits of phase-specific load allocation are more apparent on feeders, which have considerably unbalances (active power and power factor). Otherwise, phase-average load allocation can provide adequate performance.
- Step-wise load allocation models the feeder (real and) reactive power losses more accurately (than a single load allocation and scaling based on feederhead profiles) especially during off-peak load conditions.
- Scaling loads with separate P&Q profiles (as opposed to P profile only) represents better the temporal diversity of the load model through improving the load model during off-peak conditions.
- Compared to scaling loads with feederhead net load measurements, scaling loads with estimated feeder native load improves the load modeling accuracy through more accurate representation of losses.
- The load modeling methods analyzed in this research area only require feederhead SCADA measurements and customer monthly energy consumption that are available for many distribution utilities.

10.8. Load Modeling in DMS

This research area is discussed in detail in an EPRI Technical Report [98].

Distribution utilities are increasingly deploying distribution management systems (DMS) that perform power flow simulations with detailed load modeling methods. The implementation of those load modeling methods and power flow simulations is typically proprietary for the commercial DMS and hence, limited information is available between different commercial DMS. As a result, it is not clear how DMS load modeling may differ from conventional distribution load modeling methods. It is also not clear how the power flow results may differ between DMS and commercial distribution planning software.

This research area assessed the load modeling and power flow simulation of a commercial DMS and compared them to those of a distribution planning software. The objective of this research area was to provide distribution utilities with increased understanding of the potential differences in load models and power flow results between DMS and planning software and approaches to reconcile the differences. The findings from this research area can help to increase the understanding and confidence of these DMS capabilities.

Literature relevant for the assessment of DMS functions discussed in this research area was reviewed. The review encompassed numerous reports and other publications from EPRI, NREL, Argonne National Laboratory, and other entities, as well as the documentation of a commercial DMS tool. The key lessons learned include:

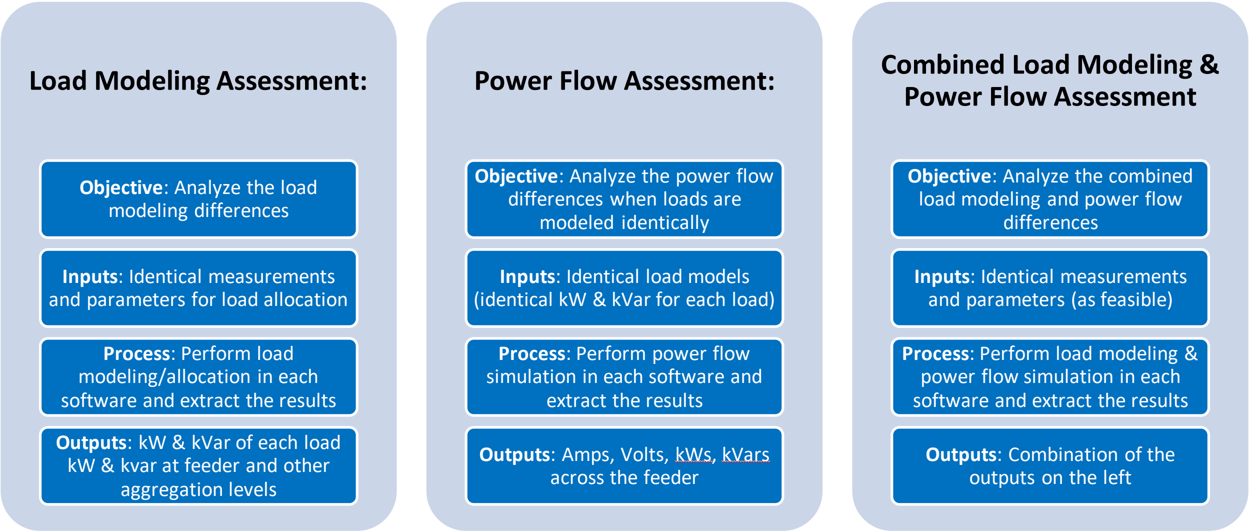
- Many DMS functions and applications have been proposed in the literature and are available from commercial DMS. For this research area, the most relevant DMS functions were:
    - *DMS load modeling (*also referred to as *load estimation)*,
    - *DMS power flow (*also referred to as *online power flow)*, and (to some extent)

    - Distribution state estimation (DSSE).
- These three functions are not clearly defined in the literature because commercial DMS have proprietary implementations with limited documentation of how particular functions are implemented. As a result, the implementation of the three relevant functions is likely to differ between commercial DMS vendor software.
- Load modeling in a commercial DMS may be performed by the DSSE function (potentially without a separate load modeling engine). Conversely, load modeling in a DMS may be performed by a separate engine but may involve aspects resembling DSSE. In other words, *DMS load modeling may have fundamental differences to conventional load modeling in distribution planning software* making apples to apples comparison challenging.
- No literature was identified on past assessments evaluating commercial DMS load modeling, power flow, DSSE, comparing DMS with planning tools, or related topics. In other words, *the research performed in this research area seems to (one of) the first publicly available assessments of its type*.

The assessment of differences between planning and operational tools was conducted on a single utility feeder that is introduced in Section 3.1.1 The following data was available for the assessment (for details, see [98]).

- Utility feeder planning model, which was converted into OpenDSS to minimize unknowns in load modeling and power flow simulation
- DMS results obtained from the utility commercial DMS for two "DMS snapshots" taken at: June 3, 2021 at 13:09 and June 29, 2021 at 17:18
- AMI data available for most customers on the feeder
- SCADA data from the feederhead and other available locations

The assessment consisted of three parts illustrated in Figure 10-12: 1) Load Modeling Assessment, 2) Power Flow Assessment, and 3) Combined Load Modeling and Power Flow Assessment.



Load Modeling Assessment:

Objective: Analyze the load modeling differences

Inputs: Identical measurements and parameters for load allocation

Process: Perform load modeling/allocation in each software and extract the results

Outputs: kW & kVar of each load kW & kvar at feeder and other aggregation levels

Power Flow Assessment:

Objective: Analyze the power flow differences when loads are modeled identically

Inputs: Identical load models (identical kW & kVar for each load)

Process: Perform power flow simulation in each software and extract the results

Outputs: Amps, Volts, kWs, kVars across the feeder

Combined Load Modeling & Power Flow Assessment

Objective: Analyze the combined load modeling and power flow differences

Inputs: Identical measurements and parameters (as feasible)

Process: Perform load modeling & power flow simulation in each software and extract the results

Outputs: Combination of the outputs on the left

Figure 10-12. Three parts of the DMS load modeling assessment

The DMS load modeling and power flow results from the two DMS snapshots were compared to five OpenDSS cases:

- **"DMS Loads":** Model each load directly with its DMS kW and kvar value. By using identical load models in the two software, this case allowed focus on comparing the power flow results as illustrated for the "Power Flow Assessment" part in Figure 10-12.
- **"AMI Loads":** Model each load directly with its AMI measurements during the DMS snapshot time instances. For simplicity, loads without AMI were modeled in OpenDSS with their DMS values. This case was analyzed to provide context to how closely DMS and OpenDSS load modeling methods may align with the actual load values as measured by AMI.
- **"I-Phase":** Allocate feederhead phase currents (from DMS) based on customer monthly kWh (from AMI) and feeder aggregated monthly average power factor (from AMI). This case represents a common load modeling practice applied by some utilities in their planning software.
- **"PQ-Phase":** Allocate feederhead phase kW and kvar (from DMS) based on customer monthly kWh (from AMI) and feederhead phase-average power factor (from DMS). This case represents another common load modeling practice applied by some utilities in their planning software.
- **"PQ-Total":** Allocate feederhead total kWs and kvars (from SCADA) based on customer monthly kWh (from AMI) and feederhead phase-average power factors (from DMS). This case represents a third load modeling practice applied by some utilities in their planning software.

Given their different use cases, DMS and distribution planning software are expected to have a range of differences, for details of the differences relevant to the assessment here see [98]. As a result, considerable effort may be required to manage the differences between the software to perform assessments like the one discussed in this research area. To focus the analysis on the differences in load modeling and power flow simulations, considerable effort was invested in minimizing all other discrepancies between the planning and operational versions of the model as well as automating the assessment. The main components developed for addressing these aspects of the assessment are illustrated in Figure 10-13. The report for this research area [98] includes a detailed list of discrepancies addressed in this assessment including:

- Feederhead voltage magnitude
- Voltage regulator tap positions
- Capacitor switch statuses
- Feeder model topology
- Measurements used for load allocation
- Load values (when not allocated)
- Load voltage sensitivity

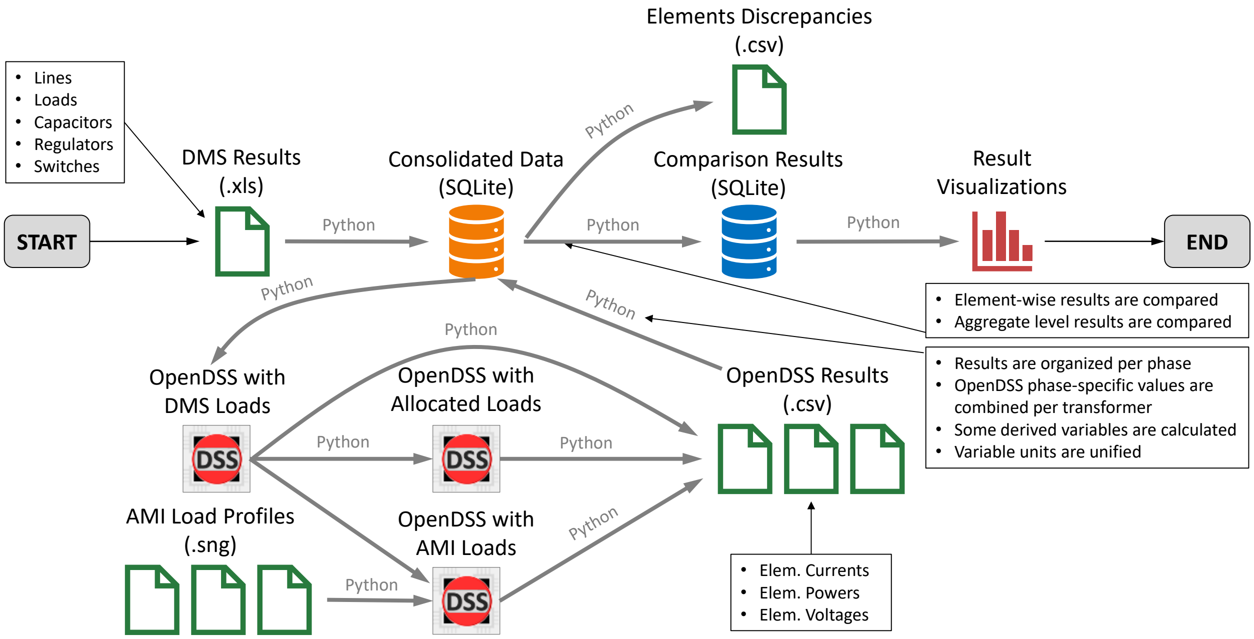


Figure 10-13. Components for the power flow and load modeling assessment including planning and DMS results

A high-level summary of the differences between DMS results and OpenDSS results is provided in Table 10-14. The key findings from the assessment were as follows.

- DMS and distribution planning software are expected to have a range of differences that can make apples to apples comparison between the software challenging given their primary applications. Details about differences relevant to the assessment and the approach followed to resolve discrepancies are discussed in detail in [98].
- Numerous factors for improving the consistency of load modeling and power flow results between DMS and planning software were identified. The documentation of these factors and their impact could be used to improve the accuracy of distribution models in planning and operation tools. Considerable effort may be required to manage the differences between the two software when performing assessments like the one discussed in this research area.
- Noticeable differences (up to +/-2%/0.02p.u.) were observed in the voltage profiles between the DMS and planning software even when all loads were modeled identically. It is worth emphasizing that these differences persisted even after considerable effort was invested in minimizing the differences between the two software. This suggests potential differences between the power flow modeling between the DMS and the planning software.
- The load modeling/allocation methods applied in DMS and planning software can have a noticeable impact on load modeling and power flow results and the differences of the results between the two software. The analysis of results illustrates the magnitude of differences that can be found between planning and operational tools with alternative load modeling methods. For simulated line currents and bus voltages, absolute differences up to 50 Amps and 10%/0.1p.u. were observed for some cases and feeder locations. However, for most feeder locations, the differences were much less.
- From the three OpenDSS load allocation methods, PQ_Phase (i.e., allocating loads based on feederhead phase kWs and kvars) yielded the best overall performance for the test case. I_Phase

(i.e. allocating based on feederhead phase currents) led to inaccurate feeder reactive power modeling, and PQ_Total (i.e. allocating based on feederhead phase-total kW and kvar) led to somewhat less accurate modeling of feeder line currents and bus voltages. As a result, PQ_Phase is the recommended load allocation method to limit the differences between planning and DMS tools, when measured feederhead phase kWs and kvars are available.

- While AMI data could theoretically allow for more detailed load modeling, considerable effort can be required to process the data and the large quantities of AMI data can introduce challenges, including handling of data quality issues and unmetered loads. In this assessment, it turned out to be challenging to properly handle unmetered loads. If AMI data is directly used to model the loads of some customers, it is recommended to allocate the loads of the remaining customers to match the feeder aggregated loads. However, if most loads are directly modeled with AMI data, allocating the limited remaining loads may lead to unrealistic values for (some of) the other allocated loads.

Table 10-14. Summary of the differences between DMS results and OpenDSS results

Case Name	OpenDSS Load Modeling Approach	Aggregated Feeder Load Differences		Simulated Feederhead Power and Current Differences			Simulated Line Current Differences	Simulated Bus Voltage Differences
		kW	kvar	kW	kvar	Amps		
DMS Loads	Use DMS load models	Small	Small	Small	Noticeable	Noticeable	Small	Noticeable
AMI Loads	Model loads with their AMI kW and kvar values	Noticeable	Noticeable	Considerable	Considerable	Considerable	Noticeable	Noticeable
I_Phase	Allocate feederhead phase currents (DMS) based on customer monthly kWh (AMI) and feeder aggregated monthly average power factor (AMI)	Small	Noticeable	Small	Considerable	Small	Noticeable	Considerable
PQ_Phase	Allocate feederhead phase kW and kvar (DMS) based on customer monthly kWh (AMI) and feederhead phase-average power factor (DMS)	Small	Small	Small	Small	Small	Noticeable	Considerable
PQ_Total	Allocate feederhead total kWs and kvars (SCADA) based on customer monthly kWh (AMI) and feederhead phase-average power factors (DMS)	Small	Small	Small	Small	Noticeable	Noticeable	Considerable

10.9. Summary

Accurate load modeling within distribution planning tools has been critical for effective system planning and analysis. However, growing penetration of DER introduces new assessment needs that require more granular and accurate load models. The work in this task highlighted the importance of leveraging the unprecedented visibility provided by AMI and other data streams to address emerging needs with updated distribution load modeling practices.

This task developed and evaluated improved, yet practical, distribution load modeling methods that leverage modern data streams. This was achieved by seven research areas addressing a broad range of distribution load modeling aspects. Research in these areas identified numerous ways for distribution planners to enhance the conventional distribution load modeling practices to better utilize the modern data.

This research has also demonstrated that increasingly granular and detailed load modeling could be supported by wider availability of AMI, DER measurements, feeder sensors, and other emerging data streams, as well as their integration to distribution planning tools. Results provided by numerous simulation scenarios demonstrated that further refinements to conventional load allocation methods can go a long way. In particular, the accuracy of load allocation methods can be improved by leveraging measurements from additional sensors, using phase-specific measurements, as well as allocating based on active and reactive power measurements. Moreover, using additional customer level information, such as native load kWh values, can lead to improvements.

The development of this task has pointed to the need to continue research for practical methods and tools to disaggregate PV and other DER, such as energy storage and electric vehicles, from net load measurements based on typically available utility data. Further research on best practices to improve consistency in load modeling within operational and planning tools is also warranted.

11. PV DYNAMIC MODELING

11.1. Task Overview

Problem Statement: High penetration of BTM PV with the revised IEEE Std 1547$^{\text{TM}}$-2018 [80] control functionalities makes validation of PV-DER models critical to ensuring bulk system stability. The Western Electricity Coordinating Council composite load model (WECC CLM) has recently been proposed, [101], and it is now available in various commercial power system simulation tools. After several updates, the latest model is called CMPLDWG and incorporates DER components by utilizing the *PVD1* model, [102], [103] Because the DER development has grown and changed after the *PVD1* model was published, it is no longer a representative model for today's distribution systems. Indeed, some renewable energy standards published in the past few years (e.g., the IEEE Std 1547$^{\text{TM}}$-2018 and California Rule 21, [104]) may significantly impact the representation of the previously developed power system models.

Objectives: Motivated by the abovementioned challenges and developments, the primary goals of this task are to: (1) determine dynamic models of the distribution system and PV systems; (2) identify and tune the key parameters used in representing an aggregate response of many individual DERs spread across a distribution system via novel estimation algorithms; and (3) demonstrate and test classification algorithms to predict responses to events so as to detect maloperation or failures of a PV inverter cluster (see Figure 11-1).

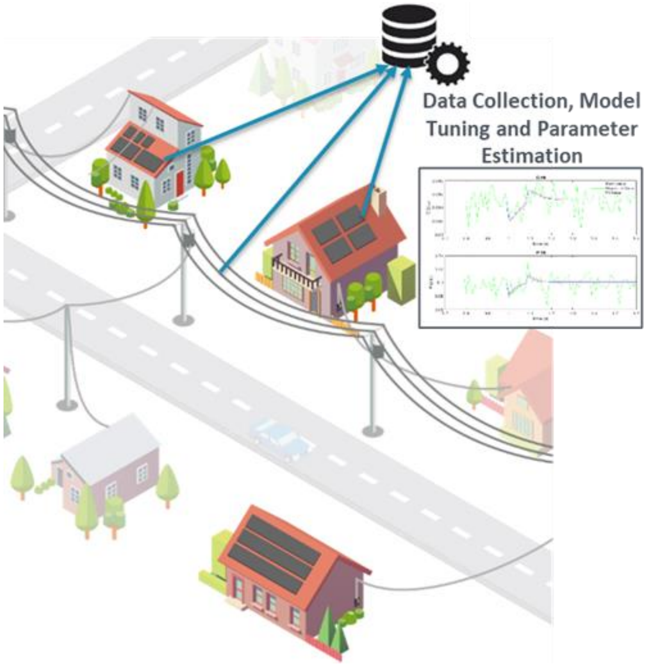


Figure 11-1. PV dynamic modeling illustration

11.2. PV Dynamic Modeling and Validation

To have a better representation of today's power system with a large number of small-scale DERs, EPRI has recently proposed the new aggregate dynamic model named "DER_A" to support power system stability studies [105]. In 2018, this aggregate model was approved for implementation in bulk power system planning studies by the WECC Model Validation Working Group. Compared to the previous power system models, DER_A provides a simpler yet more comprehensive model by considering the latest standards and functionalities (e.g., voltage and frequency control) that are related to a DER. The DER_A model can represent the aggregated dynamics behavior of many tens to hundreds of small distributed inverter-based generators (e.g., rooftop PV systems) in a distribution system. Although the DER_A model drastically reduces the dimension of model parameters from a total of 121 (for the *PVD1* model) to 48, proper setting of parameters is still critical to ensuring the accuracy of power system simulation. The DER_A model is developed for representing dynamic behaviors of inverter-based renewable generation/storage devices (e.g., wind turbines, PV systems, and batteries). It is simplified from the generic renewable energy-system models. Figure 11-2 shows the architecture of the DER_A model. It consists of five subsections based on different functions:

1) **Active power-frequency controls:** frequency and real power readings are the two inputs for this block. It outputs active current command (*Ipcmd*) for other subsections, which is calculated by using power-order (*Tpord*) and filtered terminal voltage (*Vt_filt*). *Freq_flag* is used to enable/disable the control of this subsection.
2) **Reactive power-voltage controls:** this block represents the voltage control of DER devices. Terminal voltage (*Vt*) and a user-defined voltage reference value (*Vref0*) are the inputs. The flag *Pflag* is able to decide the control mode between constant reactive power control and constant power factor control. The output of this subsection is reactive current command (*Iqcmd*).
3) **Active-reactive current priority controls:** The flag, *Typeflag*, indicates the simulated device is either a generation or a storage unit, while the other flag (i.e., *Pqflag*) determines the operation of the current limit logic between Q- or P-priority mode. In short, this subsection mainly describes the behaviors of a DER device under disturbances (e.g., fault events and intermittent nature of renewable resources).
4) **Fractional tripping controls:** this section represents the behaviors of DERs when a voltage event happens (i.e., under- and overvoltage). The major parameters are the four breakpoints (*vl0*, *vl1*, *vh0*, and *vh1*) and the flag, *Vtripflag*, which enables/disables the fraction tripping control function, while the fraction parameter, *Vrfrac*, decides how much of DER generation should be cut out with the four breakpoints during a voltage event.
5) **Voltage source representation:** the last subsection simulates a voltage source converter in a modern inverter device. In the DER_A model, it is regarded as equivalent to a voltage source behind the impedance. Therefore, the reactance, *Xe*, is used in this subsection.

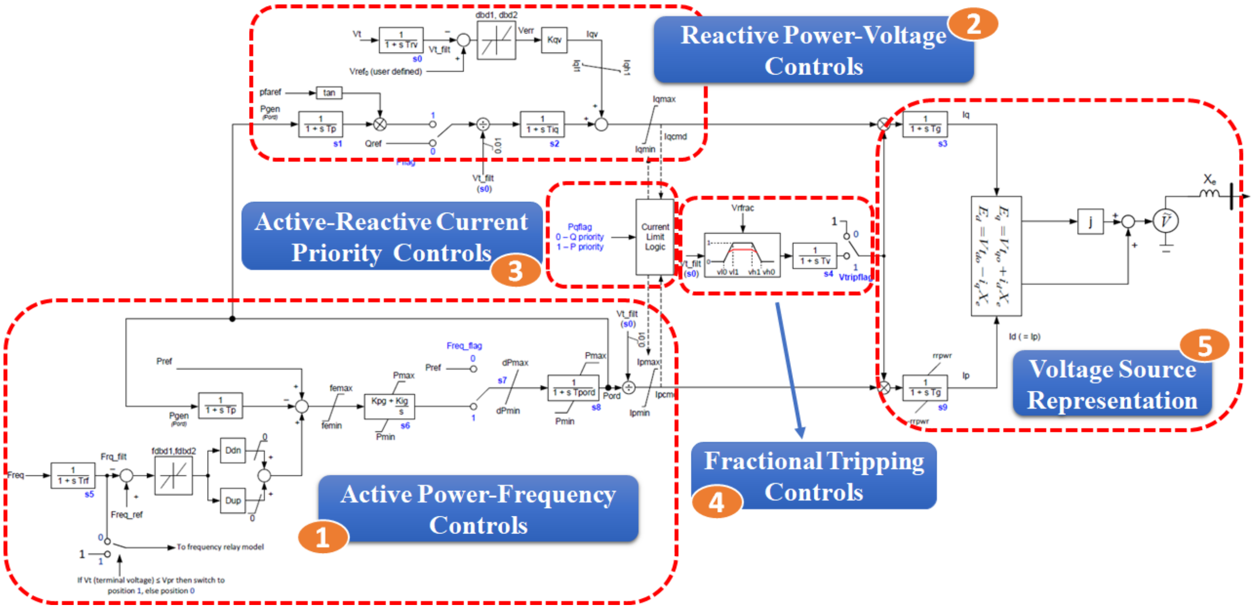


Figure 11-2. Block diagram of the DER_A model with each section of the controls highlighted

A total of 48 parameters of the DER_A model are listed in Table 2-1, which shows the parameters in each function block with their properties (e.g., time constants, gain, deadbands, etc.). Because it is not feasible to optimize all 48 parameters, we first need to identify the critical parameters to obtain an effective configuration setting of the model that will participate in the optimization process. In this study, the MATLAB® Optimization Toolbox™ is used to obtain the optimal setting of the selected DER_A model parameters.

Table 11-1. List of the DER_A model parameters

Name	Type	Section	Name	Type	Section	Name	Type	Section
Trf	Time Const.	1	*dPmin*	Deadband	1	*Tv*	Time Const.	4
Tp	Time Const.	1,2	*FreqFlag*	Switch	1	*Vrfrac*	Const.	4
Tpord	Time Const.	1	*Trv*	Time Const.	2	*Vl0*	Breakpoint	4
Kpg	Gain	1	*Tiq*	Time Const.	2	*Vl1*	Breakpoint	4
Kig	Gain	1	*Kqv*	Gain	2	*Vh0*	Breakpoint	4
Ddn	Deadband	1	*Kig*	Gain	2	*Vh1*	Breakpoint	4
Dup	Deadband	1	*dbd1*	Deadband	2	*tVl0*	Time Boundary	4
fdbd1	Deadband	1	*dbd2*	Deadband	2	*tVl1*	Time Boundary	4
fdbd2	Deadband	1	*Iql1*	Deadband	2	*tVh0*	Time Boundary	4
femax	Deadband	1	*Iqh1*	Deadband	2	*tVh1*	Time Boundary	4
femin	Deadband	1	*Iqmax*	Deadband	2	*VtripFlag*	Switch	4
Ipmax	Deadband	1	*Iqmin*	Deadband	2	*Tg*	Time Const.	5
Ipmin	Deadband	1	*Vref0*	Voltage Ref.	2	*rrpwr*	Deadband	5
Pmax	Deadband	1	*Imax*	Deadband	3	*Xe*	Reactance	5
Pmin	Deadband	1	*PqFlag*	Switch	3	*fltrp*	Config.	-
dPmax	Deadband	1	*TypeFlag*	Switch	3	*fhtrp*	Config.	-

Prior to calibration of the parameters of the DER_A model, we have implemented it in Simulink® (see Figure 11-3) and validated it against the "DERAU1" model provided by PSS®E. Figure 11-4 shows the trajectories of the bus voltage as well as dynamic real and reactive power responses obtained with our DER_A model implementation (in red) and PSSE (in blue). The results give us the confidence that we have accurately implemented DER_A.

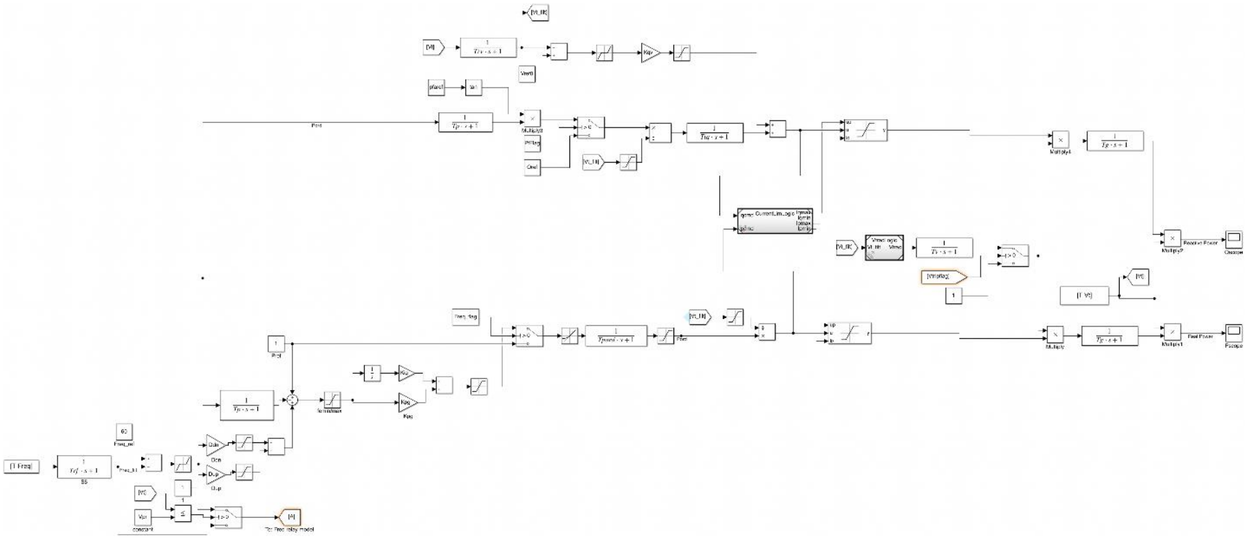


Figure 11-3. DER_A model implementation in Simulink

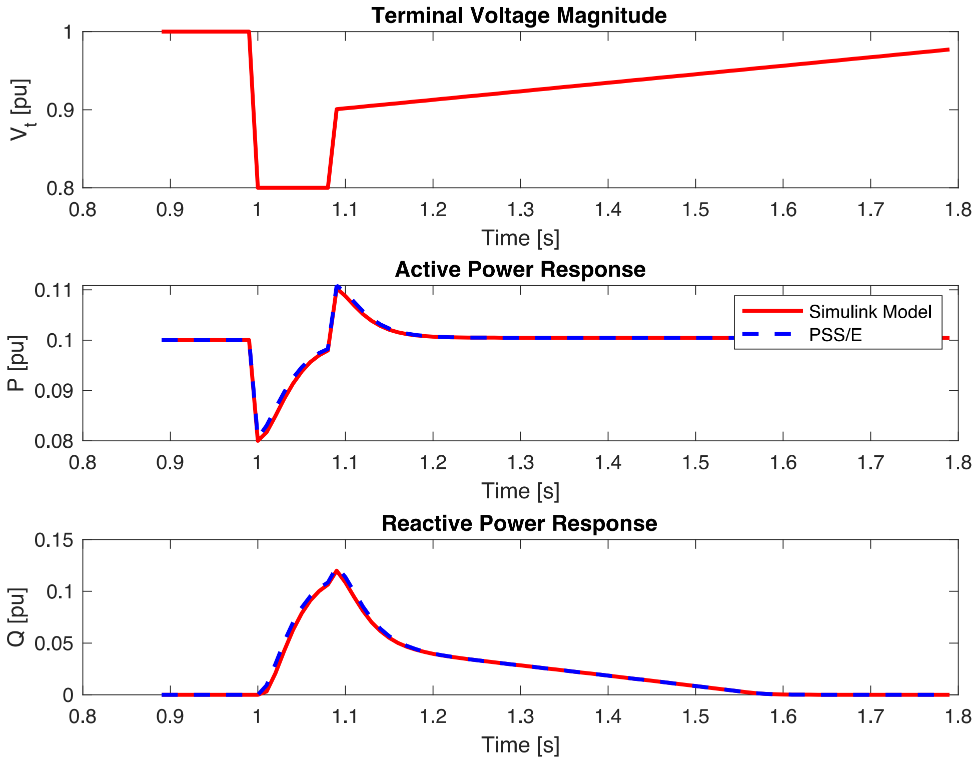


Figure 11-4. Bus voltage (top), real (middle) and reactive (bottom) power of Simulink and PSS®E models of DER_A

11.3. DER_A Parameterization and Calibration

According to different interconnection standards, NERC provides a list of default parameters value for the DER_A model [106]. The first step is to identify the critical model parameters that can impact the simulation result. Next, the MATLAB® Optimization Toolbox™ is used to obtain the optimal setting of the selected model parameters.

11.3.1. Selection of Critical Parameters

The critical parameters are defined as the DER_A model's parameters that can improve the accuracy of a power system simulation by reasonably adjusting the value of the parameters. For example, reactance, time boundaries, deadbands, and breakpoints are not appropriate parameters to be optimized because they are usually determined by the hardware limits or device/power-system standards. Second, the switch parameters (i.e., flags) are excluded as well because they are configuration settings in a power system simulation. Therefore, time constants and gain are the two kinds of parameter properties that are regarded as critical parameters in this research. To this end, we have first identified ten critical model parameters (i.e., *Trf*, *Trv*, *Tp*, *Tiq*, *Tpord*, *Tg*, *Tv*, *Kpg*, *Kig*, and *Kqv*) based on our engineering judgment and some preliminary tests we have performed.

11.3.2. Optimized DER_A Parameter Values

The MATLAB-based nonlinear least-squares (NLS) solver, *lsqnonlin*, is used for acquiring the optimal settings among all the combinations of the selected ten critical model parameters. One of the uses of the NLS algorithm is curve fitting for a closed-loop system. Because the DER_A model has a couple of feedback control loops, the NLS optimization algorithm can be utilized to achieve the goal of calibrating the model. The calibration can be done by finding the appropriate parameter values (i.e., time constants and gain) that provide the minimal error between the output and input signals.

To verify the performance of the calibration result, the CYME [107] dynamic simulation is used to simulate an IEEE standard feeder model for providing the required input of the calibrated DER_A model. The output (i.e., aggregated PV generation) of the DER_A model is further compared with the simulation results from CYME.

Because the dependencies among the ten critical parameters are unknown, different combinations of critical parameters may result in similar optimization results. Therefore, a total of all possible 1,023 ($=\sum_{k=1}^{10}\binom{10}{k}$) critical parameter combinations are used to generate the optimal parameter settings by using MATLAB's *lsqnonlin* function with the default solver (i.e., the trust-region reflective algorithm). In each trial, an optimal parameter set is generated by using the NERC's suggested default setting as the initial value. Each of the 1,023 optimal sets is further applied to the DER_A model for comparing the performance with the default model and the unit step reference signal individually. The result shows that 512 out of 1,023 sets have lower RMSE than the default setting. Finally, the set with the lowest RMSE is selected as the optimal parameter set for the DER_A model. Table 11-2 shows the optimally calibrated values (in **bold**) of the DER_A model parameters for the case with the minimum RMSE. In this case, the parameter set, *Trf*, *Trv*, *Tv*, and *Kqv*, is the combination of critical parameters that are used in the optimization process.

Table 11-2. Optimal parameter configuration for the DER_A model

Trf	*Trv*	*Tp*	*Tiq*	*Tpord*	*Tg*	*Tv*	*Kpg*	*Kig*	*Kqv*
0.44935	**0.026464**	0.02	0.02	0.02	0.02	**0.02**	0.1	10	**0.18303**

According to the optimization result, almost every test case returns different optimal settings by applying different combinations of critical parameters to the optimization toolbox, which means dependencies exist among the critical parameters.

11.3.3. Validation of the Optimized DER_A Model

The verification of model calibration is supported by a 10-second simulation on the IEEE 34-node network using CYME. Table 11-3 shows the configuration of the five PVs that are integrated into the IEEE 34-node test feeder (as shown in Figure 11-5). The voltage and frequency readings at the substation node (i.e., Node 800) are recorded during the 10-second simulation. These two data streams serve as the inputs (i.e., *Vt* and *Freq* in Figure 11-2) for the DER_A model. Throughout the simulation time window, we have varied the output of PVs in the range between [0.73, 1.0] pu, where 1.0 pu, in our case, equals 1.5 MW of aggregate PV generation. To run tests with more realistic PV data, we have added random noise to the aggregate PV generation data (i.e., *Pref*) prior to feeding it into the DER_A model that we have built in Simulink. The noise signal is created by generating a random number which is in the range of ±10% of the original PV output for each time step (i.e., 2 seconds). The input of the DER_A model, *Pref*, is calculated by summing the original aggregated PV output with the noise signal.

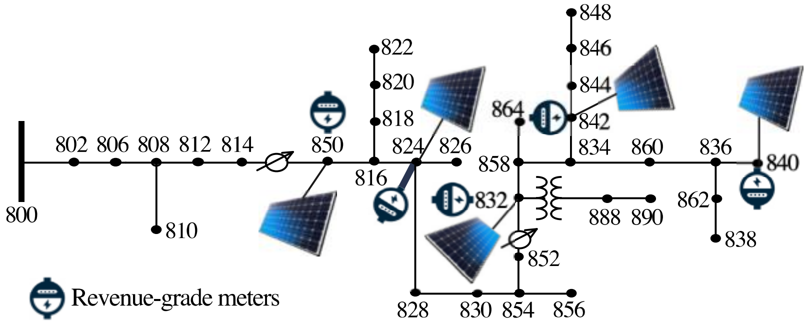


Figure 11-5. PV locations on the IEEE 34-node test feeder.

Table 11-3. IEEE 34-Node test feeder with PV generation

	Time	Node					Aggregated
	[Cycle]	850	824	832	842	840	Output
PV Output [MW]	0 – 120	0.3	0.3	0.3	0.3	0.3	1.5
	121 – 250	0	0.3	0.3	0.3	0.3	1.2
	251 – 360	0.3	0.3	0.3	0.3	0.3	1.5
	361 – 420	0	0.2	0.3	0.3	0.3	1.1
	421 – 460	0.3	0	0.3	0.3	0.3	1.2
	461 – 600	0.3	0.3	0.3	0.3	0.3	1.5

Figure 11-6 shows the comparison among the (1) noise-added PV generation data (i.e., *Pref*) from CYME, (2) output of the DER_A model with default settings, and (3) optimized DER_A model. The *Pref* represents the actual aggregated PV output from the 5 PVs in the test distribution system. The outputs of the DER_A model (dashed red and solid orange curves) should follow the *Pref* (blue curve). The zoomed-in plot shows the different transient behaviors between default and optimal settings of the DER_A model. The overall RMSEs (between Cycles 61 and 600) for the default and optimal parameter sets are found to be 0.090948 and 0.090791, respectively. Essentially, these errors represent the difference between the outputs of the two settings of the DER_A model and the aggregate PV generation from CYME. The lower RMSE corresponds to the case with the best-fit parameter values for the DER_A model calibration. According to this test result, the optimal setup of the DER_A model parameters is able to improve the accuracy of the power system simulation.

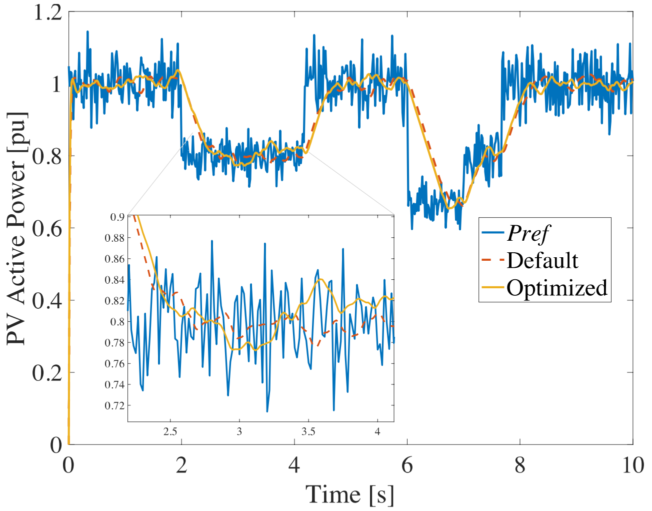


Figure 11-6. Comparison of the noise-added PV active power (Pref) with the output of the DER_A model using default settings and the optimal DER_A model.

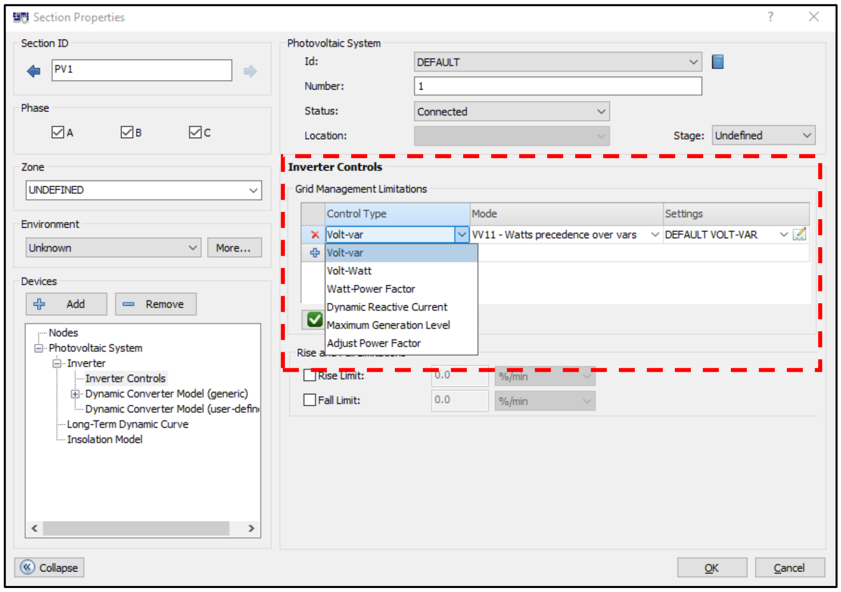
In our study, the calibrated DER_A model is verified by the real power outputs from PVs. Therefore, the result is valid for active power-frequency control and voltage-source representation function blocks, which are shown in Figure 11-2.

11.4. Detection of Maloperation or Failures

To generate a useful dataset for detecting maloperation of a PV inverter, we have refactored the Python script used to apply the different configurations of transient events and automate the CYME simulations on the IEEE 34-node test feeder. The initial dataset covers a total of 2,690 test scenarios, which are listed in Table 11-4. In the load-modification testing, four modification configurations are applied to the three scenarios individually. Thus, there are a total of $3 \times 4 \times \sum_{k=1}^{6}\binom{6}{k}$=756 cases under this test category that involves modification of 6 spot loads. CYME supports multiple operation modes for a PV inverter model as shown in Figure 11-7. The transient events are run under the three different PV inverter operation modes. According to the different operation modes, a corresponding power factor is set to the power output of each PV inverter. Because the PV inverter operation mode is set by the user rather than by switching voltage readings, it is suitable for simulating the feeder model with maloperation of PV inverter(s). All the test scenarios are run for 30 cycles with a 0.05-cycle time step during the power system simulation. Figure 11-8 shows the configuration of the CYME simulations.

Table 11-4. List of test scenarios

Inverter Operation Mode / Transient Event Type (# of Cases)	**MPPT (PF = 1)**	**Volt-var (PF = 0.7)**	**Shutoff**
Node Fault (**34**)	Single Node Fault		
Line Disconnection (**24**)	Single Overhead Line Disconnection		
PV Disconnection (**124**)	All Combinations of PV Disconnection		N/A
Spot Load Modification (**756**)	**I.** Real power only **II.** Reactive power only **III.** Combined (I&II)		**(i)** 35% increase **(ii)** 100% increase **(iii)** 20% decrease **(iv)** 50% decrease
Total Number of Cases	938	938	814



Section Properties

Section ID: PV1

Phase: ☑A ☑B ☑C

Zone: UNDEFINED

Environment: Unknown

Photovoltaic System

Id: DEFAULT

Number: 1

Status: Connected

Location:

Stage: Undefined

Inverter Controls

Grid Management Limitations

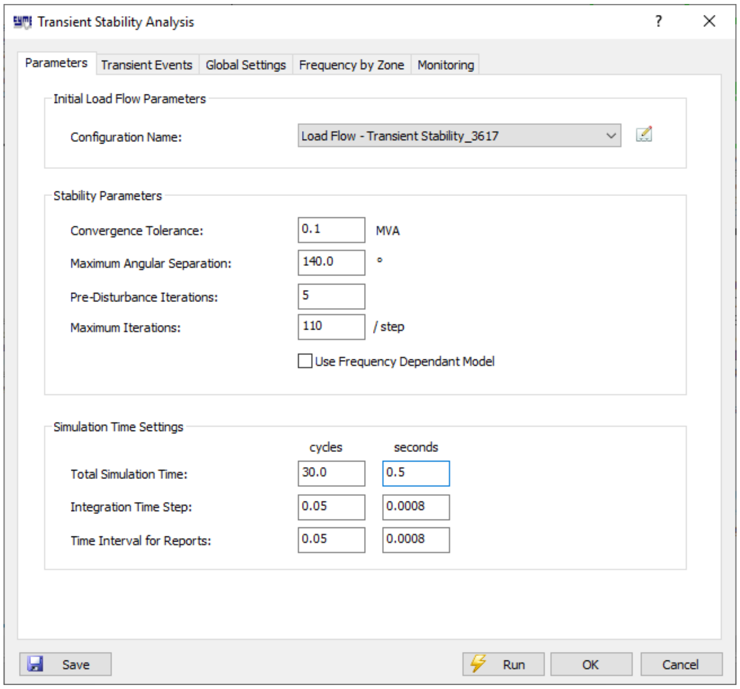
Control Type	Mode	Settings
Volt-var	VV11 - Watts precedence over vars	DEFAULT VOLT-VAR

Rise and Fall Limitations

□ Rise Limit: 0.0 %/min

□ Fall Limit: 0.0 %/min

Figure 11-7. PV inverter model in CYME



Initial Load Flow Parameters

Configuration Name: Load Flow - Transient Stability_3617

Stability Parameters

Convergence Tolerance: 0.1 MVA

Maximum Angular Separation: 140.0 °

Pre-Disturbance Iterations: 5

Maximum Iterations: 110 / step

□ Use Frequency Dependant Model

Simulation Time Settings

	cycles	seconds
Total Simulation Time:	30.0	0.5
Integration Time Step:	0.05	0.0008
Time Interval for Reports:	0.05	0.0008

Figure 11-8. Simulation parameters and time settings for all test scenarios

Because the CYME Transient Stability Analysis module only provides voltage magnitude and angle as well as frequency readings for each feeder node in the stability report, we have developed a Python program to collect the current readings of each PV inverter. Therefore, we have obtained four features (i.e., node voltage magnitude and angle, node frequency, and PV inverter's output current magnitude and angle readings) in the dataset for developing an ML model to detect the maloperation of PV systems. For training purposes, features are created for PV systems based on the power system measurement of the CYME simulation. Table 11-5 shows the 18 features defined for the PV systems.

Table 11-5. List of training dataset's features

	Duration [Cycle]	Feature Name	Description
Pre-Transient Event	6 - 200	*maxV_pre*	The maximum voltage reading during the time period before the transient event
		avgV_pre	The average voltage reading during the time period before the transient event
		minV_pre	The minimum voltage reading during the time period before the transient event
		maxF_pre	The maximum frequency reading during the time period before the transient event
		avgF_pre	The average frequency reading during the time period vbefore the transient event
		minF_pre	The minimum frequency reading during the time period before the transient event
		maxI_pre	The maximum current reading during the time period before the transient event
		avgI_pre	The average current reading during the time period before the transient event
		minI_pre	The minimum current reading during the time period before the transient event

	Duration [Cycle]	Feature Name	Description
Post-Transient Event	202 - 222	*maxV_post*	The maximum voltage reading during the time period after the transient event
		avgV_post	The average voltage reading during the time period after the transient event
		minV_post	The minimum voltage reading during the time period after the transient event
		maxF_post	The maximum frequency reading during the time period after the transient event
		avgF_post	The average frequency reading during the time period after the transient event
		minF_post	The minimum frequency reading during the time period after the transient event
		maxI_post	The maximum current reading during the time period after the transient event
		avgI_post	The average current reading during the time period after the transient event
		minI_post	The minimum current reading during the time period after the transient event

To label each instance in the dataset, the voltage readings at Sample 402. Since the power system reach a steady state after applying a transient event at Sample 402, the voltage and frequency readings (V_{402} and f_{402}) of each PV node is used with the corresponding inverter's operation mode to classify an instance (normal or abnormal). Table 11-6 shows the classification logic for each instance.

Table 11-6. Classification of instances

Operation Mode of an Inverter	Normal Operation
MPPT	$0.95 \text{ pu} \leq V_{402} \leq 1.05 \text{ pu}$
Volt-var	$V_{402} < 0.95 \text{ pu}$ OR $V_{402} > 1.05 \text{ pu}$
Shutoff	$f_{402} > 60.05 \text{ Hz}$

For the detection of inverter maloperation, we have tested and compared several common ML algorithms provided by *scikit-learn* library, which are listed below.

- ***k*-NN:** a common ML algorithm supports both unsupervised and supervised neighbors-based learning methods. For our task, the *k*-NN algorithm is configured as a supervised learning method.
- **Linear SVM:** an SVM algorithm with linear kernel. We use the default configuration for the coefficient $d = 0.025$.
- **RBF SVM:** an SVM algorithm with RBF kernel. We use the default configuration for the coefficients $\gamma = 2$ and $C = 1$.
- **GP:** a generic supervised learning method designed to solve *regression* and *probabilistic classification* problems.
- **RF:** one of the averaging algorithms based on randomized decision trees. Each tree in the ensemble is built from a sample drawn with replacement from the training set.
- **Neural Network:** the basic neural network algorithm (i.e., multilayer perceptron (MLP)) is used in the tests. *Scikit-learn* use the parameter *alpha* for regularization (i.e., L_2 regularization) term, which assists in avoiding overfitting by penalizing weights with large magnitude. The *alpha* is set to 1, while the max iteration is set to 1,000.

- **AdaBoost:** a popular boosting algorithm. The core principle of AdaBoost is to fit a sequence of weak learners (i.e., models that are only slightly better than random guessing such as small decision trees) on repeatedly modified versions of the data.
- **NB:** a simple yet powerful algorithm for predictive modeling. The model comprises two types of probabilities that can be calculated directly from the training data: (1) the probability of each class and (2) the conditional probability for each class given each *x* value.

The eight classifiers are trained by the dataset and compared the performance. The train:test split ratio is set to 80:20. In our case, we only consider the maximum power point tracking (MPPT) operation mode, resulting in a dataset that contains 756 instances for each PV. Since 5 PVs are connected to the power system, the total number of instances becomes 3,780. The performance of each classifier is measured by accuracy, which is defined as the overall true positive and true negative rates. To compare the training efficiency, we also provide the runtime of the training process for each classifier. The test result is shown in Table 11-7.

Table 11-7. Comparison of different ML classifiers with the reduced dataset

Classifier	Accuracy [%]	Runtime [s]
***k*-NN**	99.735	0.015
Linear SVM	94.444	0.101
RBF SVM	99.339	0.037
GP	98.677	6.522
RF	99.868	0.038
Neural Network (MLP)	94.180	2.440
AdaBoost	99.868	0.307
NB	98.677	0.016

The observations of the test results are summarized as follows:

- The *k*-NN classifier provides the quickest training process among the eight classifiers and balances the tradeoff between training time and accuracy the best.
- Random forest (RF) and AdaBoost classifiers return the best performance in terms of detection accuracy.
- Gaussian process (GP) model was initially assumed to be the best classifier for this task because it is becoming increasingly popular in solving classification, calibration, and regression problems. While the GP model performs relatively well in terms of accuracy, it takes the longest time to train the utilized dataset.

According to the test result on the performance of different ML classifiers, we have further created a larger dataset by including all the CYME simulations with different inverter operation modes (i.e., MPPT, volt-var, and shutoff). The size of the dataset is increased to 13,450 instances. The configuration of the training process remains the same (i.e., an 80:20 train:test split ratio), except that we shuffle the sequence of the instances before splitting the dataset. Furthermore, we repeat the shuffle, training, and testing process ten times and calculate the average accuracy and runtime values for each classifier. The test result is shown in Table 11-8.

Table 11-8. Comparison of different ML classifiers with the full dataset

Classifier	Accuracy [%]	Runtime [s]
***k*-NN**	97.60	0.300
Linear SVM	65.61	4.646
RBF SVM	78.72	5.432
GP	72.82	206.596
RF	91.75	0.075
MLP	64.61	4.363
AdaBoost	87.52	0.587
NB	64.84	0.005

In general, all the accuracy values are significantly lower than those of the previous test. However, *k*-NN can still achieve an ML accuracy of more than 97% with a 0.3-second runtime. It shows that the patterns of the PV inverters' behavior are similar within the same operation mode. Therefore, the neighboring-based algorithm can form distinctive groups when relatively similar patterns (inverter operation modes) are present in the training/testing dataset.

Decision-tree-based algorithms classify instances by sorting them in the tree. It has a similar concept to that of a clustering algorithm at high levels. However, it further differentiates each instance as the sorting process passes on to the lower level. Therefore, RF provides the second-best accuracy performance when utilizing the extended dataset. Compared to the previous test, we can infer that PV inverters' behaviors are strongly associated with different operation modes, but there are not many differences inside each operation group. Thus, the clustering-based (i.e., *k*-NN) and decision-tree-based (i.e., RF) classifiers perform better than other ML algorithms.

Another advantage of these two classifiers is that their training processes are simpler than those of others. They take much less time to develop an ML model. Because today's distribution grids are ever more complicated, massive volumes of data will most likely create computational bottlenecks for various ML applications. Regularly updating the ML model and fine-tuning its hyperparameters optimize the performance of classifiers are critical to reducing the false discoveries in ML tasks, thus ensuring the reliability of a power grid. Therefore, an ML algorithm with a short training time would be preferable for efficient utilization in today's power grids.

11.5. Summary and Future Work

In this section, we present viable approaches to calibrating an aggregate PV dynamic model and detecting maloperation/failure of PV inverter systems in modeling environment. This work resulted in the following publication [108]. More specifically, we have (1) accurately fitted the DER_A model and calibrated its parameters via a *derivative-free* nonlinear least-squares algorithm and (2) trained and tested some of the popular ML classifiers with certain pre-/post-event features, using the transient simulation data provided by CYME. To enhance the scope and generalizability of the tasks performed, the following improvement efforts are reserved for future work:

1. development of robust and scalable probabilistic ML methods to calibrate dynamic model parameters and classify and predict the behavior of inverters and control devices;
2. validation of the newly developed methods in a large-scale distribution feeder environment considering the full range of control functionalities defined in the IEEE Std 1547-2018^{TM}

3. utilization of realistic PV data and a distribution model (e.g., SMART-DS [109]) for identifying the optimized setting of aggregate DER model parameters; and
4. evaluation of more comprehensive scenarios by using higher-fidelity (e.g., electromagnetic) transient simulations under a wide range of uncertainties in feeder and PV dynamic model parameters.

12. ALGORITHM IMPLEMENTATION

12.1. Open Source Code Release

Three distribution system model calibration algorithms are included in this release, see https://github.com/sandialabs. There are two algorithms for performing phase identification: one based on an ensemble spectral clustering approach and one based on leveraging additional sensors placed on the medium voltage. Start with the CA_Ensemble_SampleScripts.py file and the SensorMethod_SampleScript.py file, respectively. The third algorithm identifies the connection between service transformers and low-voltage customers (meter-to-transformer pairing algorithm). Start with the MeterToTransPairingScripts.py file.

There is also a sample dataset included to facilitate the use of the code. The dataset will load automatically when using one of the sample scripts provided. For more details, please see Section 12.1.3.

12.1.1. Phase Identification

12.1.1.1. Spectral Clustering Ensemble Method

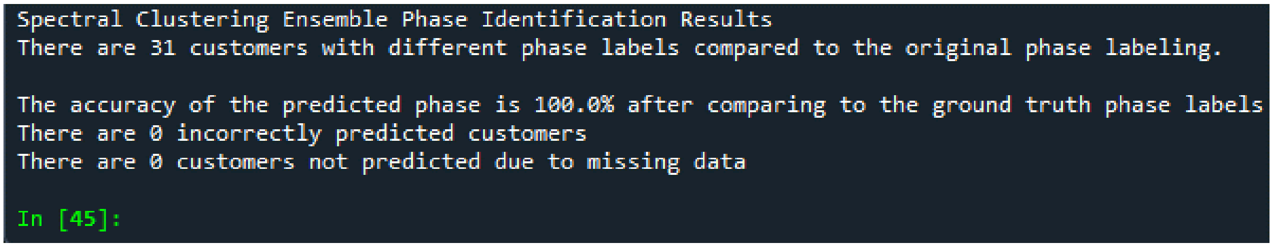
The code for the spectral clustering ensemble method is broken into three files: CA_Ensemble_SampleScripts.py, CA_Ensemble_Funcs.py, and PhaseIdent_Utils.py. Start with the CA_Ensemble_SampleScripts.py file; it will automatically load the required sample data and run the spectral clustering ensemble algorithm. CA_Ensemble_Funcs.py contains the primary functions that make up the method, and PhaseIdent_Utils.py contains helper functions.

For more details on this method, please see Section 4.3.

This algorithm performs phase identification of customers by using their voltage timeseries measurements to cluster similarly related customers, without any other data requirements of topology information, customer power measurements, or measurements from the substation or transformers. It takes as input the voltage timeseries from advanced metering infrastructure (AMI) meters which have been converted into per-unit representation and transformed into a change in voltage timeseries by taking the difference in adjacent measurements. The algorithm loops through the available data in interval sizes specified by the windowSize parameter. A windowSize of 384 may be a good place to start; values between 96 and 384 will likely work fine. Larger window sizes tend to work better, but there is a tradeoff depending on the amount of missing data in the dataset. At each iteration, customers with missing data during that window are excluded, and the remaining customers are clustered using the sklearn implementation of spectral clustering. The clustering results from each window are used to build a co-association matrix where the entries for customers that were clustered together are incremented. After all windows have been processed, the co-association matrix is normalized by dividing each entry by the number of windows where that pair of customers were both included in the window (i.e., neither customer was excluded due to missing data). The normalized co-association matrix is then used as a pre-computed affinity matrix to the spectral clustering algorithm for the final clustering determination. The number of final clusters can be set based on the feeder topology. Four to seven final clusters might be a good place to start, however this parameter may change from feeder to feeder. For example, if the feeder has voltage regulation devices a larger number of clusters may be required. The resulting clusters will be groupings by phase, but the final mapping from the cluster to a particular phase is left to a subsequent step. If the original utility labels are available and you believe they are at least 50%

accurate, then the final mapping can be done using a majority vote with the original labels. This is what is done in the sample script provided.

The two primary outputs from the spectral clustering ensemble phase identification method are finalClusterLabels from the CAEnsemble function and predictedPhases from the CalcPredictedPhaseNoLabels function. The finalClusterLabels field contains the results from the final cluster assignment; these cluster labels correspond to phase groupings but are not mapped to a particular phase. If the original phase labels are available, and you believe they are at least 50% accurate, then they can be used to do the mapping between the clusters in finalClusterLabels and particular phases. This is the predictedPhases field where each customer has been assigned a predicted phase based on the final cluster and a majority vote using the original utility labels.



```
Spectral Clustering Ensemble Phase Identification Results
There are 31 customers with different phase labels compared to the original phase labeling.

The accuracy of the predicted phase is 100.0% after comparing to the ground truth phase labels
There are 0 incorrectly predicted customers
There are 0 customers not predicted due to missing data

In [45]:
```

Figure 12-1. Output screenshot for the spectral clustering ensemble phase identification algorithm using the sample data

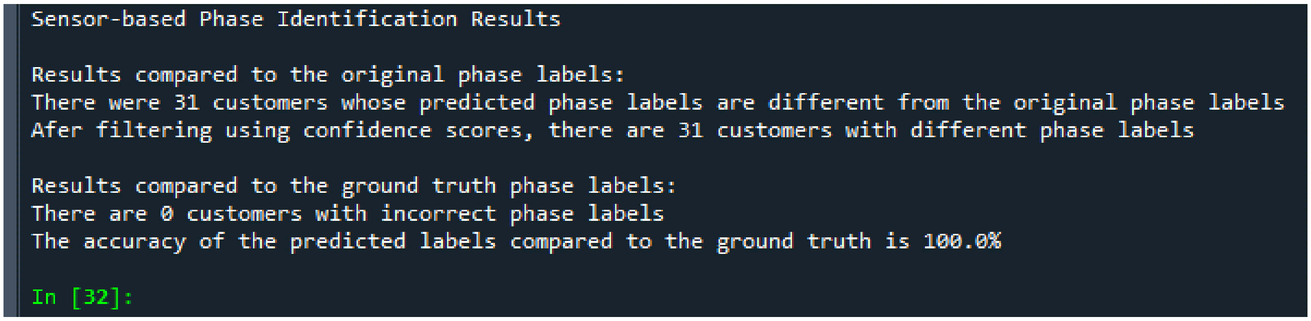
12.1.1.2. Sensor-based Method

The code for the sensor-based phase identification method is broken into three files: SensorMethod_Funcs.py, SensorMethod_SampleScript.py, and PhaseIdent_Utils.py. Sensor_Method_Funcs.py contains the primary functions for the sensor-based method. PhaseIdent_Utils.py contains helper functions. SensorMethod_SampleScript.py is the place to start; the sample data will load automatically and run the algorithm.

For more details on this method, please see Section 4.2

The sensor-based phase identification method takes voltage timeseries from AMI meters and sensors located on the medium-voltage distribution system (this research used IntelliRupters). Based on the known phases of the medium-voltage sensors located around the feeder, the phase of each AMI meter is determined based on the correlations to the phase voltage measurements from the other sensors. The voltage timeseries should be pre-processed into a per-unit representation and then converted to a change in voltage timeseries by taking the difference of adjacent measurements. A window ensemble approach is employed where intervals of data, specified by the windowSize parameter, are taken independently. Correlation coefficients are calculated between each customer and each sensor data stream (three data streams per sensor, one for each phase). Any customers with missing data during a window are excluded from consideration during that window. The other parameter to set is the CC Separation Filter threshold (CCSepThresh); this parameter filters the correlation coefficients produced by individual windows using the Correlation Coefficient Separation Score. Using a window size of 96 and a CC Separation filter value of 0.06 might be a reasonable place to start. Ranges of 96-384 for the window size and 0.02 – 0.06 for the CC Separation filter appear to be reasonable choices for the data we have tested. Once all available data has been used, the mean of the correlation coefficients is taken. The highest correlated sensors with each customer then vote on the predicted phase for each customer. The number of votes should be determined by the number of available sensors on the feeder and other considerations such as the number of voltage regulation devices in the system. Our work uses 5 votes. Finally, confidence metrics are calculated for each customer to give an indication of confidence in the phase prediction.

The primary output of the sensor-based phase identification method is the predictedPhaseLabels field. This will contain the phase labels for each customer (excluding those omitted due to missing data) which were assigned by the sensor-based method. Figure 12-2 shows the output produced by running the SensorMethod_SampleScript.py using the included sample data. The predictions are compared both to the original utility labeling and the ground-truth phase labeling.



```
Sensor-based Phase Identification Results

Results compared to the original phase labels:
There were 31 customers whose predicted phase labels are different from the original phase labels
Afer filtering using confidence scores, there are 31 customers with different phase labels

Results compared to the ground truth phase labels:
There are 0 customers with incorrect phase labels
The accuracy of the predicted labels compared to the ground truth is 100.0%

In [32]:
```

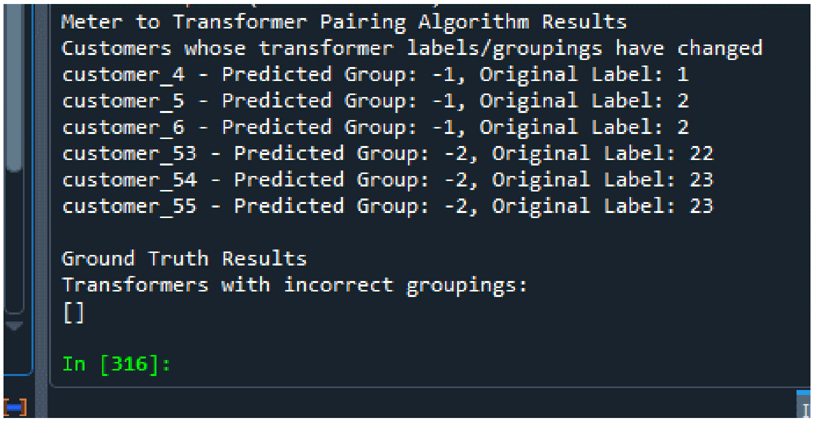
Figure 12-2. Output screenshot for the sensor-based phase identification method using the sample data

12.1.2. Meter-to-Transformer Pairing

The code that implements the meter-to-transformer pairing task is broken into three files: M2TFuncs.py, M2TUtils.py, and MeterToTransPairingScripts.py. M2TFuncs.py implements the primary functions for the methodology. M2TUtils.py implements helper functions. MeterToTransPairingScripts.py is the place to start; this file will load the sample data and run the algorithm. For more details on this method, please see Section 6 [65].

This method uses correlation coefficients analysis of the customer voltages to flag transformer groupings, which likely contain errors, and then a linear regression methodology using voltage, real power, and reactive power data is used to correctly group the customers by service transformer. This code implements the methodology described in Section 6. The input to the meter-to-transformer pairing algorithm is voltage magnitude, real power, and reactive power timeseries AMI data. The algorithm is divided into two stages. In the first stage, pairwise correlation coefficients are calculated between all pairs of customers. Then, the original transformer labels are used to inspect the correlation coefficients for each transformer grouping. If any of the pairwise correlation coefficients are below a specified threshold then the transformer is flagged for inspection in the second stage of the algorithm. Stage 1 is done using a ranked approach with multiple thresholds such that stage 1 results in a ranked list of flagged transformers, so that the earlier in the list the transformer appears, the worse the correlation coefficients were in that transformer grouping. In stage 2, a pairwise linear regression is done between all customers. This produces a mean-squared-error (MSE) value that functions as a type of goodness-of-fit metric for the regression, a resistance coefficient, and a reactance coefficient. The resistance and reactance values function as a type of distance matrix between customers. The MSE values and are used as a filter for the reactance matrix, where pairs with high MSE are discarded. The algorithm currently sets the MSE threshold by finding the minimum MSE value and adding a small amount to that value. This parameter could also be set manually if desired. The resulting, filtered, reactance matrix is used to assign new transformer groups to the flagged transformers/customers from Stage 1. Customer pairs serviced by the same transformer will have reactance "distances" lower than the reactance due to the influence of two transformers.

The primary output of the meter-to-transformer pairing algorithm is the predictedTransLabels field returned by the CorrectFlaggedTransErrors function. This field contains a list of transformer labels, unchanged labels remaining the same, and new transformer groupings labeled with negative integers. There is not a straightforward way to map the new groupings (designated with negative integers) to physical transformers; that is left as a subsequent task. The sample data produces output as shown in Figure 12-3. The two customers whose labels were changed (customer_4 and customer_53) are in new transformer groups with the other customers serviced by their respective transformers. You can see this by comparing the results in predictedTransLabels to the ground-truth transformer labels in transLabelsTrue variable. The "Transformers with incorrect groupings" list is empty because the algorithm was 100% successful in finding the correct transformer groupings. Any transformers whose groupings are incorrect when compared to the ground-truth labels would be listed here.



```
Meter to Transformer Pairing Algorithm Results
Customers whose transformer labels/groupings have changed
customer_4 - Predicted Group: -1, Original Label: 1
customer_5 - Predicted Group: -1, Original Label: 2
customer_6 - Predicted Group: -1, Original Label: 2
customer_53 - Predicted Group: -2, Original Label: 22
customer_54 - Predicted Group: -2, Original Label: 23
customer_55 - Predicted Group: -2, Original Label: 23

Ground Truth Results
Transformers with incorrect groupings:
[]

In [316]:
```

Figure 12-3. Output screenshot for the meter-to-transformer pairing algorithm

12.1.3. *Sample Data*

The sample data included in this release consists of timeseries data for 400 single-phase customers, 10 sensors, and the substation at 15-minute intervals with a total of 11,520 measurement points, or approximately four months. The customer advanced metering infrastructure (AMI) data, as well as the sensor and substation, consist of voltage magnitude, real power, and reactive power timeseries. The voltage magnitude data is in Volts and the real power and reactive power are in Watts and Var, respectively. There are also phase labels, transformer labels, and IDs for each AMI and sensor. The data is included as .npy files which is the numpy data format. This data is a subset of the dataset described in Section 3.1.1

12.2. CYME Implementation and Testing

Phase identification, secondary system topology and parameter estimation, and meter-to-transformer pairing algorithms were implemented in Python prototypes connected to the CYME software. This allows validating these algorithms on multiple utility and realistic test networks under a variety of conditions. The algorithms that were tested include those developed in this project and other state-of-the-art methods found in the literature.

12.2.1. *Phase Identification*

12.2.1.1. Algorithms

Six phase identification algorithms were implemented in CYME's Python prototype during this project. Their input data requirements are summarized in Table 12-1, wherein a checkmark (✓) indicates that the corresponding input is required.

Table 12-1. Input data requirements of the six phase identification algorithms.

Method	AMI		SCADA (Feeder)		Initial customer phases
	V	kW(h)	V	kW(h)	
ESC-GIS	✓	x	x	x	✓
ESC-SCADA	✓	x	✓	x	x
PCA	✓	x	✓	x	x
MT	✓	x	✓	x	x
LASSO	x	✓	x	✓	x
SF	x	✓	x	✓	x

12.2.1.1.1. *Ensemble Spectral Clustering with GIS Phasing (ESC-GIS)*

The ensemble spectral clustering method with GIS phasing (ESC-GIS), developed by SNL, is presented in Section 4.3.1.3 of this report. Specifically, the original approach without the co-association matrix was implemented in CYME's Python prototype [13]. Scikit-learn's spectral clustering function was used with a Gaussian kernel ($\gamma = 0.01$) [43]. Unless stated otherwise, 12 clusters were used in the k-means stage. Each window covers four days.

12.2.1.1.2. *Ensemble Spectral Clustering with SCADA Measurements (ESC-SCADA)*

The ensemble spectral clustering method with SCADA measurements (ESC-SCADA) is similar to ESC-GIS, except that the initial customer phase – used for voting – is found by computing the Pearson correlation coefficients (CCs) between the customer and the three single-phase SCADA feeder head voltage time series. The largest CC defines the initial phase.

12.2.1.1.3. *Principal Component Analysis (PCA)*

This approach, presented in [110], is similar to ESC-GIS. The main differences are that it uses principal component analysis (PCA) instead of spectral clustering, and that it doesn't use an ensemble methodology. The prototype calls scikit-learn's PCA function with two components [43]. Must-link constraints [110] are excluded to avoid dependency on a potentially erroneous network model.

12.2.1.1.4. *Multi-Tree (MT)*

The multi-tree (MT) algorithm of [111] was implemented in CYME's Python prototype. In this approach, a tree is created for each phase at the feeder head, whose root nodes are the locations of the single-phase SCADA feeder measurements. Starting from these root nodes, each tree is populated by adding the load with the highest CC to any of the existing elements of the trees, until all loads have been added.

12.2.1.1.5. Least Absolute Shrinkage and Selection Operator (LASSO)

The objective of this method is to find the phase connectivity vectors $\mathbf{X}_f$ ($f = \{a, b, c\}$) such that $\mathbf{P}\mathbf{X}_f + \mathbf{e} = \mathbf{Y}_f$ ($f = \{a, b, c\}$) where $\mathbf{P}$ and $\mathbf{Y}_f$ contain the customer AMI and substation real power measurements, respectively, and $\mathbf{e}$ is a residual vector [112]. This is solved using the least absolute shrinkage and selection operator (LASSO). Scikit-learn's LASSO function is used with a penalty coefficient $\alpha = 0.05$ [43].

12.2.1.1.6. Salient Frequency (SF)

The salient frequency (SF) approach first filters out low frequencies from the customer active power profiles using a Discrete Fourier Transform (DFT), and then extracts "salient," or distinguishing events from each customer and uses correlation coefficients with the substation real power measurements to assign a phase to each customer [113]. In CYME's Python prototype implementation, the first 10 components computed by the DFT are filtered out [113], the threshold TH [113] is equal to the number of AMI meters divided by 5, and only the load variations of the closest 20 samples are considered.

12.2.1.2. Metrics

The accuracy of the algorithms is defined as

$$\mathrm{Acc}_{\%} = \frac{1}{3} \sum_{\sigma \in \{a,b,c\}} \sum_{n \in M_\sigma^{\mathrm{ref}}} \frac{F_{\mathrm{eq}}(P_n^{\mathrm{pred}}, \sigma)}{|M_\sigma^{\mathrm{ref}}|} \times 100\% \tag{12-1}$$

where M_σ^{ref} is the set of customers with valid measurements originally connected to phase σ (in the network model for synthetic data test cases), P_n^{pred} is the predicted (identified) phase of the nth customer by the phase identification algorithm, $F_{\mathrm{eq}}(x, y)$ returns 1 if $x = y$ and 0 otherwise, and $|S|$ represents the cardinality (number of elements) of set S.

For ensemble methods (ESC-GIS and ESC-SCADA), a confidence score CS_n can be computed to get a feel of the prediction accuracy. For a given customer, CS_n represents the percentage of windows whose predicted phase equals the final predicted phase.

12.2.1.3. Synthetic Data Test Cases

Several tests using synthetic AMI data and CYME network models of real feeders are presented in this section. For each feeder, base sets of equivalent AMI and SCADA measurements are generated by solving time-series power flows on each network model for 10,000-timesteps at a 15-minute interval (around 104 days) with synthetic AMI active and reactive power profiles [32]. Key information regarding these feeders is presented in Table 12-2. In order to see the impact of the voltage regulation equipment on the algorithms, other sets of equivalent measurements are generated for the North #1, North #2, and South systems where the regulator taps are fixed during the entire simulation. They are denoted by the suffix (Fixed).

The combination of synthetic data and network models is an informative means of assessing the accuracy and robustness of the various phase identification algorithms. Unlike tests with real AMI data from utility feeders, it provides free and unequivocal ground truth (no field personnel needs to be involved); furthermore, the synthetic data can be modified in controlled ways to test the algorithms under different scenarios. The results presented here were published in [41].

Table 12-2. Properties of the four distribution test systems.

Network	Nodes	AMI Meters	Substation Voltage Regulators	Inline Voltage Regulators
EPRI's CKT5	3003	1373	0	0
North #1	2369	615	1	3
North #2	4065	963	1	6
South	1778	447	0	1

12.2.1.3.1. Ideal Conditions

The accuracy metric $\mathrm{Acc}_{\%}$ is first computed for all methods under ideal conditions (e.g., noiseless measurements, full AMI coverage, …); the corresponding results are presented in Figure 12-4.

ESC-GIS provides almost perfect results (>99%) for all systems. The accuracy of ESC-SCADA and PCA is similar: both are very accurate with no active tap changers, but their accuracy plunges for networks with multiple regulators (<60% for North #1 and #2). MT behaves similarly to ESC-SCADA and PCA but is more accurate for North #1 (84.6%). LASSO's accuracy is greater than 90% for all networks except CKT5 (67.9%). The accuracy of SF ranges from 60 to 80%.

The main reason ESC-SCADA and PCA are inaccurate for North #1 and #2 is because inline voltage regulators cause occasional but significant voltage changes that are not seen by the SCADA voltage measurements. Consequently, the step in these methods that tries to correlate the AMI and SCADA voltage measurements often fails. As for ESC-GIS, as long as an adequate number of clusters is used, the presence of voltage regulators has a negligible impact because it does not use SCADA measurements.

MT almost always correctly identifies large groups of nearby same-phase customers independently of the presence of tap-changing devices. However, it often has difficulty connecting these groups to the proper tree when tap changers are present. An extreme case is North #2, where 957 of the 963 customers were assigned to phase A while only 397 are truly on this phase.

For CKT5, the LASSO algorithm fails to identify any phase for several customers (the corresponding element in the three solution vectors $\mathbf{X}_f$ all equal 0), which explains its very low accuracy. Nevertheless, if these unidentified customers are removed from $M_{\sigma}^{\mathrm{ref}}$, $\mathrm{Acc}_{\%}$ only jumps from 67.9% to 84.4%. Since CKT5 is the network with the most customers, followed by North #2 where several customers also remain unidentified, it appears that LASSO has difficulties with larger systems.

As will be demonstrated in Section 12.2.1.3.4, SF's accuracy improves monotonically over a wide range of available samples; however, even with 10,000 samples, the accuracy is below 80% for all test networks. Figure 12-4 also shows that for a given sample count, SF is more accurate for networks with fewer customers.

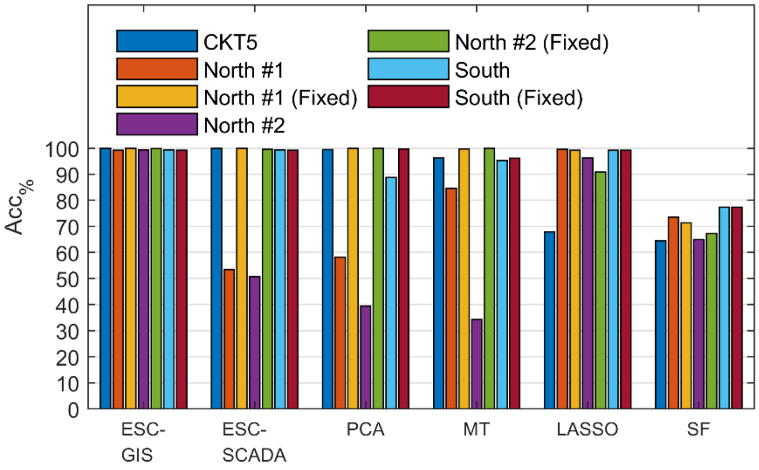


Figure 12-4. Phase identification accuracy under ideal conditions.

12.2.1.3.2. AMI Coverage

A well-known practical concern with power-based methods is low AMI coverage, e.g., due to partial AMI deployment or opt-out clauses. To assess the robustness of LASSO to different AMI coverage levels, the case study is repeated by removing different numbers of AMI meters from the base set. The location of these meters is chosen randomly. To compare with voltage-based methods, the same study is executed with ESC-GIS. The corresponding results are presented in Figure 12-5.

LASSO's accuracy decreases as the AMI coverage is reduced. The accuracy remains above 80% for the South network even with a coverage of only 50%, which shows some robustness. However, for North #2, the accuracy falls below 80% and 90% for coverage levels of 90% and 95%, respectively. This limits the applicability of LASSO in practical situations. Note that as in Figure 12-4, the presence of tap changers has a limited impact on LASSO's accuracy. As for ESC-GIS, the coverage level has no effect on accuracy.

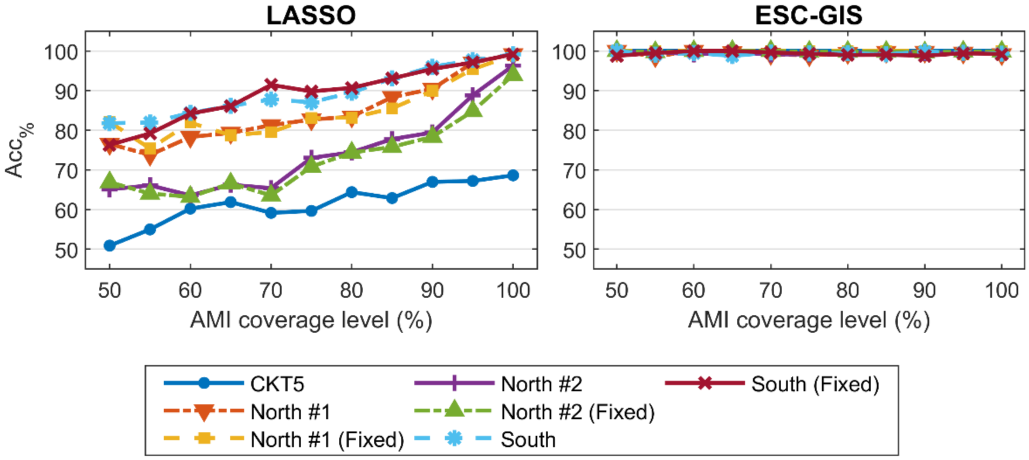


Figure 12-5. Accuracy of LASSO and ESC-GIS for different AMI coverage levels

12.2.1.3.3. *Initial Phase Mislabeling*

ESC-GIS assigns a unique phase to all customers of a given cluster using a voting scheme based on the initial guess of each customer phase. This initial customer phase typically comes from a GIS system. In the ideal conditions test case (Section 12.2.1.3.1), all initial phases were exact, which is an unrealistic assumption. Note that among the methods tested in this paper, only ESC-GIS uses this initial phasing.

The study of Section 12.2.1.3.1 is repeated by intentionally mislabeling different numbers of initial customer phases. As seen in Figure 12-6, the method remains very accurate with up to half of the phases incorrectly labeled (>95% for all networks), showing high robustness. The accuracy decreases drastically afterwards. This perfectly validates the work published in [29] during this same project.

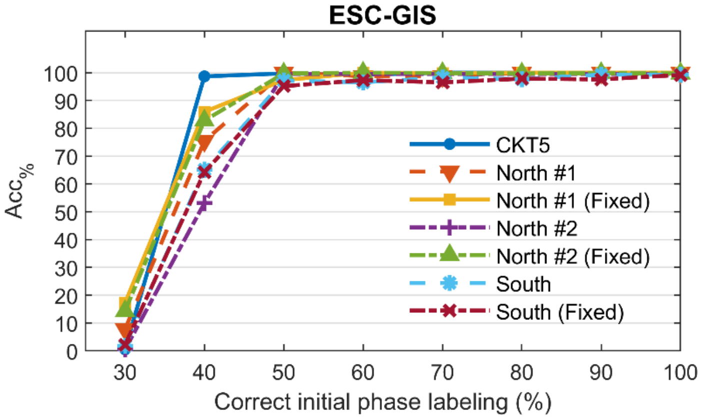


Figure 12-6. Accuracy of ECG-GIS as a function of the percentage of customers with initially mislabeled phases

12.2.1.3.4. *Number of Samples*

Figure 12-7 presents the impact of the number of measurement samples on $\mathrm{Acc}_{\%}$ on the CKT5 and South networks. For the four voltage-based methods, 1,000 samples (around 10 days of data at a 15-minute interval) is sufficient to achieve maximum accuracy under ideal conditions on all test networks. The same is true with LASSO on South (the smallest network of the set) but not CKT5 (the largest). With CKT5, the accuracy keeps increasing as a function of the sample count, although it starts plateauing between 5,000 and 10,000 samples. Finally, as mentioned in Section 12.2.1.3.1 $\mathrm{Acc}_{\%}$ monotonically increases as a function of the sample count with SF. At least 500 samples were needed for the ensemble methods to have at least one full window of four days.

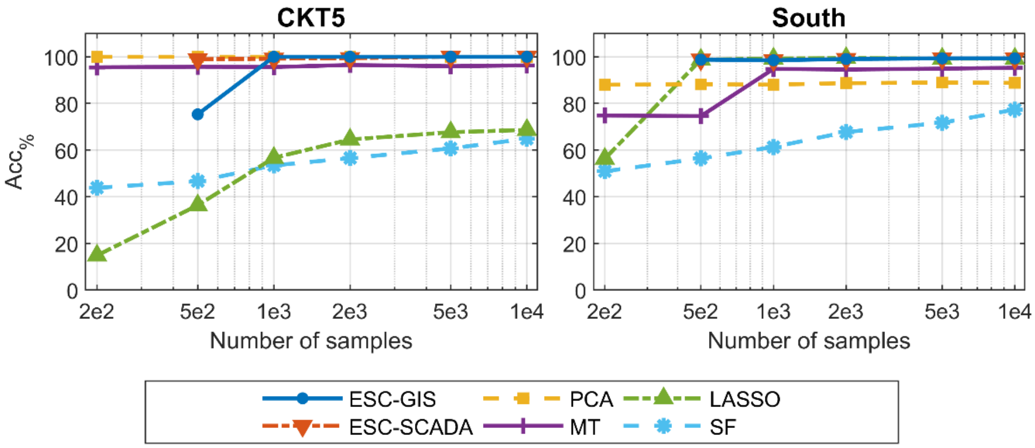


Figure 12-7. Phase identification accuracy as a function of the number of samples for the CKT5 and South networks

12.2.1.3.5. Measurement Error

All previous studies assumed ideal, noiseless, and synchronized measurements. To analyze some of the impact of imperfect measurements, the study of Section 12.2.1.3.1 is repeated by adding different levels of uniformly distributed noise (as a percentage of each measured value) to all AMI and SCADA measurements; Figure 12-8 presents the corresponding results for North #1 and North #1 (Fixed). All methods see virtually no change with an error level of 0.1%, while $\mathrm{Acc}_{\%}$ starts decreasing with a noise level of 0.2% for some of the voltage-based methods. Due to the inherent randomness, there is no clear trend. For instance, ESC-GIS has better accuracy with noises of 0.4% and 0.5% than 0.2% and 0.3%. LASSO and SF appear immune to these levels of noise, since power measurements see natural variations of much larger amplitude than the injected noise.

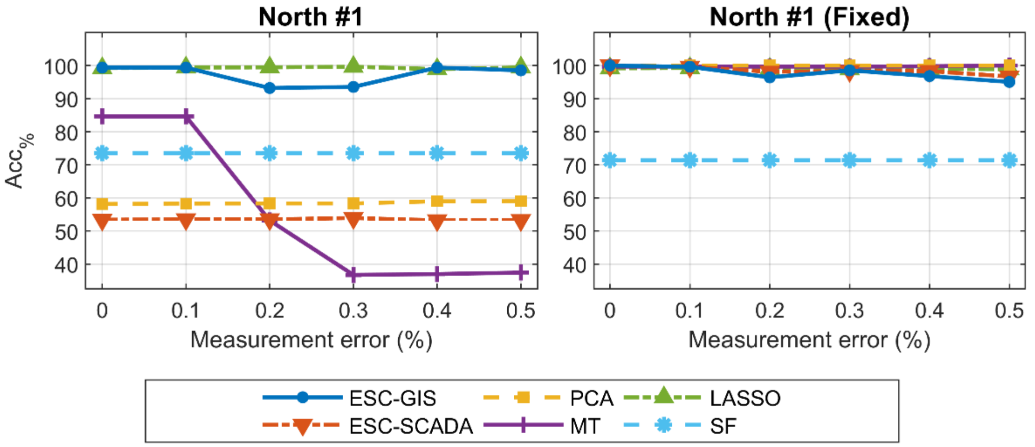


Figure 12-8. Phase identification accuracy as a function of uniformly distributed measurement error for the North #1 and North #1 (fixed) networks

12.2.1.4. Utility Data Test Case

While several scenarios were covered in the synthetic data test cases section (Section 12.2.1.3), real utility data can also be helpful to assess the phase identification algorithms under situations that would not have been otherwise envisioned.

In this section, results obtained on an AMI voltage data set provided by a North American utility are presented. Because the utility did not have access to SCADA measurements, only the ECG-GIS algorithm is considered. The set spans over five months with a 15-minute time interval and covers a single feeder with 290 metered single-phase customers. The feeder has one substation and two inline voltage regulators. There are several instances of missing data through parts the five months. Out of 11,310 potential results (39 windows times 290 customers), 2,615 could not be computed. Six clusters are used in the k-means step.

The results are summarized in Table 12-3. All customers initially labeled on phase A are predicted to be on this phase; while no additional customers are assigned on phase A. Out of 107 customers on phase B according to the original network model, ESC-GIS predicted that 29 of them should be on phase C. The confidence score of each of these 29 customers is very high ($\text{CS}_n > 89$%), with most being in the 96-97% range. According to the network model, the 29 customers belong to the same lateral. After viewing these results, the utility realized that they had planned to move this lateral to phase B and changed the model accordingly but had forgotten to do it in the field. ESC-GIS therefore correctly identified an entire mislabeled lateral. The phase identification algorithm also predicted that six customers initially connected to phase C should be on phase B, with relatively low CS_n ranging from 72.2% to 75.7%. According to the network model, all six customers are directly tapped to three-phase lines in the same part of the feeder. The utility did not confirm nor deny this prediction.

Table 12-3. Results with the utility data set using ESC-GIS

		Initial Phase (GIS)			
		A	**B**	**C**	**Total**
Predicted Phase (ESC-GIS)	**A**	124	0	0	124
	B	0	78	6	84
	C	0	29	53	82
	Total	124	107	59	290

In the spectral clustering algorithm, the final k-means stage – which finds clusters of customers on the same phase – uses the eigenvectors of a Laplacian as the input. Each eigenvector is a representation of a single customer. Visualizing them is another means of assessing the prediction quality.

The first three non-trivial dimensions of the eigenvectors of the utility data set are plotted in Figure 12-9 for a single window with few missing measurements. Most customers are neatly grouped in three clusters, implying a strong correlation between the voltage time series of customers connected on the same phase, thus giving more confidence in the predicted results. The six customers initially labeled on phase C discussed in the previous paragraph belong to a compact cluster mostly comprised of phase B meters (bottom left of Figure 12-9), strengthening the prediction. The 29 customers belonging to the mislabeled lateral are also grouped compactly with the majority of customers initially assigned on phase C (bottom right).

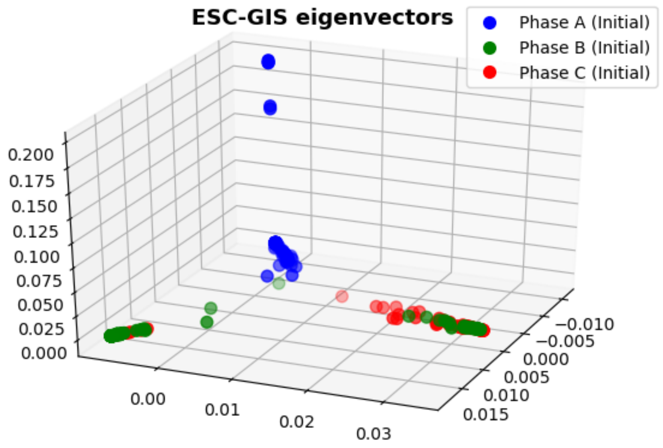


Figure 12-9. Eigenvectors of ESC-GIS on the utility data set for a single window

12.2.1.5. Discussion

12.2.1.5.1. Voltage Regulators

As outlined in Section 12.2.1.3, voltage-based phase identification methods are very sensitive to the presence of multiple voltage regulators, especially when both AMI and SCADA measurements are used. The reason is intuitive. Many voltage-based methods are derived on the assumption that the voltage variations are more similar between customers on the same phase than on different phases. Because voltage regulators cause large variations that are only seen by downstream customers, the voltage correlation between customers on the same phase but on other sides of the tap changer is weakened.

To visualize this explanation, the first three non-trivial dimensions of the eigenvectors obtained by ESC-GIS on North #2 and North #2 (fixed) under ideal conditions are shown in Figure 12-10 and Figure 12-11, respectively.

Due to the voltage regulators, many more clusters can be observed in Figure 12-10 than in Figure 12-11; nevertheless, same-phase customers are mostly grouped together, enabling proper delimitation of clusters. While clusters of the same phase are mostly closer to each other, one group of customers on phase A is closer to clusters of phases B and C. This makes it challenging to correctly map each cluster with the corresponding SCADA phase voltage measurement. Finally, as observed in Section 12.2.1.3, voltage regulators do not impact power-based methods since they have a negligible impact on load power consumption. It is therefore suggested to either use power-based methods (e.g., LASSO and SF) or voltage-based methods that do not rely on SCADA measurements (e.g., ESC-GIS) for networks with multiple voltage regulators.

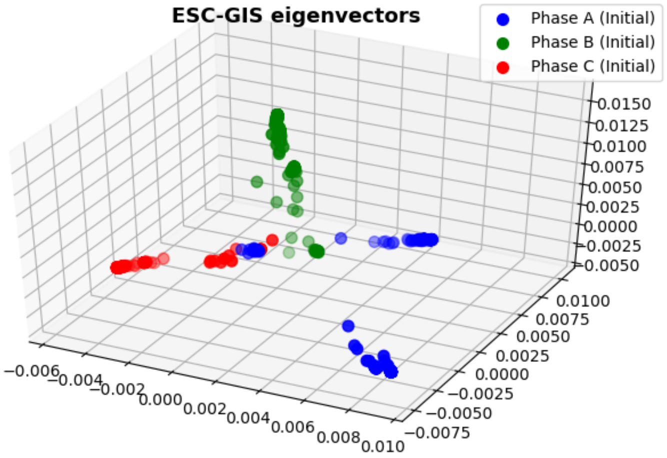


Figure 12-10. Eigenvectors of ESC-GIS on a network with active tap changers (North #2) for a single window.

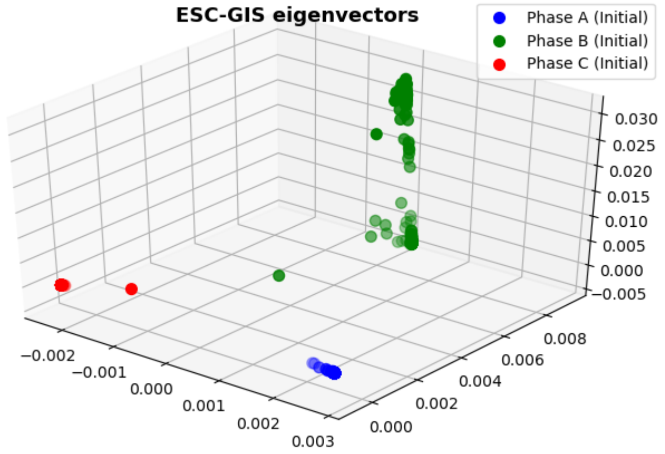


Figure 12-11. Eigenvectors of ESC-GIS on a network with fixed tap changers (North #2 [fixed]) for a single window

12.2.1.5.2. Number of Clusters

It could appear that using one cluster per phase in the k-means stage of ESC-GIS and ESC-SCADA is sufficient for networks with all customers connected between line and neutral. However, as seen in Figure 12-10, all customers connected to the same phase are not necessarily grouped in one cluster even under ideal conditions. This is particularly manifest for feeders with multiple voltage regulators. Consequently, more clusters are used with ESC-GIS and ESC-SCADA than the number of phases (i.e., 12 in Section 12.2.1.3 and six in Section 12.2.1.4).

To further substantiate this claim, the study of Section 12.2.1.3.1 is repeated with ESC-GIS using only three clusters, i.e., one per phase. The accuracy of North #1 and North #2 is only 68.9% and 66.4%, respectively; whereas $\mathrm{Acc}_{\%} > 99\%$ for all five other networks. Accuracies greater than 99% were obtained for North #1 and North #2 using 12 clusters. On the other hand, using too many clusters may also deteriorate the accuracy, especially for networks with fewer customers. As an example, the study of Section 12.2.1.4 is repeated with 12 clusters instead of six. Only six of the 29 customers belonging to the mislabeled lateral discussed in Section 12.2.1.4 are correctly identified when using 12 clusters; whereas all are identified with only six clusters. The creation of smaller clusters within the ESC-GIS algorithm may therefore also prevent the detection of multiple highly correlated mislabeled customers, e.g., belonging to the same lateral.

12.2.1.5.3. Missing Measurements

Section 12.2.1.3.2 showed and discussed the impact of AMI coverage, where loads were either fully metered or unmetered. In practice, in addition to customers with no AMI metering, AMI voltage and/or power time series will often comprise single or multiple missing measurements, e.g., due to communication failures and system maintenance.

Many phase identification methods require all customers being identified to have a full measurement set for the entire period of study. For methods requiring large data sets to achieve accurate predictions, it can be difficult to define a long enough study period where all meters have no missing measurements.

Ensemble methods such as ESC-GIS and ESC-SCADA therefore have a practical advantage. In these methods, multiple smaller windows (e.g., 4 days) are studied individually. If one or multiple measurements are missing, the customer is only ignored for the corresponding window(s). The final voting stage is still applied, except with fewer windows for customers with missing measurements. However, caution must be taken when including windows with several excluded customers, as they are more prone to yield inaccurate predictions. This may unintentionally decrease the confidence scores.

12.2.1.5.4. Measurement Distortion

The study of Section 12.2.1.3.5 considered the impact of measurement noise on accuracy by adding different levels of uniformly distributed noise to otherwise ideal and synchronized measurement sets. While informative, this does not represent the full spectrum of measurement distortions that can affect practical phase identification studies. For instance, the set of SCADA and/or AMI measurements used at a given time step may not be synchronized. These measurements may represent instantaneous values at the end of the interval, peak or average values for the entire interval, etc. Moreover, many SCADA meters provide power measurements while AMI systems often send energy measurements, requiring approximations to convert to equivalent power measurements.

12.2.1.5.5. Sample Counts

Section 12.2.1.3.4 showed that unlike with SF, a relatively small number of samples (e.g., 1,000) is needed for voltage-based methods to reach their peak accuracy. This is to be expected because SF identifies all customers individually (through their salient variations) as opposed to grouping them. It therefore requires a considerable number of samples to find sufficient high-frequency components (salient events) for each AMI meter to be associated with the corresponding SCADA meter.

While several methods can handle years of measurements with little computational cost, using too much data comes with practical concerns. For one, customer phases are not always static. An example thereof is provided in Section 12.2.1.4. Another issue is algorithms that can only use complete data sets (see Section 12.2.1.5.3). Longer periods increase the possibility of having to remove customers due to missing measurements.

12.2.2. *Secondary System Topology and Parameter Estimation*

12.2.2.1. Task Overview

End customers are generally served by connecting to a service transformer on a secondary distribution system. However, for some utilities, such a secondary distribution system is not modeled or is only modeled with limited details in their networks. This makes it difficult to study the impacts of connecting behind-the-meter (BTM) technologies (such as electric vehicles, rooftop PVs, etc.) at customers' locations and/or to study the voltage variations and service reliabilities of the secondary distribution systems.

One can build the model for secondary distribution systems from manual inspections, but it requires significant man hours and resources to do so. Alternatively, the availability of voltage and power measurements from smart meters (AMI) enables some methodologies to estimate the topology (a customer is served by connecting directly to the distribution transformer, or there are intermediary nodes between the customer and the distribution transformer) and the parameters of the secondary distribution systems (length and resistance of lines).

The objective of this task is to develop a prototype using the CYME software that meets the following performance requirements/criteria:

- Accuracy: Over 90% success rate to estimate line resistance within +/- 5% error to actual resistance values
- Algorithm efficiency and computation speed: Scalability to large distribution networks (i.e., with over 500 service transformers/secondary networks)
- Robustness
    - Ability to handle "noisy" measurements
    - Ability to handle missing reactive power measurements (due to capabilities of AMI meters)

12.2.2.2. Regression-based Estimation Method

CYME's Python prototype implements the methodology based on [62], which is formulated based on linear regression model for parameter estimation. The method assumes the availability of AMI measurements of voltage, real power and reactive power. The connectivity of each meter to the distribution transformer is also assumed to be known, which is usually available from the GIS system.

For a pair of customers which are connected to a common node, one can write the following equation of approximated voltage drop

$$V_1 - V_2 = I_{R_1} R_1 + I_{X_1} X_1 - I_{R_2} R_2 - I_{X_2} X_2 + \epsilon,$$

where $V_1, V_2 \in \mathbb{R}^{n \times 1}$ are the time series of voltage measurement with n data points for customer 1 and 2 respectively. $I_{R_1}, I_{R_2} \in \mathbb{R}^{n \times 1}$ are the time series of currents computed from real power, and $I_{X_1}, I_{X_2} \in \mathbb{R}^{n \times 1}$ are the currents computed from reactive power for customer 1 and 2 respectively. Finally, $R_1, R_2, X_1, X_2 \in \mathbb{R}$ are the line resistance and reactance to be estimated between the two customers and the common immediate upstream node.

Based on the approximated voltage drop equation, one can write the following regression model

$$y = \mathbf{X}\beta + \epsilon, \text{ where}$$

$$\mathbf{X} = \left[I_{R_1},\ -I_{R_2},\ I_{X_1},\ -I_{X_2}\right] \in \mathbb{R}^{n\times 4}$$

$$\beta = [R_1,\ R_2,\ X_1, X_2]^T \in \mathbb{R}^{4\times 1}$$

$$y = V_1 - V_2 \in \mathbb{R}^{n\times 1}$$

Using the ordinary least squares (OLS) estimator, one can estimate the parameter β by

$$\beta = (X^T X)^{-1} X^T y$$

Note that the model can be modified if reactive power measurement is not available (hence I_{X_1}, I_{X_2} cannot be computed). However, to estimate the line reactance, the X/R ratio must be assumed.

The following is a high-level sketch (illustrated in Figure 12-12) of how the methodology is systematically applied to a distribution network.

Step 1

1.1. For all meters that are served by a transformer, get all combinations of meter pairs.

1.2. For each pair of meters, use the OLS estimator to estimate the parameters and calculate the Pearson coefficient.

1.3. After Pearson coefficient is computed for all combinations, pick the pair which has the highest value which indicates best fit.

1.4. Replace the pair of customers by a "virtual node/meter" and approximate the voltage and power data for this "virtual node". Repeat 1.1 until a single "virtual node" is left for this transformer. (see an example in Figure 12-13 – Step 1)

1.5. Repeat 1.1-1.4 for all transformers in the network.

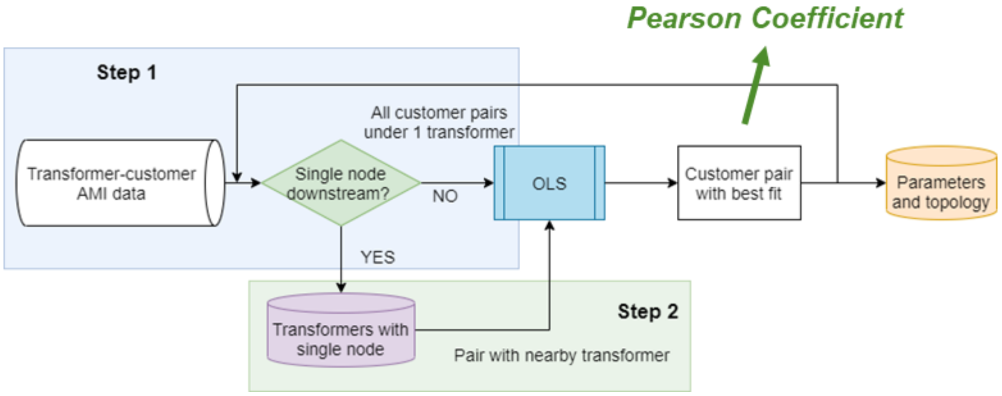

Pearson Coefficient
Step 1
All customer pairs under 1 transformer
Transformer-customer AMI data
Single node downstream?
NO
OLS
Customer pair with best fit
Parameters and topology
YES
Step 2
Transformers with single node
Pair with nearby transformer

Figure 12-12. High-level sketch of the regression-based estimation method

Step 2

2.1. Each transformer in the network should now have only 1 "virtual node". (Note, a few technical assumptions are made in order to apply the OLS estimator in Step 2. Refer to [62] for details)

2.2. Pair 1 transformer with another transformer in a specified neighborhood (by electrical distance) and on the same phase (multiple pairs may be possible).

2.3. For all pairs identified in 2.2, use the OLS estimator to estimate the parameters between the single "virtual node" and the transformer node (see an example in Figure 12-13 – Step 2). Pick the pair that has the best fit.

2.4. Repeat 2.1-2.3 for all transformers in the network.

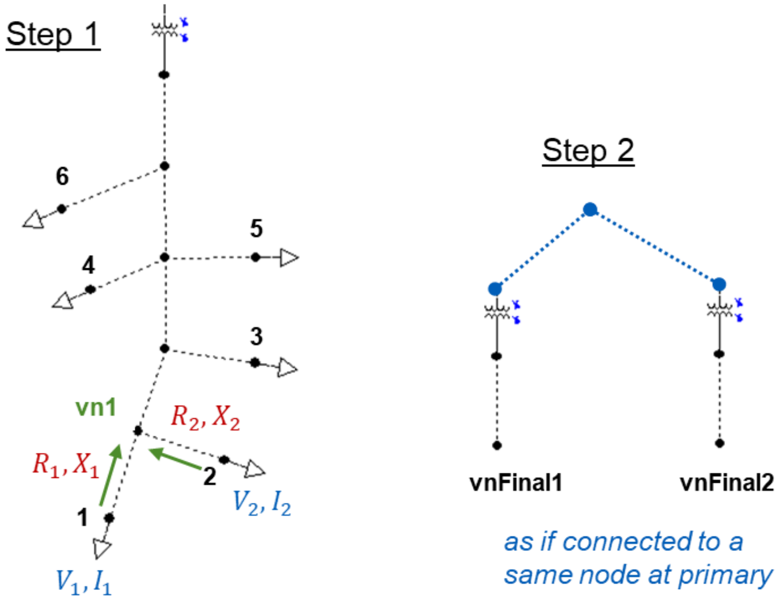


Figure 12-13. Illustrative example for Step 1 and 2 of the parameter estimation methodology

12.2.2.3. Synthetic Results

In testing the methodology and the prototype, the CKT5 network of EPRI is adopted with some modifications.

1. Lines of secondary distribution networks in the original CKT5 network have the same length. For testing purposes, lines are modified to have different lengths.
2. Some distribution transformers are selected to add a short line between the secondary node of the transformer and its first immediate downstream section. Line lengths are assigned randomly for different transformers.

Synthetic measurement data for 110 days with 15-minute intervals is used. Voltage measurements are generated from synthetic active and reactive power profiles by solving time-series power flows on the CKT5 network.

12.2.2.3.1. Noiseless measurement data

The following scenarios were run to test the performance of the prototype, assuming noiseless measurement data.

a. Use voltage (V), real power (P) and reactive power (Q) data

b. Use only V and P data, and X/R ratio assumed to be 1:4.678 (which is the actual ratio for the lines used in CKT5)

c. Use only V and P data, and X/R ratio assumed to be 1:3

d. Use only V and P data, and X/R ratio assumed to be 1:5

e. Use only V and P data, and X/R ratio assumed to be 1:7

Figure 12-14 shows the histograms of empirical measure of errors observed between the estimated and actual line resistance for all scenarios (*for better visualization purpose, the y-axis is capped at 300 for all histograms*). The legend entries show the success rates to have <5% estimation errors for all scenarios.

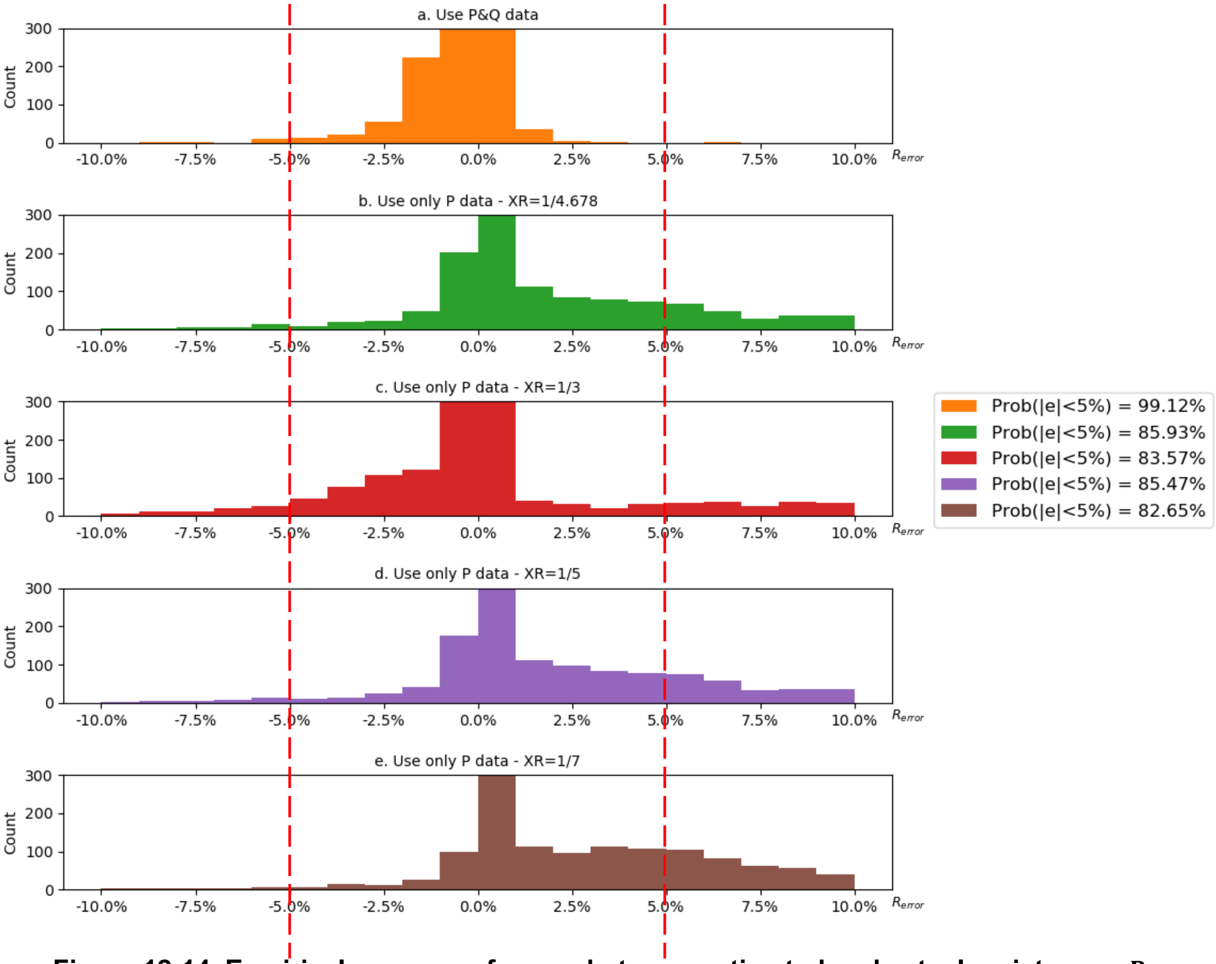


Figure 12-14. Empirical measure of errors between estimated and actual resistances *R*

It is observed from Figure 12-14 that,

- The availability of Q data results in an over 99% of success rate to have <5% estimation errors
- In absence of Q data, success rate is lower than when Q data is available, even when using the true X/R ratio
- In absence of Q data, as the assumed X/R ratio moves away from the true value, the success rate of estimation decreases

12.2.2.3.2. Estimation of X/R ratio of line impedance

In the previous section, in absence of reactive power measurement, the X/R ratio must be assumed to apply the prototype to estimate the line resistance and reactance. The estimation accuracy depends on how close the assumed X/R ratio is to the actual value. CYME developed a novel approach to estimate the X/R ratio of the line impedance from the voltage and active power measurement data, based on *Extended Kalman Filter (EKF)* estimation.

The EKF model

Denote the state variable $x(k) = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6\,]^T$, where x_1 is the voltage difference of the customer pair, x_2 and x_3 are the line resistance of customer 1 and customer 2 respectively, x_4 is the line X/R ratio to estimate, and x_5 and x_6 are the mean ratio of reactive power to active power of customer 1 and 2 respectively.

Denote the output measured $z(k) = [V_1(k) - V_2(k) \quad \widetilde{R_1} \quad \widetilde{R_2}]^T$, where V_1 and V_2 are the measured voltage data of customer 1 and 2 respectively, $\widetilde{R_1}$ and $\widetilde{R_2}$ are the estimated line resistance from linear regression for customer 1 and 2 respectively. The active power measurement for customer 1 and 2 are also given as P_1 and P_2 respectively.

Then we can derive the following state-space model:

$$x(k+1) = \Phi[x(k), k] + \Gamma\, w(k)$$

$$z(k) = H[x(k), k] + v(k)$$

where $\Phi[x(k), k] = \begin{bmatrix} -I_{r1}(k)x_2(k)\,(1 + x_4(k)x_5(k)) \;+\; I_{r2}(k)x_3(k)(1 + x_4(k)x_6(k)\,) \\ x_2(k) \\ x_3(k) \\ x_4(k) \\ x_5(k) \\ x_6(k) \end{bmatrix}$

$$H[x(k), k] = \left[x_1(k) \quad \frac{1}{k}\sum_{i=0}^{k} x_2(i) \quad \frac{1}{k}\sum_{i=0}^{k} x_3(i)\right]^T$$

$$\Gamma = \mathcal{I}_{6\times 6}$$

$$\mathbb{E}\{w(k)\} = \mathcal{O}_{6\times 1}$$

$$\mathbb{E}\{v(k)\} = \mathcal{O}_{3\times 1}$$

$$cov\{w(k), w(j)\} = Q \in \mathbb{R}^{6\times 6}, k \neq j$$

$$cov\{v(k), v(j)\} = R \in \mathbb{R}^{3\times 3}, k \neq j$$

Here, $\mathcal{I}$ and $\mathcal{O}$ refer to identity and zero matrices respectively. In $\Phi[x(k), k]$, I_{r1} and I_{r2} are considered given as control inputs and can be calculated by P_1/V_1 and P_2/V_2, respectively.

At $k = 0$, the algorithm of EKF is initialized with $\hat{x}(0) = [0,\ 0.01,\ 0.01,\ 0.2,\ 0, 0]^T$.

1. Predict states at $(k+1)$ using $\hat{x}(k+1|k) = \Phi[\hat{x}(k), k]$
2. Calculate

$$V(k+1|k) = \frac{\partial \Phi}{\partial \hat{x}} V(k) \frac{\partial \Phi^{\mathrm{T}}}{\partial \hat{x}} + \Gamma\, Q\, \Gamma^T$$

3. Calculate

$$P(k+1) = \left[\frac{\partial H}{\partial \hat{x}} V(k+1|k) \frac{\partial H^T}{\partial \hat{x}} + R\right]^{-1}$$

$$V(k+1) = V(k+1|k) - V(k+1|k) \frac{\partial H^T}{\partial \hat{x}} P(k+1) \frac{\partial H}{\partial \hat{x}} V(k+1|k)$$

4. Calculate Kalman filter gain

$$K(k+1) = V(k+1) \frac{\partial H^T}{\partial \hat{x}} R^{-1}$$

5. Update the prediction of states at $(k+1)$ using

$$\hat{x}(k+1) = \hat{x}(k+1|k) + K(k+1)\{z(k+1) - H[\hat{x}(k+1|k), k+1]\}$$

6. Repeat Step 1 by setting $k = k+1$

Figure 12-15 shows the empirical distributions of estimated X/R ratio by the EKF model, using 15-min interval measurements of 2 days, 10 days, 20 days, 30 days, 42 days, and 68 days, respectively.

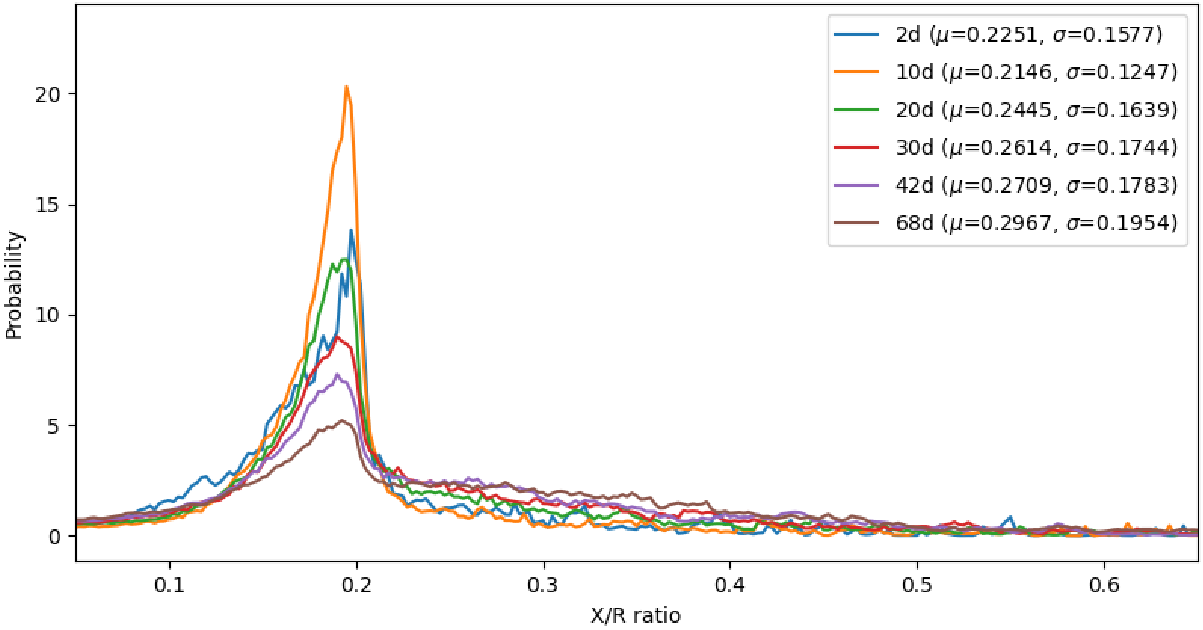


Figure 12-15. Empirical distributions of estimated X/R ratio from EKF estimation

It is observed from Figure 12-15 that,

- In all scenarios, an observable spike of probabilities occurs near X/R = 0.2, which is the true X/R value
- The scenario using 10 days data gives the best estimation with the least error % and the least deviation
- The estimation is less accurate when there is not enough data for the estimation process (i.e. using only 2 days data)

- Unlike linear regression, it is not always beneficial to have a large amount of data for the EKF model. In this experiment, the estimation is less accurate and starts to have increasing error % as the number of data increases. This is mainly due to the fact that in the EKF model, we use the mean Q/P in the state vector. As the number of data increases, if the actual Q/P ratio has a larger variation from the mean value (which is the case for the synthetic data used in the test cases), the estimated X/R values are more likely to deviate, hence making the estimation less accurate (see Figure 12-16 as an example)

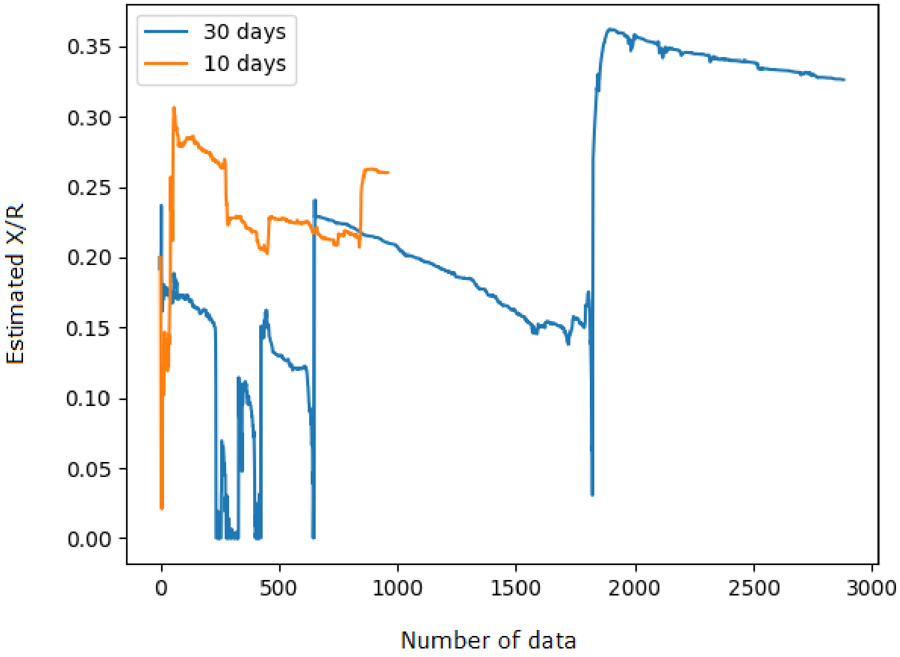


Figure 12-16. Estimated X/R ratio from the EKF model using 10-days vs. 30-days data of *P* and *V*

12.2.2.3.3. Noisy measurement data

In this section, the prototype is evaluated for robustness to measurement errors in the V, P, and Q data which is assumed available. The following scenarios are considered:

a. Use $P, Q,$ and V noiseless data (baseline case)

b. Noiseless V data, P, Q data with 5% errors (noises with a uniform distribution)

c. Noiseless V data, P, Q data with 10% errors (noises with a uniform distribution)

d. Noiseless V data, P, Q data with 20% errors (noises with a uniform distribution)

e. V data with ±0.25V errors, P, Q data with 5% errors (all noises have uniform distributions)

f. V data with ±0.5V errors, P, Q data with 5% errors (all noises have uniform distributions)

g. V data with ±0.25V errors, P, Q data with 10% errors (all noises have uniform distributions)

h. V data with ±0.25V errors, P, Q data with 20% errors (all noises have uniform distributions)

Figure 12-17 shows the histograms of empirical measure of errors observed between the estimated and actual resistance for all scenarios above.

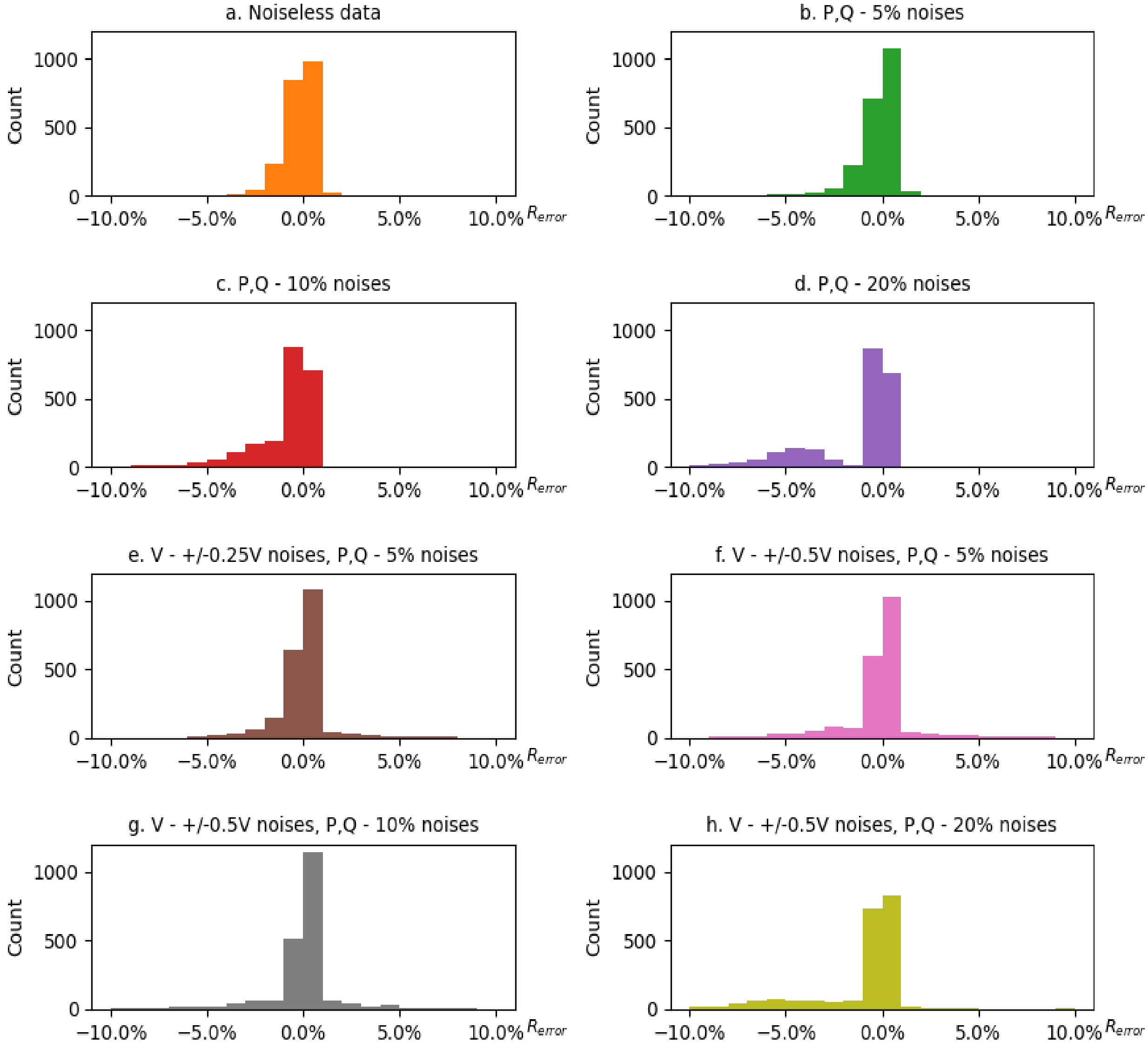


Figure 12-17. Empirical measure of errors between estimated and actual resistances *R*

Table 12-4 summarizes the success rates of <5% estimation errors in Figure 12-17 and compares with that of the baseline scenario.

Table 12-4. Results of success rate with noisy measurement data

Scenarios	Success Rate (%)	
	Results (<5% errors)	Delta
a. Noiseless $\boldsymbol{P}, \boldsymbol{Q}$, and $\boldsymbol{V}$ data (Baseline)	**99.12**	**-**
b. Noiseless $\boldsymbol{V}$ data, $\boldsymbol{P}, \boldsymbol{Q}$ data with 5% errors	98.58	-0.54
c. Noiseless $\boldsymbol{V}$ data, $\boldsymbol{P}, \boldsymbol{Q}$ data with 10% errors	96.53	-2.59
d. Noiseless $\boldsymbol{V}$ data, $\boldsymbol{P}, \boldsymbol{Q}$ data with 20% errors	87.23	-11.89
e. $\boldsymbol{V}$ data with ±0.25V errors, $\boldsymbol{P}, \boldsymbol{Q}$ data with 5% errors	97.22	-1.90
f. $\boldsymbol{V}$ data with ±0.5V errors, $\boldsymbol{P}, \boldsymbol{Q}$ data with 5% errors	92.83	-6.29
g. $\boldsymbol{V}$ data with ±0.5V errors, $\boldsymbol{P}, \boldsymbol{Q}$ data with 10% errors	90.83	-8.29
h. $\boldsymbol{V}$ data with ±0.5V errors, $\boldsymbol{P}, \boldsymbol{Q}$ data with 20% errors	85.89	-13.23

It is observed from Table 12-4 that,

- If the voltage measurement is accurate, the prototype can tolerate up to 20% measurement errors in the active and reactive power data to maintain ~90% estimation accuracy.
- If all measurement data are noisy, the prototype can tolerate up to 0.5V errors in voltage measurements, and at the same time, up to 10% errors in active and reactive power data to maintain ~90% estimation accuracy.

CYME's prototype has been demonstrated to have met the objective and milestones of this task. Some further work is possible to explore related to this task, especially on integrating the Kalman filtering concept into the parameter estimation prototype to enhance the robustness when reactive power data is missing (an iterative approach may be necessary, and the convergence is to be studied).

12.2.3. Meter-to-transformer Pairing

12.2.3.1. Task Overview

For the tasks of phase identification and parameter estimation of secondary distribution systems, we use the AMI measurement data of customers on the distribution network. It is assumed that the connectivity information of each customer and its serving distribution transformer is known, which is usually from utilities' GIS records. However, in certain situations, the connectivity information in GIS records may be wrong, i.e. a customer is labelled to be served by transformer A, but in reality the customer is served by transformer B. Such inaccuracies may happen due to repairs and/or network reconfiguration without updating the GIS records properly.

The accuracy of GIS records is important to achieve highly accurate results for the tasks of phase identification and parameter estimation of secondary distribution systems. One can do field verification to verify the connectivity information of meter and the service transformer for all GIS records, but this requires significant man hours and resources to do so, especially for a large distribution network. Alternatively, one can adopt an approach to estimate the connectivity pairing of customer meter to its service transformer by leveraging the AMI measurement data.

CYME's objective is to develop a prototype that can estimate the customer meter-to-transformer pairing information based on AMI measurement data. The prototype uses the CYME software and should meet the following performance requirements/criteria:

- Accuracy: Over 90% success rate in identifying the mapping of meters to distribution transformers
- Algorithm efficiency and computation speed: Scalability to large distribution networks (i.e. with over 500 distribution transformers / secondary networks)
- Robustness: Ability to handle "noisy" measurements

12.2.3.2. Methodology

Two algorithms are implemented in CYME's prototype.

12.2.3.2.1. ICA method

The first algorithm ("ICA") is based on the work by Wu et al [114], where Fast Independent Component Analysis (FastICA) technique is applied to extract features from the time series of voltage data of all meters in the neighborhood, and the features (denoted by a mixing matrix) are then clustered using the Kernel K-means. The mixing matrix is time-independent and reflects the similarity between voltage data series. Customers identified within the same cluster are labelled to be served by the same transformer, which is determined by the most common transformer according to the existing GIS records.

The following describes a high-level sketch of the methodology.

1. Pick a meter m and its transformer xfo_m labelled in GIS record
2. Search for nearby K transformers based on geographic distance to meter m
3. Collect all N meters that are labelled in GIS record to connect to any transformer in K
4. Collect voltage measurement data X_t for all N meters, where each meter has M data points
5. Use FastICA method to compute the mixing matrix A, where

   $X_t = AS_t + \overline{X}_t e^T$, where $X_t, S_t \in R_{N\times M}$, $A \in R_{N\times N}, \overline{X_t} \in R_{N\times 1}$, $e \in 1_{N\times 1}$
6. Use the Kernel K-means and silhouette scores to determine the proper number of clusters in A, denoted as k^*
7. For each cluster, collect the list of connected transformers in GIS for all meters in this cluster and compute their empirical counts.
   1. If $k^* \leq K$, append the collected the list of transformers as well as their empirical counts to each meter, and consolidate the list/counts.
   2. If $k^* > K$, add an '*unknown*' transformer to the list with a count of '*uc*' (which is a percentage of the highest empirical count among all transformers in this cluster), append the list as well as empirical counts of transformers to each meter, and consolidate the list/counts

8. Repeat Step 1 until all meters in the network are picked
9. Each meter should now have a list of transformers associated with empirical counts
    1. Label a meter with the transformer that has the most count
    2. If the transformer "*unknown*" has the most count, label this meter for "field verification required"
    3. If the top two transformers have similar counts (within some threshold), then also label with meter for "field verification required"

The following illustrates a simple example for the ICA technique.

The top chart in Figure 12-18 shows time-series voltage measurements for 3 meters (out of a collection of $N = 40$ meters) with $M = 50$ data points each shown, where 2 meters are connected to the same transformer (blue and orange), and the third meter (green) is connected to a different transformer.

The ICA analysis extracts a total of 40 features from the time-series data, as shown in the bottom right chart of Figure 12-18, as well as a mixing matrix with support on the 40 extracted features. The chart at bottom left of Figure 12-18 shows the mixing weights for each feature for the 3 customers. Mixing weights for blue and orange customers are quite similar because they are served by the same transformer, while the mixing weights for the green customer look different.

Based on the differences of mixing weights, customers on the same transformers will likely be grouped into the same cluster by the Kernel K-means.

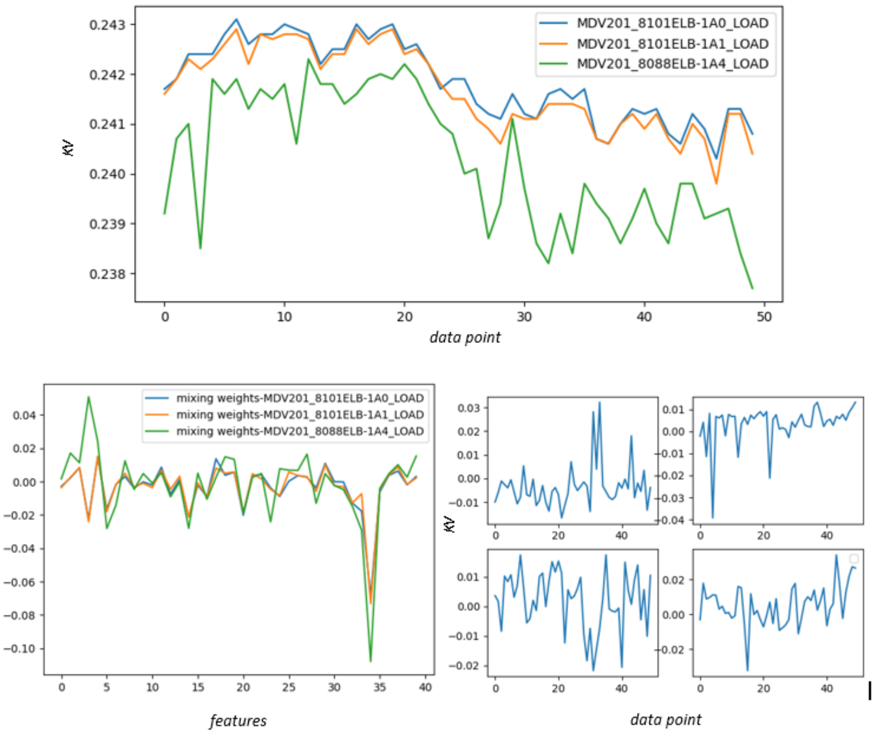


Figure 12-18. Time series voltage measurement of 3 meters (top); mixing weights corresponding to the 3 meters (bottom left); and features extracted by ICA (bottom right)

12.2.3.2.2. R2 Method

The second algorithm ("R2") is based on the work by Luan et al [115], where the Pearson correlation coefficient R2 is calculated for any two series of voltage measurements. R2 coefficients are calculated pair-wisely among all meters. The coefficients are then ranked to determine the transformer to which each meter is connected.

Note that the original R2 algorithm is limited to only two transformers. In reality, when a meter is incorrectly labeled in the GIS records, to find the correct transformer it connects to, more than 2 transformers should be searched for in the neighborhood. Because of this, the algorithm has been improved to handle this situation.

The high-level sketch of this algorithm is similar to that of ICA algorithm, except that a correlation matrix W for a group of meters is used for clustering rather than the mixing matrix A (Step 5 and 6).

$$W_{N\times N} = \begin{bmatrix} 1 & r_{12} & \cdots \\ r_{21} & 1 & \cdots \\ \vdots & \vdots & \vdots \\ r_{n1} & r_{n2} & \cdots \end{bmatrix},\ where\ r_{ij} = \begin{cases} \dfrac{\sum(x_i - \overline{x_\iota})(x_j - \overline{x_J})}{\sqrt{\left(\sum(x_i - \overline{x_\iota})^2\ \sum\left(x_j - \overline{x_J}\right)^2\right)}}, & if\ i < j \\ r_{ji}, & if\ i > j \end{cases}$$

Here, x_i and x_j are the time-series voltage measurement data for meters i and j.

For the same time-series voltage measurements in the example of ICA (as in top chart of Figure 12-18), Figure 12-19 shows the R2 coefficients for each of the 3 meters to all other meters in the meter collection. As expected, the blue and orange customers have similar coefficient curves, where the green customer has a different one.

Again, based on the differences of the R2 coefficients, customers on the same transformers will likely be grouped into the same cluster by the Kernel K-means.

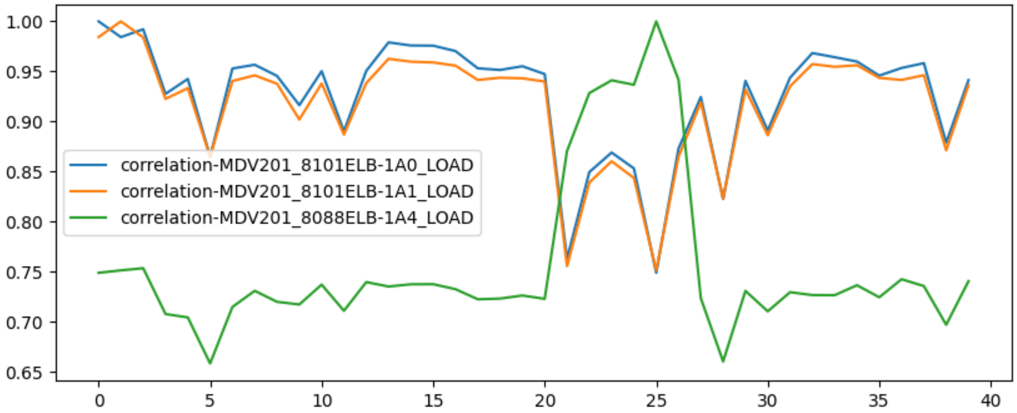


Figure 12-19. R2 coefficients between each of the 3 meters selected and all other meters in the group

12.2.3.2.3. Kernel K-means

As mentioned in the two algorithms, Kernel K-means technique is used for clustering the mixing matrix for the ICA method and the R2 coefficient matrix for the R2 method.

A brief discussion on the difference between the standard K-means and the Kernel K-means is given below. The objective function used in the standard K-means can be written as,

$$f_{kmeans} = \sum_{j=1}^{k} \sum_{x_i \in C_j} \left\| x_i - \mu_j \right\|^2, \mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i.$$

Here C_j refers to the cluster j, x_i refers to the vector of data point i belonging to cluster j, and μ_j is the center of C_j. The vector x_i is often of high dimension (in relation to the total number of meters that are clustered), and the Euclidean distance used in standard K-means is known to lose its meaning in measuring uncorrelated variables in a high dimensional space. This may result in clusters that are too close to separate. To overcome this difficulty, instead of using a summation of point-wise squared errors of x_i to the cluster center in the objective function, we map x_i into a higher dimensional metric space equipped with inner product, denoted by a function ϕ. The objective function is then modified to the following,

$$f_{kernel} = \sum_{j=1}^{k} \sum_{x_i \in C_j} \left\| \phi(x_i) - \mu_j \right\|^2, \mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} \phi(x_i).$$

To compute the $\left\| \phi(x_i) - \mu_j \right\|^2$ term in the equation above for a cluster C, we can expand the norm to,

$$\left\| \phi(x_i) - \mu_j \right\|^2 = \langle \phi(x_i), \phi(x_i) \rangle - \frac{2}{|C|} \sum_{x_j \in C} \langle \phi(x_i), \phi(x_j) \rangle + \frac{1}{|C|^2} \sum_{x_j, x_k \in C} \langle \phi(x_j), \phi(x_k) \rangle.$$

Here, the operator $\langle , \rangle$ denotes the inner product of two vectors, which can be represented by a kernel function (Gaussian kernel, polynominal kernel, etc.). Hence the objective function can be evaluated without an explicit mapping function ϕ, and the algorithm of standard K-means clustering can still be adopted. In this implementation, the Gaussian kernel function is used.

12.2.3.3. Synthetic Results

The same test network as well as the synthetic AMI measurement dataset are used to test both algorithms in CYME's prototype. However, only the voltage measurements are necessary, which are generated from synthetic active and reactive power profiles by solving time-series power flows for 30 days with a 15-minute interval on the CKT5 network.

12.2.3.3.1. Test with Ground Truth

For the initial test, we assume that all meter-to-transformer pairings are correctly labeled in the GIS records (the "Ground Truth"). Then from the voltage measurement data, we estimate the transformer that each customer meter is connected to. Finally, we compute the accuracy rate as the number of correctly estimated pairings over the total number of meter-to-transformer pairings in the network. In CKT5, there are 1,373 pairings in total, hence

$$accuracy = \frac{number\ of\ correctly\ estimated\ pairing}{1373} \times 100\%$$

Figure 12-20 and Table 2-1 show the accuracy results of the two algorithms for different numbers of transformers searched in the neighborhood (up to 10 transformers). We also compare with the results obtained by using Kernel K-means versus the standard K-means clustering.

The figure and the data table below show that accuracies remain constant as the number of transformers in the neighborhood is increased when using Kernel K-means, especially the ICA method. The reduction of accuracy using the standard K-means can be explained by the minor difference in Euclidean distance between clusters. As more transformers in the neighborhood are being searched, meters have more possible clusters are possible to group into, which increases the possibility of false estimation.

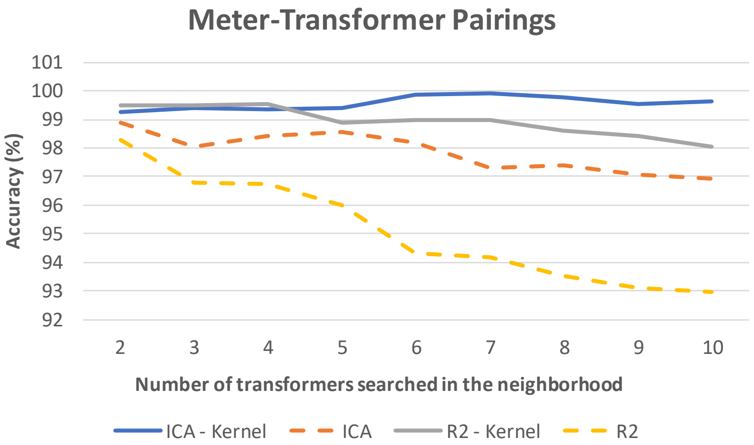


Figure 12-20. Accuracy of meter-to-transformer pairing algorithms using 30-days measurement (with comparison between Kernel K-means and K-means)

Table 12-5. Results of accuracy of meter-to-transformer pairing algorithms

Method	Accuracy (%)								
Nb. of xfos	2	3	4	5	6	7	8	9	10
ICA - Kernel	99.67	99.42	99.34	99.42	99.85	99.93	99.78	99.56	99.64
ICA	98.91	98.03	98.40	98.54	98.18	97.31	97.38	97.09	96.94
R2 - Kernel	99.49	99.49	99.56	98.91	98.98	98.98	98.62	98.40	98.03
R2	98.27	96.80	96.72	95.99	94.32	94.17	93.52	93.08	92.94

12.2.3.3.2. Test with Incorrect GIS Records

For this experiment, we test the algorithms for a couple of scenarios with different levels of incorrect pairings in the existing GIS records. Ten transformers are searched for to collect the meters in the neighborhood. For each meter, a list of possible transformers is returned, with the empirical counts (probabilities) of transformers that the meter may connect to. From these probabilities, one can have two possible outcomes for each meter.

1. **Confirmed**. There is a strong indication of 1 transformer, which has a high empirical probability. For this outcome, if we compare this identified transformer with the "ground truth", we will further have two possibilities.
   a. Successful identification: the identified transformer is the true transformer that serves the meter, regardless of whether the existing pairing information in GIS is correct or not (for example, the upper graph of Figure 12-21)

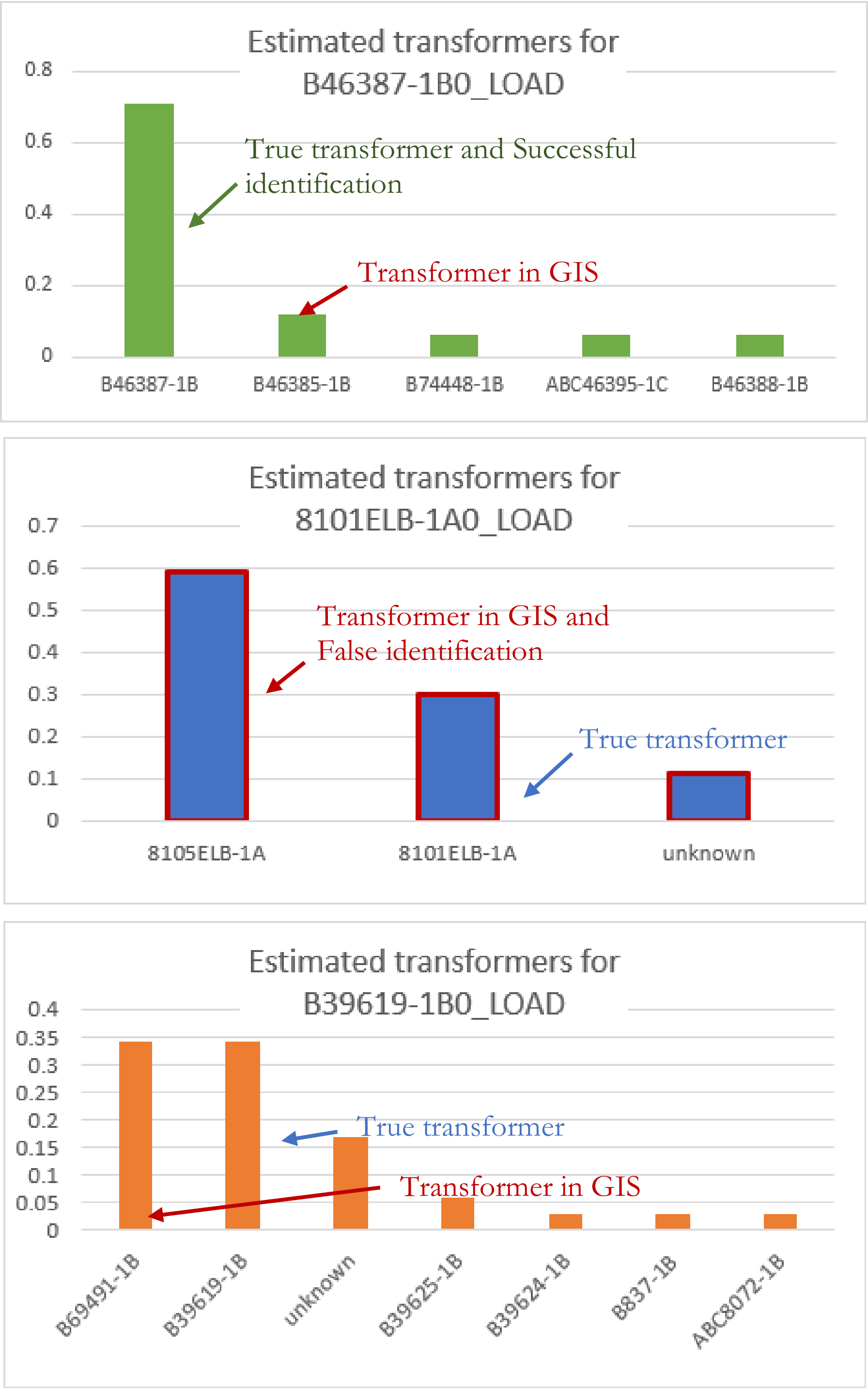
   b. False identification: the identified transformer is not the true transformer serving the customer (for example, the middle graph of Figure 12-21)

2. **Flag for further verification**. In this outcome, the algorithm identifies at least two possible transformers that have similar probabilities (for example, the bottom graph of Figure 12-21), or the “*unknown*” transformer is returned with a high probability. Hence further verification in field is required to determine the correct transformer.

Table 12-6 shows the results of both algorithms for different levels of incorrect entries in GIS records up to 30%. As the number of incorrect pairings in GIS increases, the percentage of pairings flagged by the algorithms for further verification increases linearly. If we assume that flagged pairings will be confirmed by field verifications for the correct pairing, we can include them into the rate of successful identification, which increases the overall successful rate to close to 95% even for the 30% case. Overall, the ICA method performs slightly better than the R2 method.

Table 12-6. Accuracy of meter-to-transformer pairing algorithms with inaccurate GIS records

GIS Error %	Method	Confirmed		Successful Identification		False Identification		To Verify	
0%	ICA	1366	*99.49%*	1363	*99.27%*	3	*0.22%*	7	*0.51%*
	R2	1339	*97.52%*	1329	*96.80%*	10	*0.73%*	34	*2.48%*
10%	ICA	1278	*93.08%*	1263	*91.99%*	15	*1.09%*	95	*6.92%*
	R2	1254	*91.33%*	1235	*89.95%*	19	*1.38%*	119	*8.67%*
20%	ICA	1183	*86.16%*	1146	*83.47%*	37	*2.69%*	190	*13.84%*
	R2	1150	*83.76%*	1108	*80.70%*	42	*3.06%*	223	*16.24%*
30%	ICA	1072	*78.08%*	1007	*73.34%*	65	*4.73%*	301	*21.92%*
	R2	1057	*76.98%*	982	*71.52%*	75	*5.46%*	316	*23.02%*


Estimated transformers for B46387-1B0_LOAD

True transformer and Successful identification

Transformer in GIS

Estimated transformers for 8101ELB-1A0_LOAD

Transformer in GIS and False identification

True transformer

Estimated transformers for B39619-1B0_LOAD

True transformer

Transformer in GIS

Figure 12-21. Illustration of possible outcomes by the algorithms (top – successful detection; middle – false detection; bottom – flag for further verification)

12.2.3.3.3. Test with Incorrect GIS Records and Noisy Measurements

In the last experiment, noise is added to the voltage measurements to test the robustness of the algorithms. The following scenarios are considered:

a. Use noiseless V data (baseline case)

b. V data with zero-mean and 1V standard deviation noises

c. V data with zero-mean and 1.5V standard deviation noises

d. V data with zero-mean and 2.5V standard deviation noises

e. V data with zero-mean and 5V standard deviation noises

Given that the ICA method performs better than the R2 method, only the ICA method is tested with the noisy data, assuming that 30% of meter-to-transformer pairings in GIS records are incorrect. Further, parameters and threshold values are calibrated by trial and error, such that one can optimize the judgment of whether estimated transformers for a meter can be confirmed (whether successful or false) or field verifications are required. This leads to improved results for the baseline case with noiseless data comparing to those in Table 12-6.

Figure 12-22 and Table 12-7 show the results of all test scenarios:

1. Percentages of confirmed and flagged for verifications remain relatively constant independent of noise levels
2. False identification rate increases with the noise levels, hence success rate decreases
3. The algorithm can achieve >90% overall success rate after field verification is done

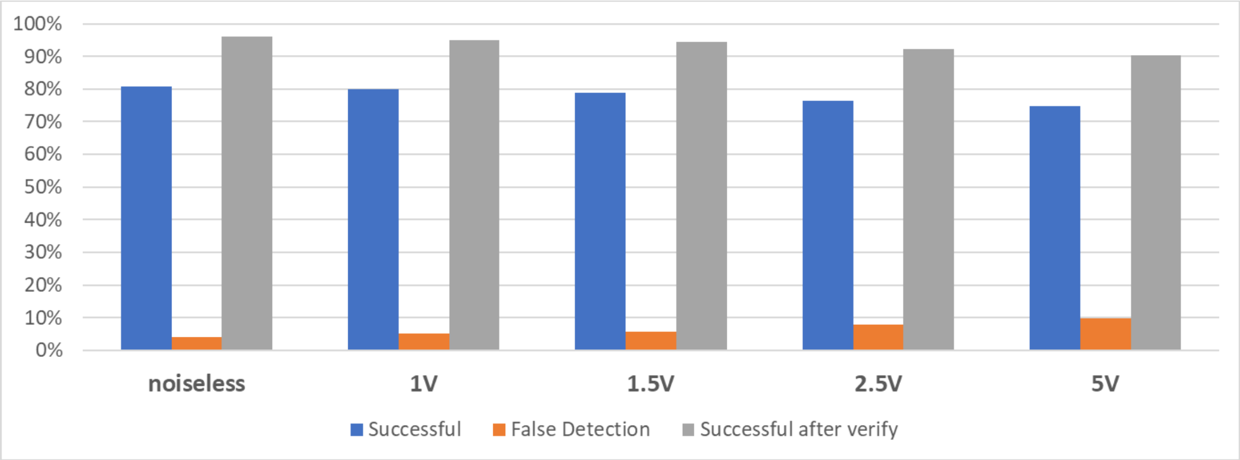


Figure 12-22. Accuracies (success rate, false identification rate, and success rate after field verifications) of ICA method on noiseless and noisy measurement data (with 30% of inaccurate information in GIS records)

Table 12-7. Results of accuracy of meter-to-transformer pairing algorithms with 30% inaccurate GIS records and noisy data

Test Case	Method	Confirmed		Successful Identification		False Identification		To Verify		Successful after verification	
Noiseless	ICA	1165	*84.81%*	1109	*80.77%*	111	*4.04%*	153	*15.19%*	1262	*95.96%*
Noises with 1V std	ICA	1169	*85.14%*	1099	*80.04%*	140	*5.10%*	134	*14.86%*	1233	*94.90%*
Noises with 1.5V std	ICA	1160	*84.45%*	1082	*78.81%*	155	*5.64%*	136	*15.55%*	1218	*94.36%*
Noises with 2.5V std	ICA	1154	*84.05%*	1047	*76.26%*	214	*7.79%*	112	*15.95%*	1159	*92.21%*
Noises with 5V std	ICA	1161	*84.56%*	1027	*74.80%*	268	*9.76%*	78	*15.44%*	1105	*90.24%*

To conclude, CYME's prototype has been demonstrated to have met the objective and milestones of this task. Some further work is possible to explore related to this task, especially on optimization of parameters and threshold values used in both ICA and R2 algorithms. This could be done creating some supervised learning model and training it with a set of accurate and inaccurate meter-pairings along with the ground truth.

Additionally, as seen in Table 12-7, if we look at the success rate without any field verification, the ICA algorithm is able to identify >80% of correct meter-to-transformer pairings from noiseless measurement where only 70% of existing GIS records are correct. If we feed the resulted pairings back to the ICA algorithm and iteratively run the process, we may expect to further increase the success rate without field verifications.

12.3. Open Modeling Framework (OMF)

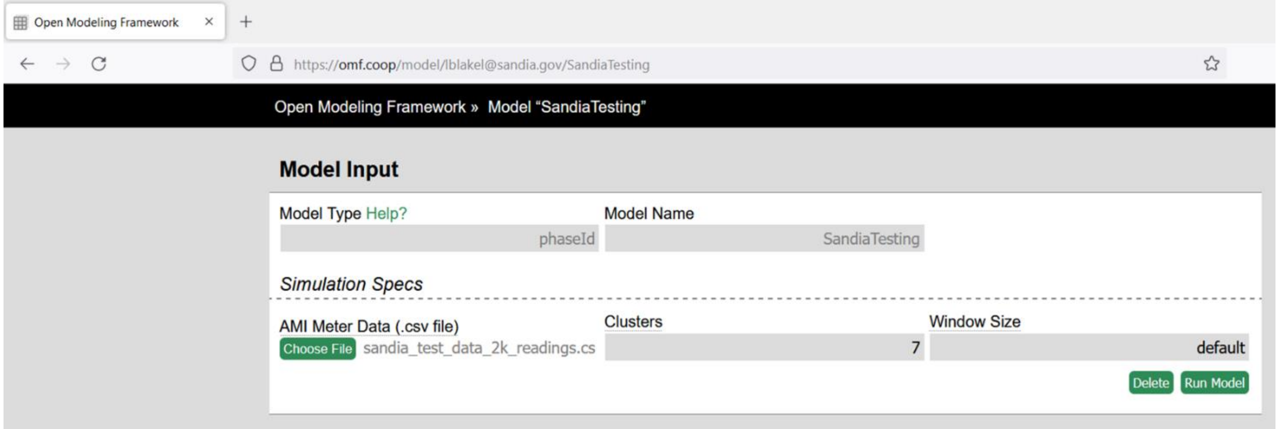
In collaboration with the National Rural Electric Cooperative Association (NRECA), the spectral clustering ensemble phase identification tool was implemented into their Open Modeling Framework (OMF) tool suite. The OMF tool suite is a free and open-source web application primarily directed toward electric cooperatives but is public-facing and open for use by everyone. NRECA has created and maintained the OMF with support from the US Department of Energy, and the system provides access to 30+ techno-economic grid planning models to an active user base of over 200 utilities, vendors, and researchers.

Part of this collaboration was the work described in Section 4.3.5 detailing the Modified Silhouette Coefficients and the Adjusted Rand Index. That work directly enabled the implementation of the phase identification algorithm into the OMF tool by providing individual confidence scores for each customer's phase prediction and enabling us to provide specific recommendations for the window size parameter and the minimum dataset requirements to the end-user of the OMF tool.

The phase identification algorithm is now fully live and available for use on https://omf.coop, labeled as the 'phaseid' tool in the menu. It can also be accessed directly via https://omf.coop/newModel/phaseId/iMoFi. Figure 12-23 shows the initial view of the GUI dashboard provided by the OMF tool when a user launches the phase identification tool. The

'Help?' icon links to the OMF wiki, Figure 12-24, which provides a brief description and user guide for the phase identification tool. The wiki also links back to the Sandia code released on Github, as well as the conference and journal papers related to the phase identification algorithm. The wiki is hosted on the Github repository for the OMF tool which can be found at https://github.com/dpinney/omf/wiki/Models-~-phaseId. The OMF tool suite is open-source and licensed under the GPL v2.0 license.

Note that in Figure 12-23, the 'Choose File' button is pre-populated with a sample dataset for new users. This is the same dataset released on the Sandia Github page. Of course, the user can supply their own dataset once they are familiar with the tool. Likewise, there are default values for the number of final clusters (7).


Open Modeling Framework ×
+
https://omf.coop/model/lblakel@sandia.gov/SandiaTesting
Open Modeling Framework » Model "SandiaTesting"

Model Input

Model Type Help?: phaseId

Model Name: SandiaTesting

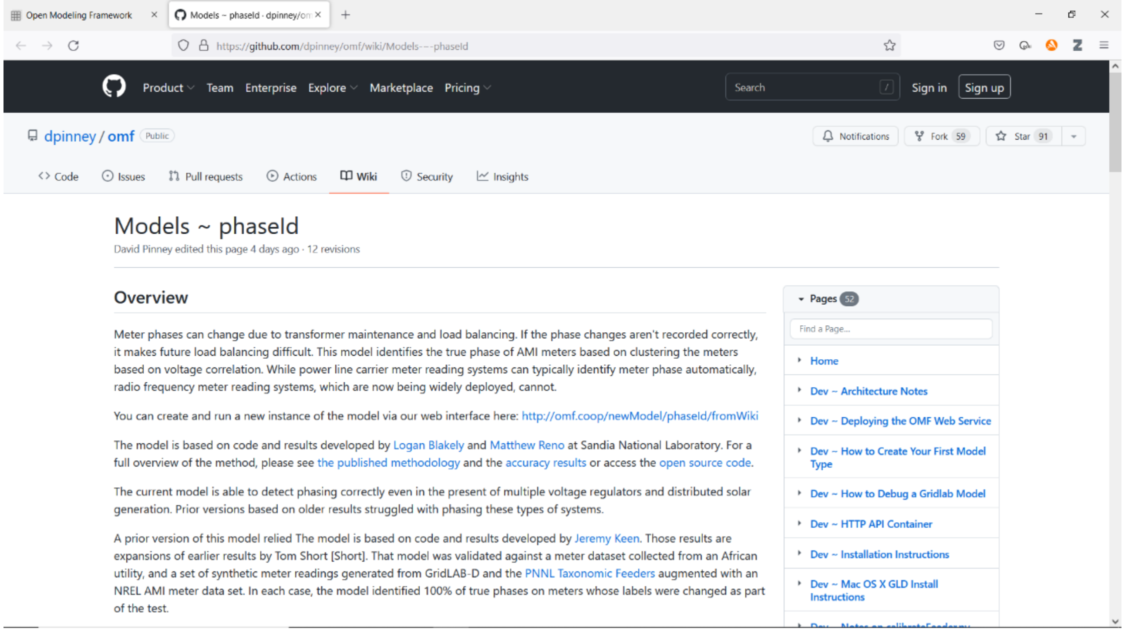
Simulation Specs

AMI Meter Data (.csv file): Choose File sandia_test_data_2k_readings.cs

Clusters: 7

Window Size: default

Figure 12-23 - OMF example GUI



Models ~ phaseId

David Pinney edited this page 4 days ago · 12 revisions

Overview

Meter phases can change due to transformer maintenance and load balancing. If the phase changes aren't recorded correctly, it makes future load balancing difficult. This model identifies the true phase of AMI meters based on clustering the meters based on voltage correlation. While power line carrier meter reading systems can typically identify meter phase automatically, radio frequency meter reading systems, which are now being widely deployed, cannot.

You can create and run a new instance of the model via our web interface here: http://omf.coop/newModel/phaseId/fromWiki

The model is based on code and results developed by Logan Blakely and Matthew Reno at Sandia National Laboratory. For a full overview of the method, please see the published methodology and the accuracy results or access the open source code.

The current model is able to detect phasing correctly even in the present of multiple voltage regulators and distributed solar generation. Prior versions based on older results struggled with phasing these types of systems.

A prior version of this model relied The model is based on code and results developed by Jeremy Keen. Those results are expansions of earlier results by Tom Short [Short]. That model was validated against a meter dataset collected from an African utility, and a set of synthetic meter readings generated from GridLAB-D and the PNNL Taxonomic Feeders augmented with an NREL AMI meter data set. In each case, the model identified 100% of true phases on meters whose labels were changed as part of the test.

Pages 52
Find a Page...
Home
Dev ~ Architecture Notes
Dev ~ Deploying the OMF Web Service
Dev ~ How to Create Your First Model Type
Dev ~ How to Debug a Gridlab Model
Dev ~ HTTP API Container
Dev ~ Installation Instructions
Dev ~ Mac OS X GLD Install Instructions

Figure 12-24 - OMF wiki for the phase identification tool

Figure 12-25, Figure 12-26, and Figure 12-27 show the output when the user runs the tool on the provided sample dataset. Figure 12-25 shows the first output of the confusion matrix, populated to the dashboard. This shows the number of customers predicted to have a different phase from their initial label.

Figure 12-26 shows the histogram of the confidence scores (Modified Silhouette Coefficients) for the sample dataset. Note that all values are above 0.2, signifying high confidence in the predictions. Figure 12-27 shows the last two outputs on the dashboard. The first is a color bar representing the total number of customers, how many were predicted to have different phase labels than the input data (9% for the sample dataset), how many were unchanged, and how many were omitted due to missing data from the analysis (0% for the sample dataset). Then there is a table showing the customer ID, the original phase, the predicted phase, and the confidence score for each customer. Customers with different predicted phase labels compared to their original label are highlighted in red.

The confidence score histogram is saved to the working directory as an image file, and the output table is saved as a csv file.

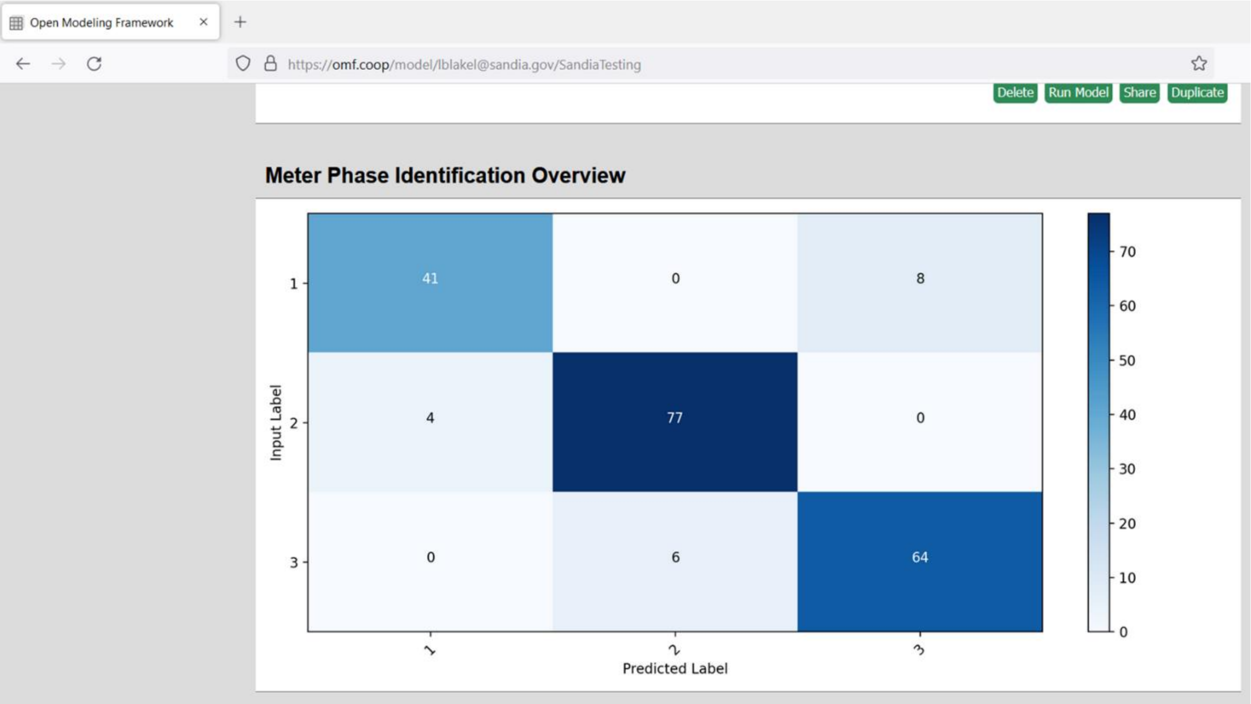
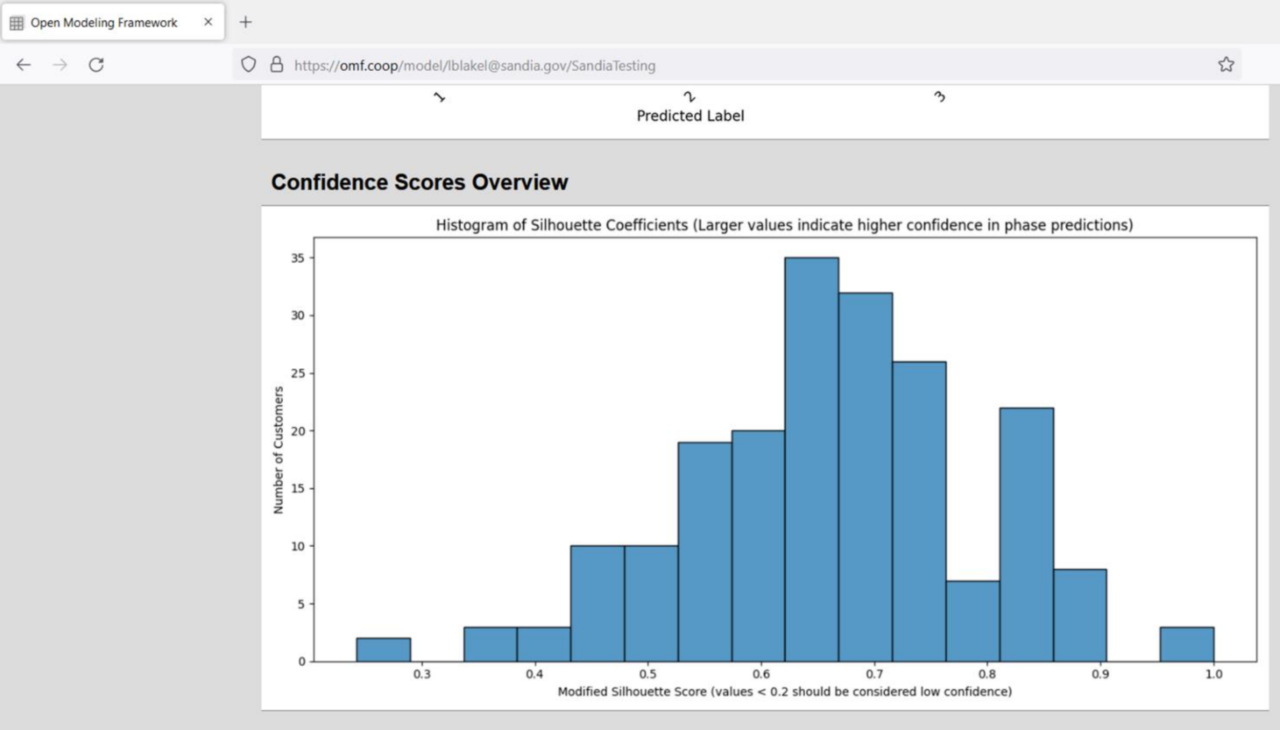
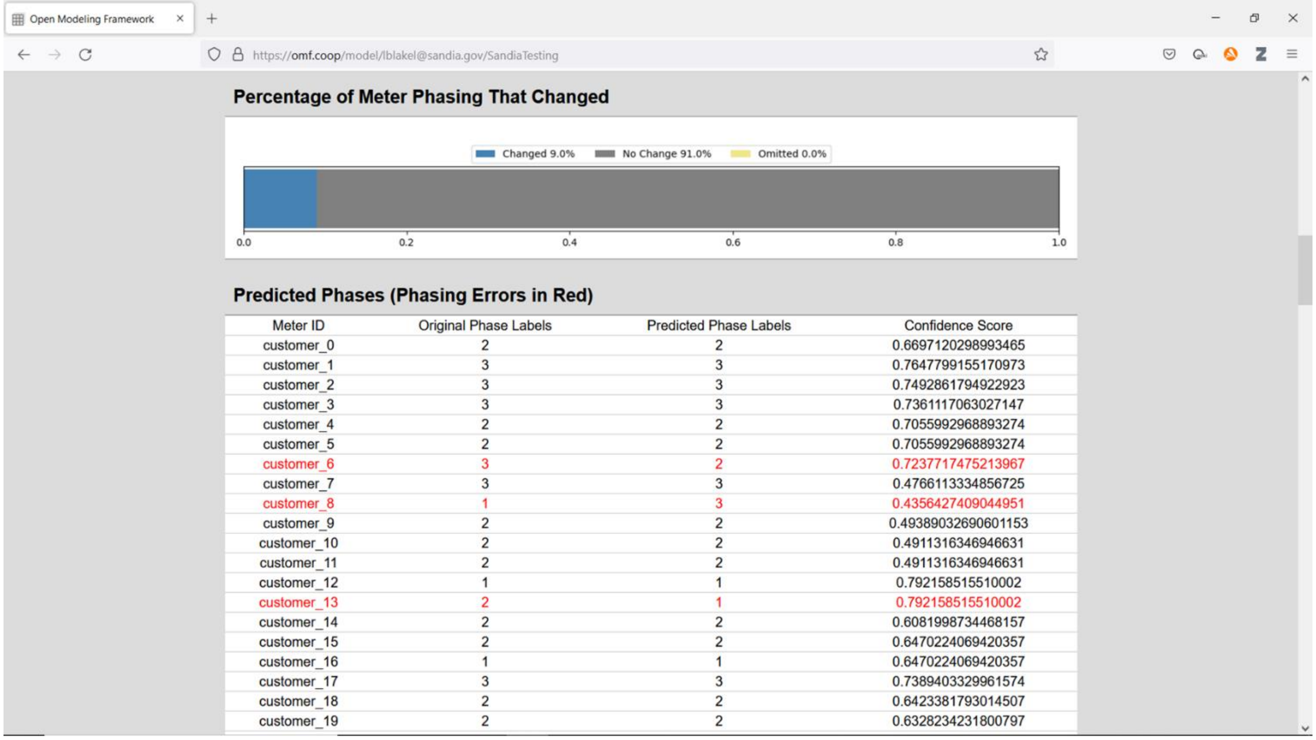


Figure 12-25 - OMF phase identification tool, confusion matrix output



Confidence Scores Overview

Figure 12-26 - OMF phase identification tool, confidence score output histogram



Percentage of Meter Phasing That Changed

Predicted Phases (Phasing Errors in Red)

Meter ID	Original Phase Labels	Predicted Phase Labels	Confidence Score
customer_0	2	2	0.6697120298993465
customer_1	3	3	0.7647799155170973
customer_2	3	3	0.7492861794922923
customer_3	3	3	0.7361117063027147
customer_4	2	2	0.7055992968893274
customer_5	2	2	0.7055992968893274
customer_6	3	2	0.7237717475213967
customer_7	3	3	0.4766113334856725
customer_8	1	3	0.4356427409044951
customer_9	2	2	0.49389032690601153
customer_10	2	2	0.4911316346946631
customer_11	2	2	0.4911316346946631
customer_12	1	1	0.792158515510002
customer_13	2	1	0.792158515510002
customer_14	2	2	0.6081998734468157
customer_15	2	2	0.6470224069420357
customer_16	1	1	0.6470224069420357
customer_17	3	3	0.7389403329961574
customer_18	2	2	0.6423381793014507
customer_19	2	2	0.6328234231800797

Figure 12-27 - OMF phase identification tool, percent of labels changed color bar and algorithm output by customer

The phase identification tool is fully live and available for usage at no cost by coops, researchers, and vendors. Having this tool available in this way puts the research funded by this project directly into the hands of the utilities and their partners who can benefit from this work.

12.4. Integration with Utility Tools

Utility #1 is implementing the sensor-based phase identification algorithm for widespread use into their system. Sandia has facilitated their implementation in the form of advice and further algorithm development as issues or questions have come up.

In preparation for putting the phase identification into their normal process, they have been working through the field verification of the feeders that the project team used for testing. Sandia provided the utility with the algorithm predictions, and the utility provided personnel and time to field-verify the algorithm predictions. This is further detailed in Section 4.2.5. The algorithm results for each of the four feeders was shown to be 100% accurate for customers predicted to be incorrect in the utility model. Each field verification provides additional confidence in the algorithm performance and additional insights into the types and frequencies of the errors that are identified.

The utility uses the IBM Cognos platform as their data analytics platform. This platform provides the functionality for a data input pipeline, data analytics, and a dashboard visualization. As seen in Figure 12-28, they have put together a draft for the phase identification algorithm which runs in the Cognos platform. The draft leverages the code that Sandia released as opensource (see Section 12.1) The algorithm has been shown to match the outputs from Sandia using the same dataset. One of the challenges with this initial version is the runtime which is several hours. This is mainly an implementation issue.

They are in the process of developing a revised complete implementation that can be applied to their entire service area. This includes developing a data input pipeline to convert raw AMI records from their database system into an array format for input into the phase identification algorithm. This may be the primary bottleneck in the runtime as it involves calls to a database outside the Cognos architecture.

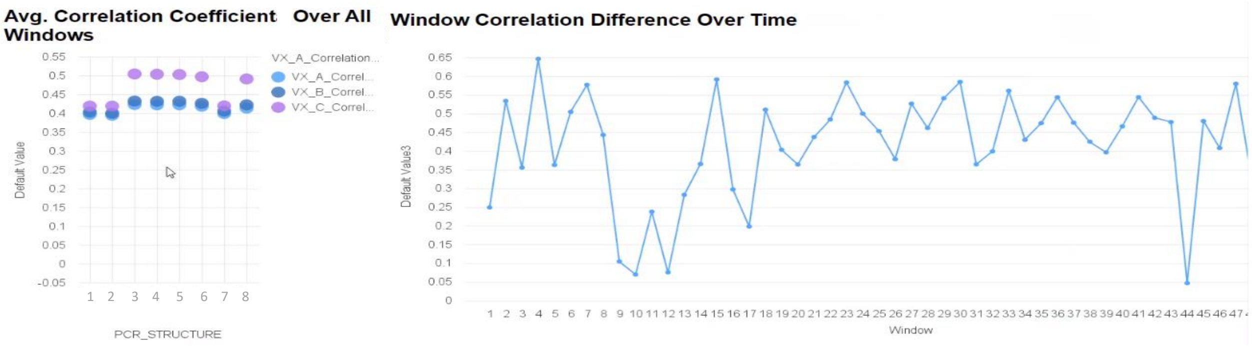


Figure 12-28 - Phase identification in Utility #1's reporting platform for using AMI data and recloser SCADA measurements to identify customers that are on the wrong phase in their database

12.5. Summary

Three of the distribution system model calibration algorithms developed during this project were released as open source on the Sandia National Laboratories GitHub, https://github.com/sandialabs. Those were the Spectral Clustering Ensemble phase identification algorithm, the Sensor-Based phase identification algorithm, and the two-stage meter-to-transformer pairing algorithm. CYME implemented and tested three different types of algorithms within CYME. Those were phase identification algorithms, secondary system topology and parameter estimation algorithms, and meter-to-transformer pairing algorithms. NRECA implemented the spectral clustering ensemble phase identification algorithm into their OMF platform, and that algorithm is now freely available for use by utilities. In addition, Utility #1 is in the process of implementing the sensor-based phase identification algorithm for widespread use within their system. These four different algorithm implementation items demonstrate the utility of the algorithms developed over the course of this project in the real world.

13. CONCLUSIONS

Uncertainty in distribution grid modeling leads to overly conservative or inaccurate decisions regarding PV integration and limits acceptance of higher levels of PV. Simulation tools are often severely limited in their effectiveness by the model accuracy. This project developed methods to efficiently process grid measurements and Big Data to substantially increase the precision and accuracy of distribution system models. There is a great industry need for more accurate feeder model phasing information, better spatial and temporal load models, phase-specific load models, more detailed reactive power load modeling, and better ways to represent feeder net load in the presence of exiting DER. The applications of the improved models include more accurate interconnection screening (reducing PV interconnection costs and expensive mistakes that impact safety and reliability) and increased confidence in public-facing hosting capacity maps (providing optimal locations to cost effectively allow high PV penetrations and improve the distribution system reliability and performance). The proposed methods leverage readily available utility measurement data, feeder models, and other information. The methods are demonstrated on real U.S. utility feeder models with field measurement data from SCADA, AMI, and other sources.

It was shown in [116] that errors in grid models can result in significant hosting capacity errors. Locational hosting capacity analysis was performed for 1379 potential PV interconnection locations with different levels of model errors for size of service transformers, customer to service transformer connection, missing existing PV, phase label errors, errors in secondary low-voltage system topology, difference in voltage regulation settings, and incorrect state of capacitors. As seen in Figure 13-1, the discrepancies between the model and actual operations of the distribution system in the field can result in up to 100% error in the locational hosting capacity estimate.

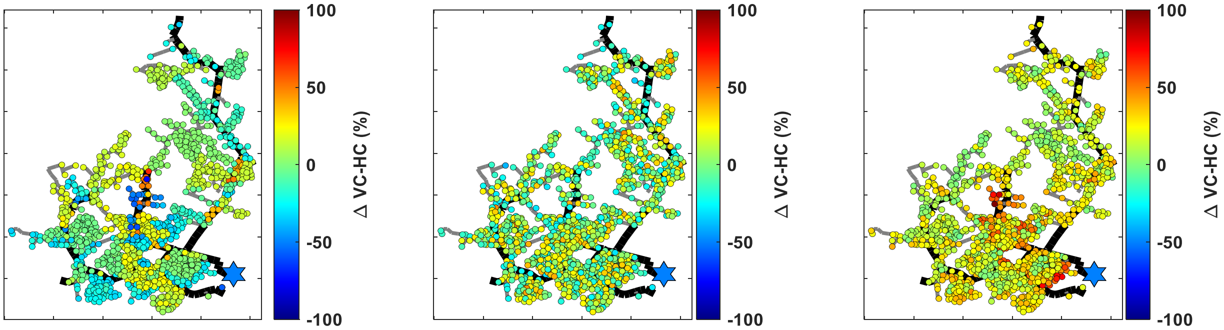


Figure 13-1 - The percent error in estimated locational voltage-constrained hosting capacity (VC-HC) when there are a) phase label errors, b) errors in secondary system topology, and c) errors in substation voltage regulation settings in the distribution system model. [116]

13.1. Accomplishments and Final Deliverables

During the course of the research, several key accomplishments achieved were:

- Demonstrated phase identification performing with at least 90% accuracy on test systems and utility feeders
- Implemented and validated meter-to-transformer pairing algorithm with >90% success rate on test systems and utility feeders
- Showed the ability to identify location, estimate size, and estimate parameters of PV systems in a medium-size distribution feeder with an error of less than 5%

- Demonstrated phase identification, secondary system parameter estimation, and meter-to-transformer pairing algorithms in CYME
- Demonstrated data-driven methods for identifying voltage regulator and switching capacitor states
- Demonstrated the ability to determine control parameters for behind-the-meter smart inverters for their power factor and volt-var curve
- Demonstrated improved, yet practical, distribution load modeling methods that leverage modern data streams
- Deployment of phase identification algorithm in NRECA's Open Modeling Framework (OMF) for easy use by coops
- Demonstration with a utility partner of phase identification integrated into their environment and existing toolset

Final deliverables resulting from the project are:

- Publication with recommendations to utilities for AMI data recording guidelines along with a listing of value that can be realized at different levels of data resolution
- Publicly published open-source algorithms for phase identification and meter-to-transformer pairing

13.2. Project Publications

Journal Papers

1. M. Lave, M. J. Reno, J. Peppanen, "Distribution System Parameter and Topology Estimation Applied to Resolve Low-Voltage Circuits on Three Real Distribution Feeders", IEEE Transactions on Sustainable Energy, 2019. [62]
2. K. Mason, M.J. Reno, L. Blakely, S. Vejdan, S. Grijalva, "A Deep Neural Network Approach for Behind-the-Meter Residential PV Size, Tilt and Azimuth Estimation", Solar Energy, 2020. [82]
3. L. Blakely, M. J. Reno, "Phase Identification Using Co-Association Matrix Ensemble Clustering" IET Smart Grid, 2020. [36]
4. F. Therrien, L. Blakely, and M. J. Reno, "Assessment of Measurement-Based Phase Identification Methods", IEEE Open Access Journal of Power and Energy, 2021. [41]
5. S. Talkington, S. Grijalva, and M. J. Reno, "Power Factor Estimation of Distributed Energy Resources Using Voltage Magnitude Measurements", Journal of Modern Power Systems and Clean Energy, 2021. [87]
6. R. D. Trevizan, C. Ruben, A. Rossoni, S. C. Dhulipala, A. S. Bretas, and N. G. Bretas, "μPMU-based Temporal Decoupling of Parameter and Measurement Gross Error Processing in DSSE", Electricity, 2021. [67]
7. J. A. Azzolini, M. J. Reno, and K. A. W. Horowitz, "Evaluating Distributed PV Curtailment Using Quasi-Static Time-Series Simulations," IEEE Open Access Journal of Power and Energy, 2021. [117]
8. S. Talkington, S. Grijalva, M. J. Reno, and J. A. Azzolini, "Solar PV Inverter Reactive Power Disaggregation and Control Setting Estimation", IEEE Transactions on Power Systems, 2022. [90]

Reports

1. *Enhanced Load Modeling: Leveraging Expanded Monitoring and Metering*. Palo Alto, CA: EPRI, 2019. [Online]. Available: https://www.epri.com/research/products/000000003002015283 [33]
2. *Enhanced Load Modeling: Study of Practical Load Modeling Methods on Distribution Feeders with Behind-The-Meter PV Systems*. Palo Alto, CA: EPRI, 2020. [Online]. Available: https://www.epri.com/research/products/000000003002018983 [96]
3. *Enhanced Load Modeling –Assessment of Load Modeling Methods to Represent Highly Unbalanced Loading Conditions*, Palo Alto, CA: EPRI , 2020. [Online]. Available: https://www.epri.com/research/products/000000003002019861 [97]
4. *Enhanced Load Modeling: Survey Results on Industry Load Modeling Practices*. Palo Alto, CA: EPRI, 2020. [Online]. Available: https://www.epri.com/research/products/000000003002020036 [95]
5. *Enhanced Load Modeling: Methodology for Estimating the Parameters of Voltage Sensitive Load Models Based on AMI Data*. Palo Alto, CA: EPRI, 2021. [Online]. Available: https://www.epri.com/research/products/000000003002021211 [34]
6. *Enhanced Load Modeling: Improved Reactive Power Load Modeling*, EPRI, 2021. [Online]. Available: https://www.epri.com/research/products/000000003002022354 [76]
7. *Enhanced Load Modeling: Comparison of Load Modeling and Power Flow in DMS and Planning Software*. Palo Alto, CA: EPRI, 2021. [Online]. Available: https://www.epri.com/research/products/000000003002021516 [98]
8. J. A. Azzolini and M. J. Reno, "Analysis of Reactive Power Load Modeling Techniques for PV Impact Studies", Sandia National Laboratories, SAND2022-7033R. [118]

Conference Papers

1. L. Blakely, M. J. Reno, W. Feng, "Spectral Clustering for Customer Phase Identification Using AMI Voltage Timeseries", IEEE Power and Energy Conference at Illinois (PECI), Champaign, IL, 2019. [13]
2. L. Blakely, M. J. Reno, and K. Ashok, "AMI Data Quality And Collection Method Consideration for Improving the Accuracy of Distribution System Models", IEEE Photovoltaic Specialists Conference (PVSC), Chicago, IL, 2019. [29]
3. L. Blakely, M. J. Reno, and J. Peppanen, "Identifying Common Errors in Distribution System Models", IEEE Photovoltaic Specialists Conference (PVSC), Chicago, IL, 2019. [2]
4. K. Ashok, M. J. Reno, D. Divan, and L. Blakely, "Systematic Study of Data Requirements and AMI Capabilities for Smart Meter Connectivity Analytics", IEEE Smart Energy Grid Engineering (SEGE), Oshawa, Ontario, Canada, 2019. [32]
5. J. Deboever, M. Hernandez, J. Peppanen, P. Siratarnsophon, and M. J. Reno, "Impact of AMI Data Time Granularity on Quasi-Static Time-Series Load Flow Simulation", IEEE Transmission & Distribution Conference & Exposition, Chicago, IL, 2020. [119]
6. K. Ashok, M. J. Reno, D. Divan, and L. Blakely, "Secondary Network Parameter Estimation for Distribution Transformers", IEEE Innovative Smart Grid Technologies (ISGT), Washington DC, 2020. [63]
7. L. Blakely, M. J. Reno, "Identifying Errors in Service Transformer Connections", IEEE PES General Meeting, Montreal, Canada, 2020. [64]

8. S. Grijalva, A. U. Khan, J. Sihno-Mbeleg, C. Gomez-Peces, M. J. Reno, L. Blakely, "Estimation of PV Location in Distribution Systems based on Voltage Sensitivities", IEEE North American Power Symposium (NAPS), 2020. [84]
9. P. Siratarnsophon, M. Hernandez, J. Peppanen, J. Deboever, M. Rylander, M. J. Reno, "Improved Load Modelling for Emerging Distribution System Assessments", CIRED Berlin Workshop, 2020. [120]
10. B. Pena, L. Blakely, and M. J. Reno, "Parameter Tuning Analysis for Phase Identification Algorithms in Distribution System Model Calibration", IEEE Power and Energy Conference at Kansas State University (KPEC), 2021. [40]
11. L. Blakely, M. J. Reno, B. Jones, and A. Furlani Bastos, "Leveraging Additional Sensors for Phase Identification in Systems with Voltage Regulators", IEEE Power and Energy Conference at Illinois (PECI), 2021. [35]
12. L. Blakely and M. J. Reno, "Identification and Correction of Errors in Pairing AMI Meters and Transformers", IEEE Power and Energy Conference at Illinois (PECI), 2021. [65]
13. C. Francis, R. D. Trevizan, M. J. Reno, and V. Rao, "Topology Identification of Power Distribution Systems Using Time Series of Voltage Measurements", IEEE Power and Energy Conference at Illinois (PECI), 2021. [66]
14. C. Gomez-Peces, S. Grijalva, M. J. Reno, and L. Blakely, "Estimation of PV Location based on Voltage Sensitivities in Distribution Systems with Discrete Voltage Regulation Equipment", IEEE PowerTech, 2021. [84]
15. C.-C. Sun, M. Korkali, E. M. Stewart, V. Donde, and M. J. Reno, "Optimization-Based Calibration of Aggregated Dynamic Models for Distributed Energy Resources", 2021 IEEE PES General Meeting, 2021. [108]
16. J. A. Azzolini, and M. J. Reno, "Impact of Load Allocation and High Penetration PV Modeling on QSTS Curtailment Studies", 2021 IEEE PES General Meeting, 2021. [77]
17. R. D. Trevizan and M. J. Reno, "Distribution System State Estimation Sensitivity to Errors in Phase Connections", IEEE Photovoltaic Specialists Conference (PVSC), 2021. [54]
18. J. Peppanen, M. Hernandez, J. Deboever, M. Rylander, and M.J. Reno, "Distribution Load Modeling – Survey of the Industry State, Current Practices and Future Needs", IEEE North American Power Symposium (NAPS), 2021. [121]
19. S. Talkington, S. Grijalva, M. J. Reno, and J. Azzolini, "Recovering Power Factor Control Settings of Solar PV Inverters from Net Load Data", IEEE North American Power Symposium (NAPS), 2021. [122]
20. C. Rocha, J. Deboever, J. Peppanen, M. Hernandez and M. Rylander, "Improved Reactive Power Load Modeling for Distribution Planning", IEEE PES T&D Conference & Exposition, New Orleans, LA, 2022. [123]
21. D. R. Garcia, B. Poudel, A. Bidram, M. J. Reno, "Substation-level Circuit Topology Estimation Using Machine Learning," IEEE Innovative Smart Grid Technologies (ISGT), 2022. [124]
22. B. D. Pena, L. Blakely, and M. J. Reno, "Data-Driven Detection of Phase Changes in Evolving Distribution Systems", IEEE Texas Power and Energy Conference (TPEC), 2022. [31]
23. S. Talkington, S. Grijalva, and M. J. Reno, "Sparse Time Series Sampling for Recovery of Behind-the-Meter Inverter Control Models", IEEE Innovative Smart Grid Technologies (ISGT), 2022. [93]

24. J. Yusuf, J. A. Azzolini, and M. J. Reno, “Data-Driven Methods for Voltage Regulator Identification and Tap Estimation”, IEEE Kansas Power & Energy Conference (KPEC), 2022. [125]
25. R. D. Trevizan and M. J. Reno, “Detection of False Data Injection Attacks in Power System State Estimation Using Sensor Encoding”, IEEE Kansas Power & Energy Conference (KPEC), 2022. [126]
26. J. A. Azzolini, S. Talkington, M. J. Reno, S. Grijalva, L. Blakely, D. Pinney, and S. McHann, “Improving Behind-the-Meter PV Impact Studies with Data-Driven Modeling and Analysis”, IEEE Photovoltaic Specialists Conference (PVSC), 2022. [116]

13.3. Path Forward

The ability to detect errors and calibrate distribution system models will continue to be an ongoing challenge for utilities. Continued adoption of smart grid technologies and controls, such as electric vehicle chargers or advanced distribution management systems, means that there will be even more parameters, settings, and components to model in the future. Fortunately, sensors and communication are also becoming cheaper, which allows for more visibility into distribution system operations and provides for additional model calibration capabilities. These algorithms will need to continue to improve in the future for additional features and more efficient processing as the data coming back from the distribution system continues to grow in size.

With growing numbers of distributed PV, the aggregate impacts to the bulk power system are becoming more important. Future work could investigate the ability to use distributed measurements to calibrate aggregate dynamic models of PV for transient and stability simulations of the system. This includes using new data sources, such as PMU and point-on-wave sensors, as they become more common in the distribution system.

To be truly effective, these model calibration algorithms must be integrated into the utility’s processes. Utilities may have information stored in a variety of locations, such as SCADA measurements with DMS, AMI data in billing, component information in GIS, and PV information in their interconnection database. It is necessary to be able to access and query each piece of information and process it in a streamlined fashion with automated reports generated for the utilities. In addition, it will be necessary to refine utility processes to manage the grid model data across the different utility IT systems, including planning and operational software.

Future work could include hardware and sensing research to expand local analytic capabilities. Most model calibration algorithms right now are performed at a central location based on large historical datasets from many sensors. As the number and sampling frequency of sensors increases, computational efficiency improvements can be made by doing some analytics locally in a distributed or hierarchical fashion. While fog computing using edge devices is becoming more common in certain grid controls, it may take some intense research to be able to apply model calibration using distributed computing architectures. This could also include improvements to sensing hardware based on the ongoing positioning, navigation, and timing (PNT) research that would allow for more synchronized measurements around the distribution system that are important for measuring small phase angle changes.

REFERENCES

[1] E. M. Stewart *et al.*, "Integrated Multi-Scale Data Analytics and Machine Learning for the Distribution Grid and Building-to-Grid Interface," *Lawrence Livermore National Laboratory*, vol. LLNL-TR-727125, 2017.

[2] L. Blakely, M. J. Reno, and J. Peppanen, "Identifying Common Errors in Distribution System Models," in *Proceedings of the IEEE 46th PVSC*, Chicago, IL, USA, 2019.

[3] Y. Liao, Y. Weng, M. Wu, and R. Rajagopal, "Distribution Grid Topology Reconstruction: An Information Theoretic Approach," *North American Power Symposium (NAPS)*, Oct. 2015.

[4] Y. Weng, Y. Liao, and R. Rajagopal, "Distributed Energy Resources Topology Identification via Graphical Modeling," *IEEE Transactions on Power Systems*, vol. 32, no. 4, Jul. 2017.

[5] G. Cavraro, R. Arghandeh, K. Poolla, and A. von Meier, "Data-driven Approach for Distribution Network Topology Detection," *IEEE Power & Energy Society General Meeting*, Jul. 2015.

[6] A. J. Berrisford, "A Tale of Two Transformers: An Algorithm for Estimating Distribution Secondary Electric Parameters Using Smart Meter Data," *26th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, May 2013.

[7] J. D. Watson, J. Welch, and N. R. Watson, "Use of Smart-meter Data to Determine Distribution system Topology," *Journal of Engineering*, vol. 2016, no. 5, pp. 94–101, 2016.

[8] J. Peppanen, M. J. Reno, M. Thakkar, S. Grijalva, and R. G. Harley, "Leveraging AMI Data for Distribution System Model Calibration and Situational Awareness," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 2050–2059, Jan. 2015.

[9] W. Luan, J. Peng, M. Maras, B. Harapnuk, and J. Lo, "Smart Meter Data Analytics for Distribution Network Connectivity Verification," *IEEE Transactions on Smart Grid*, vol. 6, p. 1, Jul. 2015.

[10] R. Mitra *et al.*, "Voltage Correlations in Smart Meter Data," *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1999–2008, 2015.

[11] T. A. Short, "Advanced Metering for Phase Identification, Transformer Identification, and Secondary Modeling," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 651–658, Jun. 2013, doi: 10.1109/TSG.2012.2219081.

[12] F. Olivier, A. Sutera, P. Geurts, R. Fonteneau, and D. Ernst, "Phase Identification of Smart Meters by Clustering Voltage Measurements," *Power Systems Computation Conference (PSCC)*, 2018.

[13] L. Blakely, M. J. Reno, and W. Feng, "Spectral Clustering for Customer Phase Identification Using AMI Voltage Timeseries," *Power and Energy Conference at Illinois (PECI)*, Feb. 2019.

[14] Y. Ma *et al.*, "Phase Identification of Smart Meters by Spectral Clustering," *IEEE Conference on Energy Internet and Energy System Integration (EI2)*, Oct. 2018.

[15] W. Wang, N. Yu, B. Foggo, J. Davis, and J. Li, "Phase Identification in Electric Power Distribution Systems by Clustering of Smart Meter Data," in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2016, pp. 259–265, doi: 10.1109/ICMLA.2016.0050.

[16] B. Foggo and N. Yu, "A Comprehensive Evaluation of Supervised Machine Learning for the Phase Identification Problem," *World Academy of Science, Engineering and Technology International Journal of Computer and Systems Engineering*, vol. 12, no. 6, 2018.

[17] M. Sheinin, Y. Y. Schechner, and K. N. Kutulakos, "Computational Imaging on the Electric Grid," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2363–2372, doi: 10.1109/CVPR.2017.254.

[18] X. Zhang and S. Grijalva, "A Data-Driven Approach for Detection and Estimation of Residential PV Installations," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2477–2485, Sep. 2016, doi: 10.1109/TSG.2016.2555906.

[19] P. Jokar, N. Arianpoo, and V. C. M. Leung, "Electricity Theft Detection in AMI Using Customers' Consumption Patterns," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 216–226, Jan. 2016.

[20] S. McLaughlin, B. Holbert, A. Fawaz, R. Berthier, and S. Zonouz, "A Multi-Sensor Energy Theft Detection Framework for Advanced Metering Infrastructures," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1319–1330, Jun. 2013.

[21] R. Jiang, R. Lu, Y. Wang, J. Luo, C. Shen, and X. S. Shen, "Energy-theft Detection Issues for Advanced Metering Infrastructure in Smart Grid," *Tsinghua Science and Technology*, vol. 19, no. 2, pp. 105–120, Apr. 2014.

[22] A. Gúzman, A. Arguello, J. Quirós-Tortós, and G. Valverde, "Processing and Correction of Secondary System Models in Geographic Information Systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3482–3491, 2018.

[23] R. Mohan, T. Cheng, A. Gupta, and V. Garud, "Solar Energy Disaggregation Using Whole-house Consumption Signals," *NILM Workshop*, Jun. 2014.

[24] D. Chen and D. Irwin, "SunDance: Black-box Behind-the-Meter Solar Disaggregation," *Proceedings of the Eighth International Conference on Future Energy Systems*, pp. 45–55, May 2017.

[25] S. Iyengar, N. Sharma, D. Irwin, P. Shenoy, and K. Ramamritham, "A Cloud-based Black Box Solar Predictor for Smart Homes," *ACM Transactions on Cyber-Physical Systems*, 2016.

[26] "Smart Meters and Smart Meter Systems: A Metering Industry Perspective," in *EEI-AEIC-UTC, Washington, DC, USA*, 2011.

[27] M. Lave, M. J. Reno, R. J. Broderick, and J. Peppanen, "Full-Scale Demonstration of Distribution System Parameter Estimation to Improve Low-Voltage Circuit Models," in *IEEE Photovoltaic Specialists Conference (PVSC)*, Washington, DC, USA, 2017.

[28] J. Peppanen, M. J. Reno, R. J. Broderick, and S. Grijalva, "Distribution System Model Calibration with Big Data from AMI and PV," *IEEE Transacation on Smart Grid*, vol. 7, no. 5, pp. 2497–2506, Sep. 2016.

[29] L. Blakely, M. J. Reno, and K. Ashok, "AMI Data Quality And Collection Method Consideration for Improving the Accuracy of Distribution System Models," in *IEEE PVSC*, Chicago, IL, USA, 2019.

[30] J. Peppanen, X. Zhang, S. Grijalva, and M. J. Reno, "Handling Bad or Missing Smart Meter Data Through Advanced Data Imputation," *IEEE Power & Energy Society Innovative Smart Grid Technologies (ISGT)*, 2016.

[31] B. D. Pena, L. Blakely, and M. J. Reno, "Data-Driven Detection of Phase Changes in Evolving Distribution Systems," in *TPEC*, 2022.

[32] K. Ashok, M. J. Reno, D. Divan, and L. Blakely, "Systematic Study of Data Requirements and AMI Capabilities for Smart Meter Connectivity Analytics," in *IEEE SEGE*, Oshawa, Ontario, Canada, 2019.

[33] *Enhanced Load Modeling: Leveraging Expanded Monitoring and Metering*. Palo Alto, CA: EPRI, 2019.

[34] *Enhanced Load Modeling: Methodology for Estimating the Parameters of Voltage Sensitive Load Models Based on AMI Data*. Palo Alto, CA: EPRI, 2021.

[35] L. Blakely, M. J. Reno, B. Jones, A. Furlani Bastos, and D. Nordy, "Leveraging Additional Sensors for Phase Identification in Systems with Voltage Regulators," in *Power and Energy Conference at Illinois (PECI)*, 2021.

[36] L. Blakely and M. J. Reno, "Phase Identification Using Co-Association Matrix Ensemble Clustering," *IET Smart Grid*, no. Machine Learning Special Issue, Jun. 2020.

[37] R. Mitra *et al.*, "Voltage Correlations in Smart Meter Data," in *ACM SIGKDD*, Sydney, NSW, Australia, 2015, pp. 1999–2008.

[38] T. A. Short, "Advanced Metering for Phase Identification, Transformer Identification, and Secondary Modeling," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 651–658, Jun. 2013, doi: 10.1109/TSG.2012.2219081.

[39] B. K. Seal and M. F. McGranaghan, "Automatic Identification of Service Phase for Electric Utility Customers," *IEEE Power and Energy Society General Meeting*, Jul. 2011.

[40] B. D. Pena, L. Blakely, and M. J. Reno, "Parameter Tuning Analysis for Phase Identification Algorithms in Distribution System Model Calibration," in *IEEE Kansas Power and Energy Conference (KPEC)*, 2021, doi: 10.1109/KPEC51835.2021.9446218.

[41] F. Therrien, L. Blakely, and M. J. Reno, "Assessment of Measurement-Based Phase Identification Methods," *IEEE Open Access Journal of Power and Energy*, 2021.

[42] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and an Algorithm," in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, Cambridge, MA, USA, 2001, pp. 849–856.

[43] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, no. 85, pp. 2825–2830, 2011.

[44] D. Arthur and S. Vassilvitskii, "k-means++: The Advantages of Careful Seeing," in *Symposium on Discrete Algorithms (SODA)*, New Orleans, LA, USA, 2007, pp. 1027–1035.

[45] S. X. Yu and J. Shi, "Multiclass Spectral Clustering," in *Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV)*, Nice, France, 2003, vol. 2.

[46] S. Vega-Pons and Ruiz-Shulcloper, "A Survey of Clustering Ensemble Algorithms," *International Journal of Pattern Recognition*, vol. 25, no. 3, pp. 337–372, 2011.

[47] X. Zhang, L. Jiao, F. Liu, L. Bo, and M. Gong, "Spectral Clustering Ensemble Applied to SAR Image Segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 7, pp. 2126–2136, 2008.

[48] Y. Ma *et al.*, "Phase Identification of Smart Meters by Spectral Clustering," in *IEEE EI2*, Beijing, China, 2018.

[49] A. J. Izenman, "Nonhierarchical or Partitioning Methods: Silhouette Plots," in *Modern Multivariate Statistical Techniques*, Springer, 2008.

[50] "American National Standards Institute," 2015. [Online]. Available: http://www.nema.org/stds/c12-20.cfm.

[51] A. Monticelli, "State Estimation in Electric Power Systems: A Generalized Approach," *Springer Science & Business Media*, vol. 507, 1999.

[52] Smarter Grid Solutions, "Fundamental Research Challenges for Distribution State Estimation to Enable High-Performing Grids," New York State Energy Research and Development Authority (NYSERDA), New York, NY, Report Number 18-37, Oct. 2018.

[53] J. Peppanen, J. Grimaldo, M. J. Reno, S. Grijalva, and R. G. Harley, "Increasing Distribution System Model Accuracy with Extensive Deployment of Smart Meters," in *IEEE PES General Meeting*, 2014, doi: 10.1109/PESGM.2014.6939840.

[54] R. D. Trevizan and M. J. Reno, "Distribution System State Estimation Sensitivity to Errors in Phase Connections," in *IEEE 48th Photovoltaic Specialists Conference (PVSC)*, 2021, doi: 10.1109/PVSC43889.2021.9518900.

[55] J. Peppanen, M. Bello, and M. Rylander, "Service Entrance Hosting Capacity," in *IEEE 7th World Conference on Photovoltaic Energy Conversion (WCPEC)*, 2018, doi: 10.1109/PVSC.2018.8547340.

[56] J. Peppanen, S. Grijalva, M. J. Reno, and R. J. Broderick, "Distribution System Low-Voltage Circuit Topology Estimation using Smart Metering Data," *IEEE PES Transmission & Distribution Conference & Exposition, Dallas, TX*, 2016.

[57] J. Peppanen, M. J. Reno, R. J. Broderick, and S. Grijalva, "Distribution System Secondary Circuit Parameter Estimation for Model Calibration," Sandia National Lab (SNL-NM), Technical Report SAND2015-7477, 2015.

[58] J. Peppanen, M. J. Reno, R. J. Broderick, and S. Grijalva, "Secondary Circuit Model Creation and Validation with AMI and Transformer Measurements," in *North American Power Symposium (NAPS)*, 2016.

[59] J. Peppanen, M. J. Reno, R. J. Broderick, and S. Grijalva, "Secondary Circuit Model Generation Using Limited PV Measurements and Parameter Estimation," in *IEEE PES General Meeting*, 2016.

[60] C. W. Brice, "Comparison of Approximate and Exact Voltage Drop Calculations for Distribution Lines," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-101, no. 11, pp. 4428–4431, Nov. 1982.

[61] W. Kersting, *Distribution System Modeling and Analysis*, 3rd ed. Taylor & Francis, 2012.

[62] M. Lave, M. J. Reno, and J. Peppanen, "Distribution System Parameter and Topology Estimation Applied to Resolve Low-Voltage Circuits on Three Real Distribution Feeders," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 3, pp. 1585–1592, 2019.

[63] K. Ashok, M. J. Reno, D. Divan, and L. Blakely, "Secondary Network Parameter Estimation for Distribution Transformers," in *IEEE Innovative Smart Grid Technologies (ISGT)*, Washington DC, 2020.

[64] L. Blakely and M. J. Reno, "Identifying Errors in Service Transformer Connections," in *PES General Meeting*, 2020.

[65] L. Blakely and M. J. Reno, "Identification and Correction of Errors in Pairing AMI Meters and Transformers," in *IEEE Power and Energy Conference at Illinois (PECI)*, 2021.

[66] C. Francis, V. Rao, R. D. Trevizan, and M. J. Reno, "Topology Identification of Power Distribution Systems Using Time Series of Voltage Measurements," in *IEEE Power and Energy Conference at Illinois (PECI)*, 2021, pp. 1–7, doi: 0.1109/PECI51586.2021.9435253.

[67] R. D. Trevizan, C. Ruben, A. Rossoni, S. C. Dhulipala, A. S. Bretas, and N. G. Bretas, "Micro-PMU-based Temporal Decoupling of Parameter and Measurement Gross Error Processing in DSSE," *Electricity*, vol. 2, no. 4, pp. 423–438, Dec. 2021.

[68] Z. K. Pecenak, H. V. Haghi, C. Li, M. J. Reno, V. R. Disfani, and J. Kleiss, "Aggregation of Voltage-Controlled Devices During Distribution Network Reduction," *IEEE Transactions on Smart Grid*, 2020.

[69] C. Li, R. Disfani, H. V. Haghi, and J. Kleissl, "Optimal Voltage Regulation of Unbalanced Distribution Networks with Coordination of OLTC and PV Generation," in *IEEE PES General Meeting*, 2019.

[70] R. Yan, Y. Li, T. K. Saha, L. Wang, and M. I. Hossain, "Modeling and Analysis of Open-Delta Step Voltage Regulators for Unbalanced Distribution Network with Photovoltaic Power Generation," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 2224–2234, 2018.

[71] F. Bai, R. Yan, T. K. Saha, and D. Eghbal, "An Excessive Tap Operation Evalutaion Approach for Unbalanced Distribution Networks With High PV Penetration," *IEEE Transactions on Sustainable Energy*, vol. 12, no. 1, pp. 169–178, 2021.

[72] F. Bai, R. Yan, T. K. Saha, and D. Eghbal, "A New Remote Tap Position Estimation Approach for Open-Delta Step-Voltage Regulator in a Photovoltaic Integrated Distribution Network," *IEEE Transactions on Power Systems*, vol. 33, no. 4, pp. 4433–4443, 2018.

[73] K. Jiseong, W. M. Grady, A. Arapostathis, J. C. Soward, and S. C. Bhatt, "A Time-Domain Procedure for Locating Switched Capacitors in Power Distribution Systems," *IEEE Transactions on Power Delivery*, vol. 17, no. 4, pp. 1044–1049, 2002.

[74] K. Hur and S. Santoso, "Distance Estimation of Switched Capacitor Banks in Utility Distribution Feeders," *IEEE Transactions on Power Delivery*, vol. 22, no. 4, pp. 2419–2427, 2007.

[75] H. Khani, M. Moallem, and S. Sadri, "On Tracking and Finding the Location of Switched Capacitor Banks in Distribution Systems," in *Transmission & Distribution Conference & Exposition: Asia and Pacific*, 2009.

[76] EPRI, *Enhanced Load Modeling: Improved Reactive Power Load Modeling*. Palo Alto, CA: Electrical Power Research Institute, 2021.

[77] J. A. Azzolini and M. J. Reno, "Impact of Load Allocation and High Penetration PV Modeling on QSTS-Based Curtailment Studies," in *IEEE PES General Meeting*, 2021.

[78] J. A. Azzolini, M. J. Reno, and M. Lave, "Visualization Methods for Quasi-Static Time-Series (QSTS) Simulations with High PV Penetration," in *IEEE 46th Photovoltaic Specialists Conference (PVSC)*, 2019, pp. 0892–0899, doi: 10.1109/PVSC40753.2019.8980550.

[79] D. Feldman, V. Ramasamy, R. Fu, A. Ramdas, J. Desai, and R. Margolis, "U.S. Solar Photovoltaic System and Energy Storage Cost Benchmark," National Renewable Energy Laboratory, Technical Report TP-6A20-77324.

[80] IEEE, "Standard for Interconnection and Interoperability of Distributed Energy Resources with Associated Electric Power Interfaces," IEEE Std. 1547-2018 (Revision of IEEE Std 1547-2003), 2018.

[81] J. Seuss, M. J. Reno, R. J. Broderick, and S. Grijalva, "Analysis of PV Advanced Inverter Functions and Setpoints under Time Series Simulation," *Sandia National Laboratories, SAND2016-4856*, 2016.

[82] K. Mason, M. J. Reno, L. Blakely, S. Vejdan, and S. Grijalva, "A Deep Neural Network Approach for Behind-the-Meter Residential PV Size, Tilt, and Azimuth Estimation," *Solar Energy*, vol. 196, pp. 260–269, Jan. 2020.

[83] S. Grijalva, A. U. Khan, J. S. Mbeleg, C. Gomez-Peces, M. J. Reno, and L. Blakely, "Estimation of PV Location in Distribution Systems Based on Voltage Sensitivities," in *IEEE North American Power Symposium*, 2021.

[84] C. Gomez-Peces, S. Grijalva, M. J. Reno, and L. Blakely, "Estimation of PV Location Based on Voltage Sensitivities in Distribution Systems with Discrete Voltage Regulation Equipment," in *IEEE PowerTech*, Madrid, Spain, 2021.

[85] M. U. Qureshi, S. Grijalva, M. J. Reno, J. Deboever, X. Zhang, and R. J. Broderick, "A Fast Scalable Quasi-Static Time Series Analysis Method for PV Impact Studies Using Linear Sensitivity Model," *IEEE Transactions on Sustainable Energy*, vol. 10, no. 1, pp. 301–310, 2019.

[86] M. U. Qureshi, S. Grijalva, and M. J. Reno, "A Fast Quasi-Static Time Series Simulation Method for PV Smart Inverters with Var Control Using Linear Sensitivity Model," in *IEEE Photovoltaic Specialists Conference*, 2018.

[87] S. Talkington, S. Grijalva, and M. J. Reno, "Power Factor Estimation of Distributed Energy Resources Using Voltage Magnitude Measurements," *Journal of Modern Power Systems and Clean Energy*, vol. 9, no. 4, pp. 859–869, 2021.

[88] K. Christakou, J. LeBoudec, M. Paolone, and D. Tomozei, "Efficient Computation of Sensitivity Coefficients of Node Voltages and Line Currents in Unbalanced Radial Electrical Distribution Networks," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 741–750, 2013.

[89] F. Bu, K. Dehghanpour, Y. Yuan, Z. Wang, and Y. Guo, "Disaggregating Customer-level Behind-the-Meter PV Generation Using Smart Meter Data and Solar Exemplars," *IEEE Transactions of Power Systems*, 2021.

[90] S. Talkington, S. Grijalva, M. J. Reno, and J. Azzolini, "Solar PV Inverter Reactive Power Disaggregation and Control Setting Estimation," *IEEE Transactions on Power Systems (Early Access)*, 2022.

[91] K. Manohar, B. W. Brunton, J. N. Kutz, and S. L. Brunton, "Data-driven Sparse Sensor Placement for Reconstruction: Demonstrating the Benefits of Exploiting known Patterns," *IEEE Control Systems Magazine*, vol. 38, no. 3, pp. 63–86, 2018.

[92] E. Clark, T. Askham, S. L. Brunton, and J. N. Kutz, "Greedy Sensor Placement with Cost Constraints," *IEEE Sensors Journal*, vol. 19, no. 7, pp. 2642–2656, 2019.

[93] S. Talkington, S. Grijalva, and M. J. Reno, "Sparse Time Series Sampling for Recovery of Behind-the-Meter Inverter Control Models," in *IEEE ISGT*, Washington, DC, USA, 2022.

[94] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.

[95] *Enhanced Load Modeling: Survey Results on Industry Load Modeling Practices*. Palo Alto, CA: EPRI, 2020.

[96] *Enhanced Load Modeling: Study of Practical Load Modeling Methods on Distribution Feeders with Behind-The-Meter PV Systems*. Palo Alto, CA: EPRI, 2020.

[97] *Enhanced Load Modeling - Assessment of Load Modeling Methods to Represent Highly Unbalanced Loading Conditions*. Palo Alto, CA: EPRI, 2020.

[98] *Enhanced Load Modeling: Comparison of Load Modeling and Power Flow in DMS and Planning Software*. Palo Alto, CA: EPRI, 2021.

[99] M. Lave, J. E. Quiroz, M. J. Reno, and R. J. Broderick, "High Temporal Resolution Load Variability Compared to PV Variability," *IEEE Photovoltaic Specialists Conference*, 2016.

[100] X. Zhang, S. Grijalva, and M. J. Reno, "A Time-Variant Load Model Based on Smart Meter Data Mining," in *IEEE PES General Meeting*, 2014.

[101] X. Wang, Y. Wang, D. Shi, J. Wang, and Z. Wang, "Two-stage WECC Composite Load Modeling: A Double Deep Q-Learning Networks Approach," *IEEE Transactions on Smart Grid*, vol. 11, no. 5, pp. 4331–4344, Sep. 2020.

[102] WECC, "WECC Composite Load Model with DG Specification," MVWG, Feb. 2015.

[103] B. Mather and F. Ding, "Distribution-connected PV's Response to Voltage Sags at Transmission-scale," in *IEEE PVSC*, Portland, OR, USA, 2016.

[104] State of California, "California's Rule 21."

[105] EPRI, "The New Aggregated Distributed Energy Resources (DER_A) Model for Transmission Planning Studies," Electrical Power Research Institute, Technical Report, 2018.

[106] NERC, "Reliability Guideline: Parameterization of the DER_A Model," North American Electric Reliability Corporation, Technical Report, Sep. 2019.

[107] CYME, "CYME Scripting Tool With Python." CYME Eaton.

[108] C. C. Sun, M. Korkali, E. M. Steward, V. Donde, and M. J. Reno, "Optimization-Based Calibration of Aggregated Dynamic Models for Distributed Energy Resources," in *IEEE PES General Meeting*, 2021.

[109] NREL, "SMART-DS: Synthetic Models for Advanced, Realistic Testing: Distribution Systems and Scenarios," NREL, Models.

[110] W. Wang, N. Yu, B. Foggo, J. Davis, and J. Li, "Phase Identification in Electric Power Distribution Systems by Clustering of Smart Meter Data," in *15th IEEE ICMLA*, Anaheim, CA, USA, 2016, pp. 259–265, doi: 10.1109/ICMLA.2016.0050.

[111] F. Olivier, A. Sutera, P. Geurts, R. Fonteneau, and D. Ernst, "Phase Identification of Smart Meters by Clustering Voltage Measurements," *Power Systems Computation Conference (PSCC)*, 2018.

[112] X. Tang and J. V. Milanovic, "Phase Identification of LV Distribution Network with Smart Meter Data," in *IEEE PES GM*, Portland, OR, USA, 2018.

[113] M. Xu, R. Li, and F. Li, “Phase Identification with Incomplete Data,” *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 2777–2785, Jul. 2018.

[114] S. Wu, W. Hu, and Y. Dong, “A Method for Identifying Connectivity Relationship in Low-Voltage Area Based on Voltage Big Data,” in *IEEE PES Innovative Smart Grid Technologies Asia*, 2019.

[115] W. Luan, J. Peng, M. Maras, B. Harapnuk, and J. Lo, “Smart Meter Data Analytics for Distribution Network Connectivity Verification,” *IEEE Trans. Smart Grid*, vol. 6, p. 1, Jul. 2015.

[116] J. A. Azzolini, S. Talkington, M. J. Reno, S. Grijalva, L. Blakely, and D. Pinney, “Improving Behind-the-Meter PV Impact Studies with Data-Driven Modeling and Analysis,” in *IEEE Photovoltaic Specialists Conference (PVSC)*, 2022.

[117] J. A. Azzolini, M. J. Reno, and K. A. W. Horowitz, “Evaluating Distributed PV Curtailment Using Quasi-Static Time-Series Simulations,” *IEEE Open Access Journal of Power and Energy*, 2021.

[118] J. A. Azzolini and M. J. Reno, “Analysis of Reactive Power Load Modeling Techniques for PV Impact Studies,” Sandia National Laboratories, SAND2022-7033R, 2022.

[119] J. Deboever, M. Hernandez, J. Peppanen, P. Siratarnsophon, and M. J. Reno, “Impact of AMI Data Time Granularity on Quasi-Static time-Series Load Flow Simulation,” in *IEEE Transmission & Distribution Conference & Exposition*, Chicago, IL, USA, 2020.

[120] P. Siratarnsophon, M. Hernandez, J. Peppanen, J. Deboever, M. Rylander, and M. J. Reno, “Improved Load Modelling for Emerging Distribution System Assessments,” in *CIRED Berlin Workshop*, 2020.

[121] J. Peppanen, M. Hernandez, J. Deboever, M. Rylander, and M. J. Reno, “Distribution Load Modeling - Survey of the Industry State, Current Practices and Future Needs,” in *IEEE North American Power Symposium (NAPS)*, 2021.

[122] S. Talkington, S. Grijalva, M. J. Reno, and J. Azzolini, “Recovering Power Factor Control Settings of Solar PV Inverters from Net Load Data,” in *IEEE North American Power Symposium*, 2021.

[123] C. Rocha, J. Deboever, J. Peppanen, M. Hernandez, and M. Rylander, “Improved reactive Power Load Modeling for Distribution Planning,” in *IEEE PES T&D Conference & Exposition*, New Orleans, LA, USA, 2022.

[124] D. R. Garcia, B. Poudel, A. Bidram, and M. J. Reno, “Substation-level Circuit Topology Estimation Using Machine Learning,” in *IEEE Innovative Smart Grid Technologies (ISGT)*, 2022.

[125] J. Yusuf, J. A. Azzolini, and M. J. Reno, “Data-Driven Methods for Voltage Regulator Identification and Tap Estimation,” in *IEEE Kansas Power and Energy Conference (KPEC)*, 2022.

[126] R. D. Trevizan and M. J. Reno, “Detection of False Data Injection Attacks in Power System State Estimation Using Sensor Encoding,” in *IEEE Kansas Power and Energy Conference*, 2022.

DISTRIBUTION

Email—Internal

Name	Org.	Sandia Email Address
Matthew Reno	8813	mjreno@sandia.gov
Logan Blakely	8813	lblakel@sandia.gov
Rodrigo Trevizan	8811	rdtrevi@sandia.gov
Joseph Azzolini	8813	jazzoli@sandia.gov
Summer Ferreira	8812	srferre@sandia.gov
Technical Library	01911	sanddocs@sandia.gov

Email—External

Name	Company Email Address	Company Name
Guohui Yuan	Guohui.yuan@ee.doe.gov	DOE SETO
Mert Korkali	korkali1@llnl.gov	Lawrence Livermore National Lab
Chih-Che Sun	sun31@llnl.gov	Lawrence Livermore National Lab
Jouni Peppanen	jpeppanen@epri.com	EPRI
Santiago Grijalva	sgrijalva@ece.gatech.edu	Georgia Institute of Technology
Feng Li	FengLi1@eaton.com	CYME International T&D
Francis Therrien	FrancisTherrien@eaton.com	CYME International T&D