12-2023

# Model-Free Methods to Analyze Pmu Data in Real-Time for Situational Awareness and Stability Monitoring

Sean Kantra

skantra@clemson.edu

MODEL-FREE METHODS TO ANALYZE PMU DATA IN REAL-TIME FOR
SITUATIONAL AWARENESS AND STABILITY MONITORING

_____

A Dissertation
Presented to
the Graduate School of
Clemson University

_____

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Electrical Engineering

_____

by
Sean Derek Kantra
December 2023

_____

Accepted by:
Dr. Ramtin Hadidi, Committee Chair
Dr. Richard Groff
Dr. Daniel Noneaker
Dr. Sukumar Brahma
Dr. John Wagner

ABSTRACT

This dissertation presents and evaluates model-free methodologies to process Phasor Measurement Unit (PMU) data. Model-based PMU applications require knowledge of the system topology, most frequently the system admittance matrix. For large systems, the admittance matrix, or other system parameters, can be time-consuming to integrate into supporting PMU applications. These data sources are often sensitive and can require permissions to access, delaying the implementation of model-based approaches. This dissertation focuses on evaluating individual model-free applications to efficiently perform functions of interest to system operators for real-time situational awareness. Real-time situational awareness is evaluated with respect to central digitization where the PMU data is archived, and delays from telecommunication and system architecture are not considered.

The PMU data available to utilities is often a subset of the overall system. Even without full observability, PMU data for observable portions of the system provides valuable, high-resolution information about the current system state. Methods are needed that can analyze and generate critical insight about the system in real-time to assist in detection and mitigation of major system events. All chapters address methodologies that can derive their output solely from the PMU signals. These methodologies are evaluated for their reliability and computational efficiency, considering a specific task of interest.

Inter-area oscillations and poorly damped electromechanical modes are dangerous when undetected for extended periods of time, eventually leading to blackouts when unstable parameters are present. Prony Analysis and Matrix Pencil Method were selected

in Chapter 4 for their proven effectiveness of estimating the dominant modes of an input signal; for purposes of this dissertation, the signal of interest for oscillation analysis is real power. The speed of convergence, accuracy of the methods, and viability when applied to utility PMU data were assessed to determine suitability to online system operation. Matrix Pencil Method was determined to provide more robust and computationally efficient estimation of key system modes for both simulated and real utility PMU data.

The biorthogonal discrete wavelet transform, which can correlate frequency data to a time-domain solution, was utilized in Chapter 3 to create a methodology for event detection and classification for a subset of selected events. The derived methodology was shown to be effective for identification and classification of load and capacitor switch events, as well as breaker operation and faults.

Methods to mimic the power flow Jacobian from discrete measurements are derived to assess system stability and eigenvalues in Chapter 2. These methods were effective for fast detection of unstable system parameters. Chapter 5, the most significant contribution of this dissertation, details derivations of a mathematical reduced system model and power flow Jacobian variants for more robust instability detection, system weak point identification, mitigation techniques, and state estimation capabilities. Considering the functions of all evaluated and developed model-free methodologies, event detection, event classification, detection of poorly damped oscillatory modes, and instability detection and mitigation can be achieved for situational awareness.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

Table of Contents (Continued) Page

LIST OF TABLES

List of Tables (Continued)

LIST OF FIGURES

List of Figures (Continued)

List of Figures (Continued)

Figure                                                                                    Page

List of Figures (Continued)

Figure                                                                              Page

CHAPTER ONE

INTRODUCTION

## 1.1    PMU Overview

Phasor measurement units (PMUs) have seen increased implementation and installation in industry power systems. Due to this, PMUs are seeing increased use in applications for protection, controls, and situational awareness [1][2]. PMU devices can return a variety of different quantities, including the current and voltage magnitude of each phase, the corresponding phase angles of these quantities, the sequence voltage and current magnitudes, the phase angles of the sequence voltages and currents, and the frequency at the bus of installation [3]. Many PMUs installed in existing systems report phasor and frequency data at a 30 Hz rate, in accordance with IEEE Standard C37.118-2005, which allows a report rate up to half of the nominal system frequency: 30Hz for a 60 Hz system [4][5]. Many PMUs were installed while this standard was in place. The 60Hz report rate was included and supported by PMUs considering the updated IEEE Standard C37.118-2011 [6], as well as subsequent standards. This dissertation considers the 30Hz report rate, as this is the more restrictive case and is often encountered in utility data. The phase angles are calculated by comparing system waveforms to an absolute time-synchronized reference waveform via Global Positioning System (GPS) [4]. The recorded readings at a specific point in time are compared to the position of the time synchronized absolute reference during calculation. The developed methodologies in this dissertation consider compliance

with PMU data that is streamed by the industry through openPDC [7], which streams PMU

data as individual signals without relationship to system topology.

The increased number of PMUs in the power grid necessitates a need for effective

and high-speed analytic techniques to return viable data near real-time. These methods

need to capitalize on the speed at which the PMU data is returned while converging to a

solution before the next PMU measurements are received, when critical, or within a

timeframe that is suitable to its purpose and allows consumption of all incoming PMU data.

For large systems, the convergence of many applications considering system topology,

such as state estimation, do not resolve in time to utilize the full report rate of the PMUs.

This dissertation considers model-free applications and focuses on the incoming PMU data

for meaningful analysis.

## 1.2    Power Flow Jacobian Utilization for Instability Identification

The power flow Jacobian, derived and calculated when implementing Newton-

Raphson state estimation, requires system topology in order to be utilized. In [8] and [9],

stability of the system is assessed by monitoring eigenvalues and determining when the

Jacobian is reaching singularity. Singularities in the Jacobian lead the matrix to be non-

invertible when used in the Newton-Raphson method. The singularities give an upper

bound to the loading and voltage conditions that would induce instability, since a

singularity in the steady state model assumes that the stable solution can be reached by the

dynamic system; confirming these bounds with greater accuracy requires detailed

generator models and a representation of the control systems [8]. Monitoring the Jacobian

for developing singularities has been reliably used to determine the stability of the system,

with singularity indicating an unstable system. The relation of the power flow Jacobian to Bifurcation and nose curves was presented in [9]. In [10], Singular Value Decomposition (SVD) is used to monitor trends toward singularity and instability in the system. Sources [11] and [12] show the effectiveness and reduced computational complexity of using the decoupled load-flow Jacobian for state estimation. The discrete Jacobian approximations derived in this dissertation utilize the decoupled Jacobian form for computational speed. The decoupled Jacobian only accounts for the partial derivative of real power with respect to bus voltage angle, $\partial P_i / \partial \delta_i$, and the partial derivative of reactive power with respect to bus voltage $\partial Q_i / \partial |V_j|$. The significance of singularity is that when the inverse Jacobian is used to calculate the next iteration's voltage magnitudes and angles, the operation cannot be completed due to the singularity. A zero singularity in the Jacobian will result in an infinite term when taking the inverse. This implies that even an infinite change in bus voltage angle will not change the real power output, and an infinite change in voltage magnitude will be unable to change the reactive power at a given bus. The terms that are leading to the singularity can also be analyzed to show system weak points during operation.

If the system were known and all buses had a PMU placed in the system, the Jacobian could be fully generated for every cycle that the PMU data is returned. However, in large systems, this is not practical and would be expensive to implement and maintain. It is likely that the system PMUs are limited and spaced out, requiring different techniques to analyze the PMU data in real-time. Even with 150 PMUs present in a 3000-bus system, the computational time for convergence of the power flow Jacobian, iteratively calculated

3

during the Newton-Raphson power flow algorithm, would not be substantially reduced. Novel methods to mimic the power flow Jacobian are presented in Chapter 2 as the Decoupled Discrete-Time Jacobian Eigenvalue Approximation (DDJEA) and Expanded Discrete-Time Jacobian Approximation (EDDJA) methods. Chapter 5 presents a derivation to generate a mathematical system topology that can be used to perform Newton-Raphson state estimation. This topology is leveraged to generate two variants of the power flow Jacobian that are compared to determine system weak points and flag divergent system parameters.

## 1.3 Prony Analysis and Matrix Pencil Method for Detection of Undamped Power Oscillations

Another critical situation to consider is the fast and accurate detection of undamped or poorly damped power system oscillations. Sources [13-20] addressed the effectiveness of using Prony analysis to estimate power system modes. In [13], Prony analysis was compared to other methods to estimate the damping ratio and frequency of a power oscillation, including Matrix Pencil Method (MPM) and Hankel Total Least Square (HTLS) method. The slowest dominant system mode and the eigenvalues associated with this mode can be distinguished. Frequency domain decomposition (FDD) is applied to the power spectral density matrix and is compared with Prony Analysis to show that the FDD can adequately match the results of Prony Analysis with decreased computational expense; the methods were applied to ringdown data for a 17-machine model proving applicability for larger systems [14]. The electromechanical mode of a generator is approximated through Prony Analysis to identify inter-area oscillations [15]. Prony Analysis has been

4

applied to a Wide Area Monitoring Systems (WAMS) in order to estimate the dominant eigenvalue, frequency, and damping ratio [16]. Inter-area and electromechanical oscillations can cause cascading line tripping and blackouts if they are not detected. Overall, Prony Analysis is one of the most proven techniques to identify poorly damped oscillations. Further research also considers Matrix Pencil Method to cross compare with Prony Analysis. A major limitation of Prony Analysis that is not shared by the Matrix Pencil Method is that MPM does not have to guess the number of system modes or require any previous analysis to determine the adequate number of modes [17]. Although Prony Analysis properly models the fundamental frequency of a signal, HTLS and MPM outperform Prony when computing the damping ratio on signals with noise [18]. In Prony Analysis, the z-domain transfer function can easily be overdetermined. From the eigenvalues calculated by converting the z-domain function to state space, the eigenvalue that most closely matches the max in the power spectral density can be selected for a quick estimation. The scaling weights of each term can also be used to eliminate terms from the model in the case that the model is overdetermined. The number of terms is unknown, but [19] proposes a method to increase the number of terms until a minimum number of modes is found through a feedback loop. This research focuses on analyzing the real power at each bus to monitor dominant system modes, eigenvalues, and damping ratios to give valuable insight for distinguishing a healthy system from one that is marginally stable or presenting a trend toward instability. Although discussed for a different application in Section 1.4, wavelet transforms can also be used to extract the damping ratio from a signal of interest

[21][22]. Prony Analysis and MPM are evaluated in Chapter 3 with respect to both simulated and utility PMU data.

## 1.4    Event Detection and Classification

Detection and classification of events is crucial to understanding the cause of changes in the system. Either detection or classification can prove difficult during system operation, depending on the complexity of the data and the event of interest. In [23], both modeling and analysis of a variety of power system events are discussed. These events encompass automatic control events, including resetting of governor or exciter setpoints, protection relay operation during fault, under-frequency relay operation, on-load tap changer switching, and shunt capacitor bank switching. In [23], manual control events are also addressed, including connection or disconnection of generators, transmission lines, shunt capacitors, and FACTs devices. Several autonomous disturbance events are considered, including load changes, tripping of generator due to fault, tripping of transmission line due to fault, and lightning strike. In [24], a combined approach utilizing a wavelet transform and neural network is used to identify and classify breaker switching, capacitor switching, low impedance fault, primary arc, and lightning strike events. It should be noted that some events, such as transmission line tripping due to fault, can be interpreted as two overlapping events or a single consolidated event. In [25], a methodology using supervised machine learning is proposed to detect the presence of a fault, define the fault type (single line to ground, line to line, and three phase to ground), and determine if redundant identification can be made for other PMUs with visibility of the event. Some events can pose challenges to simulate accurately, such as high impedance faults (HIFs);

HIF simulation requires a complex non-linear model, like the mathematical model proposed in [26]. HIFs are truly only defined for distribution systems and are not considered in the transmission system simulations, but it was essential to the scope of this dissertation to evaluate a tool that has been proven capable of distinguishing HIF from other system events when analyzing PMU data.

Some PMU applications solely focus on event detection [22][27-30]. In [27], a machine learning algorithm considering modal dynamics is built to support real-time event detection. It is stated in [27] that event identification of a wide range of events is a more difficult problem to solve due to system nonlinearity, so the derived methodology focuses solely on detection. Methodologies developed in [22][28] leveraged the wavelet transform. In [28], the developed algorithm can classify the event as a real or reactive power event, but a more refined classification is not provided; the general location of the event can also be determined. The wavelet transform is used in [22] to monitor system stability by calculating the damping ratio for real power; the advantages of the multilevel decomposition are also addressed for distinguishing between two events that occur within close proximity to each other. Statistical analysis and the peak valley algorithm are utilized in [29] for general event detection. A combination of the short-time Fourier transform (STFT), residual modelling, and linear regression were applied in [30] for comprehensive event detection capabilities.

The methodologies detailed in [24-25][31-40] address both event detection and event classification utilizing PMU data, where the event classification is often limited to a subset of defined events. The application developed in [31] analyzed the third order

harmonic and effects on transformer vector groups to accurately distinguish high impedance fault events. Wavelet transforms were utilized in [24][32]. The Morlet wavelet transform was shown to be effective when used to analyze wide area monitoring systems for identification and classification of poorly damped inter-area and electromechanical oscillations, showing characteristics in the 0.1-0.8Hz range [32]. Applications involving neural networks were also demonstrated to be effective for both event detection and classification, assuming high quality data labels [33-36]. The applications in [33][34] are tailored to accurately detect and classify two specific events: malfunctioned capacitor bank switching and malfunctioned regulator on-load tap changer switching. The convolutional neural network algorithms in [35][36] were designed to detect and classify more general system behavior, such as normal operation, line fault, line outage, transformer outage, and frequency and oscillation events. Machine learning applications for event detection and classification are detailed in [25][37]. As previously noted, [25] implemented targeted classification relating to fault types and affected system areas. In [37], the classification algorithm was built considering four types of logged events: line outage, transformer outage, frequency event, and oscillation event. Additional statistical analytics methods were considered for accurate detection and classification of system events, such as swinging door trending [38], density based spatial clustering of applications with noise combined with a decision tree considering power system analysis [39], and a combination of spectral kurtosis and support vector machines considering short-term energy and group delay [40].

Due to the high impedance fault being one of the most difficult events to accurately detect and classify, additional research was focused on finding methodologies suited to accurately identify HIFs in transient data [41-48]. The high resistive value, non-linearity of resistance, and low current contribution make distinguishing this event from regular system operation difficult. Wavelet transforms are utilized in [41-44] to distinguish HIFs from other system events through decomposition of transient data sets. The use of statistical analysis and pattern recognition, in addition to the discrete wavelet transform, to detect HIF was presented in [43][44]. Time domain approaches, such as ground ratio relays and harmonic distortions, were used in [45][46] for HIF detection. However, these approaches can be ineffective when the system is not balanced or the HIF does not draw enough current to cause a significant deviation. Kalman filtering was used in [47], but the method relies on several assumptions to be related back to the time domain. The Choi–Williams distribution is used in [46], instead of the wavelet transform, for purposes of speed and effectiveness, but the proposed algorithm only works in radial systems and does not take into account current distortion of a real power system.

Depending on the selection of the mother wavelet, the wavelet transform has been proven effective for both event detection and classification for a variety of implementations. Wavelet transforms were also proven useful for HIF detection in both PMU and transient datasets. Although HIF detection is not considered in transmission systems, discrete wavelet transforms are established as a viable candidate for high accuracy event detection, with the ability to identify more subtle event signatures. Utilizing the wavelet transform allows for a relation between the time-domain and frequency domain.

Frequency domain solutions can be difficult to relate changes in fundamental frequency back to the time domain. The wavelet transform circumvents this issue, relating both the time and frequency domain through a multi-level decomposition of the input signal to various cutoff frequencies. Due to this, the wavelet transform was selected to build the event detection and classification methodology explored in this dissertation.

## 1.5    Bridging the Gap in Model-Free Applications

PMUs stream time synchronized data at a report rate that significantly exceeds supervisory control and data acquisition (SCADA) measurements and traditional state estimation solutions, making PMUs more fit to track power system dynamics in real-time [2]. In order to fully leverage the high report rate and data resolution of the PMU devices, applications need to converge near real-time to fully consume the PMU data available. Model-free applications eliminate the requirement to align and integrate system topology for implementation and can be designed for fast convergence. However, model-free applications may lose context to take targeted actions in the real system. It is noted in [49] that the static state estimator for the Great Britain National Grid returns a full solution of all system states every 60 seconds. PMU data in [12] was leveraged to flag windows where transient behavior was present to trigger utilization of a different window for state estimation and improve accuracy, but the speed of convergence was not materially affected.

There are a variety of model-based applications that are designed to be computationally efficient and consume more, or all, of the PMU data streams available. In [50-52], applications for linear state estimators are explored. In [50], a weighted least

square (WLS) solution implementing Gauss-Newton was developed to increase the computational speed of state estimation and reduce impact from bad measurements. A similar approach to fully utilize the PMU data in real-time for state estimation is developed in [51], using a least absolute value (LAV) estimator to increase performance in the presence of bad data. A proof of concept is demonstrated in [52] to show viability of using PMUs for a linear state estimator in the distribution system. These applications require detailed system topology, as well as full system observability, which will limit implementation in most systems.

Hybrid state estimators utilize inputs from both SCADA and PMUs to improve calculation efficiency and accuracy compared to traditional state estimation [53-55]. In [53], bad data rejection and improvements to computational efficiency were accomplished by using a Schweppe-type estimator with an iteratively reweighted least squares solution. The solution convergence is increased by a factor of 1000 times in [54] by employing a K-Nearest Neighbors (K-NNs) search on historical data to set an initial guess based on historical system states, with the application built on data sets back to 2005. A model-free state estimator is presented in [56] that uses a neural network, but this application requires abundant and clean historical data to be accurate. Although [53][54] appear to converge near real-time, a high degree of observability and historical data can limit implementation, and [55] was calculated to take 0.5 seconds to converge for a 181-bus test system. Hybrid state estimators are more practical for implementation in utility systems and present a valuable upgrade with respect to static state estimators, but implementation requires extensive data integration and may still be unable to consume the full report rate of PMUs.

Some model-based applications to assess voltage stability were presented in [57-59]. In [57][58], a WAMS application is presented that can monitor voltage stability and considers dynamic conditions that are beyond the capabilities of static state estimation. In [59], a calculation is performed to monitor the system's proximity to reactive generation thresholds. In addition to the admittance matrix, these applications require full observability and integration of simplified generator models [57][58] or generator limits [59]. Model-free applications to assess voltage stability are discussed in [60-63]. In [60][61], the Lyapunov exponents of voltage magnitudes are calculated to flag when a voltage signal is exponentially diverging. Although this implementation is computationally efficient, utilizing only the voltage magnitude does not leverage the full potential of the PMU data. Additionally, mitigation actions do not appear to be clear. A methodology to map changes in real and reactive power to changes in bus voltage magnitude and bus voltage angle is developed in [62][63]. It is shown that, when given a sufficient window length, SVD of the derived matrix results in a nearly identical maximum singular value when compared to implementing SVD on the true Jacobian matrix. The application was shown to be effective in identifying major events and topology changes in the real system. However, utilizing the data to mitigate unstable system conditions was unclear and for an IEEE 300 bus system, 8640 data samples took 100 seconds to process, which is reflective of the window length and performance required for processing PMU data from large systems.

Chapter 5 presents the development of an application that derives a mathematical system topology and load flow to construct two Jacobian variants. These Jacobian variants

are analyzed via SVD and comparing the system state to detect system weak points, identify the type of instability (angular or voltage), and show that the application can be used to mitigate unstable load conditions. The application is designed to be scalable and flexible and accommodate the PMU measurements available in the system. There is no need for integration of historical datasets. Lastly, the computational efficiency and sub-cycle convergence of the application is assessed to determine feasibility for large systems, with 112 PMUs from a large utility system integrated into the application.

## 1.6    Additional Approaches and Addendums to Utilizing PMU Data

Various other methods and algorithms to effectively utilize PMU data are presented in [64-68]. Quick detection of unstable conditions is necessary, but there is also a need to relay what the data means to both power engineers and system operators. Retrieving archived data related to events and system conditions has been a value application for system operators [64]. The National Renewable Energy Laboratory, a government institution, released a detailed explanation of viable methods for analyzing PMUs as well as explaining the output signals that PMUs return [65]; This text also details prominent analysis tools like Yule-Walker and Matrix Pencil Method. A very similar detailed reference is [41], with more focus on visualization and relation to system operators. The overwhelming amount of data processed by system operators is addressed in [66] with an algorithm that poses to archive system characteristics during a major event for cross comparison to present system operations, automated report generation after an event has occurred, and classifications of system characteristics with respect to previous events to aid in solution. This form of big data management takes a practical approach that appeals

13

to system operators. Another similar solution allows for simulation of previous events, permitting the system operator to change variables to the simulated system to discern what could have been done to change the outcome [67]. The methodology proposed in [68] utilizes the STFT and thresholds over a small moving window to detect certain system events, but then utilizes a combination of Yule Walker and MPM to detect anomalies for further offline analysis. In [2], a novel implementation of fuzzy logic was applied to PMU data to monitor voltage stability and enhance situational awareness through proposed PMU placement. PMU data is being used and researched with respect to a wide range of power system use cases and applications. Many tools and approaches have been developed to make use of the data or investigate the data's viability to solve power system issues. The list of assumptions to create these methods can be very broad and can limit implementation in a practical system. The research conducted in this dissertation avoids assumptions that could limit sensible implementation.

## 1.7 Overview of Dissertation Research Objectives

The goal of this research is to develop real-time applications that can individually fulfill a desired component necessary for situational awareness, both succinctly and accurately. In order for this to be achieved, there must be methods that can correctly detect and identify system events. There must also be methods to give system stability bounds and system weak points. Then another set of methods can be used to identify slow system oscillations that are leading to unstable conditions, which are difficult to detect unless a specialized application is considered. Lastly, a methodology with flexibility and

practicality to identify and mitigate unstable system conditions in the real system needs to be considered.

The Discrete-Time Decoupled Jacobian Eigenvalue Approximation (DDJEA) and the Expansion of the Decoupled Discrete-Time Eigenvalue Approximation (EDDJA) were established in [69] and [70] respectively. Chapter 2 of this dissertation is a detailed explanation and derivation of these methods. Least squares and linear regression are applied to PMU data in order gain a sense of connectivity for the EDDJA method. The concept of the power flow Jacobian was incorporated in this method by defining bounds for approaching singularity.

A method to detect HIF through statistical analysis was presented in [71]. Although this method is not directly utilized due to limitations in identifying the variety of events, statistical analysis and Gaussian distributions are implemented when defining the cutoff values for the output of biorthogonal wavelet transforms. Chapter 3 presents the use of biorthogonal discrete wavelets to detect and classify system events. A novel application of the discrete wavelet transform to analyze real power, reactive power, bus voltage, bus voltage angle, phase current, and phase current angle is demonstrated. By comparing the peak wavelets during an event across each signal of interest, event detection and identification can be determined. Other time-domain techniques are incorporated to increase the accuracy of identification and decrease the possibility of an incorrect conclusion. The biorthogonal wavelet transform is also applied in an application utilizing a shorter window of observation (5 seconds). This modified application also employs

power system analytics in parallel to gain useful insight into system events and enhance situational awareness.

Methodologies to calculate the dominant frequencies and damping ratios of the real power signal are discussed in Chapter 4. The Matrix Pencil Method (MPM) and Prony Analysis are compared to ascertain effectiveness and speed when calculating the dominant frequency and damping ratio of a signal. Both methods are powerful tools for identifying poorly damped oscillatory system modes. Performance with respect to a known signal, engineered to test reliability and detection of unstable parameters, and viability when applied to real utility data are assessed.

Chapter 5 presents a novel application to create a reduced system topology, notated as the Mathematical Synthetic System Model (MSSM). Where Chapter 2 applies discrete approximations, Chapter 5 leverages the MSSM to derive power flow Jacobians through direct measurement and computation via the Newton-Raphson power flow algorithm. A novel methodology is presented that demonstrates the ability to detect system weak points and divergent parameters through SVD and comparison between the two Jacobian variants. The methodology is utilized to show immediate detection of unstable conditions and is leveraged to mitigate unstable load conditions. Computational efficiency is demonstrated for the IEEE 39 bus system, simulated with observability of the 10 generators. The methodology is also applied to real utility PMU data for an equivalent 20-machine and 50-machine derivation. Flexibility and computational scalability of the algorithm is demonstrated further by incorporating 112 PMUs with different configurations from the utility dataset for real-time convergence.

CHAPTER 2

DEVELOPMENT OF A METHODOLOGY FOR INSTABILITY DETECTION USING
DISCRETE JACOBIAN APPROXIMATION

## 2.1 Discrete Jacobian Approximation for Real Power

When system topology, connectivity, and loads are not known, the true power flow

Newton-Raphson method and associated Jacobian matrix cannot be applied. However, it

is possible to develop a reduced matrix that has a similar function as the decoupled

Jacobian. More specifically, it would function as a discrete decoupled Jacobian eigenvalue

approximation (DDJEA) matrix. This method is then expanded upon to show that even

without system connectivity, a general sense of connectivity can be gained from the PMU

data over large sample periods: this method is called the Expanded Discrete Decoupled

Jacobian Approximation (EDDJA). A fundamental assumption for both methods is that the

Jacobian should not change significantly between two cycles, unless under a significant

system event. For the case of a fault, the Jacobian approximation of the previous cycle will

have an increased error, but this can be used to aid in event detection. Both the DDJEA and

EDDJA methods utilize a net-power approach. This is due to the lack of system

connectivity and load information. However, after the derivation of the EDDJA matrix, it

will be shown that mathematically it can carry out a similar function in the Newton

Raphson method. This will be addressed after the derivation of the EDDJA matrix. In order

to prove the effectiveness of these algorithms, they will be directly substituted into the

Newton-Raphson method in place of the Jacobian to show that the approximation can

produce almost identical results for a test case. For equations below, variable N represents

the total number of buses in the system, and variable n represents the total number of buses with a PMU installed. The mathematical derivations detailed in this chapter were published in [69][70] as part of this research. The formal definition of the real power portion of the decoupled Jacobian is presented in **Equation 1** and **Equation 2**.

$$\Delta P_i = \sum_{j=1}^{N} \frac{\partial P_i}{\partial \delta_j} * \Delta \delta_j \tag{1}$$

$$\frac{\partial P_i}{\partial \delta_j} = -\left|Y_{ij} V_i V_j\right| \sin\left(\theta_{ij} + \delta_j - \delta_i\right)(2) \tag{2}$$

$P_i$ denotes the real power injection at bus (i) for a specific measurement. $\Delta P_i$ denotes the difference between the target value and the value considering the current iteration's values. With respect to the discrete matrices, $\Delta P_i$ denotes the change between the current measurement and the previous measurement. $\Delta \delta_j$ is the difference of voltage angle at bus (j) between two measurements or iterations in radians. The partial derivative of real power at a bus (i) with respect to bus (j) voltage angle is shown in **Equation 2**. $Y_{ij}$ denotes the $Y_{bus}$ p.u. admittance value between bus (i) and bus (j) in p.u. In the case of i=j, the $Y_{ij}$ element is the diagonal term of the matrix at row (i). $V_i$ and $V_j$ are the voltages at bus (i) and bus(j), respectively, in p.u. $\Theta_{ij}$ represents the phase angle for the $Y_{ij}$ term in radians. The $\delta_i$ and $\delta_j$ terms are the voltage angles at buses (i) and (j), respectively, in radians.

The proposed method utilizes a model-free approach to mimic the general form of the decoupled Jacobian without needing full system connectivity or loading conditions. The method was developed considering the use of only openPDC data as a window to the PMU output to assist in real-time situational awareness. For example, consider a 4000-bus system with 150 PMUs installed at a subset of selected buses. In this case, utilizing the

PMU data to aid Newton-Raphson iterations would not significantly increase the speed of convergence of the Jacobian matrix. Even if the algorithm was used to only create a decoupled power flow Jacobian, two matrices of size 4000x4000 would need to be generated (considering the aforementioned system size): one for real power and one for reactive power. This would lead to most of the PMU data never being used since the solution for the power flow Jacobian is unlikely to converge before the next measurement. Given that the system in this case is large, it is very possible that of the 150 installed PMUs, none of the PMUs are directly connected. This would lead to 0 terms in the off-diagonal when developing a reduced discrete Jacobian form. Furthermore, considering the other system buses is computationally expensive, so an approximation method would be helpful when considering a real-time application.

For this, a new expression for $\Delta P_i$ is required with some fundamental assumptions. This method assumes that the time step between measurements is small. Standard PMU measurements typically return at a 30Hz or 60Hz rate, which would return a measurement every one or two cycles. For one or two cycles, the power flow Jacobian should not change significantly unless under a major system event. The approximation method proposed will be inaccurate for large time steps of seconds or minutes. Since the model-free approach does not consider connectivity or topology, all changes in net power at a bus for the initial approximation are assumed to be due to the change in that particular bus's voltage angle. This leads to an eigenvalue matrix that mimics the form of the power flow Jacobian. Instead of using N buses for the system, the reduced method only utilizes the n buses with PMUs installed. The time step will be denoted $\Delta t$. In order to test the viability of this method, the

19

DDJEA of the previous iteration will be used to predict the change in real power at the current time step. For purposes of notation, the prediction will be notated as (t+Δt). This is also the time at which the most recent measurement is being taken. Then the predicted and actual values will be compared to assess whether the approximation accurately predicts the next state when given either the change in real power or change in bus voltage angle. **Equation 3** shows a general format and then **Equations 4-8** demonstrate the rest of the derivation.

$$\Delta P_i(t + \Delta t) = \frac{\Delta P_i(t)}{\Delta \delta_i(t)} * \Delta \delta_i(t + \Delta t) \tag{3}$$

$$\Delta P_i(t + \Delta t) = \frac{P_i(t) - P_i(t - \Delta t)}{\delta_i(t) - \delta_i(t - \Delta t)} * \left(\delta_i(t + \Delta t) - \delta_i(t)\right) \tag{4}$$

$$P_{i_{predicted}}(t + \Delta t) = P_{i_{actual}}(t) + \Delta P_i(t + \Delta t) \tag{5}$$

$$P_{i_{actual}}(t + \Delta t) = |V_{i_1}| * |I_{i_1}| * \cos(\delta_{V i_1} - \delta_{I i_1}) \tag{6}$$

$$\begin{bmatrix} \Delta P_1(t + \Delta t) \\ \vdots \\ \Delta P_n(t + \Delta t) \end{bmatrix} = \begin{bmatrix} \frac{\Delta P_1(t)}{\Delta \delta_1(t)} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{\Delta P_n(t)}{\Delta \delta_n(t)} \end{bmatrix} \begin{bmatrix} \Delta \delta_1(t + \Delta t) \\ \vdots \\ \Delta \delta_n(t + \Delta t) \end{bmatrix} \tag{7}$$

$$DDJEA = \begin{bmatrix} \frac{\Delta P_1(t)}{\Delta \delta_1(t)} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{\Delta P_n(t)}{\Delta \delta_n(t)} \end{bmatrix} \tag{8}$$

The $P_{ipredicted}$ term indicates the predicted power at bus (i) while the $P_{iactual}$ term represents the direct power measurement at a particular cycle for the same bus. It is important to emphasize that the most recent measurement in all of these derivations is considered to be at time (t+Δt). This would mean that $\delta_i(t)$ denotes the bus voltage phase angle at bus (i) of the previous measurement. $V_{i1}$ and $I_{i1}$ denote the positive sequence

voltage and current at bus (i) respectively. Similarly, $\delta_{Vi1}$ and $\delta_{Ii1}$ are the phase angles of the positive sequence bus voltage and current at bus (i).

**Equation 8** shows the form of the Discrete Decoupled Jacobian Eigenvalue approximation matrix. **Equation 9** shows a similar representation to **Equation 7**. **Equation 10** shows the format of the real power decoupled power flow Jacobian to cross compare with **Equation 8**.

$$\begin{bmatrix} \Delta P_1(t + \Delta t) \\ \Delta P_2(t + \Delta t) \\ \vdots \\ \Delta P_N(t + \Delta t) \end{bmatrix} = \begin{bmatrix} \frac{\partial P_1}{\partial \delta_1} & \frac{\partial P_1}{\partial \delta_2} & \cdots & \frac{\partial P_1}{\partial \delta_N} \\ \frac{\partial P_2}{\partial \delta_1} & \frac{\partial P_2}{\partial \delta_2} & \cdots & \frac{\partial P_2}{\partial \delta_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_N}{\partial \delta_1} & \frac{\partial P_N}{\partial \delta_2} & \cdots & \frac{\partial P_N}{\partial \delta_N} \end{bmatrix} \begin{bmatrix} \Delta \delta_1(t + \Delta t) \\ \Delta \delta_2(t + \Delta t) \\ \vdots \\ \Delta \delta_N(t + \Delta t) \end{bmatrix} \tag{9}$$

$$Decoupled\ Real\ Power\ Jacobian = \begin{bmatrix} \frac{\partial P_1}{\partial \delta_1} & \frac{\partial P_1}{\partial \delta_2} & \cdots & \frac{\partial P_1}{\partial \delta_N} \\ \frac{\partial P_2}{\partial \delta_1} & \frac{\partial P_2}{\partial \delta_2} & \cdots & \frac{\partial P_2}{\partial \delta_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial P_N}{\partial \delta_1} & \frac{\partial P_N}{\partial \delta_2} & \cdots & \frac{\partial P_N}{\partial \delta_N} \end{bmatrix} \tag{10}$$

For any buses that are not directly connected, the corresponding $Y_{bus}$ values will be zero. The Expansion of the Discrete Decoupled Jacobian Approximation (EDDJA) gives time-varying weights and values to off-diagonal terms by applying linear models to large sample sizes. These weights can help account for changes in system topology and events as they occur in real-time and modify the EDDJA values to reduce error from the algorithm. This brings back the consideration of connectivity to the discrete model. Off-diagonal terms are only zero for all measurements when the $Y_{bus}$ is zero in the traditional power flow Jacobian. These weighted terms in **Equation 11** give a reduced form of system connectivity, allowing more than one element to predict the change in power at a bus.

Although the model is reduced, statistically significant terms can be used in the final representation to give a general sense of system connectivity not considered in the DDJEA method. The proposed final format of the EDDJA method is shown in **Equation 11** and the EDDJA matrix is highlighted in **Equation 13**. **Equation 12** illustrates the calculation for the change in real power at bus (i).

$$\begin{bmatrix} \Delta P_1(t+\Delta t) \\ \vdots \\ \Delta P_n(t+\Delta t) \end{bmatrix} = \begin{bmatrix} \alpha_{11}(t)\frac{\Delta P_1(t)}{\Delta \delta_1(t)} & \alpha_{12}(t)\frac{\Delta P_1(t)}{\Delta \delta_2(t)} & \cdots & \alpha_{1n}(t)\frac{\Delta P_1(t)}{\Delta \delta_n(t)} \\ \alpha_{21}(t)\frac{\Delta P_2(t)}{\Delta \delta_1(t)} & \alpha_{22}(t)\frac{\Delta P_2(t)}{\Delta \delta_2(t)} & \cdots & \alpha_{2n}(t)\frac{\Delta P_1(t)}{\Delta \delta_1(t)} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n1}(t)\frac{\Delta P_n(t)}{\Delta \delta_1(t)} & \alpha_{n2}(t)\frac{\Delta P_n(t)}{\Delta \delta_2(t)} & \cdots & \alpha_{nn}(t)\frac{\Delta P_n(t)}{\Delta \delta_n(t)} \end{bmatrix} \begin{bmatrix} \Delta \delta_1(t+\Delta t) \\ \vdots \\ \Delta \delta_n(t+\Delta t) \end{bmatrix} \quad (11)$$

$$\Delta P_i(t+\Delta t) = \sum_{j=1}^{n} \alpha_{ij}(t) * \frac{\Delta P_i(t)}{\Delta \delta_j(t)} * \Delta \delta_j(t+\Delta t) \quad (12)$$

$$EDDJA = \begin{bmatrix} \alpha_{11}(t)\frac{\Delta P_1(t)}{\Delta \delta_1(t)} & \alpha_{12}(t)\frac{\Delta P_1(t)}{\Delta \delta_2(t)} & \cdots & \alpha_{1n}(t)\frac{\Delta P_1(t)}{\Delta \delta_n(t)} \\ \alpha_{21}(t)\frac{\Delta P_2(t)}{\Delta \delta_1(t)} & \alpha_{22}(t)\frac{\Delta P_2(t)}{\Delta \delta_2(t)} & \cdots & \alpha_{2n}(t)\frac{\Delta P_1(t)}{\Delta \delta_1(t)} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n1}(t)\frac{\Delta P_n(t)}{\Delta \delta_1(t)} & \alpha_{n2}(t)\frac{\Delta P_n(t)}{\Delta \delta_2(t)} & \cdots & \alpha_{nn}(t)\frac{\Delta P_n(t)}{\Delta \delta_n(t)} \end{bmatrix} \quad (13)$$

Once terms are shown to be statistically significant, a time-varying weight is given to each term. This weight is calculated through a Least Squares method using overdetermined equations. The number of statistically significant terms and the role these terms play in accuracy of the model will be discussed after the derivation. The $\alpha_{ij}$ term denotes the time varying weights associated with each term in the EDDJA matrix. Unlike the $Y_{bus}$ terms, $\alpha$ terms change between every cycle, but this is due to their function in also accounting for variability in the reduced system.

Unlike DDJEA, which is a real-time application with no start up or offline analysis required, EDDJA requires offline analysis over a large data set in order to determine

statistically significant terms. Linear models must be built for every bus with a PMU installed in the system. Consider a 20-minute window where PMU measurements are read. This would result in a [35,999 x n ] matrix, at a 30 Hz report rate, for each bus with a PMU in the system in order to determine whether each column is statistically significant when predicting the real power change. The linear model is derived in **Equations 14-25.**

$$\Delta P_{i_{linear\ model}} = B_0 + B_1 * X_{i1} + B_2 * X_{i2} \dots + B_n * X_{in} \tag{14}$$

**Equation 14** builds a linear model for the change in real power at bus (i). In this equation, the (i) term references the particular bus number of interest when constructing the linear model. The (j) term denotes the bus being analyzed to determine if it is statistically significant to bus (i). Each $X_{ij}$ term can be expanded as shown in **Equation 15**. The specific case is given considering whether the column vector used to determine whether the PMU at bus (2) can be used to determine the change in power at bus (1). The 36,000 prior measures are utilized when constructing this column vector.

$$X_{12_{samples}}(dimension\ 35999\ x\ 1) = \begin{bmatrix} x_{12}(t + \Delta t) \\ x_{12}(t) \\ x_{12}(t - \Delta t) \\ \vdots \\ x_{12}(t - 35{,}997\Delta t) \end{bmatrix} \tag{15}$$

$$x_{12}(t + \Delta t) = \frac{\Delta P_1(t)}{\Delta \delta_2(t)} * \Delta \delta_2(t + \Delta t) \tag{16}$$

$$\Delta P_{1_{samples}}(dimensions\ 35999\ x\ 1) = \begin{bmatrix} \Delta P_{1actual}(t + \Delta t) \\ \Delta P_{1actual}(t) \\ \Delta P_{1actual}(t - \Delta t) \\ \vdots \\ \Delta P_{1actual}(t - 35{,}997\Delta t) \end{bmatrix} \tag{17}$$

The model is applied so that each bus is individually tested and analyzed in order to determine whether it is statistically significant to the overall model. Each $x_{ij}$ term is effectively the product of the change in power at bus (i) divided by the change in bus voltage angle at bus (j) of the previous measurement in radians and the value of the bus voltage angle at bus (j) of the most recent observation. If this term were used without a weight, then all the change in power would be assumed to be influenced by bus (j). **Equation 16** presents the definition of the independent variable for the linear model. The test was coded in R to build a linear model with all terms included. Each $X_{ij}$ term is a column vector that contains all $x_{ij}$ terms as per **Equation 15**. The column vectors are used to predict the actual power change at the bus, held in a similar column vector shown in **Equation 17.** For each PMU, there are (n) $X_{ij}$ column vectors incorporated for the linear model to determine whether there are other buses than the diagonal term that can be used to reduce error from the model. $\Delta P_{i_{samples}}$ is a single column vector that stores the actual change in power at bus (i) for all the time increments. The linear model is built so that the estimate of $\widehat{\Delta P_i}$ is minimized considering all instances T, where T is a specific point in time. **Equation 18** shows the notation for estimating the change in power at PMU 1 considering all PMUs in the system. **Equation 19** shows the general expression for how the data is related in R code. **Equation 20** demonstrates the simple linear model relation more clearly, assuming that the linear model is true.

$$\widehat{\Delta P_1(T)} = \widehat{\beta_{10}} + \sum_{j=1}^{n} \widehat{\beta_{1j}} * X_{1j}(T) \tag{18}$$

$$\overbrace{\begin{bmatrix} \Delta P_{1actual}(t + \Delta t) \\ \Delta P_{1actual}(t) \\ \Delta P_{1actual}(t - \Delta t) \\ \vdots \\ \Delta P_{1actual}(t - 35{,}997\Delta t) \end{bmatrix}}^{\Delta P_{1samples}} =$$

$$\begin{bmatrix} \overbrace{x_{11}(t + \Delta t)}^{X_{11}} & \overbrace{x_{12}(t + \Delta t)}^{X_{12}} & \cdots & \overbrace{x_{1n}(t + \Delta t)}^{X_{1n}} \\ x_{11}(t) & x_{12}(t) & \cdots & x_{12}(t) \\ x_{11}(t - \Delta t) & x_{12}(t - \Delta t) & \cdots & x_{1n}(t - \Delta t) \\ \vdots & \vdots & \cdots & \vdots \\ x_{11}(t - 35{,}997\Delta t) & x_{12}(t - 35{,}997\Delta t) & \cdots & x_{1n}(t - 35{,}997\Delta t) \end{bmatrix} \tag{19}$$

$$\widehat{\Delta P_{1samples}} = \widehat{\beta_0} + \widehat{\beta_{11}}X_{11} + \widehat{\beta_{12}}X_{12} + \widehat{\beta_{13}}X_{13} \ldots \widehat{\beta_{1n}}X_{1n} \tag{20}$$

The appropriate $\beta$ values are approximated from the data to individually reduce the sum of least squares for each term in the model. The issue with this model, in regards the final format desired by **Equation 11**, is the $\beta$ values are constant, not time varying. However, these $\beta$ values can be individually tested to determine which PMUs are significant when estimating the change in real power at a bus. Furthermore, not all $\beta$ values used to build the model are statistically significant, so reducing them from the model increases efficiency. In order to determine significance, the $p_{value}$ of each term is calculated, as well as the $p_{value}$ of the overall model. When all unnecessary terms are dropped from the model, the final result must be tested for a sufficient $p_{value}$.

In many statistical applications, a $p_{value}$ less than 0.05 or 0.01 is sufficient to show that a term is statistically significant to the model. However, in this case there are factors to consider for lowering the required level of significance for both the terms and models.

The duality of adding all significant terms to the linear model is that it will not necessarily improve the accuracy when applied to the time-varying model. The nature of EDDJA and DDJEA matrices is that they fluctuate over time and their weights should similarly fluctuate. The $\beta$ values calculated to determine the linear model do not play a role in the implementation of the EDDJA matrix other than determining which terms should be included. Furthermore, the accuracy of the EDDJA method is dependent upon the assumption that matrix does not change drastically over a short period of time unless experiencing a system event. These terms do give a sense of a reduced system connectivity, but adding too many terms to the model is not ideal, as will be discussed in the application of overdetermined equations to predict temporary linear models by Least Squares Estimation. Computational speed is another concern when considering which values should be considered. If the model can be reduced without affecting the desired accuracy, that specific term should be given a weight of zero so that only the most significant terms are considered. **Equation 21** and **Equation 22** set in place the general criteria that were used when selecting the most significant terms.

$$For\ p_{\beta_{ij}} > 0.00001,\ \beta_{ij} = 0 \tag{21}$$

$$For\ p_{model_i} > 0.00001,\quad Reject\ Linear\ Model \tag{22}$$

The term $p_{\beta_{ij}}$ represents the test statistic for an individual term relating the significance of bus(j) voltage angle when predicting the change in power at bus(i). Since the data being used is large, the mean of the data is assumed to be approximately Gaussian when considering the Central Limit Theorem [72][73]. The standard assumption is that for normal data, the distribution will be approximately Gaussian about the mean for a sample

term. A perfect Gaussian distribution will not be obtained by a finite amount of real data, but a large sample will have Gaussian properties about the mean. In this case, the sample size is much greater than 30 samples. The Gaussian distribution of each term is used to calculate a simple hypothesis test summarized in **Equation 23**.

$$H_0: \beta_{ij} = 0 \; ; \; H_A: \beta_{ij} \neq 0 \; ; \alpha = 1x10^{-5} \rightarrow For \; p_{\beta_{ij}} > \alpha, reject \; H_0 \qquad (23)$$

The $H_0$ term is the null hypothesis, and the $H_A$ term is the alternative hypothesis. The $\alpha$ term is the level of significance for the model. The R code considers this test when calculating the $p_{value}$ via the summary of the linear model. The level of significance has a practical purpose: it limits the chance that the null hypothesis is rejected when it should not be rejected. In the context of this algorithm, the level of significance limits terms from being accepted into the final model when that individual element is not statistically significant and should be ignored. A level of significance of $1\times10^{-5}$ creates an upper bound so that there is at most a 0.001% chance that any terms placed into the model should have been excluded. A similar way of looking at this is through confidence intervals. There is a confidence interval associated with each $\beta_{ij}$ term. The confidence interval is set at $(1-\alpha)\times100\%$. A 99.999% confidence interval is constructed which has a 99.999% chance, when using a level of significance of $\alpha = 1 \times 10^{-5}$, of enclosing the true value of the beta term. If zero is not in this confidence interval, then the term is assumed to be statistically significant. For any $p_{value}$ above this threshold, the null hypothesis will not be rejected, causing the weight for that term to be indefinitely set to zero. This would also imply that no sense of connectivity can be assumed between bus (i) and (j).

A similar test is applied to the overall linear model. If PMU 1, PMU 5, and PMU 15 have β values with test statistics (p-values) lower than the specified level of significance, α, then resulting linear model would be as follows in **Equation 24**.

$$\Delta P_1 = \beta_{11}X_{11} + \beta_{15}X_{15} + \beta_{1,15}X_{1,15} \qquad (24)$$

$$For\ p_{model1} < \alpha\ \rightarrow Assume\ model\ is\ significant$$

The $p_{value}$ of the linear model is computed in software and the resulting test statistic is compared to the level of significance for the model itself presented in **Equation 24**. If the entire model is deemed statistically significant, then all significant terms are considered when building reduced connectivity for the system in relation to PMU (i).

The entire process is summarized in **Figure 2.1** to help aid in the overall understanding of how the model for each PMU is developed. Unless there is some conflict with the terms, the final model should meet the requirements. Otherwise, the model would need to be reduced further. In the event that too many terms are added to a predictive model for the change in real power at a PMU, the model would be reduced to only the most significant terms. The reason for this will be explained in the derivation of the $\alpha_{ij}$ terms.
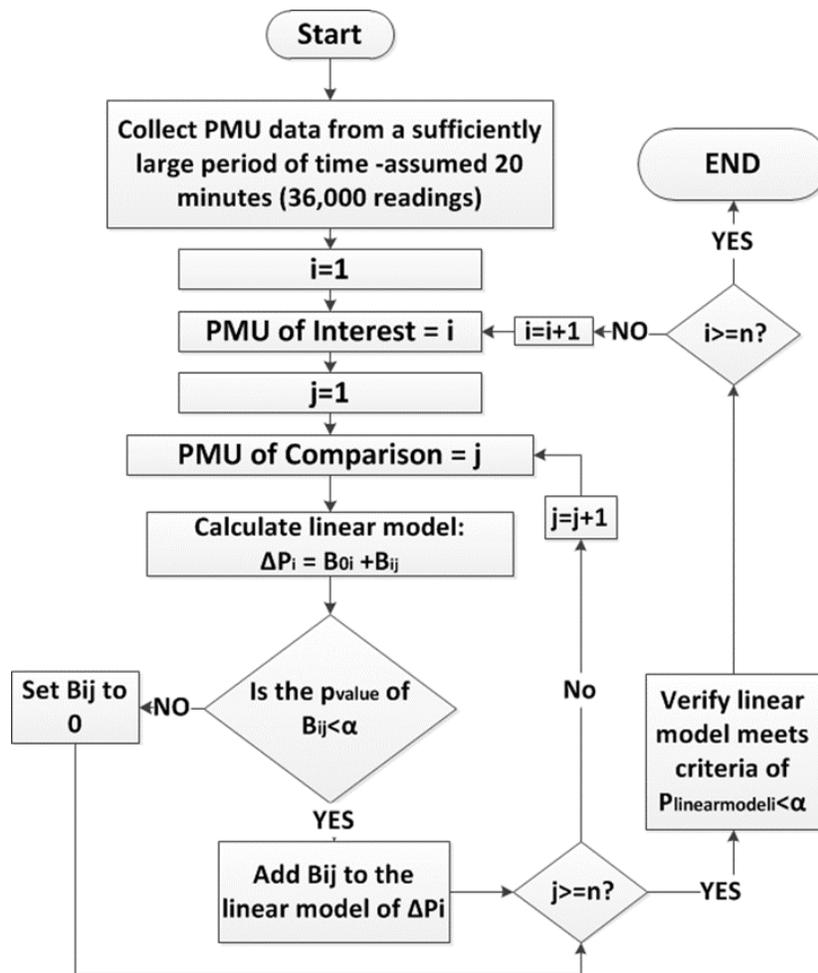
**Figure 2.1 Application of Linear Models to Derive EDDJA Significant Terms**

Another common way of referencing the effectiveness of a model is by looking at the coefficient of correlation, $R^2$. The model is built over a large sample of fluctuating data; this is not necessarily ideal for producing a high $R^2$, especially when considering that the data could contain events that caused fundamental terms to change entire orders of magnitude. This would cause the term to have one highly inaccurate reading around the time of the fault when using the last sample's measurements to estimate the system's behavior and several other inaccurate measurements as the system converges to a new

29

equilibrium following the event. Since the coefficient of correlation is a percentage of the variation in the model that can be explained [72], most models below $R^2 = 0.9$ are disregarded. However, in this case it is important to recognize that this is not the case since extreme outliers can drastically affect this number. Considering real industry PMU data, the full linear model yielded a $R^2$ value of 0.573. The $p_{value}$ for the model was calculated to be less than $2.2 \times 10^{-16}$. From the $R^2$ value alone, the model may not seem adequate, but the $p_{value}$ is the lowest possible value that R-code can compute, meaning the significance of the model is effectively 100%, with a $p_{value}$ approaching a limit close to zero. A true value of 0 can never be reached, as per the nature of the Gaussian distribution. The reduced model had an even more unsettling $R^2$ value of 0.091 over the entire period of 4225 sample measurements per PMU channel, but the $p_{value}$ of the model was not changed by the reduction, remaining at less than $2.2 \times 10^{-16}$. By looking at the data over a shorter time span, it was shown that an outlier was the source of error. If only the first 100 measurements were considered, the full model (53 PMUs) yielded an $R^2$ value of 1. The reduced model, considering only two PMUs, produced an $R^2$ value of 0.9996 when building the linear regression with only the first 100 measurements. The $p_{value}$ for both models remained the same at $2.2 \times 10^{-16}$. When the first thousand measurements were considered for reduced linear model, the $R^2$ value of the model plummeted to 0.033 and a $p_{value}$ of $4.32 \times 10^{-8}$ with a near singularity in the EDDJA matrix being the culprit. However, when the set of (1000 to 2000) measurements were considered for building the reduced model, the result yielded an $R^2$ of 0.953 and a $p_{value}$ at the minimum possible value. Along smaller time segments, a linear model will not be affected by outliers in the data unless those outliers are enclosed

in the sample set. This is the basis for applying the Least Squares estimation over short time periods in order to reduce error in the EDDJA itself. Although the Least Squares approximation is not time-varying at a particular point, building a linear model of a set length that shifts through the data allows the weights in the EDDJA method to update. Jacobians can change drastically over time so applying a static or constant relationship between buses with PMUs over a large period of time is not reasonable.

Once the significant terms of each PMU are determined, these terms are placed back into the EDDJA matrix. All other terms are set to zero. Consider a case where the system contains 4 PMUs. PMU 1 and PMU 4 are derived to be significant to each other. Likewise, PMU 2 and PMU 3 are shown to be linearly related in the regression model. **Equation 25** shows the format of the EDDJA method considering these relationships.

$$
\begin{bmatrix} \Delta P_1(t + \Delta t) \\ \Delta P_2(t + \Delta t) \\ \Delta P_3(t + \Delta t) \\ \Delta P_4(t + \Delta t) \end{bmatrix} =
$$

$$
\begin{bmatrix} \alpha_{11}(t)\frac{\Delta P_1(t)}{\Delta \delta_1(t)} & 0 & 0 & \alpha_{14}(t)\frac{\Delta P_1(t)}{\Delta \delta_4(t)} \\ 0 & \alpha_{22}(t)\frac{\Delta P_2(t)}{\Delta \delta_2(t)} & \alpha_{23}(t)\frac{\Delta P_2(t)}{\Delta \delta_3(t)} & 0 \\ 0 & \alpha_{32}(t)\frac{\Delta P_3(t)}{\Delta \delta_2(t)} & \alpha_{33}(t)\frac{\Delta P_3(t)}{\Delta \delta_3(t)} & 0 \\ \alpha_{41}(t)\frac{\Delta P_4(t)}{\Delta \delta_1(t)} & 0 & 0 & \alpha_{44}(t)\frac{\Delta P_4(t)}{\Delta \delta_4(t)} \end{bmatrix} \begin{bmatrix} \Delta \delta_1(t + \Delta t) \\ \Delta \delta_2(t + \Delta t) \\ \Delta \delta_3(t + \Delta t) \\ \Delta \delta_4(t + \Delta t) \end{bmatrix} \quad (25)
$$

A similar symmetry to the decoupled power flow Jacobian can be seen in **Equation 25,** but this model considers only the buses with PMUs for analysis. In order to calculate the weighted coefficients of each row, a Least Squares estimate is applied. This is shown

for only terms to predict $\Delta P_1$ in **Equations 26-29.** Both **Equations 26** and **27** consider the notation defined in **Equation 16**.

$$\begin{bmatrix} \Delta P_1(t - 2\Delta t) \\ \Delta P_1(t - \Delta t) \\ \Delta P_1(t) \end{bmatrix} = \begin{bmatrix} x_{11}(t - 2\Delta t) & x_{14}(t - 2\Delta t) \\ x_{11}(t - \Delta t) & x_{14}(t - \Delta t) \\ x_{11}(t) & x_{14}(t) \end{bmatrix} \begin{bmatrix} \alpha_{11}(t) \\ \alpha_{14}(t) \end{bmatrix} \tag{26}$$

$$A = \begin{bmatrix} x_{11}(t - 2\Delta t) & x_{14}(t - 2\Delta t) \\ x_{11}(t - \Delta t) & x_{14}(t - \Delta t) \\ x_{11}(t) & x_{14}(t) \end{bmatrix} \tag{27}$$

$$B = \begin{bmatrix} \Delta P_1(t - 2\Delta t) \\ \Delta P_1(t - \Delta t) \\ \Delta P_1(t) \end{bmatrix} \tag{28}$$

$$\begin{bmatrix} \alpha_{11}(t) \\ \alpha_{14}(t) \end{bmatrix} = (A^T A)^{-1} * A^T * B \tag{29}$$

Assume that $N_s$ is the number of significant terms in a particular row. In order to run the EDDJA method, a window containing $N_s+1$ previous measurements is required to implement the matrix for situational awareness and stability assessment. This is where the number of PMUs selected for the model becomes important. The most significant advantage of this method is the rate at which PMU data is obtained. The entire method hinges on the assumption that the EDDJA matrix will remain similar over a short time period, unless a serious system event occurs that causes significant changes in the bus voltage, drastically affects the $Y_{bus}$, or causes bus voltage angles to diverge quickly. These events would include events such as faults, line outages, or large load changes. Other system events like capacitor switching and load changes are standard system operation, so these events should not be flagged unless there is a slow trend toward instability and system collapse. By introducing the effect of samples that were previously measured, it is possible

to introduce error into the model if the window runs over an event. The DDJEA method is only dependent on the previous sample to estimate the next incoming value. The EDDJA method allows an $N_s+1$ size window where the measurements can be obscured by one significant event or quickly changing parameters.

The number of terms in the EDDJA matrix should be reduced to only essential elements for two critical reasons. First, the accuracy of the method depends on how recent the measurement is. If a model was built with 10 terms in a single row, then there would be data as old as 0.37 seconds influencing the weights given to each term for a given row in the EDDJA matrix. This is not an ideal window when applying these weights to the most recent sample. Even if a model may be more accurate when estimating specific segments of time, the error introduced by a critical event causes error over the $N_s+1$ window and this can cause the method to be less effective than the DDJEA matrix. An ideal number appears to be 2-3 significant terms per row, if any of the off-diagonal terms can be statistically linked. Even though error can be introduced over short time periods from an extreme outlier in the data, the overall effect on the accuracy of considering the EDDJA matrix for both simulated and real data is presented in **Table 2.1** and **Table 2.2.** When an extreme outlier is detected, a cross-comparison can be run between the two methods and the DDJEA method is implemented until the $(N_s+1) \times \Delta t$ window clears.

Data was obtained from a Kundur Two-Area System [74] simulation over 90 seconds with a fault simulated for 0.1 seconds and typical load changes. PMU data retrieved via openPDC from a utility was also considered to test whether the method could be applied to a large system. A total of 53 PMUs received complete measurements over

33

the 147.5 second sample, and the accuracy of the method is shown through application to

both data sets in reference to predicting the $\Delta P_i$ term of the next iteration.

Table 2.1. DDJEA Accuracy for Real Power Estimation

| Calculation Source | Percent Error of Real Power ($P_{err}$) for Measurements | |
| --- | --- | --- |
| | *Mean Percent Error ($P_{err}$)* | *Median Percent Error ($P_{err}$)* |
| Kundur Two-Area System Simulation | 0.054% | $9.51 \times 10^{-6}$% |
| Utility PMU Data Measurements | 0.1977% | 0.1003% |

Table 2.2. EDDJA Accuracy for Real Power Estimation

| Calculation Source | Percent Error of Real Power ($P_{err}$) for Measurements | |
| --- | --- | --- |
| | *Mean Percent Error ($P_{err}$)* | *Median Percent Error ($P_{err}$)* |
| Kundur Two-Area System Simulation | 0.0012% | $1.25 \times 10^{-7}$% |
| Utility PMU Data Measurements | 0.0034% | 0.0013% |

**Table 2.3** highlights the magnitude of error reduced from the system by using

EDDJA compared to DDJEA.

Table 2.3. EDDJA Error Reduction Compared to DDJEA

| Calculation Source | Percent Error Reduction of Real Power ($P_{err}$) Measurements | |
| --- | --- | --- |
| | *Mean Percent Error Magnitude Reduction* | *Median Percent Error Magnitude Reduction* |
| Kundur Two-Area System Simulation | 45x | 76x |

| Calculation Source | Percent Error Reduction of Real Power ($P_{err}$) Measurements | |
| --- | --- | --- |
| | *Mean Percent Error Magnitude Reduction* | *Median Percent Error Magnitude Reduction* |
| Utility PMU Data Measurements | 58x | 77x |

It can be observed from **Table 2.1** and **Table 2.2** that both matrices function to accurately predict the change in real power when using the change in bus voltage angle of the previous iteration as input. The performance of the algorithms is proof of their effectiveness in regard to accomplishing a similar function to the decoupled power flow Jacobian. **Table 2.3** illustrates the magnitude of error reduced from the DDJEA model by implementing the EDDJA algorithm. For real power, the EDDJA method reduces a similar amount of error from both the simulation and the industry system in comparison to DDJEA. The error is greater when using either method on the PMU data from industry. This is mostly due to the complexity and amount of change constantly occurring in the real system in comparison to the simulated system. The real power output is kept constant for loads while the reactive power varies to meet system conditions. After the reactive power portion of the EDDJA matrix is derived in Section 2.2, a mathematical proof can be performed to show that the EDDJA matrix can be substituted into the Newton-Raphson method.

## 2.2    Discrete Jacobian Derivation for Reactive Power Estimation

Similar to the derivation for real power, the reactive power potion of the DDJEA matrix requires a substitution to mimic the format of the decoupled power flow Jacobian

for reactive power presented in **Equations 30-31.** Shared notation in 2.2 continues the same

format as 2.1.

$$\Delta Q_i = \sum_{j=1}^{N} |V_i| * \frac{\partial Q_i}{\partial |V_j|} * \left| \frac{\Delta V_j}{V_j} \right| \tag{30}$$

$$|V_i| * \frac{\partial Q_i}{\partial |V_j|} = -|V_i||Y_{ij}V_iV_j| \sin(\theta_{ij} + \delta_j - \delta_i) \tag{31}$$

The term $\Delta V_j$ represents the change in per unit voltage at bus (j). The term $\Delta Q_i$ is

the change in reactive power over one iteration of the Newton Raphson method. Since the

PMU measurements are taken over a very short time period, the derivation can be written

as follows in **Equations 32-37.**

$$Q_{i_{actual}}(t + \Delta t) = Q_{i_{actual}}(t) + \Delta Q_{i_{actual}}(t + \Delta t) \tag{32}$$

$$Q_{i_{actual}}(t + \Delta t) = V_{i1} * I_{i1} * \sin(\delta_{v1i} - \delta_{I1i}) \tag{33}$$

$$Q_{err} = \frac{Q_{actual} - Q_{predicted}}{Q_{actual}} * 100 \tag{34}$$

$$\Delta Q_i (t + \Delta t) = \frac{\Delta Q_i(t)}{|\Delta V_i(t)|} |V_i(t)| * \frac{|\Delta V_i(t+\Delta t)|}{|V_i(t+\Delta t)|} \tag{35}$$

$$\Delta Q_i(t + 1) = \frac{Q_i(t) - Q(t - \Delta t)}{|V_i(t) - V_i(t - \Delta t)|} * |V_i(t)| * \frac{|(V_i(t+\Delta t) - V_i(t))|}{|V_i(t+\Delta t)|} \tag{36}$$

$$\begin{bmatrix} \Delta Q_1(t + \Delta t) \\ \vdots \\ \Delta Q_n(t + \Delta t) \end{bmatrix} = \begin{bmatrix} \frac{\Delta Q_1(t)}{|\Delta V_1(t)|} * |V_1(t)| & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{\Delta Q_n(t)}{|\Delta V_n(t)|} * |V_n(t)| \end{bmatrix} \begin{bmatrix} \left| \frac{\Delta V_1(t+\Delta t)}{V_1(t+\Delta t)} \right| \\ \vdots \\ \left| \frac{\Delta V_n(t+\Delta t)}{V_n(t+\Delta t)} \right| \end{bmatrix} \tag{37}$$

The performance of the EDDJA method is compared to the DDJEA method to

assess how well they both predict the change in reactive power with the change in bus

voltage as an input. Unlike the real power supplied to loads, the reactive power has more

flexibility to balance voltage and system conditions. **Equation 37** displays the reactive

counterpart of the DDJEA expression. Then the off-diagonal terms need to be generated for the EDDJA method. In order to do so, the EDDJA matrix in **Equation 39** should resemble the decoupled Jacobian for reactive power presented in **Equation 38**.

$$
\begin{bmatrix} \Delta Q_1(t+\Delta t) \\ \vdots \\ \Delta Q_N(t+\Delta t) \end{bmatrix} = \begin{bmatrix} \frac{\partial Q_1}{\partial |V_1|}*|V_1| & \frac{\partial Q_1}{\partial |V_2|}*|V_1| & \cdots & \frac{\partial Q_1}{\partial |V_n|}*|V_1| \\ \frac{\partial Q_2}{\partial |V_1|}*|V_2| & \frac{\partial Q_2}{\partial |V_2|}*|V_2| & \cdots & \frac{\partial Q_2}{\partial |V_n|}*|V_2| \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Q_n}{\partial |V_1|}*V_n & \frac{\partial Q_n}{\partial |V_2|}*|V_n| & \cdots & \frac{\partial Q_n}{\partial |V_n|}*|V_n| \end{bmatrix} \begin{bmatrix} \frac{|\Delta V_1(t+\Delta t)|}{|V_1(t+\Delta t)|} \\ \vdots \\ \frac{|\Delta V_n(t+\Delta t)|}{|V_n(t+\Delta t)|} \end{bmatrix}
\tag{38}
$$

$$
\begin{bmatrix} \Delta Q_1(t+\Delta t) \\ \vdots \\ \Delta Q_n(t+\Delta t) \end{bmatrix} =
$$

$$
\begin{bmatrix} \alpha_{11}(t)\frac{\Delta Q_1(t)}{|\Delta V_1(t)|}*|V_1(t)| & \alpha_{12}(t)\frac{\Delta Q_1(t)}{|\Delta V_2(t)|}*|V_1(t)| & \cdots & \alpha_{1n}(t)\frac{\Delta Q_1(t)}{|\Delta V_n(t)|}*|V_1(t)| \\ \alpha_{21}(t)\frac{\Delta Q_2(t)}{|\Delta V_1(t)|}*|V_2(t)| & \alpha_{22}(t)\frac{\Delta Q_2(t)}{|\Delta V_2(t)|}*|V_2(t)| & \cdots & \alpha_{2n}(t)\frac{\Delta Q_2(t)}{|\Delta V_n(t)|}*|V_2(t)| \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n1}(t)\frac{\Delta Q_n(t)}{|\Delta V_1(t)|}*|V_n(t)| & \alpha_{n2}(t)\frac{\Delta Q_n(t)}{|\Delta V_2(t)|}*|V_n(t)| & \cdots & \alpha_{nn}(t)\frac{\Delta Q_n(t)}{|\Delta V_n(t)|}*|V_n(t)| \end{bmatrix} \begin{bmatrix} \frac{|\Delta V_1(t+\Delta t)|}{|V_1(t+\Delta t)|} \\ \vdots \\ \frac{|\Delta V_n(t+\Delta t)|}{|V_n(t+\Delta t)|} \end{bmatrix}
\tag{39}
$$

The method to implement the linear regression was explained in detail in Section 2.1. The general definitions will be covered for consistency. All previous variables are denoted by a Q in order to differentiate the terminology in **Equations 40-45**, with all other notation previously described applying to these equations. The example is shown specifically for building the linear model to establish a sense of connectivity for PMU 1. The $x_{ij}$ and $X_{ij}$ elements are also demonstrated to show how to build a column vector to determine if PMU 2 is statistically significant in determining the reactive power output of PMU 1.

$$X_{12Q_{samples}}(dimension\ 35999\ x\ 1) = \begin{bmatrix} x_{12Q}(t + \Delta t) \\ x_{12Q}(t) \\ x_{12Q}(t - \Delta t) \\ \vdots \\ x_{12Q}(t - 35,997\Delta t) \end{bmatrix} \qquad (40)$$

$$x_{12Q}(t + \Delta t) = \frac{\Delta Q_1(t)}{|\Delta V_2(t)|} * |V_1(t)| * \left|\frac{\Delta V_1(t+\Delta t)}{V_1(t+\Delta t)}\right| \qquad (41)$$

$$\Delta Q_{1_{samples}}(dimensions\ 35999\ x\ 1) = \begin{bmatrix} \Delta Q_{1actual}(t + \Delta t) \\ \Delta Q_{1actual}(t) \\ \Delta Q_{1actual}(t - \Delta t) \\ \vdots \\ \Delta Q_{1actual}(t - 35,997\Delta t) \end{bmatrix} \qquad (42)$$

$$\widehat{\Delta Q_{1samples}} = \widehat{\beta_{10Q}} + \widehat{\beta_{11Q}}X_{11Q} + \widehat{\beta_{12Q}}X_{12Q} + \widehat{\beta_{13Q}}X_{13Q} \ldots \widehat{\beta_{1nQ}} \qquad (43)$$

$$For\ p_{\beta_{ijQ}} > 1 * 10^{-5},\ \beta_{ijQ} = 0 \qquad (44)$$

$$For\ p_{model_{iQ}} > 1 * 10^{-5},\quad Reject\ Linear\ Model \qquad (45)$$

When setting up the final matrix, all significant regression terms are considered in the model. For the sake of using the same example, it is assumed there are 4 PMUs in a system. PMU 1 and PMU 4 make up one pair of statistically correlated variables that is assumed to have some form of connectivity. PMU 2 and PMU 3 are also significant when determining the next system state at each PMU respectively. **Equation 46** demonstrates the final product of the offline operations.

$$\begin{bmatrix} \Delta Q_1(t + \Delta t) \\ \Delta Q_2(t + \Delta t) \\ \Delta Q_3(t + \Delta t) \\ \Delta Q_4(t + \Delta t) \end{bmatrix} =$$

$$\begin{bmatrix} \alpha_{11}(t)\frac{\Delta Q_1(t)}{|\Delta V_1(t)|}|V_1| & 0 & 0 & \alpha_{14}(t)\frac{\Delta Q_1(t)}{|\Delta V_4(t)|}|V_1| \\ 0 & \alpha_{22}(t)\frac{\Delta Q_2(t)}{|\Delta V_2(t)|}|V_2| & \alpha_{23}(t)\frac{\Delta Q_2(t)}{|\Delta V_3(t)|}|V_2| & 0 \\ 0 & \alpha_{32}(t)\frac{\Delta Q_3(t)}{|\Delta V_2(t)|}|V_3| & \alpha_{33}(t)\frac{\Delta Q_3(t)}{|\Delta V_3(t)|}|V_3| & 0 \\ \alpha_{41}(t)\frac{\Delta Q_4(t)}{|\Delta V_1(t)|}|V_4| & 0 & 0 & \alpha_{44}(t)\frac{\Delta Q_4(t)}{|\Delta V_4(t)|}|V_4| \end{bmatrix} \begin{bmatrix} \frac{|\Delta V_1(t+\Delta t)|}{V_1(t+\Delta t)} \\ \vdots \\ \frac{\Delta V_n(t+\Delta t)}{V_n(t+\Delta t)} \end{bmatrix} \quad (46)$$

Least squares analysis is then applied to find the time varying weights over an $N_s+1$ running window for all rows. **Table 2.4** and **Table 2.5** show the mean and median absolute percent error of both the DDJEA and EDDJA methods, respectively, utilizing the same data sets that the real power portion of the EDDJA method was applied to. The effectiveness of EDDJA for removing error from the predicted values is shown in **Table 2.6.**

**Table 2.4. DDJEA Accuracy for Reactive Power Estimation**

| Calculation Source | Percent Error of Predicted and Actual Reactive Power State ($Q_{err}$) | |
| --- | --- | --- |
| | *Mean Percent Error* | *Median Percent Error* |
| Kundur Two-Area System Simulation | 0.1884% | $9.104\times10^{-7}$% |
| Utility PMU Data Measurements | 0.8916% | 0.3342% |

**Table 2.5. EDDJA Accuracy for Reactive Power Estimation**

| Calculation Source | Percent Error of Predicted and Actual Reactive Power State ($Q_{err}$) | |
| --- | --- | --- |
| | *Mean Percent Error* | *Median Percent Error* |
| Kundur Two-Area System Simulation | $3.91 \times 10^{-5}$% | $1.137 \times 10^{-11}$% |
| Utility PMU Data Measurements | 0.1435% | 0.1027% |

**Table 2.6. EDDJA Error Reduction Compared to DDJEA for Reactive Power**

| Calculation Source | Percent Error Reduction of Real Power ($Q_{err}$) Measurements | |
| --- | --- | --- |
| | *Mean Percent Error Magnitude Reduction* | *Median Percent Error Magnitude Reduction* |
| Kundur Two-Area System Simulation | 4,818x | 80,070x |
| Utility PMU Data Measurements | 6.21x | 3.25x |

These results show that the simulation has a much more linear trend than the actual data obtained from industry. Although the EDDJA method was able to reduce a similar amount of error between both the simulated and industry case for real power, the algorithm was not able to reduce the same amount of error from the change in reactive power for industry data as it could for simulation when compared to the DDJEA method. This was expected due to the fluctuating nature of reactive power in a large-scale power system with dynamic load. The EDDJA model was still able to reduce error from the model despite the added complexity of the real PMU data.

Since the formal decoupled power flow Jacobian cannot be derived without system information, a more mathematically tangible way of proving the viability of this method to function similarly to the Jacobian is to replace all non-zero terms of the decoupled power flow Jacobian with EDDJA terms, utilizing the power flow Jacobian for an initial iteration. Then the final convergence of both matrices can be checked. **Figure 2.2** shows the diagram for the system which will be analyzed and all pertinent system information.
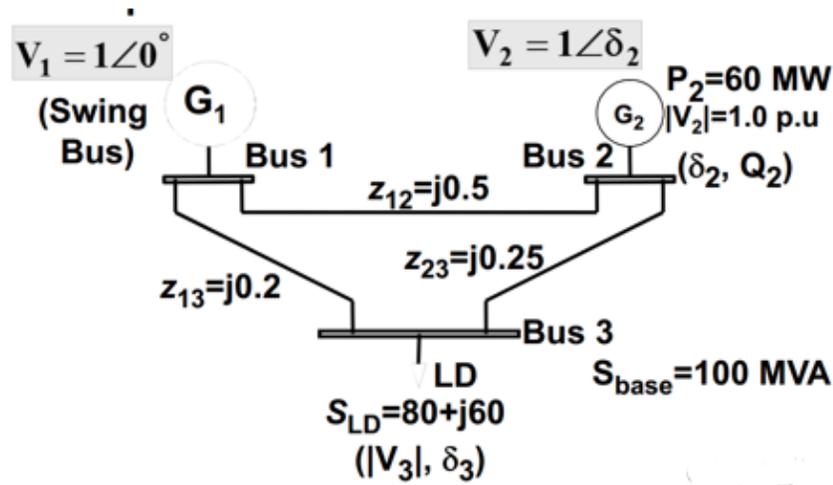


**Figure 2.2. Three Bus Test System**

Since the voltage at bus (2) has a set value, the decoupled Jacobian does not require the $\Delta Q_2$ terms. The Newton Raphson method takes the form shown in **Equation 47**. Bus (1) is set as the swing bus with a constant voltage of $1\angle 0°$.

$$\begin{bmatrix} \Delta P_2 \\ \Delta P_3 \\ \Delta Q_3 \end{bmatrix} = \begin{bmatrix} \dfrac{\partial P_2}{\partial \delta_2} & \dfrac{\partial P_2}{\partial \delta_3} & 0 \\ \dfrac{\partial P_3}{\partial \delta_2} & \dfrac{\partial P_3}{\partial \delta_3} & 0 \\ 0 & 0 & |V_3|\dfrac{\partial Q_3}{\partial |V_3|} \end{bmatrix} \begin{bmatrix} \Delta \delta_2 \\ \Delta \delta_3 \\ \dfrac{|\Delta V_3|}{|V_3|} \end{bmatrix} \tag{47}$$

The EDDJA matrix is substituted into the Newton Raphson algorithm in the format presented in **Equation 48.**

$$
\begin{bmatrix} \Delta P_2^i \\ \Delta P_3^i \\ \Delta Q_3^i \end{bmatrix} = \begin{bmatrix} \alpha_{11}^i \frac{\Delta P_{2\,prediction}^i}{\Delta \delta_2^i} & \alpha_{12}^i \frac{\Delta P_{2\,prediction}^i}{\Delta \delta_3^i} & 0 \\ \alpha_{21}^i \frac{\Delta P_{3\,prediction}^i}{\Delta \delta_2^i} & \alpha_{22}^i \frac{\Delta P_{3\,prediction}^i}{\Delta \delta_3^i} & 0 \\ 0 & 0 & \alpha_{33}^i |V_3^i| \frac{\Delta Q_{3\,prediction}^i}{\Delta |V_3^i|} \end{bmatrix} \begin{bmatrix} \Delta \delta_2^{i+1} \\ \Delta \delta_3^{i+1} \\ \frac{|\Delta V_3^{i+1}|}{|V_3^{i+1}|} \end{bmatrix} \quad (48)
$$

In the context of the substitution, $\Delta P_l$ and $\Delta P_{l\,prediction}^i$ have different definitions. $\Delta P_l^i$ is the difference between the final load/generation value, $P_{desired}$, and the current value, $P_{calculated}^i$ at bus ($l$). The notation for (i) now denotes the iteration value. The (i+1) value would be the next values in the iteration. $\Delta P_{l\,prediction}^i$ is the difference between the current value obtained for real power at bus ($l$) and the value calculated at bus ($l$) during the previous iteration, as indicated in **Equations 49 and 50**.

$$
\Delta P_l^i = P_{desired} - P_{calculated}^i \quad (49)
$$

$$
\Delta P_{l\,prediction}^i = P_{calculated}^i - P_{calculated}^{i-1} \quad (50)
$$

The alphas are calculated using the inverse EDDJA function identically in the inverse EDDJA matrix. Every time a prediction is made, the matrix updates, as does the decoupled power flow Jacobian. All other steps are identical to the Newton-Raphson algorithm, but the EDDJA matrix is simply substituted in place of the Jacobian by applying least squares analysis and directly calculating discrete differences in key values between iterations. The final convergence of both algorithms is displayed in **Table 2.7.**

**Table 2.7. EDDJA Applied in Newton Raphson Method**

| Matrix Used | $V_2$ | $V_3$ | $P_2$ | $P_3$ | $Q_3$ |
|---|---|---|---|---|---|
| Decoupled Jacobian | $1 \angle (3.4680°)$ | $0.9226 \angle (-3.9898°)$ | 0.6000 | -0.8000 | -0.6000 |
| EDDJA Matrix | $1 \angle (3.4680°)$ | $0.9226 \angle (-3.9897°)$ | 0.6000 | -0.8000 | -0.6000 |

This validates that the use of the EDDJA matrix can perform a similar function to the Jacobian if system connectivity and topology are known. The EDDJA matrix is not the decoupled power flow Jacobian, but it can be used in a modified implementation to perform an accurate substitution. Chapter 5 presents an expanded methodology to derive a reduced system model that can leverage PMU data to perform Newton-Raphson load flow analysis.

## 2.3    Implementation of DDJEA and EDDJA for Instability Detection

The traditional power flow Jacobian is used to ascertain the weak points of the system. This is done, as previously discussed in the introduction, by monitoring the Jacobian for singularity. Since these reduced matrices do take into account some fundamental assumptions like accounting for net power and only being applicable over a short period of time, tools must be developed to detect the presence of a singularity before one occurs. It is not practical to wait for conditions indicative of imminent system collapse before flagging the possible event. In a similar fashion to applying Singular Value Decomposition (SVD) to the power flow Jacobian, SVD and eigenvalue analysis can be applied to the matrices to ascertain when a singularity is being approached.

In Newton-Raphson, a singular matrix is unable to be inverted or used to carry out power flow calculations. More practically, this indicates system instability. If the inverse

eigenvalue approaches infinity, then only an infinite change in bus voltage angle would translate to a finite change in power. Vice versa, no possible change in power would be able to change the bus voltage angle.

In order to catch a singularity before one occurs in the EDDJA and DDJEA methods, some mathematical tools were developed. The eigenvalues of both matrices can be calculated over time and statistical analysis can be run to flag extreme outliers during operation. Assuming that the data is Gaussian about the mean for a large sample size, this also allows for calculations relating to the probability that an eigenvalue should be observed, even if it is outside the bounds of normal system operation. This analysis can also be used to give a definitive bound for regular system operations. When a change has occurred over a longer period of time, it is pertinent to flag trends in the data. The expressions below in **Equations 51-62** can help to process the data**.**

$$AI_{DDJEA_i}(t) = \Delta\left(\frac{\Delta P_i(t)}{\Delta\delta_i(t)}\right) = \left|\left(\frac{\Delta P_i(t)}{\Delta\delta_i(t)} - \frac{\Delta P_i(t-\Delta t)}{\Delta\delta_i(t-\Delta t)}\right)\right| - \left|\left(\frac{\Delta P_i(t-\Delta t)}{\Delta\delta_i(t-\Delta t)} - \frac{\Delta P_i(t-2\Delta t)}{\Delta\delta_i(t-2\Delta t)}\right)\right| \quad (51)$$

$$\left|AI_{DDJEA_i}(t)\right| - \left|AI_{DDJEA_i}(t-\Delta t)\right| > 0 \xrightarrow{yields} bus(i)\ DDJEA\ is\ accelerating \quad (52)$$

$$\left(\left|\frac{\Delta P_i(t)}{\Delta\delta_i(t)}\right| - \left|\frac{\Delta P_i(t-\Delta t)}{\Delta\delta_i(t-\Delta t)}\right|\right) > 0 \xrightarrow{yields} bus(i)\ DDJEA\ is\ increasing \quad (53)$$

$$AI_{EDDJA_i}(t) = \left|\left(eig_i(t) - eig_i(t-\Delta t)\right)\right| - \left|\left(eig_i(t-\Delta t) - eig_i(t-\Delta 2t)\right)\right| \quad (54)$$

$$\left|AI_{EDDJA_i}(t)\right| - \left|AI_{EDDJA_i}(t-\Delta t)\right| > 0 \xrightarrow{yields} bus(i)\ EDDJA\ is\ accelerating \quad (55)$$

$$\left(|eig_i(t)| - |eig_i(t-\Delta t)|\right) > 0 \xrightarrow{yields} bus(i)\ EDDJA\ is\ increasing \quad (56)$$

$$\mu_{eig_i} \cong \frac{1}{total}\sum_{m=0}^{total}|eig_i(m*\Delta t)| \quad (57)$$

$$s^2_{eig_i} \cong \frac{1}{total-1}\sum_{m=0}^{total}(|eig_i(m*\Delta t)| - \mu_{eig_i})^2 \quad (58)$$

$$|eig_i(t + \Delta t)| > \mu_{eig_i} + IQR * 1.5 \rightarrow \quad Flag\ Outlier \tag{59}$$

$$|eig_i(t + \Delta t)| < \mu_{eig_i} - IQR * 1.5 \rightarrow \quad Flag\ Outlier \tag{60}$$

$$|eig_i(t + \Delta t)| > \mu_{eig_i} + 2.5\sqrt{s^2_{eig_i}} \quad \rightarrow \quad Flag\ Anomoly \tag{61}$$

$$|eig_i(t + \Delta t)| < \mu_{eig_i} - 2.5\sqrt{s^2_{eig_i}} \quad \rightarrow \quad Flag\ Anomoly \tag{62}$$

Where:

$AI_{DDJEA_i}$ is the acceleration indicator for the DDJEA method at bus (i), and

$AI_{EDDJA_i}$ is the acceleration indicator for the EDDJA method at bus (i).

The variable $eig_i(t + \Delta t)$ is the eigenvalue of the EDDJA method at Bus (i) for the most present measurement. Both of these variables serve a similar purpose. By comparing the eigenvalues of each matrix respectively, a value greater than zero in **Equations 52** and **55** indicates that the eigenvalue at that particular bus is accelerating. **Equations 53** and **56** are used to determine whether the eigenvalues are simultaneously increasing or decreasing in magnitude. The term $\mu_{eig_i}$ represents the mean magnitude that a particular bus eigenvalue takes. Assuming a large sample size from the incoming data, the data will be approximately Gaussian around the mean, allowing probabilities to be calculated even if they have not been observed if the standard deviation is known. The term $s^2_{eig_i}$ is the variance of the eigenvalue at bus(i). This term can also be used to find the standard deviation which is $\sqrt{s^2_{eig_i}}$. IQR is the interquartile range, or the range between the 75th Percentile and 25th percentile of the data. These terms can be used to help determine outliers in the incoming data. When an eigenvalue falls outside of the bounds for

determining an outlier, the value should be monitored and in linear models, any nearly singular should be removed from the Ns+1 window. If the eigenvalue at a specific point in time is not in either of the bounds presented in **Equations 59** and **60**, then the system is running within the bounds of typical operation. When the eigenvalue falls outside of these bounds it should be analyzed for any behavior that is divergent from the mean, especially if the divergence is accompanied by acceleration. When the measurement falls outside of the ranges presented in **Equations 61** and **62** the bus should be closely monitored and flag if there is any further divergence from the mean. Consulting the Z-norm, it is clear that the chances of observing an eigenvalue outside of 2.5 standard deviations from the mean is approximately 1%. If there are multiple measurements that flag an anomaly with a divergent trend, then system instability will eventually occur if conditions do not resolve.

The criteria for flagging a bus eigenvalue can have several conditions once the eigenvalue is outside of the IQR centered on the mean. An eigenvalue increasing and accelerating outside of normal system operations will cause instability to occur quickly if the value is diverging from the mean. This necessitates an immediate solution. Likewise, an eigenvalue accelerating and decreasing outside of the normal system operation range will cause the system to go unstable quickly if the conditions do not improve. Other persistent divergent trends that show a slow trend away from the mean should also be flagged, since these could be indicative of inter-area power oscillations that are slowly going unstable or an unstable load condition. If the eigenvalue is increasing and accelerating toward the mean, then the system is instead returning to equilibrium. This allows system operators to sort convergent positive trends from divergent unstable trends.

Once an unstable eigenvalue is identified, the EDDJA matrix elements can be analyzed to determine which terms are most critically driving the change, as well as whether dynamic or voltage instability will occur if conditions do not change. Dynamic instability trends in the real power portion of the EDDJA matrix and voltage instability trends in the reactive power portion of the EDDJA matrix can be analyzed further. Event identification is addressed in a later chapter. The general process for using DDJEA and EDDJA for instability detection is summarized in **Figure 2.3**.
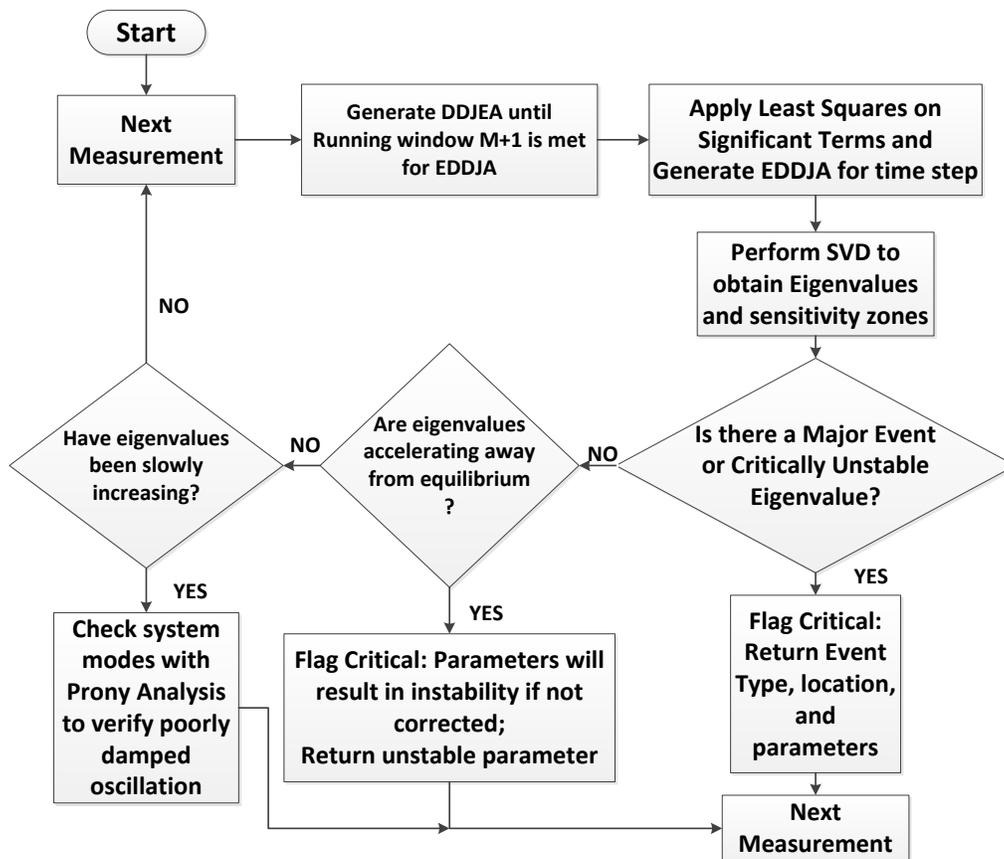


Figure 2.3. Utilizing the EDDJA Algorithm for Instability Detection

This process was applied to a simulated data set for both the EDDJA and DDJEA methods for an unstable case in order to determine how effectively they could relay the

47

urgency once an unstable condition has been introduced. **Figure 2.4** and **Figure 2.5** show the output that can be derived from the methods to determine an unstable eigenvalue and how accurately these methods can relay a need for the operators to act. A fault is placed into the system at the $1500^{th}$ measurement at PMU 4. **Table 2.8** is used to decode the meaning of the values at each measurement. **Figure 2.4** shows the output of the DDJEA algorithm. **Figure 2.5** shows the output of the EDDJA algorithm, highlighting its advantages in identifying the unstable eigenvalue.
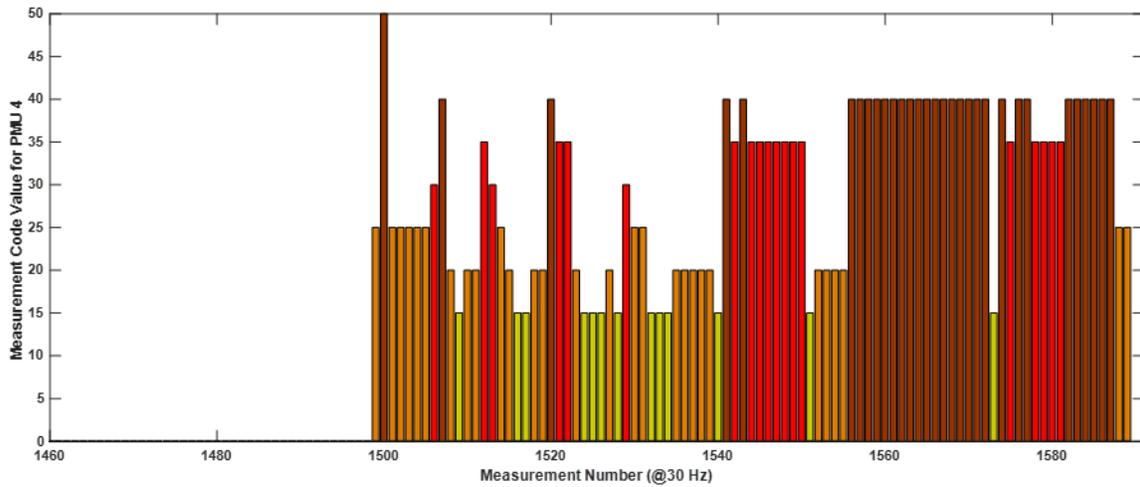


**Figure 2.4. DDJEA Analysis Output for Unstable Case**
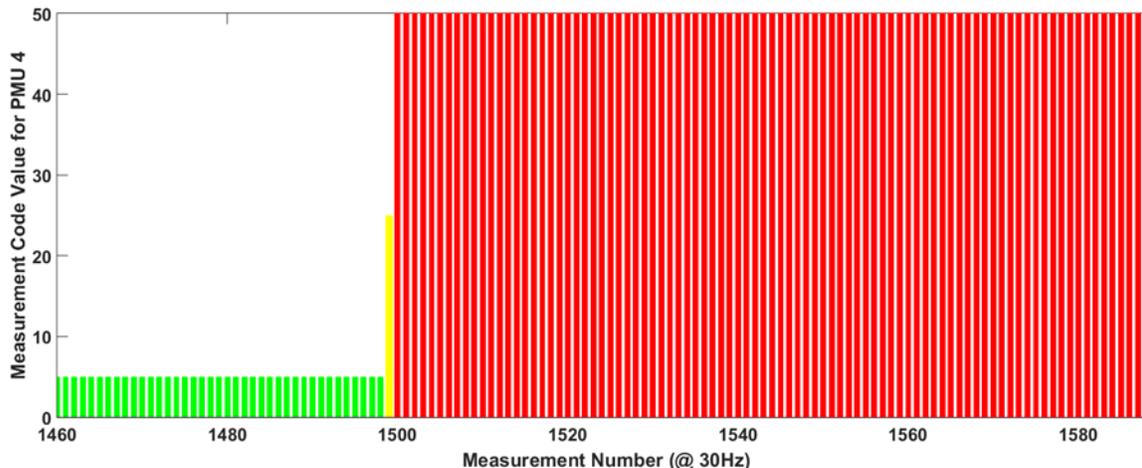
**Figure 2.5. EDDJA Analysis Output for Unstable Case**

**Table 2.8. DDJEA and EDDJA Analysis Code Value Interpretation**

| Code Value | Figure 2.4 and Figure 2.5 Decode Table | |
|---|---|---|
| | *Interpretation of Value* | *Action* |
| 0 | System is at equilibrium range | No action necessary |
| 5 | Bus eigenvalue is converging to a new equilibrium point | No action necessary |
| 10 | Slightly divergent trends detected in Eigenvalue | No action unless this pattern continues |
| 15 | Eigenvalue is converging from unstable parameters | No action unless divergence occurs after event |
| 20 | Bus eigenvalue is marginally converging from unstable parameters | FLAG |
| 25 | Eigenvalue at bus is increasing toward dynamic instability | FLAG CRITICAL |
| 30 | Eigenvalue is accelerating toward dynamic instability | FLAG CRITICAL |
| 35 | Dynamic and voltage instability parameters detected | FLAG CRITICAL |
| 40 | Approaching singularity; System will go unstable soon without solution | FLAG CRITICAL |
| 50 | Major system event and eigenvalue in range of singularity | FLAG HIGHEST PRIORITY |

The EDDJA algorithm more accurately and reliably shows the unstable eigenvalue. Any value greater than 15 can cause instability eventually, so both methods are adequate when identifying the bus causing the unstable condition. The EDDJA method has two advantages. Since the matrix encompasses a sense of connectivity between buses, both the location and cause are more identifiable. Not only will the bus most sensitive to the event be flagged, but the role of other buses and parameters can be used to address the particular issue. Furthermore, the EDDJA matrix yields eigenvalues that more accurately reflect the system state. With the increased number of terms, the cause of the eigenvalue's divergence can be analyzed using more data so that a more effective solution can be reached. Event identification and complimentary methods will be addressed in the next chapters. EDDJA and DDJEA both excel in rapid event detection. The deviation of the eigenvalue from the range it should be able to take assist when identifying an immediate threat. Although the methods do not function as well for their intended purpose of prediction during serious events, near singularities and large deviations in the eigenvalue that occur during such events also function to identify critical threats within one cycle. Slow trends toward instability can be flagged by monitoring which eigenvalues are slowly changing and diverging from any plausible equilibrium point.

Chapter 5 bridges the gap between model-free and model-based applications analyzing the power flow Jacobian. A reduced system model is created, with a mathematical $Y_{Bus}$ and modelled loads. Differences in current between correlated buses are explained with current injections modelled as an equivalent impedance. Ultimately, the power flow Jacobian is classically derived leveraging this mathematical model and variants

of the power flow Jacobian are analyzed to determine system weak points and divergent

parameters, including an actionable process for mitigating unstable parameters.

## CHAPTER THREE

## POWER SYSTEM EVENT DETECTION AND CLASSIFICATION UTILIZING THE BIORTHOGONAL WAVELET

Event detection and classification is fundamental to understanding the cause of system changes. If the events leading up to undesirable or unstable parameters can be articulated, then the appropriate solution will often be much easier for operators to discern and implement. The wavelet transform combines a time-domain and frequency-domain approach by breaking down the signal through an effective low pass and high pass decomposition. The general format of the wavelet transform is described in **Equation 63.**

$$W_\psi(a,d) = \int_{-\infty}^{\infty} x(t)\psi_{a,d}(t)dt \tag{63}$$

$\psi(t)$ is the mother wavelet function, with $\psi_{a,d}$ being made up of orthonormal basis vectors to decompose x(t), the desired signal for analysis. The proposed methodology will utilize the 1-D biorthogonal discrete wavelet transform. This will ultimately result in a high and low frequency component decomposition. The general equations for deriving the discrete wavelet transform coefficients are shown below in **Equations 64-68** [75].

$$f(t) = \sum_{n=-\infty}^{\infty} a_0[n]\phi(t-n) \tag{64}$$

$$a_{j+1}[n] = \sum_{k=-\infty}^{\infty} h[k-2n]a_j[k] \tag{65}$$

$$d_{j+1}[n] = \sum_{k=-\infty}^{\infty} g[k-2n]a_j[k] \tag{66}$$

$$\phi(t) = \sqrt{2} * \sum_{n=-\infty}^{\infty} h[n]\phi(2t-n) \tag{67}$$

$$\psi(t) = \sqrt{2} \sum_{n=-\infty}^{\infty} g[n]\phi(2t-n) \tag{68}$$

In **Equation 64**, $a_0[n]$ is the discrete sampled signal of f(t). The term $\phi(t-n)$ is an orthonormal vector to the mother wavelet, $\psi(t)$. **Equation 65** shows the formula for

the approximation coefficients, and **Equation 66** shows the formula for the detail coefficients. The function $h[n]$ is a low pass filter and $g[n]$ denotes a high pass filter.

The identification and classification of events during system operation is vital when investigating the cause of undesirable system parameters to facilitate an appropriate solution. The objective was to detect and identify faults, load switching, capacitor switching, and breaker operation. Simultaneous events, or events in close proximity, were not considered. Additionally, only the subset of in-scope events is incorporated into this derivation to demonstrate the process. Wavelet transforms have been applied to transient and PMU data for accurate detection of a variety of system events, including HIF. It should be noted that HIFs in simulation are non-linear, require complex models to simulate, and are only defined for distribution systems, excluding them from the scope of this transmission application. Regardless, identification of HIFs demonstrated that the wavelet transform should be sufficient to distinguish more pronounced system events. Alternative solutions leveraging machine learning or neural networks require intensive training before a model can be built, and these models may not work on another system. This section seeks to build an intuitive and repeatable approach that can relate the frequency and time domain characteristics of the PMU signals and overlap power system fundamentals for reliable event detection and identification. In order to facilitate this objective, a set of measured and derived signals from each PMU were analyzed after biorthogonal wavelet transform decomposition. The "bior3.5" wavelet transform from the Matlab wavelet toolbox was used for the decompositions [76]. The Kundur Two-Area system [74], shown in **Figure 3.1**, was used for the following test case, with 4 PMUs placed in the system near the

generators. The model is built using Power Systems Simulator in Matlab and has inputs for various power system stabilizer models [76]. This model is also ideal for monitoring power oscillations.
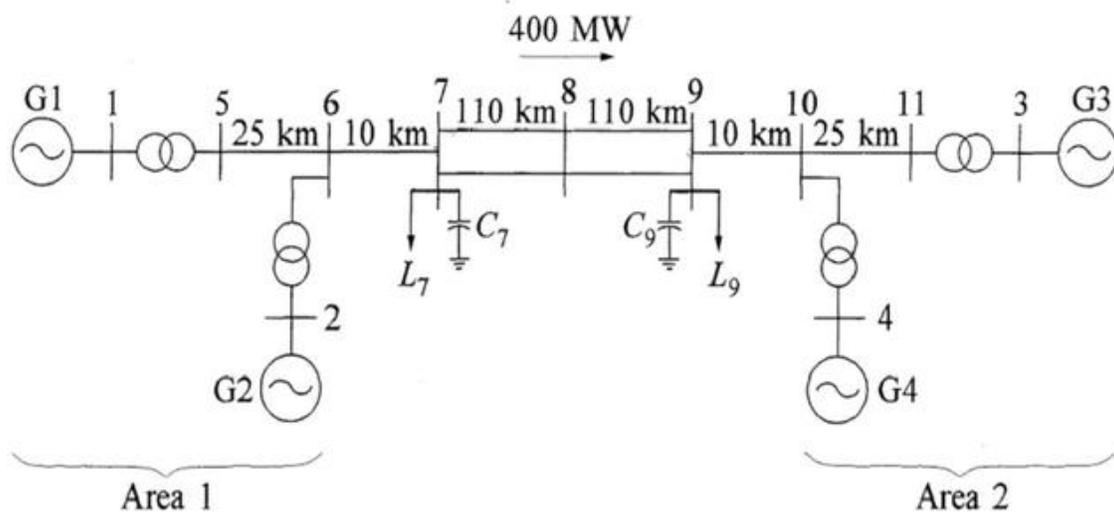


**Figure 3.1. Kundur Two-Area System Diagram [74]**

The biorthogonal wavelet transform was applied to signals for real power, reactive power, bus voltage magnitude, discrete derivative of bus voltage angle, discrete derivative of current, discrete derivative of current angle, and frequency. The high pass wavelet coefficients, from **Equation 66**, were analyzed over a moving window in order to gain insight into events from the PMU data directly. In each analyzed window, the values returned are the local peak wavelet coefficients, recorded only when a threshold value has been crossed, indicating an event. For normal system operation, the peak wavelet coefficient is approximately zero. Once an event has been detected, the peak value is held for a few cycles until the peak wavelet coefficient returns to approximately zero and drops below the threshold. The wavelet coefficients are influenced by both the size of the data

54

window used for computation and the magnitudes of the p.u. values for the current and voltage. Larger loads will change the current profile of the system and p.u. bus voltages significantly. A safe threshold for the detail coefficient relating to p.u. real power was determined to be 0.001, where any high pass coefficient value less than this threshold is disregarded as noise. Any high pass coefficient value over this threshold is assumed to be an event and is processed for classification. Although this bound is sufficient as an initial guess, the threshold should be tested considering the target system data and desired window length to calculate the standard deviation of the wavelet coefficients from 0. The general methodology was initially published in [70] as part of this research.

The application of this methodology to detect and classify events requires calculating and analyzing the peak wavelet coefficients of each signal of interest during a system event. These coefficients, listed below, are analyzed in order to positively identify the event.

| | |
|---|---|
| $\alpha_P$ | The peak wavelet coefficient associated with the real power signal |
| $\alpha_Q$ | The peak wavelet coefficient associated with the reactive power signal |
| $\alpha_V$ | The peak wavelet coefficient associated with the positive sequence voltage magnitude signal |
| $\alpha_I$ | The peak wavelet coefficient associated with the current magnitude |
| $\alpha_F$ | The peak wavelet coefficient associated with the frequency signal |
| $\alpha_{V\Delta\Theta}$ | The peak wavelet coefficient associated with the discrete derivative of bus voltage phase angle |
| $\alpha_{I\Delta\Theta}$ | The peak wavelet coefficient associated with the discrete derivative of current phase angle |
| $\alpha_{eventmin}$ | This is the threshold for the real power wavelet coefficient used to determine an event, many orders above system noise |
| $\alpha_{capcutoff}$ | Lower wavelet bound for typical system capacitors with respect to $\alpha_P$ |

$$\alpha_{P/Q} = \frac{\alpha_P}{\alpha_Q} \tag{69}$$

| | |
|---|---|
| $\alpha_{P/Q}$ | Gives a margin to easily detect line removals; All other events fall near 1 |
| $\alpha_{FCL}$ | Calculated by creating a gaussian distribution over a range of data to ascertain a value of $\alpha_{P/Q}$ not within the range of other events (Typically greater than 10 but often over 1000) |

$$\alpha_{V\Delta\Theta N} = \frac{\alpha_{V\Delta\Theta}}{0.001} \rightarrow This\ is\ just\ an\ applied\ weight\ for\ some\ logical\ tests \tag{70}$$

$$\alpha_{I\Delta\Theta N} = \frac{\alpha_{I\Delta\Theta}}{0.001} \tag{71}$$

The term $\alpha_{LS}$ is used to confirm a load switch event. In instances where a load switch cannot be confirmed, the event will be considered undefined with respect to the reference PMU and the decision will be deferred, if applicable, to analysis of other PMU devices.

$$\alpha_{LS} = \frac{\alpha_{V\Delta\Theta}}{\alpha_{I\Delta\Theta}} \tag{72}$$

**Figure 3.2** shows the general algorithm for the method of applying the biorthogonal wavelet transform for event detection and identification.

This model does have some considerations to be reliable. In order to get an accurate result, the window used for analysis must be large, which could exclude the method from being applied to larger power systems due to constraints on real-time convergence, although particular signals of interest can be supported for real-time processing and overlapping windows can be configured to consume all incoming PMU data. There are an infinite number of analog values that loads, capacitors, and fault impedances can assume. The intention of this method is to provide reliable classification most of the time for significant events of interest. Additional considerations can always be made to accommodate an important edge case that occasionally is misclassified.

**Start**

Calculate wavelet coefficients over a running window

Determine peak wavelet coefficients $\alpha_P$, $\alpha_Q$, $\alpha_V$, $\alpha_A$, $\alpha_{V\Delta\theta}$, and $\alpha_{I\Delta\theta}$

$\alpha_P > \alpha_{eventmin}$ — NO → No Event

YES

$\alpha_{P/Q} > 2\,\alpha_{FCL}$ — YES → Line Removal

NO

$\alpha_I * \alpha_{V\Delta\theta N}$ && $\alpha_I * \alpha_{I\Delta\theta N}$ $> 10* \alpha_{capcutoff}$ — YES → Fault

NO

$\alpha_I * \alpha_{V\Delta\theta N}$ && $\alpha_I * \alpha_{I\Delta\theta N}$ $> .5*\alpha_{capcutoff}$ — YES → Capacitor Switch

NO

$\alpha_{LS} > 10$ — YES → Load Switch

NO

Not Definitive/Undefined Use other PMU Signal
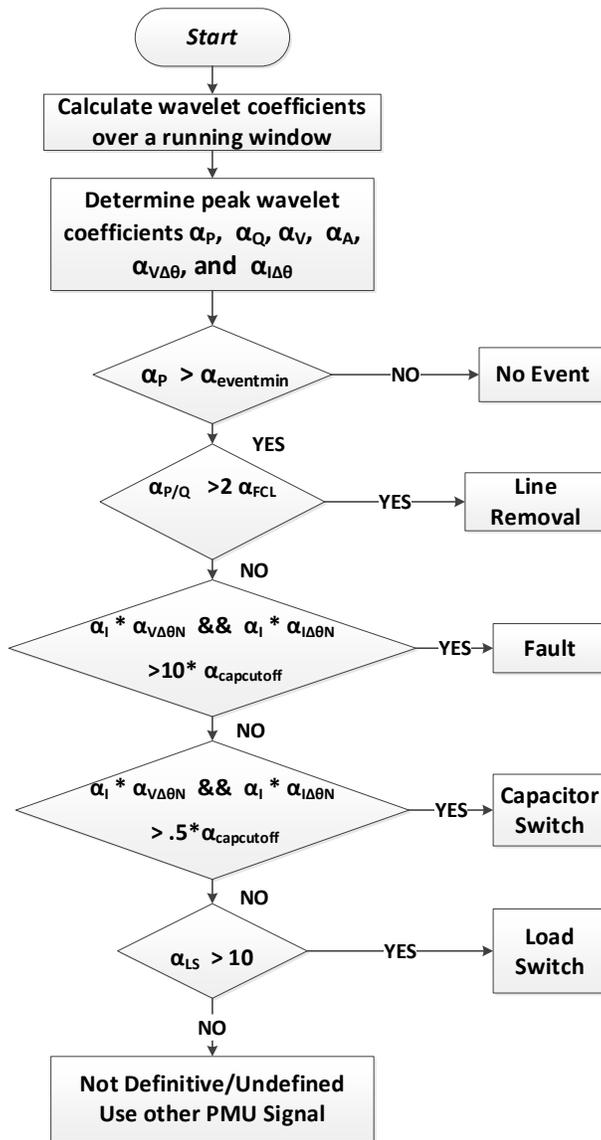
**Figure 3.2. Biorthogonal Wavelet Event Identification and Classification**

The methodology shown in **Figure 3.2** does not take into account simultaneous, or nearly simultaneous, events, such as dropping a load and then immediately switching on a capacitor or opening line breakers to clear an active fault. The classification derived from this methodology will likely favor the event with a greater system impact. The output of the above method has also been compared to the error experienced by applying the DDJEA

and EDDJA matrices to estimate the change in real power, as prediction errors typically occur during major events. Some derived wavelet coefficient variables consider time-domain power system analysis in their construction. By considering time-domain power system analysis in parallel, a smaller window can be used for calculating the wavelet coefficients. This leads to decreased computation time to execute the wavelet decomposition.

When analyzing the different event types, low impedance faults stand out from other considered events. For faults, the peak values of the high pass wavelet coefficient relating to both real and reactive power are significantly larger than the other system events being considered. Typical faults, with resistance near zero, also have the most significant impact on the peak value of the frequency wavelet coefficient; faults have a profoundly negative effect on system frequency when compared to other events being considered. Time-domain analysis can be accurately applied in the same cycle that a possible fault is detected since bus voltage will drop significantly below the previous measurement and real power will generally experience a large change. Since faults may be cleared by protection or dissipate within a short timeframe, a conservative threshold of approximately 0.1 p.u. voltage drop between two measurements is reasonable to consider flagging a fault. This is due to the possibility that for a 30Hz signal, a fault could fall at the beginning of a cycle, clearing before the measurements reflect the full impact on voltage. The zero-sequence current can also be calculated from the line currents at a bus, indicating the presence of a ground fault. Time domain analysis can be sufficient, when coupled with the wavelet decomposition, to decrease the minimum window size without forfeiting reliability.

A differentiation that can be difficult to make when analyzing the PMU data strictly through the wavelet transform is when a capacitor is switched on or a load is switched off. Unless the PMU is directly on the load bus, the trend tends to be that the real power and voltage both increase the first few cycles after the load is removed from the circuit. Furthermore, the reactive power, as measured at generators, will often reflect an immediate decrease. These three qualities are more typically associated with a capacitor switching on. Finding some topology differences in the wavelets was crucial. From the perspective of the power system, capacitors tend to have a more significant signature with respect to the reactive power than real power. Similarly, loads will tend to have a more noticeable impact on the real power measured at a bus. Independent of load, a term was defined to help distinguish the capacitor switching event from a load event.

$$\alpha_{HC} = \frac{\alpha_I \alpha_F}{\alpha_V^2 * \alpha_{V\Delta\Theta}^2} \tag{73}$$

With this variable in place, a reference is drawn for distinguishing between switching a capacitor on and load removal, or vice versa. There is also a distinction that has to be made between a large load increase and a load increase due to normal operation, as one has significantly more impact on the system. For a low impedance fault, the wavelet coefficients for the real and reactive power wavelet decomposition will have opposite slopes in reference to the most recent coefficient. The same trend applies to a normal system load. When a load substantially increases, both the reactive and real power wavelet coefficients will have aligned slopes immediately after being introduced.

$$Given\ Event:\ \alpha_P(n+1) > \alpha_P(n)\ \&\ \alpha_Q(n+1) > \alpha_Q(n) \rightarrow Large\ Load \tag{74}$$

$$\alpha_P(n+1) < \alpha_P(n)\ \&\ \alpha_Q(n+1) < \alpha_Q(n) \rightarrow Large\ Load$$

This seems counterintuitive, but from the measurements of PMUs that are not directly on the loaded bus, the general trend in the system is that there will be a net real power increase in the system and a decrease in bus voltages. Due to accounting for the load, there will be an increase in reactive power for the system. The real power and reactive power high pass wavelet coefficients can be seen in **Figure 3.3** for visualization of detail coefficients during a fault event.
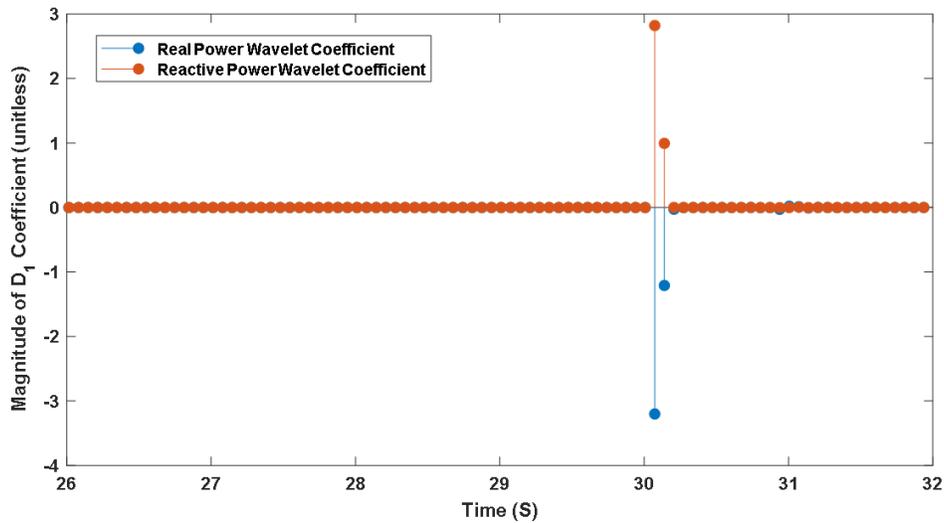


**Figure 3.3. Real and Reactive Wavelet Coefficients during Fault Event**

In the case of a heavy load, both the $\alpha_{HC}$ and $\alpha_{LS}$ term can be used to help positively identify a load switch event. In cases where the real and reactive power detail wavelet coefficients are not linearly related in their trajectory post-event, it would suffice to analyze the system's real power and bus voltage differences from the prior measurement, since a fault will have a more abrupt signature than a large load event. A standard increase in load is easily distinguished from a fault since the peak real and reactive power wavelets are well below the threshold for faults or capacitor switching.

The EDDJA and DDJEA methods also can be used adjacently to indicate when a major event has occurred, or the system conditions have changed significantly from the assumptions that were made to build the matrices. The primary assumption of both matrices is that the true power flow Jacobian should not change drastically between two cycles unless a system event has occurred, driving the solution away from the previous equilibrium point. This error should be more pronounced in the DDJEA matrix, as a corrective step to increase accuracy is not considered in its construction. This error is an indication that the model is unable to approximate the behavior of the system, generally coinciding with a fundamental change or event. **Figure 3.4** shows the relation of errors in the DDJEA eigenvalues to simulated system events.
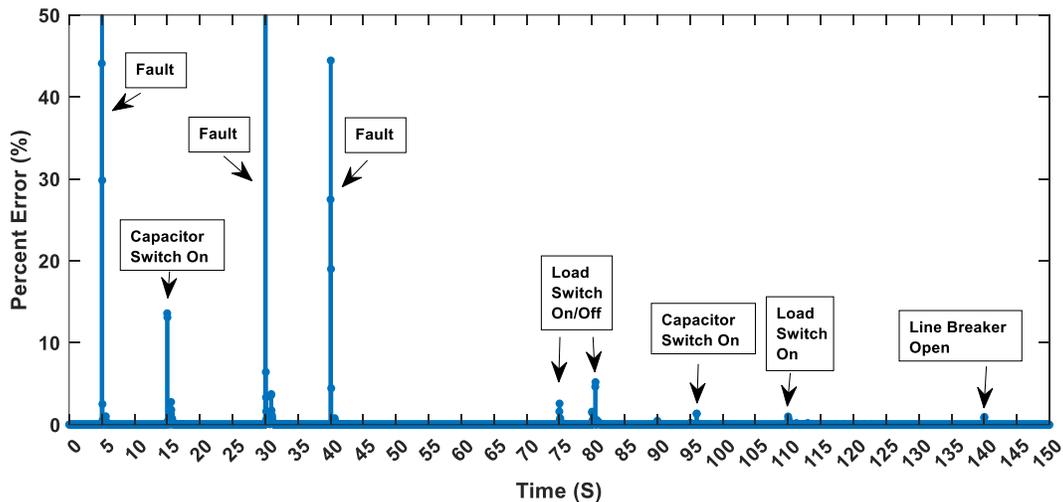


Figure 3.4. DDJEA Estimation Error as an Indicator of System Event

The highest errors over this sample period are due to faults in the system. Choosing to flag at any value greater than or equal to 5% error, every low impedance faulted case and several of the switching operations would be detected as a system event. There were

no false flags on this test case. System events that the model was able to adequately compensate for or that cause a negligible disturbance in the DDJEA estimation would not be flagged. The magnitude of the real power detail coefficient for a utility dataset was also assessed for the signal shown in **Figure 3.5**. The threshold of 0.001, previously stated for the high pass real power wavelet coefficient, is sufficient to avoid unnecessary processing, as the maximum absolute value of $\alpha_P$ for the utility real power signal, where no events were present, was 0.000197. As this value is below the threshold, no event will be detected over the observable window and no further analysis will be triggered.



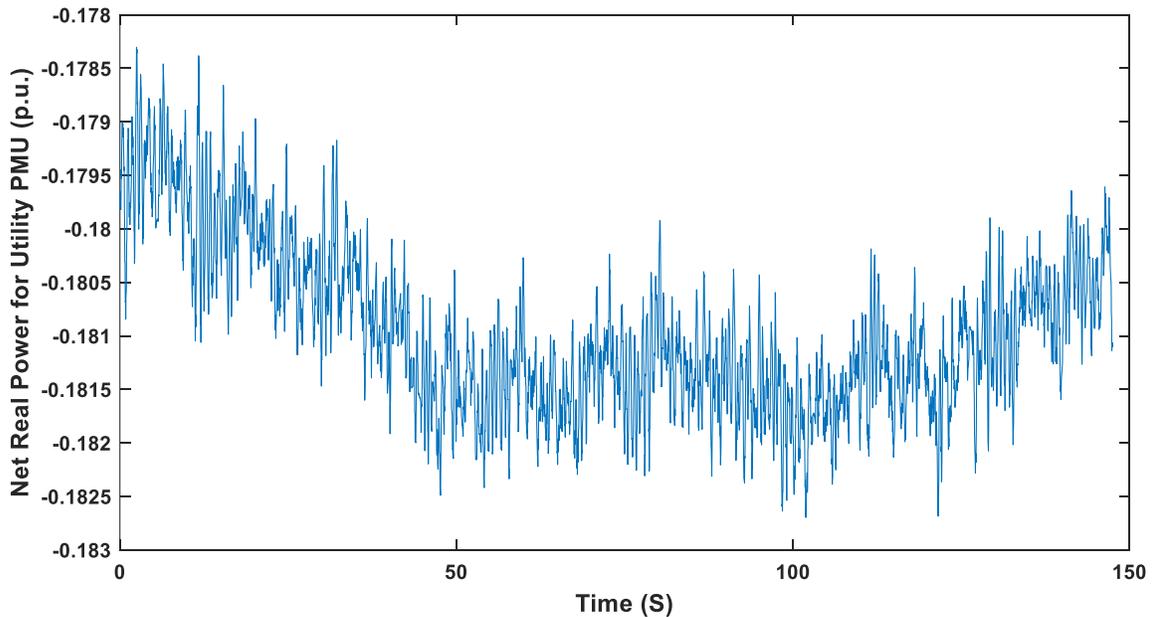**Figure 3.5. Reference Utility PMU Power Signal Processed**

## 3.1    Wavelet Event Classification Utilizing a Reduced Observability Window

Considering an i9-12900HK 2.5GHz processor and no dedicated optimization, the computational time to implement biorthogonal wavelet event classification for the previous algorithm is 0.0028 seconds for simulated data containing events and 0.0016 seconds for

utility data containing no events. The computational times are calculated in Matlab code for a single PMU. It is clear that the presence of events increases computational time, so 0.0028 seconds is used as the estimate. A modified methodology for event classification utilizing a shorter window of observability, 5 seconds with 150 data set reports, is assessed to decrease the computational burden and increase the methodology's usefulness for near real-time event classification. It should be noted that the computational time to process all signals from an individual PMU is reduced to 0.0008 seconds using this shorter time window for analysis. Since a smaller window is considered, a practical approach to analyzing the PMU signals with respect to the power system physics is implemented in parallel to assist in reliable event classification. Given the shorter window, a delay is introduced to monitor system behavior for more reliable identification during some events, such as distinguishing a load switch from a capacitor switch. The modified approach is displayed in **Figure 3.6** for use of the smaller time window. Some of the new variables are defined below (**Equations 75** and **76**).
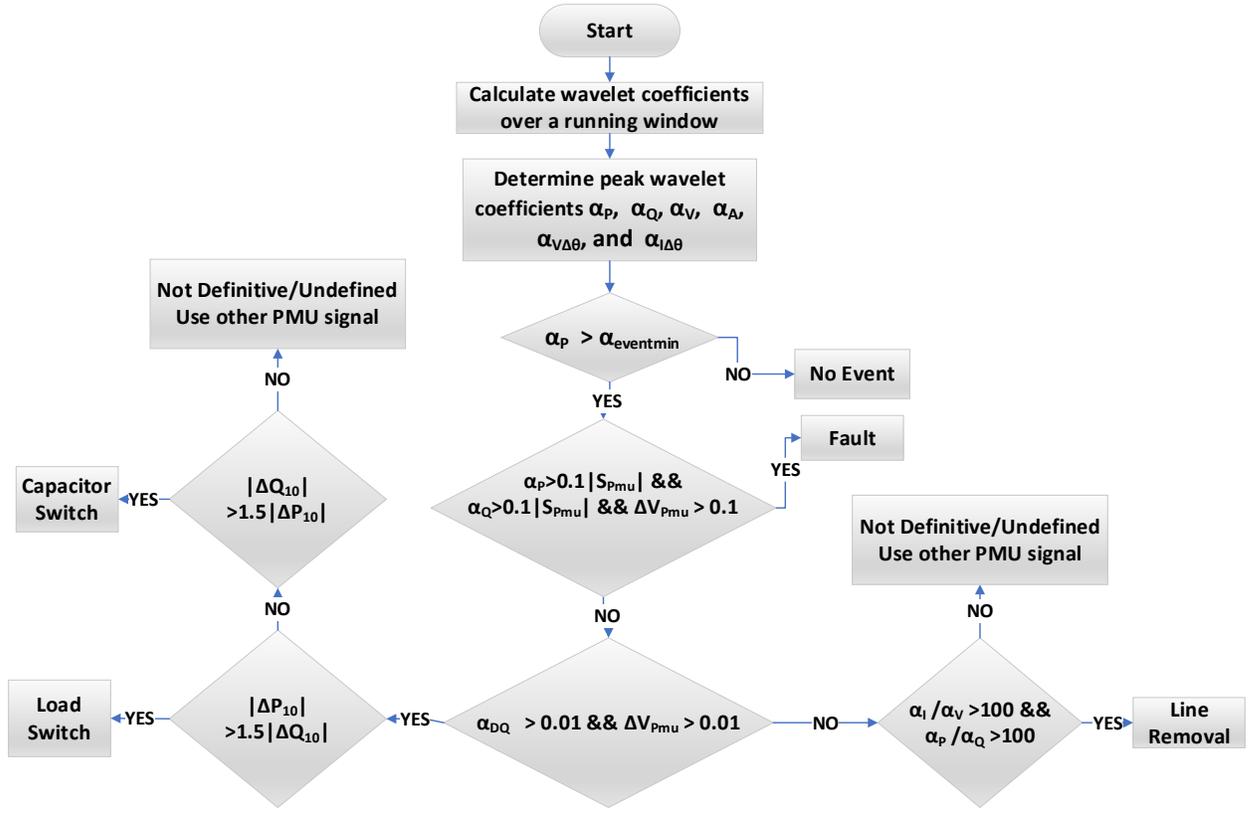
**Figure 3.6. Modified Methodology for Event Classification with Reduced Wavelet Window**

$$|\Delta P_{10}| = |P(t_{event}) - P(t_{event} + 10\Delta t)| \qquad (75)$$

$$|\Delta Q_{10}| = |Q(t_{event}) - Q(t_{event} + 10\Delta t)| \qquad (76)$$

**Equations 75** and **76** derive two useful variables for determining whether a capacitor switch or load switch has occurred. For the time-step that an event is identified, $t_{event}$, the per unit real and reactive power at each PMU is recorded, $P(t_{event})$ and $Q(t_{event})$. If the final decision is between a load or capacitor event, the real and reactive power are sampled 10 measurements later to determine the net change in the real and reactive power values, $\Delta P_{10}$ and $\Delta Q_{10}$. The reason this is reliable is that load events will typically have a

notably larger impact on net real power in the system after transients resolve. Similarly, capacitive events will typically have a notably larger impact to net reactive power, regardless of where the signal is measured from. $S_{Pmu}$ is used to denote the mean p.u. complex power, considering 10 measurements prior to an event. $\Delta V_{Pmu}$ denotes the absolute per unit change in voltage between the sample where an event is detected and the prior measurement. The variable $\alpha_{DQ}$ denotes the peak wavelet coefficient associated with the discrete derivative of the reactive power signal between two measurements. This variable was utilized with the shorter window to assist with sorting events that have less pronounced impact between measurements on the reactive power signal.

Event classification is essential for situational awareness, but there are some events where a short delay to confirm identification is acceptable. Faults are typically cleared by protection devices in a sub-second timeframe. The full and reduced wavelet event detection and classification methodologies discussed in this chapter converge in real-time for fault classification. When applicable, all other event classifications converge in an acceptable timeframe to log an event for later reference by system operators. With respect to wide deployment, utilizing a reduced window length has computational advantages. For either case, wide deployment in a large system would require architecture considerations to support near real-time event detection and classification, but implementation of either methodology is sufficient and accurate for its stated purpose.

# CHAPTER 4

## APPLICATIONS TO DETECT POORLY DAMPED POWER SYSTEM OSCILLATIONS

### 4.1 Objective Overview

As previously referenced, inter-area and electromechanical oscillations tend to have a frequency in the 0.1-0.8Hz range. Oscillations between 0.1-2 Hz tend to be caused by heavy loading conditions, inter-area load flows, or faults as consistent power to loads is generated. Several methods can be employed to estimate the dominant eigenvalues of a signal. From the eigenvalues, the damping ratio and frequency of the signal at that particular eigenvalue can be obtained. The methodologies of interest model oscillatory modes and damping factors by representing the signal in one of the following forms: **Equations 77-79**.

$$y(k) = \sum_{i=1}^{n} R_i * z_i^k \tag{77}$$

$$y(t) = \sum_{i=1}^{n} R_i * e^{\lambda_i t} \tag{78}$$

$$y(s) = \sum_{i=1}^{n} \frac{R_i}{s - \lambda_i} \quad, where \ s = j\omega \tag{79}$$

In the above equations, $R_i$ is the residue for each individual eigenvalue $\lambda_i$. This term will take a complex value for complex eigenvalues. **Equations 80-82** show further relations to determine where the eigenvalue and damping ratio are ultimately derived. This term also gives a sense of weight to the term. In the case of Prony Analysis, the model order is selected without being known between iterations so a reasonable guess to enclose all important modes is selected and hardcoded as a predetermined system order. For Prony Analysis, it is ideal to over-determine the number of modes and discard those eigenvalues

with low residue terms. Other methods of interest do not share this drawback of Prony Analysis.

$$z_i = e^{\lambda_i * \Delta t} \tag{80}$$

$$\lambda_i = \frac{\ln(z_i)}{\Delta t} \tag{81}$$

$$\lambda_i = \sigma_i + j\omega_i \tag{82}$$

The time step between samples is denoted as $\Delta t$. The eigenvalues can be determined once $z_i$ is found by taking the natural log and dividing by the time step. The sampling frequency, $f_s$, is 30Hz for PMUs, making the time step 1/30 for individual measurements. Each model proposed will use a window of 300 measurements so that the slowest system modes of interest, approximately around 0.1 Hz, can be adequately modelled by the data. Since the Nyquist frequency must be considered, these methods will be able to return oscillatory data for the 0-15Hz range, with a more conservative lower cutoff of 0.1Hz when identifying the mode characteristics. The methods analyzed in this paper are the Matrix Pencil Method (MPM) and Prony Analysis. These methods have a different way of expressing the data set in order to approximate the system modes at a given bus. Since system topology is not considered, regional monitoring will be done by flagging the monitored regions for poorly damped oscillatory modes while checking the frequency at which these modes occur. Damping ratios of slow system modes under 3% must be identified and flagged in parallel with more dynamic conditions.

The term $\sigma_i$ is the real portion of the estimated eigenvalue $\lambda_i$. Similarly, the term $\omega_i$ is the imaginary component of the complex eigenvalue. In cases where there is no oscillatory mode, $\omega_i$ will be zero. These cases are not a part of the study unless they are

positive, indicating instability, or approaching zero while having a significant residue value. The calculation of the damping ratio and associated frequency are shown in **Equation 83** and **Equation 84** respectively.

$$DR = -\frac{\sigma_i}{\sqrt{\sigma_i^2 + \omega_i^2}} \qquad (83)$$

$$f_i = \frac{\omega_i}{2\pi} \qquad (84)$$

There are various methods to attain the exponentials. Each method has a different way to solve for the valid parameters and reduce noisy modes or invalid parameters from the model. Prony analysis and MPM will be discussed in detail below. The computational expense and advantages of each method will also be discussed.

## 4.2 Prony Analysis Method

Prony Analysis is one of the most well-known techniques to approximate power system modes. A significant drawback of Prony Analysis is that the number of system modes is not known before implementation. Consider in the following equations that (n) is the number of selected system modes and (N) is the total number of measurements over a moving window. **Equation 85** shows the format of the signal of interest, the real power at a bus. **Equation 86** shows in intermediate step to solving for the poles during a sample iteration.

$$signal\ of\ interest \rightarrow y = Power(most\ recent\ 300\ measurements)_{1x300} \quad (85)$$

$$
\begin{bmatrix} y(n) \\ y(n+1) \\ y(n+2) \\ \vdots \\ y(N-1) \end{bmatrix} =
\begin{bmatrix}
y(n-1) & y(n-2) & y(n-3) & \dots & y(0) \\
y(n-2) & y(n-3) & y(n-4) & \dots & y(1) \\
y(n-3) & y(n-4) & y(n-5) & \dots & y(2) \\
\vdots & \vdots & \vdots & \dots & \vdots \\
y(N-2) & y(N-3) & y(N-4) & \dots & y(N-n-1)
\end{bmatrix}
\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_n \end{bmatrix} \quad (86)
$$

In order to attain the discrete eigenvalues, the matrix containing $\alpha$ terms needs to be solved. The pseudoinverse is an ideal function for this since the matrix is not square and using least squares did not yield consistent results. Once this is done, the alpha values can be inserted into **Equation 87** to solve for the discrete eigenvalues, $z_1$ through $z_n$.

$$z^n - (\alpha_1 z^{n-1} + \alpha_2 z^{n-2} + \alpha_3 z^{n-3} + \alpha_4 z^{n-4} \ldots + \alpha_{n-1} z^1 + a^n) = 0 \qquad (87)$$

The residues, $R_i$, can be calculated for the Prony analysis method by applying **Equation 88.** Consider $z_1$ as the eigenvalue associated with the first row and $z_n$ as the eigenvalue of the $n^{\text{th}}$ row.

$$\begin{bmatrix} y(n) \\ y(n+1) \\ y(n+2) \\ \vdots \\ y(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \\ z_1 & z_2 & z_3 & \ldots & z_n \\ z_1^2 & z_2^2 & z_3^2 & \ldots & z_n^2 \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ z_1^{N-1} & z_2^{N-1} & z_3^{N-1} & \ldots & z_n^{N-1} \end{bmatrix} \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ \vdots \\ R_n \end{bmatrix} \qquad (88)$$

The pseudo-inverse can be implemented, and the residue values can be solved. Extremely low residue values indicate that the term is modelling noise in the system, and residue values with a very low value can be removed from the model. Prony Analysis is not as robust when predicting the correct system eigenvalue and damping ratio with a significant amount of noise present. Unlike MPM, discussed further in *Section 4.3*, there is no consideration in the classic Prony Analysis application for separating the data into groups containing noise and those terms without noise. This means that the calculation for the terms containing noise occurs before these terms can be identified through residues as insignificant. Prony Analysis is computationally expensive when compared to other methods used to calculate the damping ratio; however, if the number of oscillatory modes to monitor is significantly high, it does leave the user with more modes to analyze. Once

the discrete eigenvalues are found, **Equation 81** can be applied to solve for the continuous time eigenvalues $\lambda_i$. For each iteration, the peak residue is identified. Then the residue of each individual mode is compared to the peak residue by a ratio.

$$\frac{|R_i|}{|R_{peak}|} > 10^{-4} \tag{89}$$

**Equation 89** presents a ratio that was used in the process of refining and selecting valid modes so that only those contributing to the model with some level of significance are added to the overall model. This was a methodology applied to this implementation for this data which was effective. It is not a defined standard, but practice has shown that throwing out any values below this margin will not yield a significant error. The main points to consider when evaluating this method for real-time implementation are accuracy and computational speed. The model order for Prony Analysis was set to 0.4×N. This selection agrees with the pencil length of N/3 to N/2 for the MPM [77]. Unlike MPM all modes will be calculated. Even though the window only spans 300 measurements, calculating the eigenvalues for a 120-order model becomes tedious. This limits the effectiveness of Prony Analysis as an online application. Resiliency to noise is another problem that must be addressed since Prony Analysis is at a disadvantage to methods that reduce system order significantly before calculating the terms not generated by noise.

## 4.3    Matrix Pencil Method

MPM has some notable advantages over Prony Analysis. First and foremost, MPM is faster. The pencil parameter, L, is set to some value between N/3 to N/2 in order to effectively approximate the minimum number of eigenvalues and eliminate noise from the model. For Prony analysis, the L parameter was technically used, but there was no

reduction in system order. The filtering process is a biproduct of an effective L pencil

parameter and using singular value decomposition (SVD), which will be seen below. The

overall matrix is shown in **Equation 90,** which is the matrix of inputs containing noise.

$$Y = \begin{bmatrix} y(0) & y(1) & y(2) & ... & y(L) \\ y(1) & y(2) & y(3) & ... & y(L+1) \\ y(2) & y(3) & y(4) & ... & y(L+2) \\ \vdots & \vdots & \vdots & ... & \vdots \\ y(N-L-1) & y(N-L) & y(N-L+1) & ... & y(N-1) \end{bmatrix} \tag{90}$$

The Y matrix has a size of (N-L) x (L+1). The end goal is to separate noise from

the rest of the input. The theory behind this method is presented in **Equations 91-99**. The

data is desired to be broken down into noise free matrices with all x(k) terms being noise

free.

$$Y_1 = \begin{bmatrix} x(0) & x(1) & x(2) & ... & x(L-1) \\ x(1) & x(2) & x(3) & ... & x(L) \\ x(2) & x(3) & x(4) & ... & x(L+1) \\ \vdots & \vdots & \vdots & ... & \vdots \\ x(N-L-1) & x(N-L) & x(N-L+1) & ... & x(N-2) \end{bmatrix} \tag{91}$$

$$Y_2 = \begin{bmatrix} x(1) & x(2) & x(3) & ... & x(L) \\ x(2) & x(3) & x(4) & ... & x(L+1) \\ x(3) & x(4) & x(5) & ... & x(L+2) \\ \vdots & \vdots & \vdots & ... & \vdots \\ x(N-L) & x(N-L+1) & x(N-L+2) & ... & x(N-1) \end{bmatrix} \tag{92}$$

These matrices are both size (N-L)xL and are noiseless partial representations of

the full Y matrix, which includes noise. Both matrices have one less column than the

original input that these matrices are jointly modelling without noise. $Y_1$ is the original Y

matrix with the last row removed. $Y_2$ is simply the original Y matrix with the first row

removed.

$$Y_1 = [Z_1][R][Z_2] \tag{93}$$

$$Y_2 = [Z_1][R][Z_0][Z_2] \tag{94}$$

$$Z_0 = \begin{bmatrix} z_1 & 0 & 0 & \cdots & 0 \\ 0 & z_2 & 0 & \ldots & 0 \\ 0 & 0 & z_3 & \ldots & 0 \\ \ldots & \ldots & \cdots & \ddots & 0 \\ 0 & 0 & 0 & \cdots & z_M \end{bmatrix}, size \ M \ x \ M \tag{95}$$

$$Z_1 = \begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \\ z_1 & z_2 & z_3 & \ldots & z_M \\ z_1^2 & z_2^2 & z_3^2 & \cdots & z_{M^2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ z_1^{N-L-1} & z_2^{N-L-1} & z_3^{N-L-1} & \ldots & z_M^{N-L-1} \end{bmatrix}, size \ (N-L) \ x \ M \tag{96}$$

$$Z_2 = \begin{bmatrix} 1 & z_1 & z_1^2 & \ldots & z_1^{L-1} \\ 1 & z_2 & z_2^2 & \ldots & z_2^{L-1} \\ 1 & z_3 & z_3^2 & \cdots & z_3^{L-1} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & z_M & z_M^2 & \ldots & z_M^{L-1} \end{bmatrix}, size \ M \ x \ L \tag{97}$$

$$R = \begin{bmatrix} R_1 & 0 & 0 & \cdots & 0 \\ 0 & R_2 & 0 & \ldots & 0 \\ 0 & 0 & R_3 & \ldots & 0 \\ \ldots & \ldots & \cdots & \ddots & 0 \\ 0 & 0 & 0 & \cdots & R_M \end{bmatrix}, size \ M \ x \ M \tag{98}$$

These matrices can be used in **Equation 99** to form the matrix pencil.

$$Y_2 - \lambda Y_1 = Z_1 R \ [Z_0 - \lambda I] \ Z_2 \quad \rightarrow \det (Y_1^{PI} Y_2 - \lambda I) = 0 \tag{99}$$

I is the MxM identity matrix. Assuming that M is less than L, then the rank of the eigenvalue matrix is M. This is also the rank of the desired noiseless matrix, considering only the eigenvalues of interest. $Y_1^{PI}$ is the pseudo-inverse of $Y_1$.

In order to implement the matrix pencil method, SVD is applied to the Y matrix as in **Equation 100**.

$$Y = U \, \Sigma \, V^* \tag{100}$$

The notation $V^*$ denotes the conjugate transpose of V. Both U and V are unitary matrices, where the columns of U are the left singular vectors, and the columns of V are the right singular vectors. The V matrix will be leveraged when generating the noiseless eigenvalues of interest. The value of M is determined before the final eigenvalues are solved for. There are several ways of justifying this, but the simplest way is to normalize the singular value matrix sigmas. Then a cutoff threshold is set for the data. This threshold tends to be set at some ratio to the largest singular value. Since the sigma values have been normalized, the largest singular value will be 1. In the code implemented, $10^{-4}$ was used as a cutoff threshold. **Equations 101-104** further clarify these relationships.

$$\Sigma_{\text{norm}} = \frac{\Sigma}{\sigma_{max}} \tag{101}$$

$$\Sigma_{i_{ratio}} = \frac{\sigma_{i_{norm}}}{\sigma_{max_{norm}}} = \frac{\sigma_{i_{norm}}}{1} \tag{102}$$

$$\Sigma_{i_{ratio}} < 10^{-4} \rightarrow \ Noise \ \therefore \ Remove \ Measurement \ (M \ determined) \tag{103}$$

$$M = number \ of \ \sigma_{i_{norm}} \ values \ accepted \tag{104}$$

In order to derive the eigenvalues of $Y_1^{PI}Y_2$, the relationship to the V matrix is needed.

$$V_{MPM} = V^*(1:m) \tag{105}$$

V$_{MPM}$ is the matrix that will be utilized for the MPM algorithm. It is the first 8 columns of the conjugate transpose of V. **Equations 106-108** present the definitions for the last variables needed to solve for the system modes.

$$V_{MPM} = [v_1 \ v_2 \ v_3 \ ... v_M] \tag{106}$$

$$V_1 = [v_1 \ v_2 \ ... v_{M-1}] \tag{107}$$

$$V_2 = [v_2 \ v_3 \ ... \ v_M]\tag{108}$$

The terms $v_1$ through $v_n$ are column vectors of the $V_{MPM}$ matrix. $V_1$ is the $V_{MPM}$ matrix with the last column removed and $V_2$ is the $V_{MPM}$ matrix with the first column removed.

$$Y_1^{PI}Y_2 = V_1^{PI}V_2\tag{109}$$

The eigenvalues of the constructed $V_1$ and $V_2$ matrices as presented are equivalent to the $Y_1$ and $Y_2$ matrices, given the same relation as **Equation 109**. Once the z poles are solved through this method, each the residues can be calculated through solving **Equation 110.**

$$\begin{bmatrix} y(0) \\ y(n+1) \\ y(n+2) \\ \vdots \\ y(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & ... & 1 \\ z_1 & z_2 & z_3 & ... & z_M \\ z_1^2 & z_2^2 & z_3^2 & ... & z_M^2 \\ \vdots & \vdots & \vdots & ... & \vdots \\ z_1^{N-1} & z_2^{N-1} & z_3^{N-1} & ... & z_M^{N-1} \end{bmatrix} \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ \vdots \\ R_M \end{bmatrix}\tag{110}$$

Since M should be less than L, cases where M is substantially large or equal to M are an indication that the model is not accurately estimating the modes. The solution for the residues is similar to Prony Analysis, but M is typically significantly smaller than the (n) modes used to build the Prony model.

## 4.4 Comparison of MPM and Prony Analysis

Prony Analysis has the advantage of increased computational power and higher system order, but this comes at a steep price when considering an algorithm for real-time operation. Prony Analysis is not ideal for analyzing noisy signals when compared to MPM. The system order of Prony is many magnitudes higher. The test system under study is the Kundur Two-Area system, which can simulate oscillations after major events. MPM is

more robust when working with noisy signals. The measurements from industry PMUs have a highly nonlinear characterization, even when compared to a circuit with simulated faults, standard load switches, and heavy loads. Both methods detected undamped power oscillations for simulated data, but the timing of the methods was not identical when registering both a frequency in the desired range of 0.1-0.8Hz and a damping ratio under 5%. Since the MPM is not as influenced by noise when calculating the damping ratio, a filter was applied to highlight specific frequency ranges and test whether Prony Analysis would benefit from the procedure. The unfiltered and filtered cases were compared to decide whether applying a filter to the power signal improves computational time or the accuracy of the estimates for Prony Analysis.

First the result of the unfiltered case should be discussed. **Figure 4.1** shows the dominant low frequency calculated through the MPM bounded by the two eigenvalues in the Prony Analysis matrix. The first eigenvalue is the closest eigenvalue generated by the method to the MPM eigenvalue in question. The second eigenvalue of interest from the Prony Analysis method is the eigenvalue that is approximately the eigenvalue within a bounded region that exhibits a similar damping ratio trend. The trends are difficult to see in **Figure 4.1**, but **Figure 4.2** and **Figure 4.3** show the individual relationships.
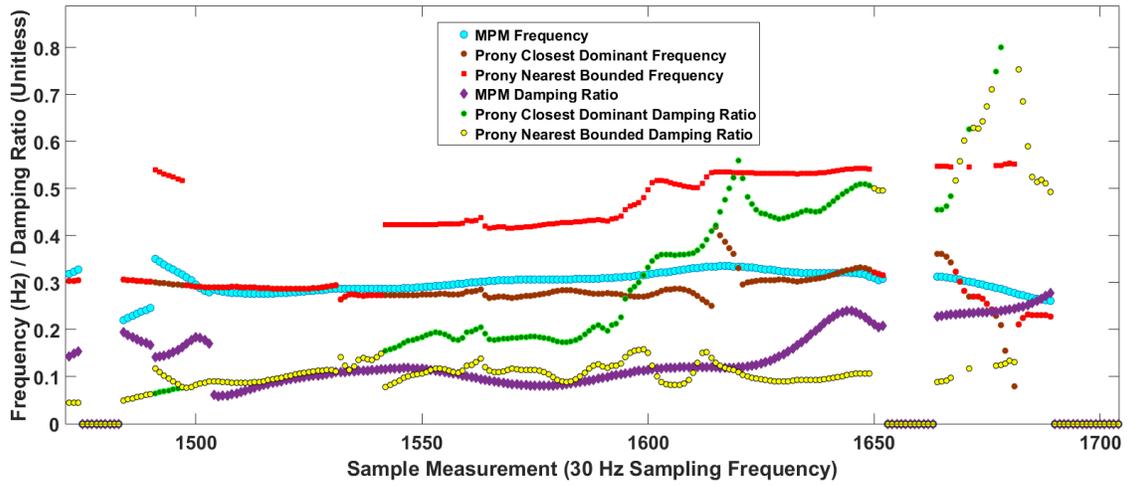
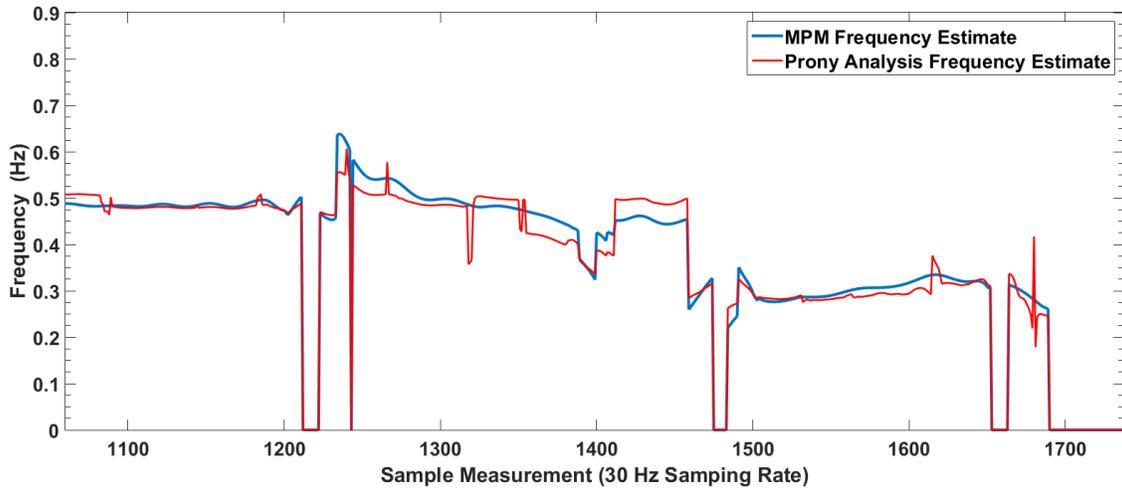**Figure 4.1. Prony Analysis Bounding MPM After Event**



**Figure 4.2. Matrix Pencil and Prony Analysis Estimate for Lowest Frequency**
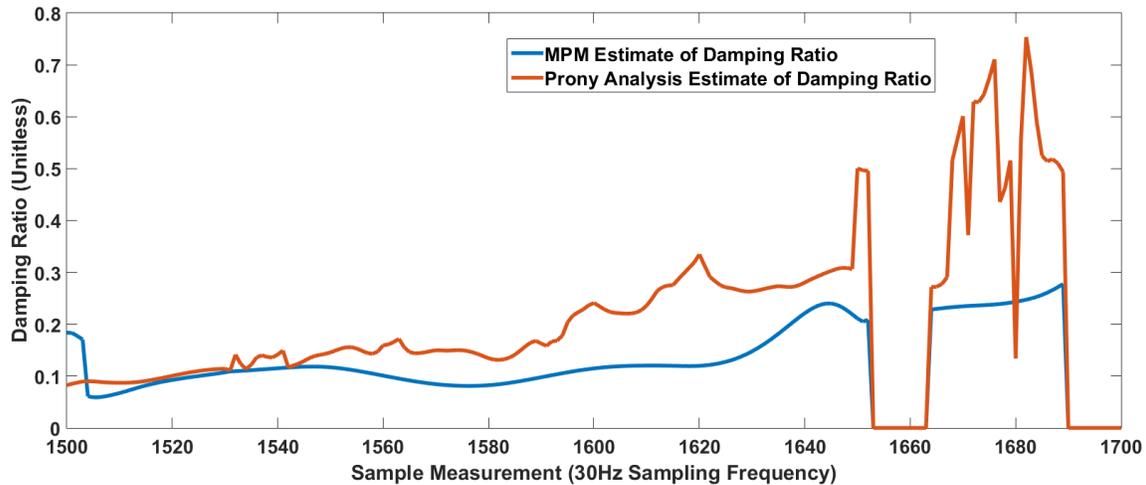
**Figure 4.3. MPM and Prony Estimate for Damping Ratio Associated with Lowest Frequency**

**Figure 4.1** demonstrates the full relationship between the methods and their underlying output. Even under significant system noise, by using both of the closest eigenvalues from Prony Analysis as a weighted bound, the MPM output is roughly mimicked in **Figure 4.2** for dominant frequency above 0.1Hz and in **Figure 4.3** for the associated damping ratio of the focused eigenvalue.

Prony Analysis gives an adequate model but even with modifications, MPM performs better under system noise. There is a direct relationship that can be drawn between the methods. An early attempt at pre-filtering the data was applied to estimate whether even under system noise, Prony Analysis could achieve nearly the name eigenvalues consistently. A low pass filter was applied to primarily pass frequencies between 0-6Hz. Several bandpass filters were also tested for performance and computational speed. Below is the comparison of running MPM on the original signal and applying Prony analysis to a filtered signal with a pass range of 0.1Hz to 6 Hz. The decibel

magnitude of the filter is presented in **Figure 4.4**. A Hamming window was applied for a nearly perfect bandpass filter requiring a moving window length of 401 samples to implement the filter to specification. The Hamming window was selected due to its high attenuation on the stopband while having a relatively short transition period from 0 to peak.

$$h(n) = \left( \frac{\sin\left(\frac{6\pi}{15}*\left(n-\frac{M}{2}\right)\right)}{\pi\left(n-\frac{M}{2}\right)} - \frac{\sin\left(\frac{0.1\pi}{15}*\left(n-\frac{M}{2}\right)\right)}{\pi\left(n-\frac{M}{2}\right)} \right) \left(0.54 - 0.46\cos\left(\frac{2\pi n}{M}\right)\right) \tag{111}$$

$$when\ n \neq \frac{M}{2} \qquad when\ n = \frac{M}{2}, \qquad h(n) = \frac{5.9}{15}$$
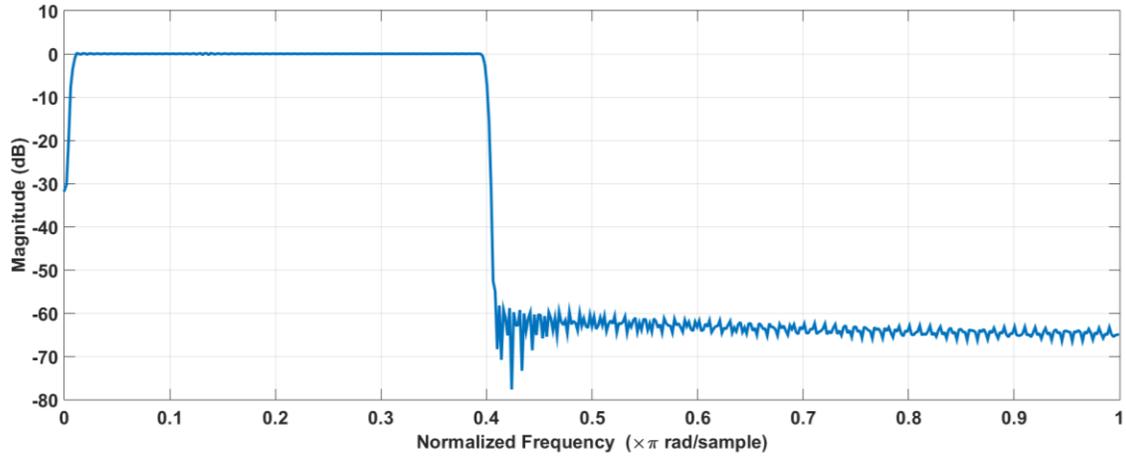


**Figure 4.4. Bandpass Filter Using Hamming Window**

The time to apply the filter to data was insignificant: even a large sample, 1500 measurements, was not time consuming at about 0.001 second to execute. The "filter" function in Matlab was used to recreate the original signal with the filter applied. There were obvious obstacles with this implementation such as accounting for a time difference despite having the same sample size. The primary concern was to reduce the time for Prony analysis to accurately analyze the sample. This method does distort the original data, but

the point was to manipulate it in such a way that Prony might be faster. The bandpass filter had the noticeable effect of removing the low frequencies. These frequencies included changing constants and magnitudes.

For the filtered signal presented in **Figure 4.5**, all low and high frequencies were removed. From experience, the signals with a high frequency are rarely of interest to oscillation monitoring, and most oscillations below 0.1Hz have a small residue. The estimate of the lowest significant system frequency and damping ratio are shown in **Figure 4.6** and **Figure 4.7**, respectively. The oscillatory modes and damping ratio can change very quickly given system conditions of a large load or fault. The filter was ultimately ineffective when trying to reduce computation time. For some cases, a filter would increase the speed temporarily but then impede the system later. Several different filters were tested with a plethora of parameters. According to the tic/toc function on Matlab, for a window length of 300 measurements, MPM would perform between 10-100 times faster on simulated data, including both regular operation and simulated system events.
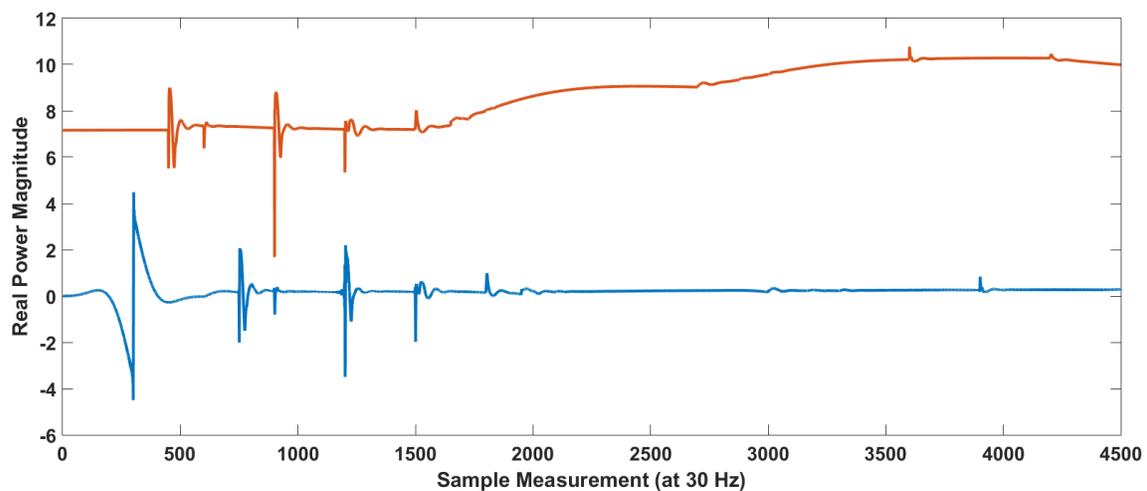


**Figure 4.5. Filtered Real Power Signal (Bottom) vs. Original Real Power Signal (Top)**

79
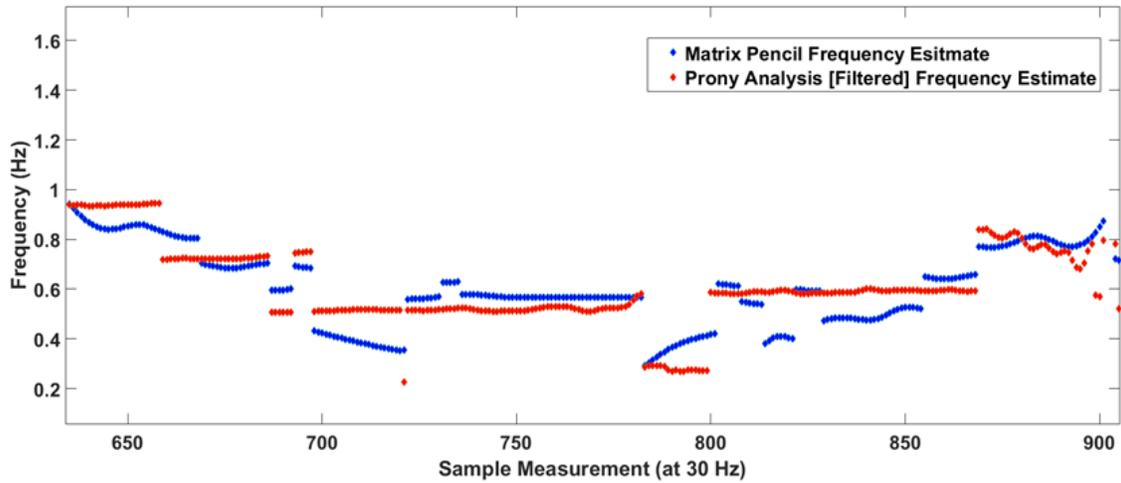
**Figure 4.6. MPM vs Prony Analysis Applied to Filtered Signal for Frequency Estimation**
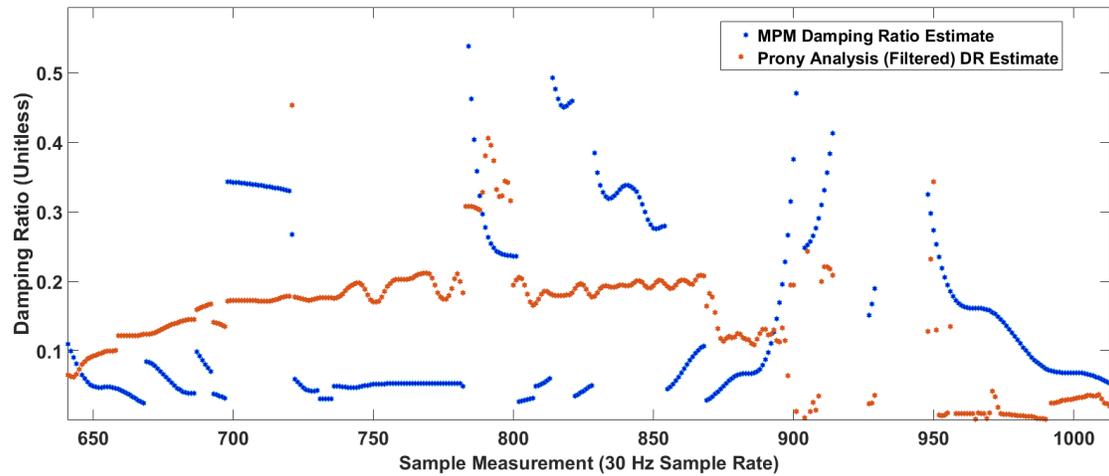


**Figure 4.7. MPM vs Prony Analysis Applied to Filtered Signal for DR Estimation**

The time difference between the methods is not substantial until an event occurs. Most MPM outputs resulted in between 1-5 oscillatory modes with excellent estimation of the (n+1) discrete time step.

Next, a mathematical oscillation is generated as the input signal for both MPM and Prony Analysis. This power signal, shown in **Equation 112**, contains an unstable mode at

0.8Hz with a positive real exponent, causing the signal magnitude to increase similarly to a real unstable power oscillation. The signal also contains two damped modes at 0.4 and 10 Hz. The resulting real power signal can be seen in **Figure 4.8**.

$$P(t) = e^{0.1*t} * \cos(2\pi * 0.8t) + 3e^{-2t}\cos(2\pi * 0.4t) + 2e^{-0.5t}\cos(2\pi * 10t) \quad (112)$$
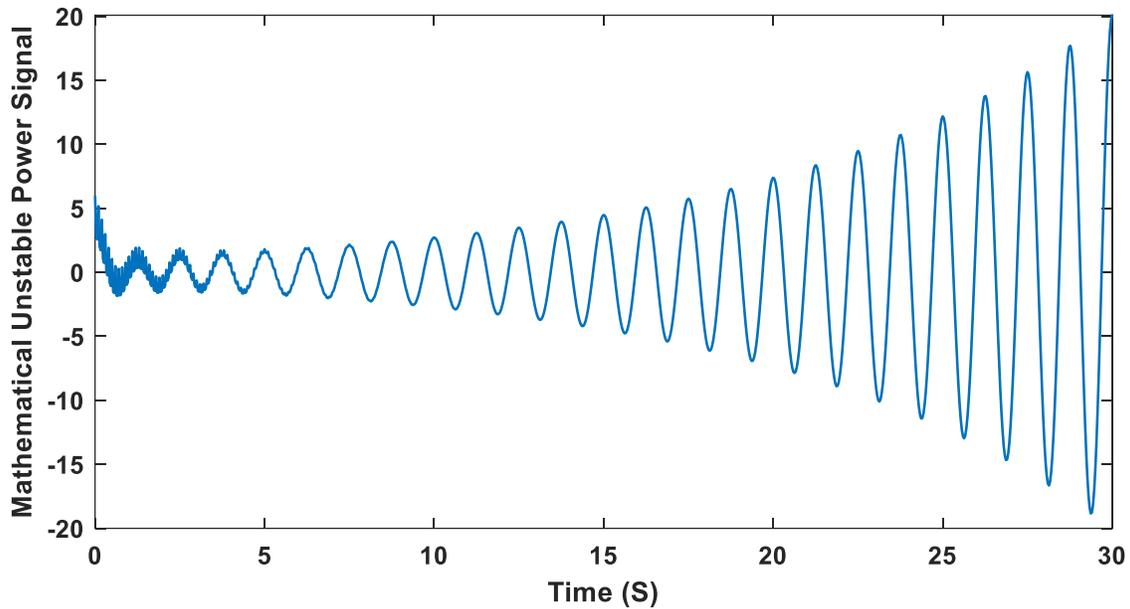


**Figure 4.8. Mathematically Derived Unstable Real Power Oscillation**

Since the signal is known, the estimation by both MPM and Prony Analysis for the dominant frequencies and real component of the eigenvalue can be compared. The analysis starts at 10 seconds, consistent with the required 10 second running window to identify 0.1Hz oscillations. **Figure 4.9** and **Figure 4.10** show the dominant frequencies detected for MPM and Prony Analysis, respectively.
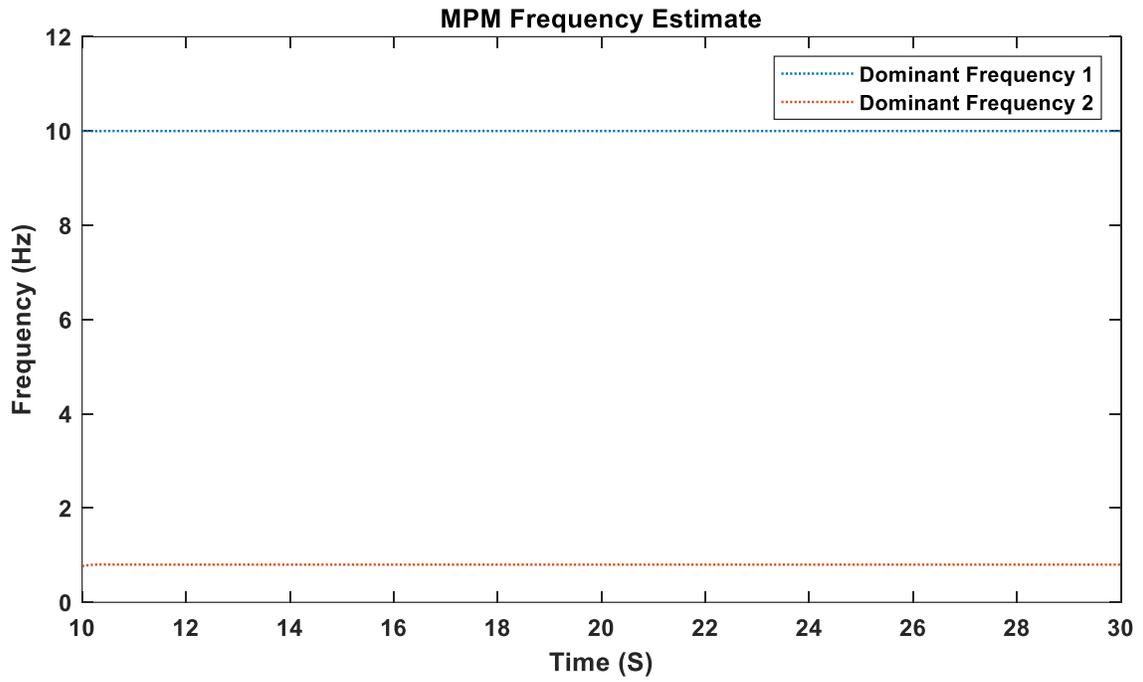
**Figure 4.9. MPM Dominant Frequency Estimates for Unstable Power Signal**
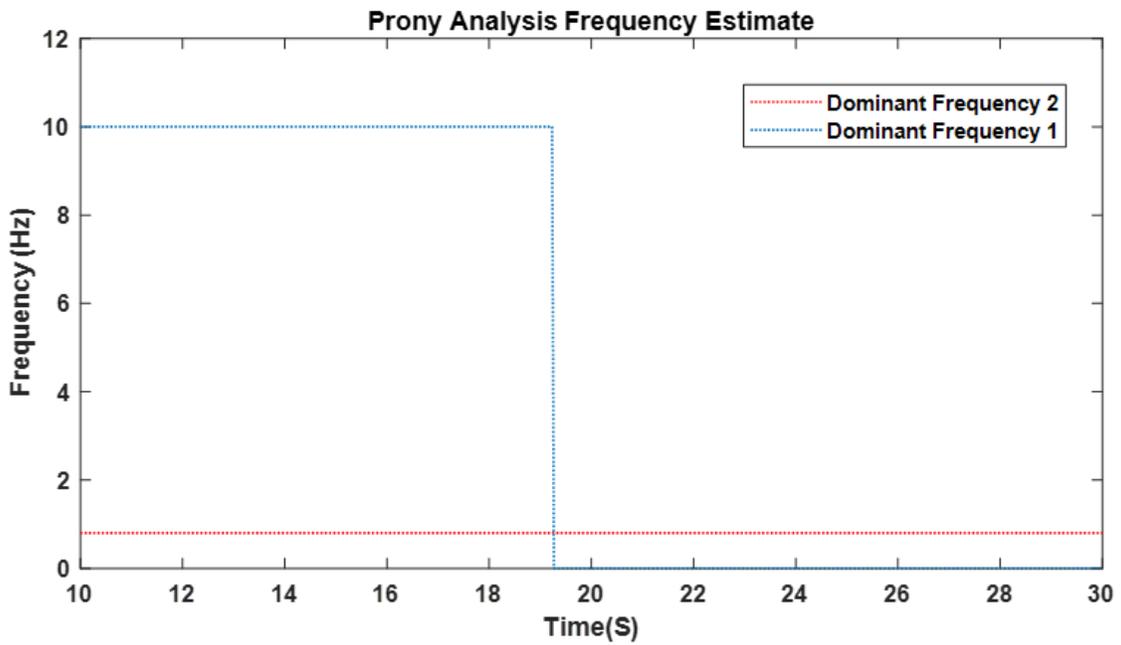


**Figure 4.10. Prony Analysis Dominant Frequency Estimates for Unstable Power Signal**

MPM detects two of the dominant frequencies over the observable window of analysis. The mode associated with 0.4Hz quickly decays and is processed as noise. Prony Analysis also excludes the frequency at 0.4 Hz, as the residue is below the threshold. Prony Analysis and MPM detect the frequency associated with the unstable mode at 0.8 Hz. It should be noted that as the oscillation grew in magnitude, Prony Analysis did lose capability to detect the 10 Hz oscillation in **Figure 4.10**, which was reported by MPM over the entire timeframe in **Figure 4.9**. **Figure 4.11** and **Figure 4.12** show each method's capability to accurately estimate the real component of the eigenvalues associated with the dominant frequencies.
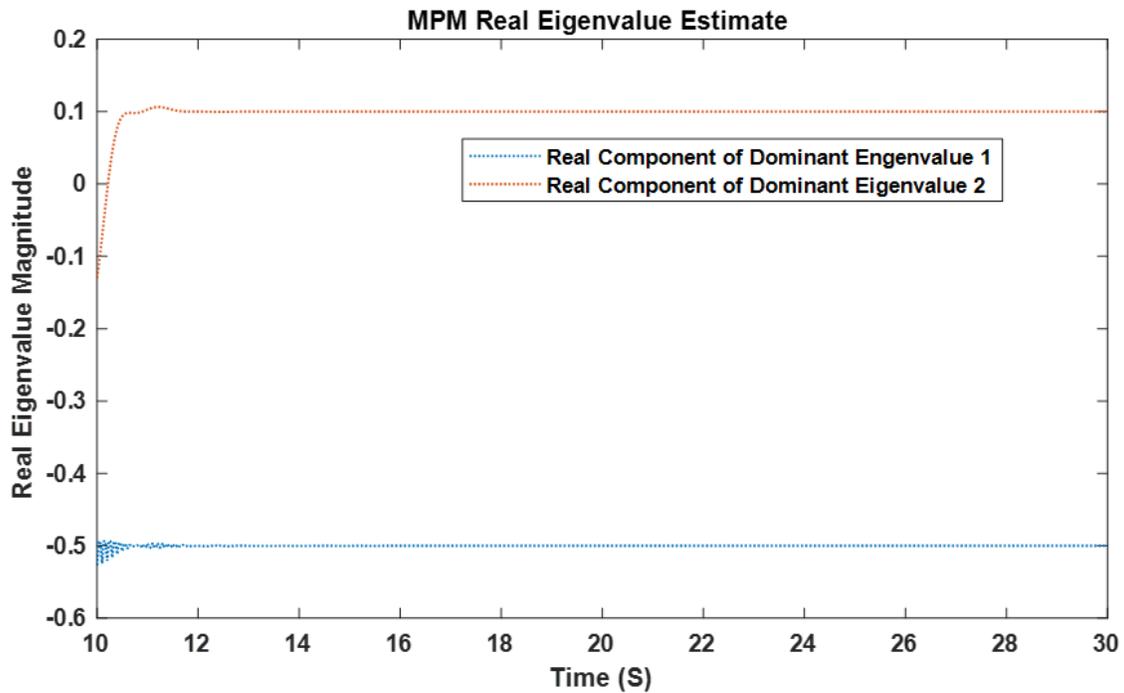


**Figure 4.11. MPM Estimate of Real Eigenvalue Component**
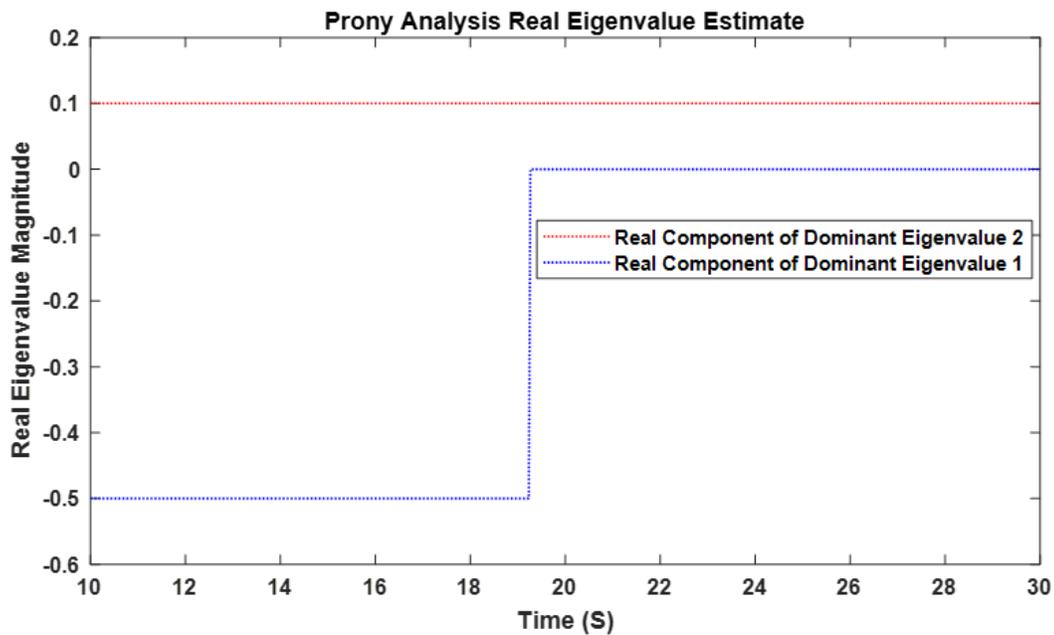
**Figure 4.12. Prony Analysis Estimate of Real Eigenvalue Component**

From **Figure 4.11** and **Figure 4.12**, it can be noted that both MPM and Prony Analysis estimate the unstable mode adequately, which is undamped with a value of 0.1, over the analyzed window. In this case, Prony Analysis accurately estimates the positive magnitude slightly faster than MPM. However, as the simulation continues, MPM maintains accurate estimates of the real component of both dominant modes. Since MPM processes the quickly decaying frequency as noise, but the signal does initially have observable magnitude, the initial inaccuracy of MPM is acceptable and is quickly rectified. MPM shows more robust and consistent mode identification and estimation considering the entire time window.

The real power signal from a utility PMU device can be seen in **Figure 4.13**. This signal is significantly more complex, so the speed and performance of MPM and Prony Analysis applied to this data is more valuable to implementation in real utility systems.
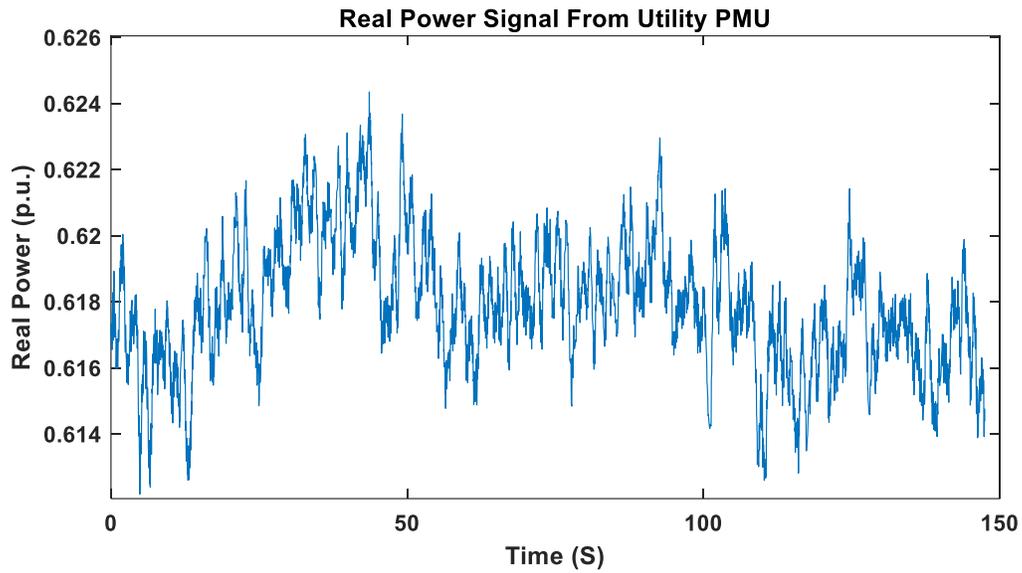
**Figure 4.13. Real Power Signal for Analysis from a Utility PMU**

**Figure 4.14** shows the lowest frequency associated with the dominant modes and the corresponding damping ratio, estimated by MPM. The frequencies and damping ratios from MPM were more consistent for the utility real power data. The closest frequency associated with a dominant mode from the Prony Analysis output, as well as the associated damping ratio, are shown in **Figure 4.15** for comparison.
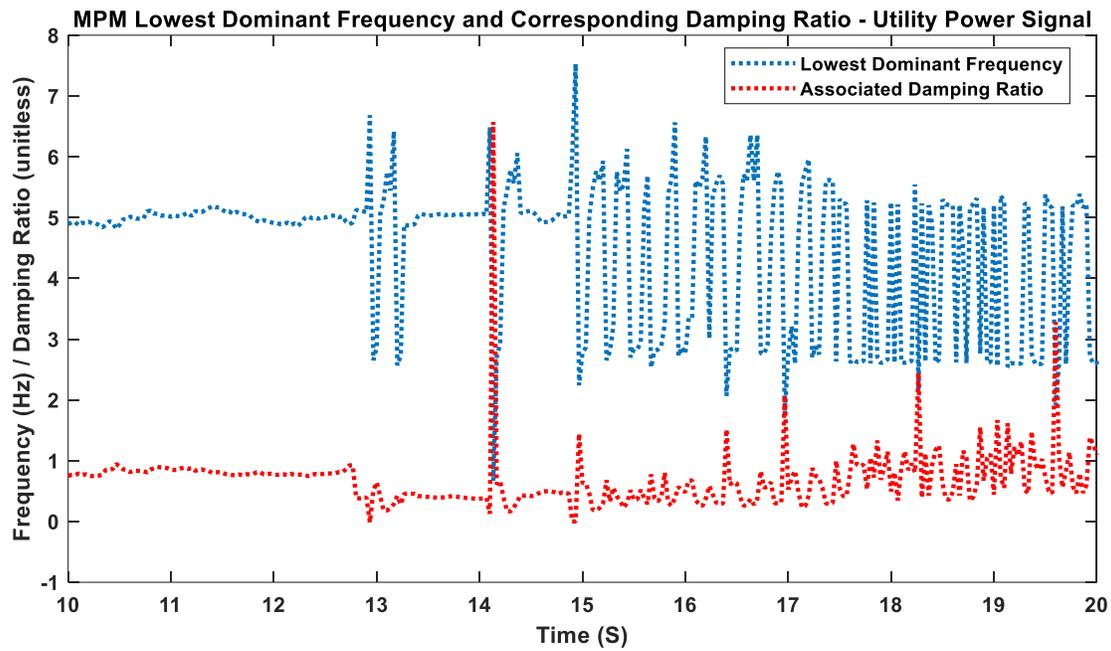
**Figure 4.14. MPM Lowest Frequency and Associated Damping Ratio for a Dominant Mode**
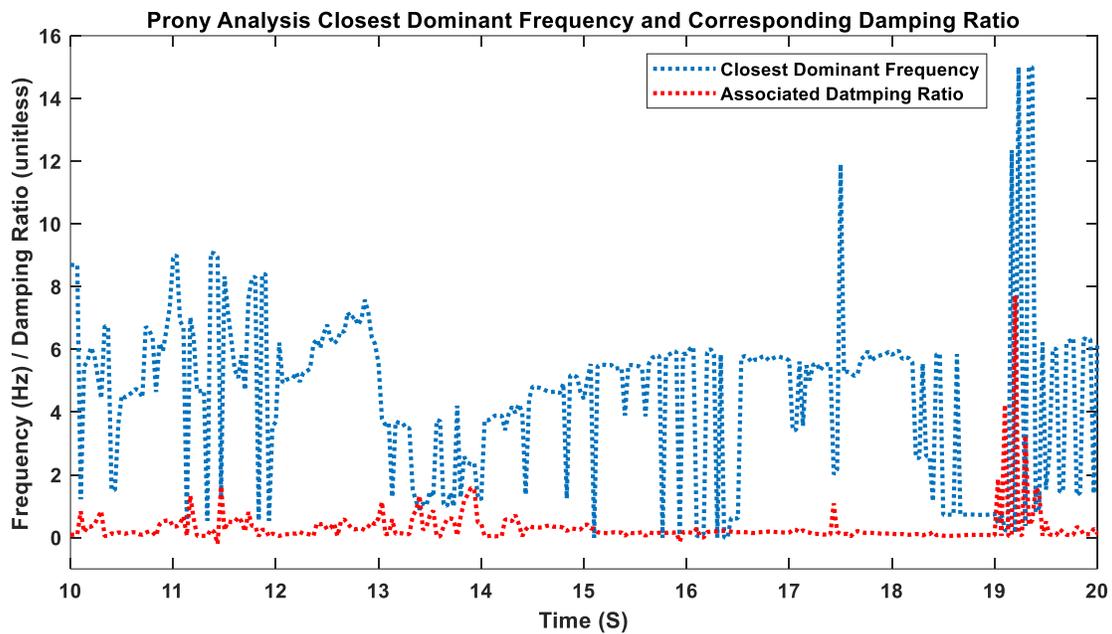


**Figure 4.15. Prony Analysis Closest Dominant Frequency and Associated Damping Ratio for a Dominant Mode**

MPM produces more consistent frequency estimates and shows lower volatility when estimating the lowest observable frequency with an associated damping ratio. Calculated from Matlab code, MPM was able to perform all analysis in 0.0053 seconds on average for the utility power signal. Prony Analysis converged in 0.0192 seconds for the utility power signal, requiring about 3.5x more computational time on average for processing real PMU data streams. The previously developed filters for electromechanical modes would not meaningfully increase computational speed or estimation for the Prony method, as the dominant frequency was outside of the filter bounds and Prony analysis derives a 120-order model before removing insignificant residues. MPM applies SVD to estimate only the modes associated with a theoretical noiseless signal. This also makes MPM more robust if noise is introduced to the signal. Both MPM and Prony Analysis are within bounds for near real-time computation when considering individual signals. Any application attempting to monitor a sufficiently large number of signals would need to consider a dedicated system and architecture. Alternatively, a moving window, such as 1 second, could be used to fully consume all incoming PMU data generated over the past 10 seconds. If executed on a large number of power signals, MPM presents a more robust and computationally efficient solution for utility implementation.

CHAPTER FIVE

SITUATIONAL AWARENESS METHODOLOGY FOR SYSTEM WEAK POINT
IDENTIFICATION AND INSTABILITY MITIGATION

This chapter seeks to bridge the gap between the model-free applications and the model-based applications that require access to the system topology. Chapter 5 presents an enhanced application and proves a more robust process than the derivation in Chapter 2. The methodology detailed in this section derives a synthetic mathematical system model that can iteratively update as new information becomes available or topology changes, to generate two Jacobian variants, directly measured from available PMU devices and iteratively calculated via the Newton-Raphson algorithm. The simulations are carried out with some limited assumptions about the system topology, but these assumptions are not required to execute the application. These assumptions were meant to be easily integrated into a utility setting, as these assumptions would be reasonably met without requiring any detailed system topology or the admittance matrix:

- PMUs in close proximity to major generators are known; flags for these generators can be articulated to system operators.

- Additional PMUs not indicated in proximity to system generators can be distinguished as separate from the PMU devices assigned to a major generator; if the PMU is still in general proximity to a generator, it can be assigned to that generator indirectly.

o A mathematical approach to designate PMUs in either close proximity or redundant PMUs is presented when processing utility PMU data for 112 signals.

This methodology does not have any minimum observability requirements. The accuracy and potential of this methodology do increase as key points in the system become observable. In particular, observability of major generators allows for more thorough identification of unstable conditions and the nature of instability (angular instability or voltage collapse). Simulated case studies were performed assuming observability of all generators. PMU data from a real and sufficiently large utility system is also incorporated to test the methodology's performance for real and simulated data. Bad data detection is not considered and was not a focus of research in this dissertation. However, errors are introduced by utilizing outdated measurements when forming the MSSM for applied state estimation computations, and the accuracy and speed of the resulting state estimation is quantified. This methodology is intended to act as a mathematical linear state estimator. Due to this and the function to flag unstable system conditions, this application is extremely time sensitive and is intended to provide a solution in real-time for all PMU measurements available in a system.

## 5.1 General Format for the Mathematical Synthetic System Model Approximation (MSSM)

The MSSM is a mathematically generated topology designed so that it can incorporate available or desired PMU measurements. The construction is designed for computational simplicity and to prioritize, if known, any PMUs located on generators.

Additional PMUs in proximity or throughout the system are also considered. The model shown in **Figure 5.1** may be modified for additional factors, but the computational efficiency and accuracy achieved by leveraging this model are reviewed in *Sections 5.4* and *5.5*. This model does lose observability in large systems if very few of the generators are observable, no PMU is in proximity to an event, or net generation cannot be aligned with critical changes to load data. The derived model is useful for monitoring the stable state of the system generators and determining when unstable parameters have been introduced to the system.
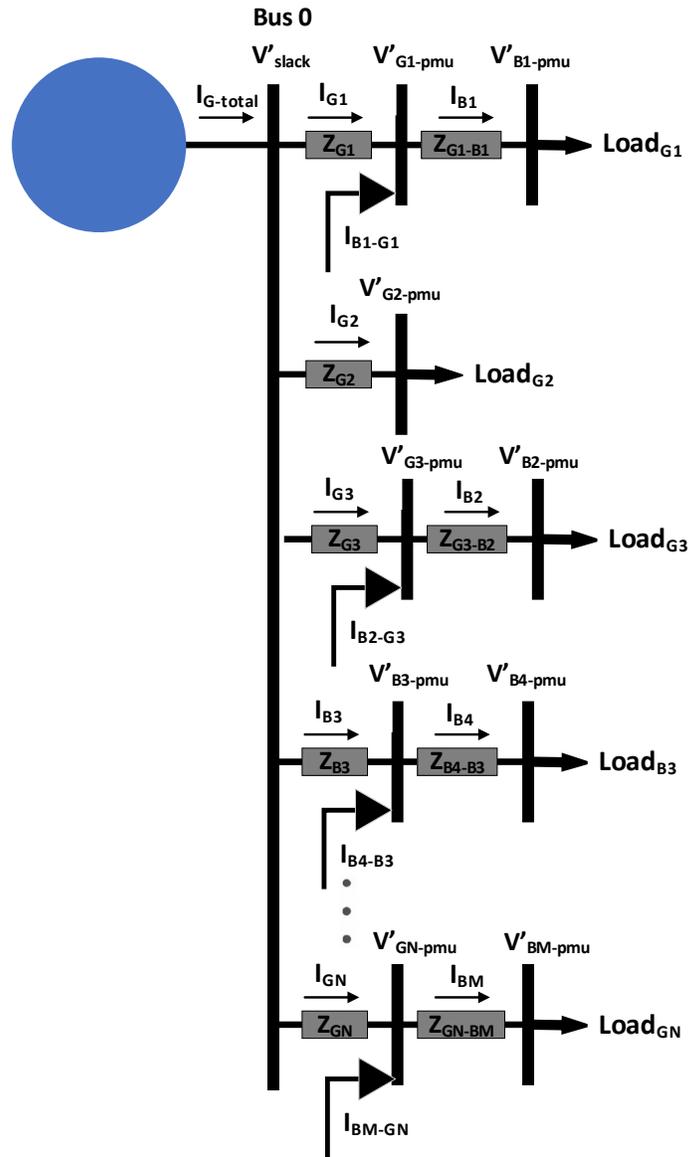
**Figure 5.1. MSSM General Format**

Brief definitions and explanations for the variables in this theoretical model include:

- $V'_{slack}$ – This is the phasor voltage of the theoretical master slack bus; the magnitude is flexible and can be set higher than $V'_{G\text{-}pmu}$ measurements, but typically the

positive sequence would be set in the range of 1.00-1.05 p.u. V and the bus voltage positive sequence phase angle would be set to 0 radians.

- $V'_{G1\text{-pmu}}$, $V'_{G2\text{-pmu}}$, $V_{GN\text{-pmu}}$ – Each of these variables represents the phasor values for voltages measured by generator PMUs; these individual generators are accounted for in the derived model as buses with a relationship to the theoretical master slack bus.

  - If not known, $V'_{B\text{-pmu}}$ measurements can also be substituted.

- $I_{G\text{-total}}$ – This variable represents the sum of all currents measured by the PMUs observing generators.

- $I_{G1}$, $I_{G2}$ … $I_{GN}$ – Each of these variables represents the phasor currents measured at the PMUs on the corresponding generators.

- $Z_{G1}$, $Z_{G2}$ … $Z_{GN}$, $Z_{B1\ ...}$ – Each of these variables represents the impedance calculated to explain the voltage drop from the theoretical master slack bus to the generator phasor voltages (or first bus assigned to a branch), considering the particular generator's phasor currents.

- $V'_{B1\text{-pmu}}$ … $V'_{BM\text{-pmu}}$ – These variables represent the phasor voltages for additional buses with PMUs in close proximity to a generator, or other system buses.

- $I_{B1\text{-}G1}$ … $I_{BM\text{-}GN}$, $I_{B3\text{-}B2}$ – These variables are derived to account for any difference in the current from the measured generator current to an additional bus as the current flows to a net load; in cases where there are multiple buses with PMUs grouped in close proximity to a generator, this variable can expand to $I_{B1\text{-}B2}$ to explain any difference in current as the model increases in complexity.

- $I_{B1}$ … $I_{Bn}$ – These variables represent the phasor currents measured by buses assigned proximity to the generator PMUs.

- $Z_{G1-B1}$, $Z_{B1-B2}$ … $Z_{GN-BM}$ – These variables represent the calculated impedance to account for voltage drops between buses assigned proximity to a generator.

- $Load_{G1}$, $Load_{G2}$ … $Load_{GN}$ – These variables represent the system load contributed for the branch relating to each generator, accounting for any differences in net load through injection along the circuit.

The general configuration of the MSSM, shown in **Figure 5.1**, is flexible and is designed to handle any number of generator PMUs and any number of PMUs installed at other buses throughout the system. This derivation can effectively be done considering all PMUs as bus PMUs. V'$_{slack}$ functions as a mathematical system reference for any derived configuration. The parameters of the MSSM can be iteratively updated as new PMU measurements become available.

## 5.2 Deriving the Synthetic Model Jacobian Approximation

As previously discussed, the Jacobian matrix, iteratively calculated during Newton-Raphson state estimation, has been utilized to assess system stability by monitoring eigenvalues and developing singularities of the inverse Jacobian [8][9]. Traditional computation of the power flow Jacobian requires system topology for implementation. The MSSM is leveraged to generate $Y_{Bus}$ components so that a model-free solution can apply similar analytics. **Equations 113-116** show the derivation for the Jacobian matrix considering a MSSM configuration with PMUs located at generators 1 to

N. All $Y_{Bus}$ and voltage magnitudes are per unit values derived from the MSSM, and the bus voltage angles are in radians.

$$P_n = |V_n| \sum_{j=1}^{N} |Y_{nj}| * |V_j| * \cos(\delta_n - \delta_j - \theta_{nj}) \tag{113}$$

$$Q_n = |V_n| \sum_{j=1}^{N} |Y_{nj}| * |V_j| * \sin(\delta_n - \delta_j - \theta_{nj}) \tag{114}$$

$$J_1 = \begin{bmatrix} \frac{\partial P_1}{\partial \delta_1} & \cdots & \frac{\partial P_1}{\partial \delta_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial P_N}{\partial \delta_1} & \cdots & \frac{\partial P_N}{\partial \delta_N} \end{bmatrix}, \ J_2 = \begin{bmatrix} \frac{\partial P_1}{\partial |V_1|} & \cdots & \frac{\partial P_1}{\partial |V_N|} \\ \vdots & \ddots & \vdots \\ \frac{\partial P_N}{\partial |V_1|} & \cdots & \frac{\partial P_N}{\partial |V_N|} \end{bmatrix}, J_3 = \begin{bmatrix} \frac{\partial Q_1}{\partial \delta_1} & \cdots & \frac{\partial Q_1}{\partial \delta_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial Q_N}{\partial \delta_1} & \cdots & \frac{\partial Q_N}{\partial \delta_N} \end{bmatrix} \tag{115}$$

$$J_4 = \begin{bmatrix} \frac{\partial Q_1}{\partial |V_1|} & \cdots & \frac{\partial Q_1}{\partial |V_N|} \\ \vdots & \ddots & \vdots \\ \frac{\partial Q_N}{\partial |V_1|} & \cdots & \frac{\partial Q_N}{\partial |V_N|} \end{bmatrix}, \ J = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \tag{116}$$

As this method only considers $Y_{Bus}$ and voltages from the MSSM model rather than the actual system topology, and these measurements are known for each iteration, the first Jacobian variant can be derived by direct measurement every time PMU measurements are received. This is separate from computing the power flow Jacobian iteratively during Newton-Raphson state estimation. The Jacobian computed directly from measured PMU values and the MSSM topology is notated in this dissertation as the Synthetic Model Jacobian Approximation (SMJA) matrix.

$$SMJA = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix}, \ Measured \tag{117}$$

## 5.3 Deriving the Synthetic Model Power Flow Jacobian

In addition to direct measurement, the MSSM $Y_{Bus}$ and derived load conditions can be used to iteratively compute a power flow Jacobian matrix. For this implementation, the PMU at generator 1, or the first PMU available, is set as a reference. The voltage magnitude

and bus voltage phase angles for all other PMUs are iteratively calculated from an initial start value. This results in a second Jacobian variant calculated via Newton-Raphson power flow. The first iteration considers the initial conditions in **Equations 118-120**. The notation "(k)" has been added to indicate that the voltage magnitudes and bus voltage phase angles are iteratively calculated and separate from the value measured by the PMU. All variables considering these iteratively calculated values are also denoted with notation "(k)" for the current iteration. The iteration is incrementally increased until the solution converges below a specified error. Additionally, all notation is built around PMUs observing generators, but bus PMUs can be substituted into the derived equations if generator locations are not known.

$$V_1(k) = V_{G1}, \ \delta_1(k) = \delta_{G1} \tag{118}$$

$$V_2(k), V_3(k), \dots V_N(k) = V_{start} \tag{119}$$

$$\delta_2(k), \delta_3(k), \dots \delta_N(k) = \delta_{start} \tag{120}$$

The values for $V_1(k)$ and $\delta_1(k)$ are held constant during the iterations. The initial conditions in **Equations 118-120** are applied to the first iteration of **Equations 121-123**. **Equation 120** is used to update the voltage magnitudes and bus voltage angles for the subsequent iteration. The values of $V_{start}$ and $\delta_{start}$ would typically be 1 p.u. and 0 radians, respectively, but can be individually modified based on other preferences or considerations.

$$\begin{bmatrix} \Delta P_1(k) \\ \vdots \\ \Delta P_N(k) \\ \Delta Q_1(k) \\ \vdots \\ \Delta Q_N(k) \end{bmatrix} = \begin{bmatrix} P_1 \\ \vdots \\ P_N \\ Q_1 \\ \vdots \\ Q_N \end{bmatrix} - \begin{bmatrix} P_1(k) \\ \vdots \\ P_N(k) \\ Q_1(k) \\ \vdots \\ Q_N(k) \end{bmatrix} \tag{121}$$

For **Equation 121**, $P_1$ through $P_N$ and $Q_1$ through $Q_N$ consider direct PMU measurements and the MSSM $Y_{Bus}$, solving **Equations 113** and **114**. $P_1(k)$ through $P_N(k)$ and $Q_1(k)$ through $Q_N(k)$ consider the $V(k)$ and $\delta(k)$ variable values of the present iteration and the MSSM $Y_{Bus}$, solving **Equations 113** and **114**.

$$
\begin{bmatrix} \Delta\delta_1(k) \\ \vdots \\ \Delta\delta_N(k) \\ \Delta V_1(k) \\ \vdots \\ \Delta V_N(k) \end{bmatrix} = \begin{bmatrix} J_1(k) & J_2(k) \\ J_3(k) & J_4(k) \end{bmatrix}^{-1} \begin{bmatrix} \Delta P_1(k) \\ \vdots \\ \Delta P_N(k) \\ \Delta Q_1(k) \\ \vdots \\ \Delta Q_N(k) \end{bmatrix}
\tag{122}
$$

Similarly, in **Equation 122**, the individual calculated Jacobian matrices, with notation $J(k)$, are determined using the MSSM $Y_{Bus}$ and present iteration of $V(k)$ and $\delta(k)$ variables applied to **Equations 115-116**.

$$
\begin{bmatrix} \delta_2(k+1) \\ \vdots \\ \delta_N(k+1) \\ V_2(k+1) \\ \vdots \\ V_N(k+1) \end{bmatrix} = \begin{bmatrix} \delta_2(k) \\ \vdots \\ \delta_N(k) \\ V_2(k) \\ \vdots \\ V_N(k) \end{bmatrix} + \begin{bmatrix} \Delta\delta_2(k) \\ \vdots \\ \Delta\delta_N(k) \\ \Delta V_2(k) \\ \vdots \\ \Delta V_N(k) \end{bmatrix}
\tag{123}
$$

The changes to the calculated bus voltage magnitude and bus voltage angle are used to update all values for the next iteration via **Equation 123**.

$$
If \left( \left\| \begin{bmatrix} \Delta P_1(k+1) \\ \vdots \\ \Delta P_N(k+1) \\ \Delta Q_1(k+1) \\ \vdots \\ \Delta Q_N(k+1) \end{bmatrix} \right\| < \epsilon \right), k = k+1, \ End
\tag{124}
$$

**Equation 124** evaluates **Equation 121** considering the updated variables. When the absolute difference of all real and reactive power estimates is below the set threshold

error, $\varepsilon$, the final values of all V(k) and $\delta$(k) variables are returned from the iterative process. The threshold error is set to a target value, such as 0.0001.

$$If \left( \left\| \begin{bmatrix} \Delta P_1(k+1) \\ \vdots \\ \Delta P_N(k+1) \\ \Delta Q_1(k+1) \\ \vdots \\ \Delta Q_N(k+1) \end{bmatrix} \right\| > \epsilon \right), k = k+1, \; Continue \tag{125}$$

If any variables exceed the specified threshold error, the iteration is updated in **Equation 125** and the process is repeated from **Equation 121**, considering the updated V(k) and $\delta$(k) variables. This is iteratively implemented until the condition in **Equation 124** is satisfied.

The final values of the V(k) and $\delta$(k) variables, as well as the MSSM $Y_{Bus}$, are leveraged to derive a calculated Jacobian, referred to in this paper as the Synthetic Model Power Flow Jacobian (SMPFJ). **Equation 126** shows this relationship.

$$SMPFJ = \begin{bmatrix} J_1(k) & J_2(k) \\ J_3(k) & J_4(k) \end{bmatrix}, \; Iteratively \; Calculated \tag{126}$$

The iterative calculation of the load-flow Jacobian, performed when generating the SMPFJ matrix, intrinsically assumes infinite inertia for the reference generator and all loads modelled as constant power [8]. No generator in the physical system can provide infinite inertia, and loads are generally more complex than constant power, but these assumptions are made when solving the standard load-flow Jacobian. Due to this, during events that place dynamic stress on the system, such as bus faults and large load increases, the iteratively calculated solution may converge to a more ideal state than realized. Under normal and stable system conditions, the SMJA and SMPFJ should be effectively identical,

and this will be reflected in the accuracy of SMPFJ to perform state estimation, assessed in the case studies for both simulated and real PMU data.

The SMJA and SMPFJ are both mathematical models leveraging MSSM for the admittance matrix; however, due to the assumptions made when deriving the SMPFJ matrix, the resulting calculations for the system states may yield more optimistic estimates during periods where unstable parameters are present in the model. It is noted in [8] that maximum loadability derived from the Jacobian should be considered an optimistic upper bound. Phrased differently, the conditions that will make the real system unstable occur before theoretical limits are met. Due to this, the SMJA, which is derived from the PMU measurements rather than iterative calculation, is used in the implementation of SVD to track developing singularities.

Once both the SMPFJ and SMJA have been formed, singular value decomposition can be applied for the SMJA matrix, as summarized in **Equations 127-129**, to monitor trends in the corresponding Σ matrix. The diagonal values in **Equation 129**, sigmas, are of particular importance in future figures that monitor the magnitude of the smallest sigmas along the diagonal. As one or many of these sigmas decrease toward zero, the inverse power flow Jacobian will trend toward singularity, indicative of instability.

$$SMJA = U\Sigma V^*$$  (127)

$$UU^* = VV^* = I$$  (128)

$$\Sigma = \begin{bmatrix} s_1 & 0 & \cdots & 0 \\ 0 & s_2 & 0 & 0 \\ \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & s_n \end{bmatrix}$$  (129)

The magnitude of the smallest sigmas, in addition to general trends for all sigmas related to the system, are critical to determining instability. These sigmas are processed for each time step, and any decrease of more than 80 percent from operating norm in any of the lowest 10 sigmas generates a flag, triggering additional analysis. A decrease of more than 90 percent from operating norm in the lowest sigma is also flagged as immediately critical and triggers additional analysis. Decreases of more than 50 percent for any sigma are broadly monitored but do not immediately generate a flag. Since the operating conditions can change over time and a stressed system is occasionally expected, a comparison between the values used to form the SMJA and SMPFJ matrices is conducted in parallel to determine if divergent conditions are developing or changes in sigmas are due to short term volatility while a new system equilibrium point is reached. The PMU voltage magnitudes and bus voltage angles are compared to determine divergence between the measured state and the calculated state.

$$\left|\left|\begin{bmatrix}V_{G1}\\\vdots\\V_{GN}\end{bmatrix}-\begin{bmatrix}V_{G1}(k)\\\vdots\\V_{GN}(k)\end{bmatrix}\right|\right| > \Delta V_{threshold} \tag{130}$$

For clarity in **Equation 130**, $V_{G1}$ is the direct measurement of the voltage magnitude relating to the PMU streaming data for generator 1, or if alternative notation is considered, bus 1. The variable $V_{G1}(k)$ is the voltage magnitude for the bus at generator 1 computed when forming the SMPFJ matrix. Similar notation is used in **Equation 131** regarding the bus voltage angles.

$$\left|\left|\begin{bmatrix}\delta_{G1}\\\vdots\\\delta_{GN}\end{bmatrix}-\begin{bmatrix}\delta_{G1}(k)\\\vdots\\\delta_{GN}(k)\end{bmatrix}\right|\right| > \Delta\delta_{threshold} \tag{131}$$

Small variations between the calculated and measured states are expected, but this difference was found to be miniscule during steady state operation. For the Sections 5.4 and 5.5, a voltage threshold of 0.05 p.u. and a bus voltage angle threshold of 0.1 radians were used to flag critical divergence. When unstable parameters are introduced and not cleared, the two solutions will increasingly diverge until system instability occurs, which will be visually shown in figures summarizing flags for voltage and angular instability through in *Section 5.4*.

**Figure 5.2** demonstrates a robust algorithm to monitor developing singularities in the inverse SMJA matrix, monitor divergence from the calculated system state, and enter mitigation protocols when persistent unstable conditions are identified. **Figure 5.3** demonstrates the logic to implement load shedding when conditions are caused by the addition of an unstable load. When the parameters are passed from **Figure 5.2**, the net real and reactive power generation is cross referenced to any system load increases to ensure that the net increases can be attributed to a corresponding load.
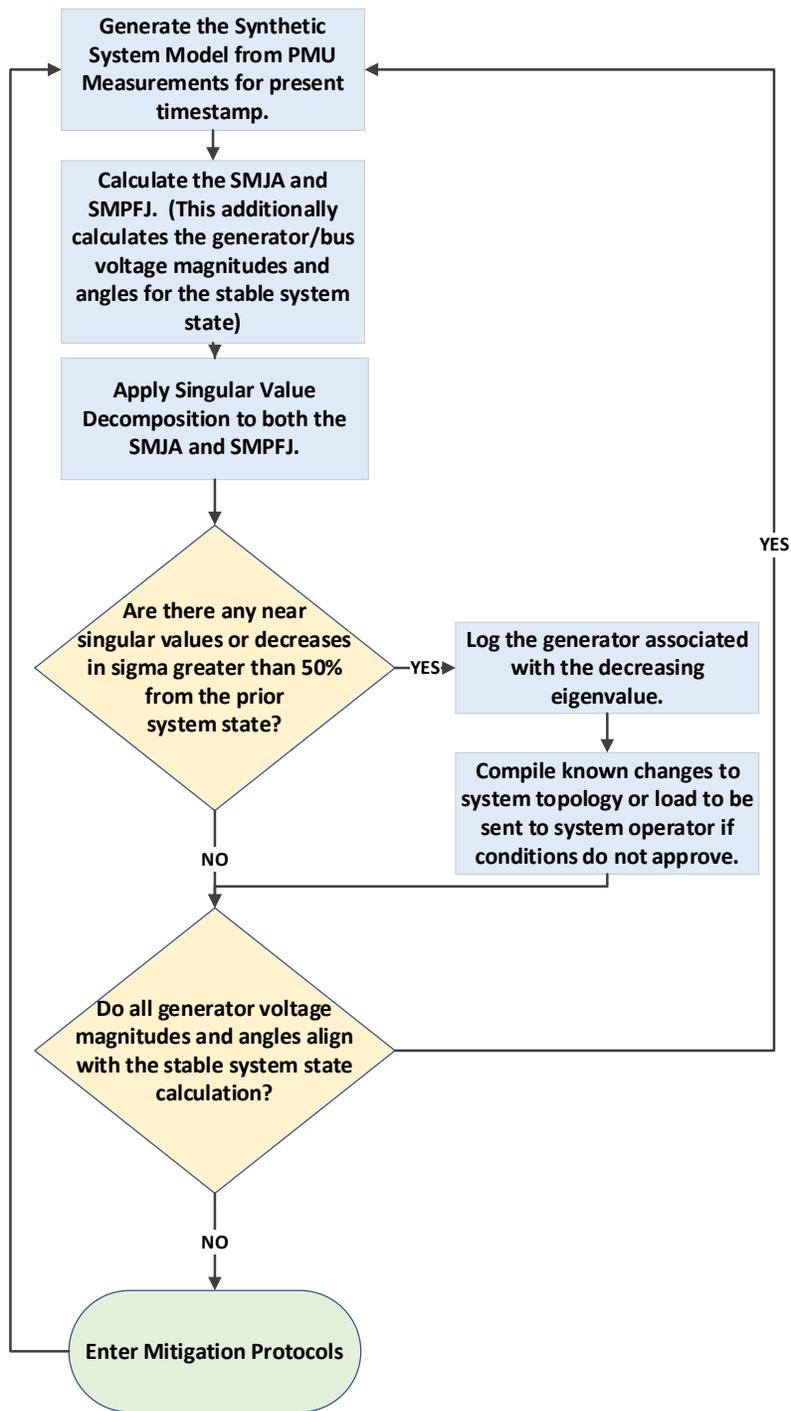
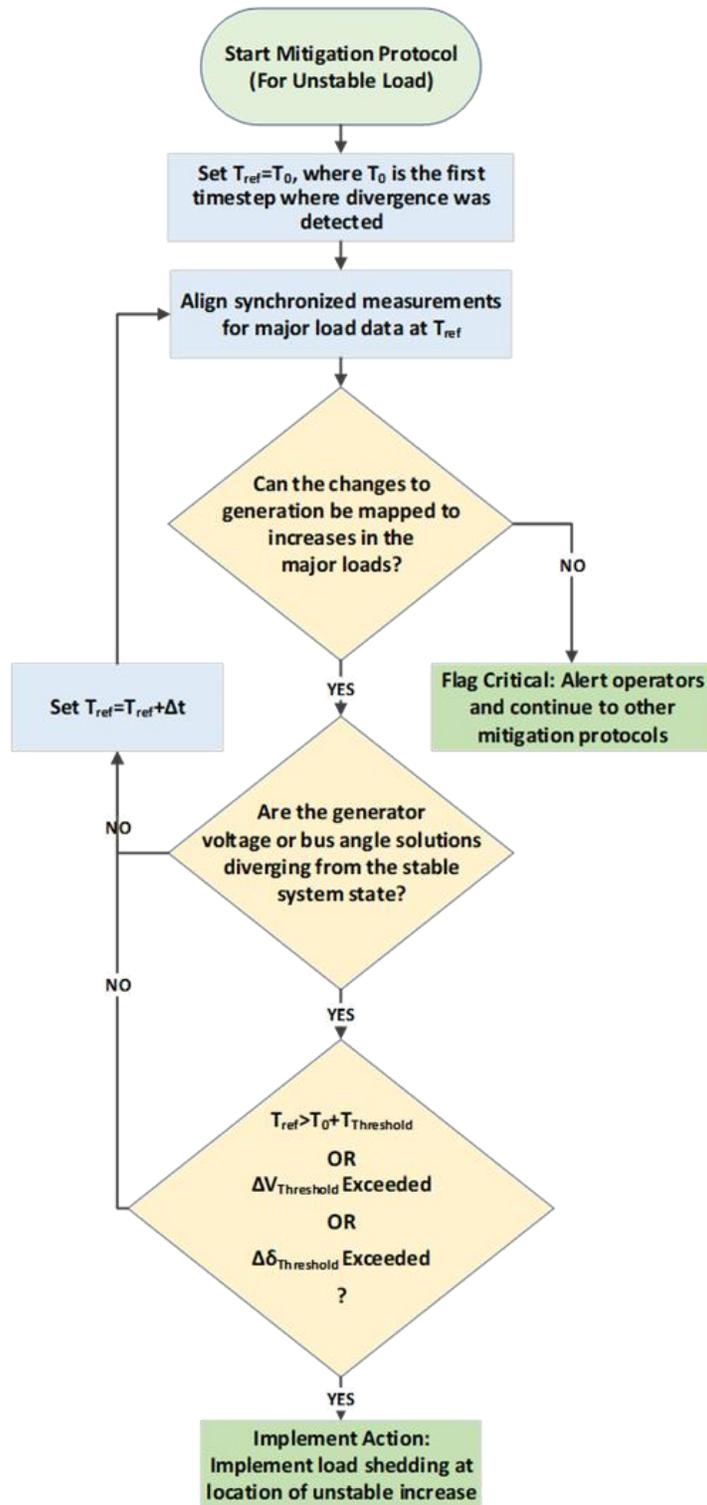**Figure 5.2. Process for Detection of Near Singularities and Flags for Divergence**

**Figure 5.3. Process for Mitigation of Unstable Load Conditions**

A major disturbance to the generator eigenvalues sets an initial reference time, $T_{ref0}$. If the SMJA and SMPFJ matrices are continually diverging for a time greater than $T_{Threshold}$, the flag will be considered critical even if the individual thresholds for voltage magnitude and bus voltage angle are not met, per **Equation 132**. $T_{Threshold}$ is set to 0.5 seconds for the case studies.

$$T_{ref} > T_{ref0} + T_{Threshold} \tag{132}$$

The implementation of load shedding in **Figure 5.3** is not meant to replace existing load shedding methods. The load shedding implemented in the case studies is primarily meant to show that the SMJA and SMPFJ matrices can be utilized to mitigate unstable parameters. Section 5.4 presents the results of this novel methodology to detect unstable system conditions, identify system weak points and diverging parameters, and mitigate instability in simulated case studies. The methodology is also assessed in Section 5.5, applied to real utility PMU data, to show the versatility and computational efficiency of the methodology when applied to more complex and realistic data streams for state estimation and analysis of the SMJA and SMPFJ matrices. The computational efficiency and accuracy of this method are quantified for both simulated and real utility data.

## 5.4    SMJA and SMPFJ Case Study Validation Using Simulated Data

The IEEE 39 Bus System, specifically the MathWorks Simulink 10-machine New-England system [78], was used to generate data for three simulated case studies. These case studies obtain test data for various system events to assess accurate detection and mitigation of unstable parameters, validating the methodology. In this study, the PMUs were placed at the bus after every generator transformer and returned sequence phasor data at a 30 Hz

report rate. This paper investigates mitigation of developing unstable conditions that lead to instability over the span of seconds rather than a few cycles. Although PMUs can be configured to clear events such as faults, these applications tend to require edge computing and consider contingencies that typical protection devices do not handle, such as utilizing PMU data to de-energize circuits where a falling conductor is detected prior to hitting the ground [79]. For typical fault conditions, protection devices will clear the fault more quickly and reliably. Mitigation of fault conditions is not considered, as this methodology is being shown as a proof of concept and not as a substitute to typical protection requiring a high-speed, sub-cycle command to open a circuit breaker. However, the ability to immediately detect the unstable parameters caused by a fault, show the flag's removal if unstable parameters are cleared, and record persistence of unstable parameters when the system is collapsing, are of interest and are demonstrated below.

### 5.4.1   Instability Due to Fault

Case Study 5.4.1 focuses on the fast detection of angular and voltage instability following different fault events. The events are summarized in **Table 5.1**. The initial three faults have a duration of 3 cycles, while the final constant fault at Bus 19 leads to system collapse.

**Table 5.1. Sequence of Events for Case Study 5.4.1**

| Event | Location | Start Time of Event (S) | End Time of Event (S) |
|-------|----------|------------------------|-----------------------|
| Fault | Bus 16 | 20 | 20.05 |
| Fault | Bus 1 | 50 | 50.05 |
| Fault | Bus 6 | 100 | 100.05 |
| Fault | Bus 19 | 180 | N/A (Continuous) |

**Figure 5.4** visualizes the disturbance to the lowest 10 sigmas generated by applying singular value decomposition to the SMJA matrix. For the first three faults at 20, 50, and 100 seconds, the sigmas decline after the fault is introduced, oscillating, and returning to the previous steady state value over time after the fault is cleared. The sub-plot in **Figure 5.4**, a zoom window during system collapse, shows the values of the lowest 10 sigmas during the constant fault at 180 seconds. In this case, the sigmas decline and a near singularity can be seen immediately before the real system collapses.

**Figure 5.5** shows that for all four faults, voltage divergence in the calculated and measured values was detected at one or more generators within one measurement of the fault being applied.
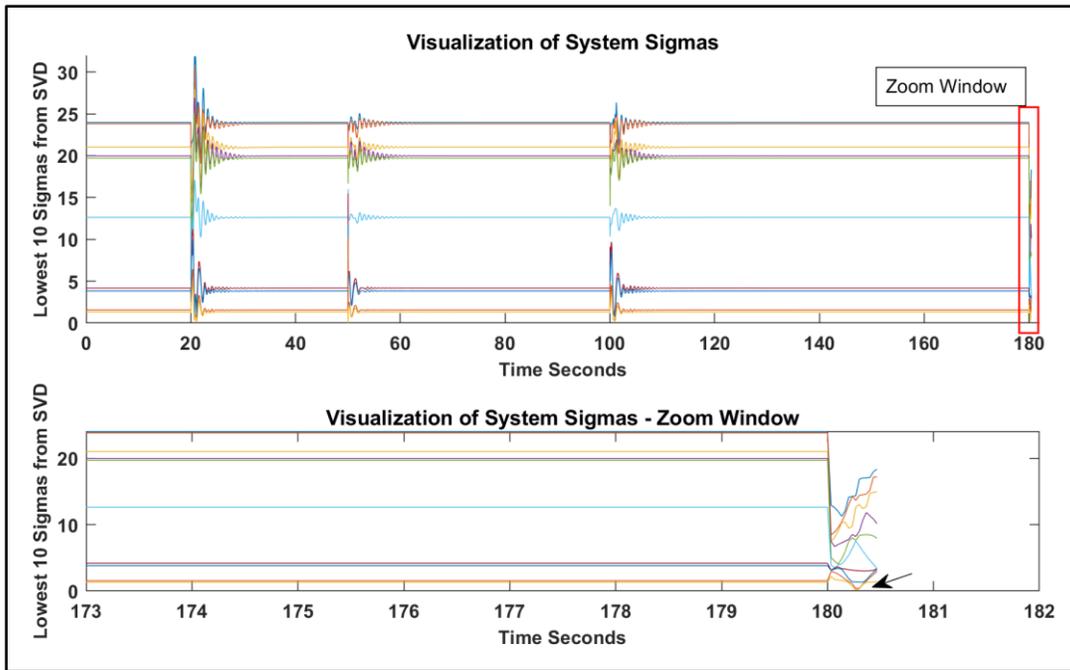
**Figure 5.4. Visualization of the Generators' Lowest 10 SVD Sigmas During Faulted Conditions (With Zoom Window)**
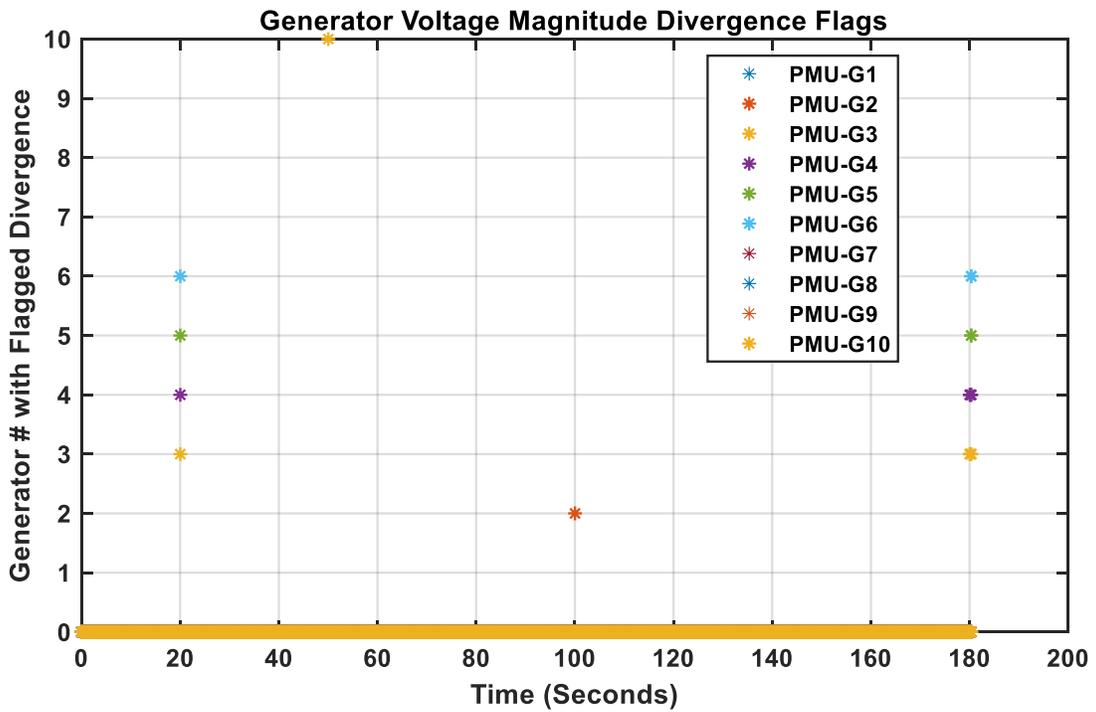


**Figure 5.5. Flags for Generator Voltage Magnitude Divergence**

Since the first three faults have a duration of 3 cycles and the PMUs are configured to a 30Hz report rate, a maximum of 2 measurements could contain the fault signature at 20, 50, and 100 seconds. In these cases, voltage instability was not identified for the initial measurement, as the faulted conditions were not yet reflected in the PMU signal. However, a flag for voltage instability was detected for at least one generator in the following measurement for all faults before they were cleared. **Figure 5.6** shows the detection of divergent voltage magnitudes for the PMUs located at generators 3, 4, 5, and 6 at 20.033 seconds. In the next measurement at 20.067 seconds, where the fault has already been cleared, divergence in the voltage magnitudes derived from the SMJA and SMPFJ matrices is no longer present.
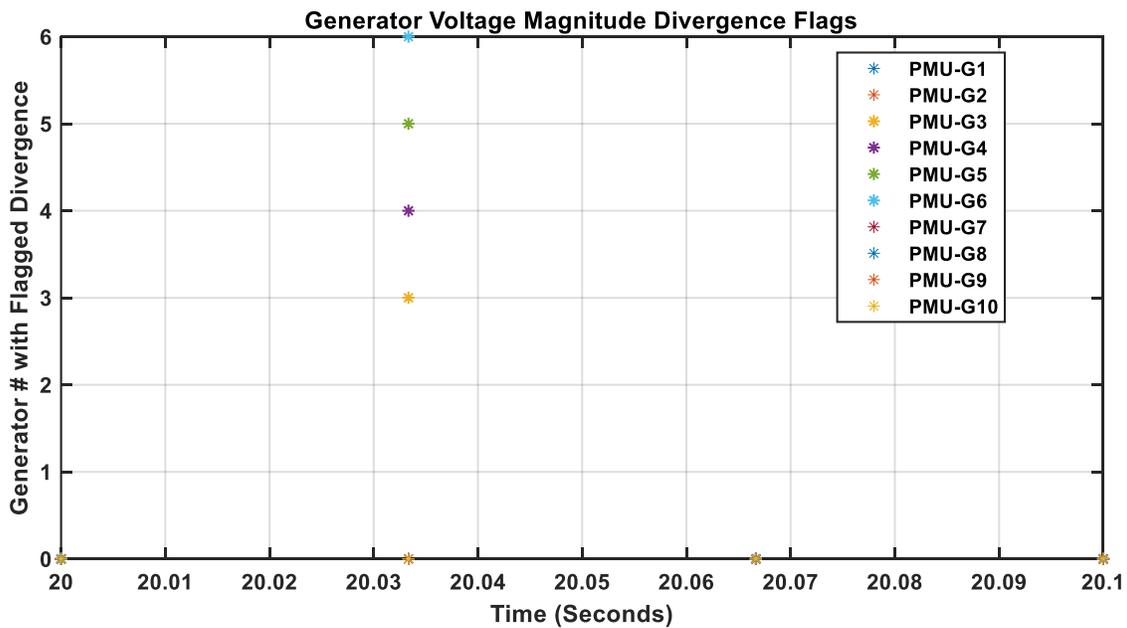


**Figure 5.6. Voltage Mag. Divergence Flags – Demonstration of Flags Resolving after Fault is Cleared**

**Figure 5.7** illustrates the voltage divergence when comparing values derived from the SMJA and SMPFJ matrices during the fault applied after 100 seconds for the PMU at generator 2. This is an underlying component to determining the flags in **Equation 130**.
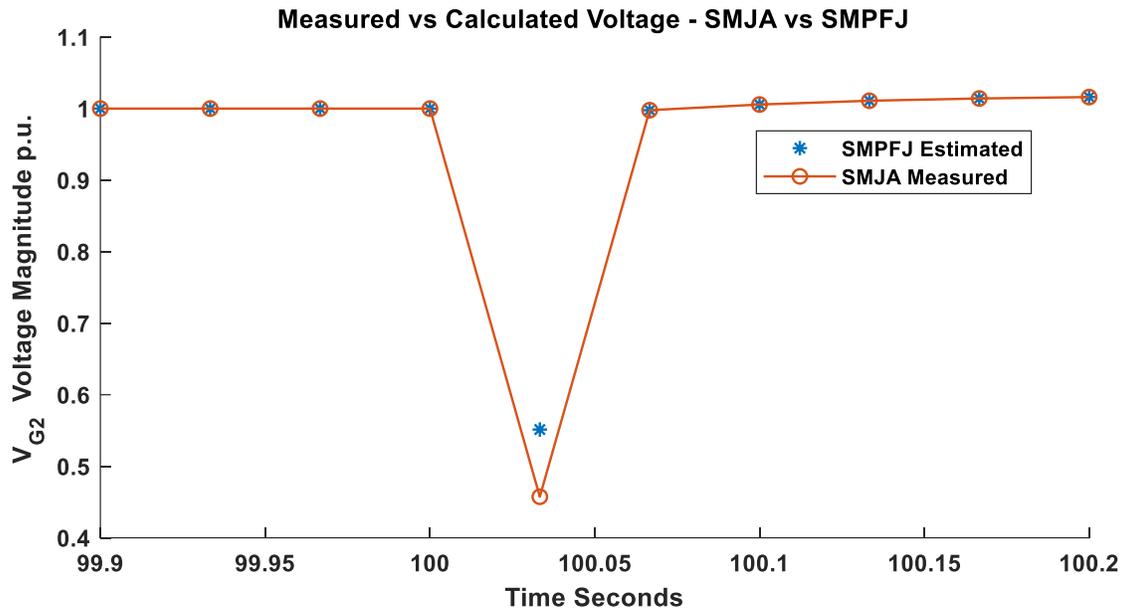


**Figure 5.7. Voltage Mag. Divergence – Demonstration of SMJA and SMPFJ Divergence During Unstable Parameters**

In **Figure 5.8**, divergence in generator bus voltage magnitude parameters can be seen within one PMU measurement following a constant fault applied at Bus 19 for the PMUs at generators 3 and 4. After 8 more consecutive measurements (16 cycles), voltage instability is also detected at generators 5 and 6. These flags are constant following initial identification at all four generators until the system collapses.
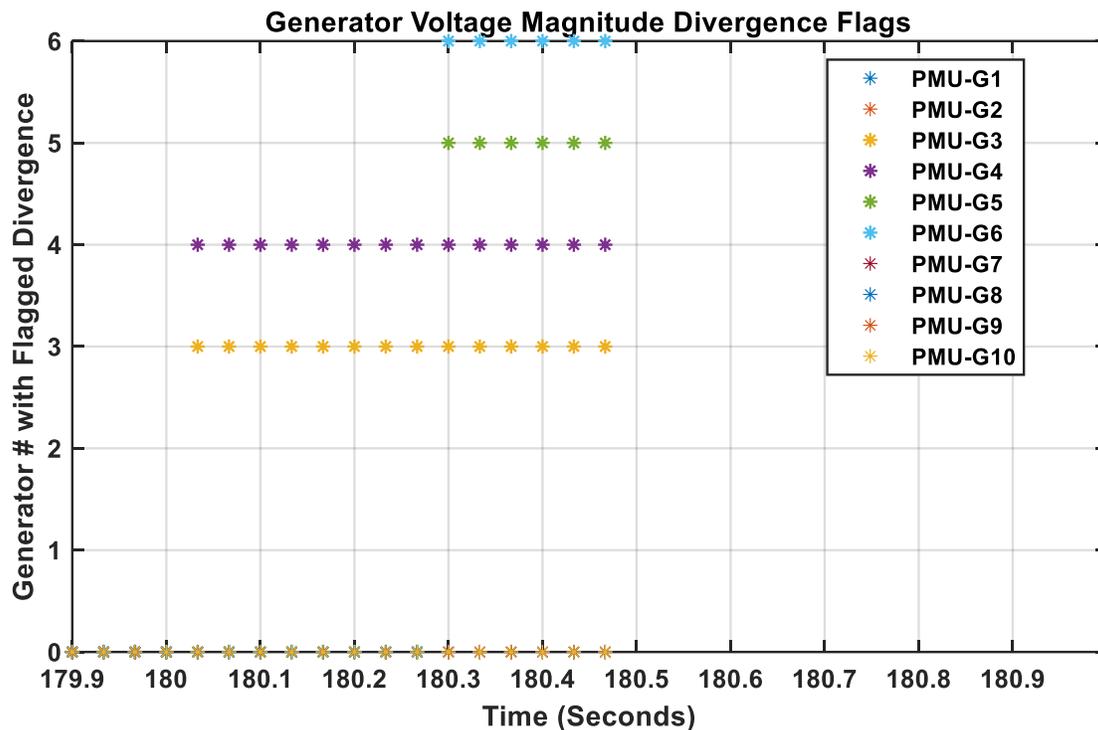
**Figure 5.8. Continuous Voltage Magnitude Divergence Flags Until System Instability**

In addition to flags for voltage magnitude, flags for divergence in generator bus voltage angle are critical to determine a system health. These flags may occur at different times or independent of each other, subject to the system conditions. Divergence in the generator bus voltage angles is not detected for the fault at 100 seconds. Angular instability is detected for the PMU located at generator 10 during the fault at 50 seconds and the PMUs located at generators 3, 4, 5, and 6 during the temporary fault at 20 seconds and constant fault at 180 seconds. These characteristics can be seen at a high level in **Figure 5.9**, and **Figure 5.10** shows higher resolution of the flags during the constant fault leading to instability.
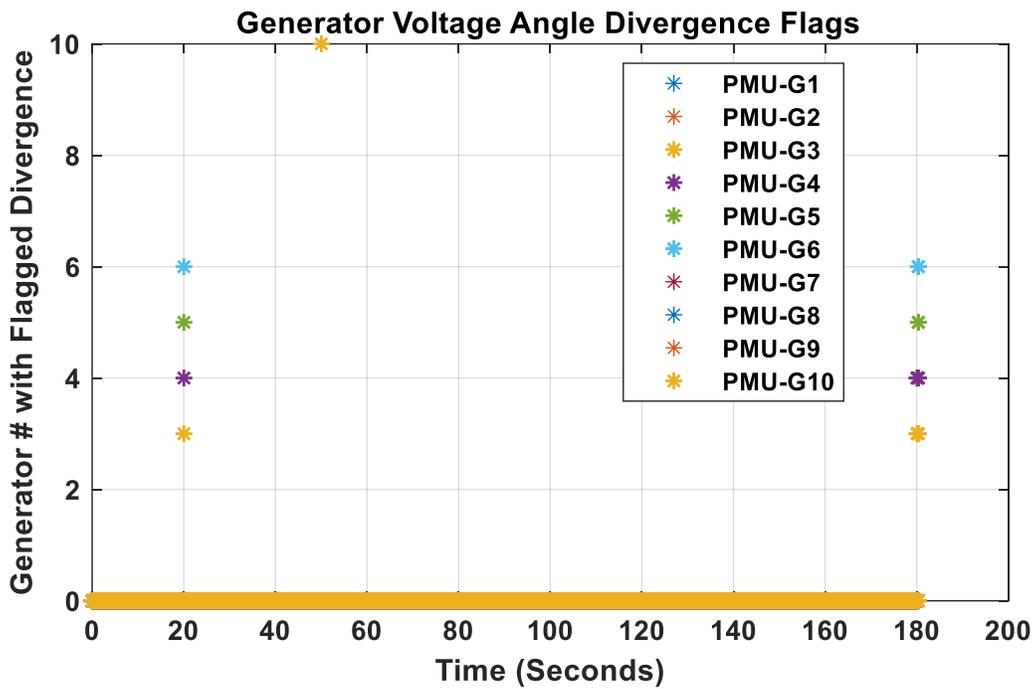
**Figure 5.9. Flags for Generator Voltage Angle Divergence**
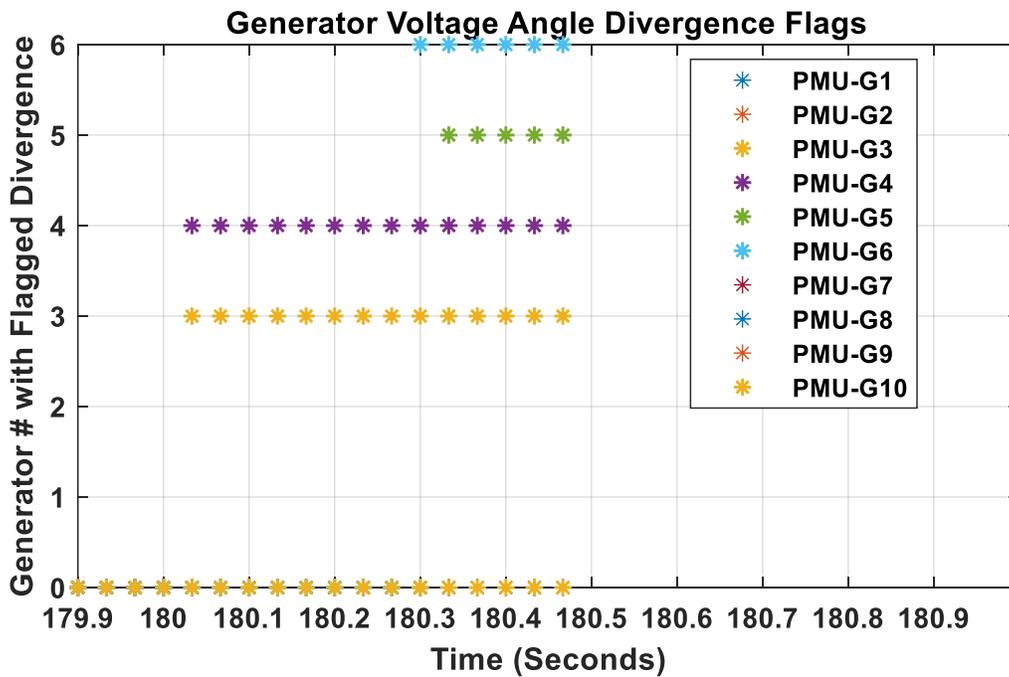


**Figure 5.10. Generator Bus Voltage Angle Divergence Flags During Continuous Fault Condition Until System Instability**

110

The divergence in bus voltage angle at generator 5, comparing bus voltage angles used to construct the SMJA and SMPFJ matrices, is displayed in **Figure 5.11** for the first fault event. As SMPFJ is calculated with the assumption of infinite inertia, the calculated model converged to a more ideal state than the system realized. After the fault clears, the divergence between the calculated and measured parameters immediately resolves.
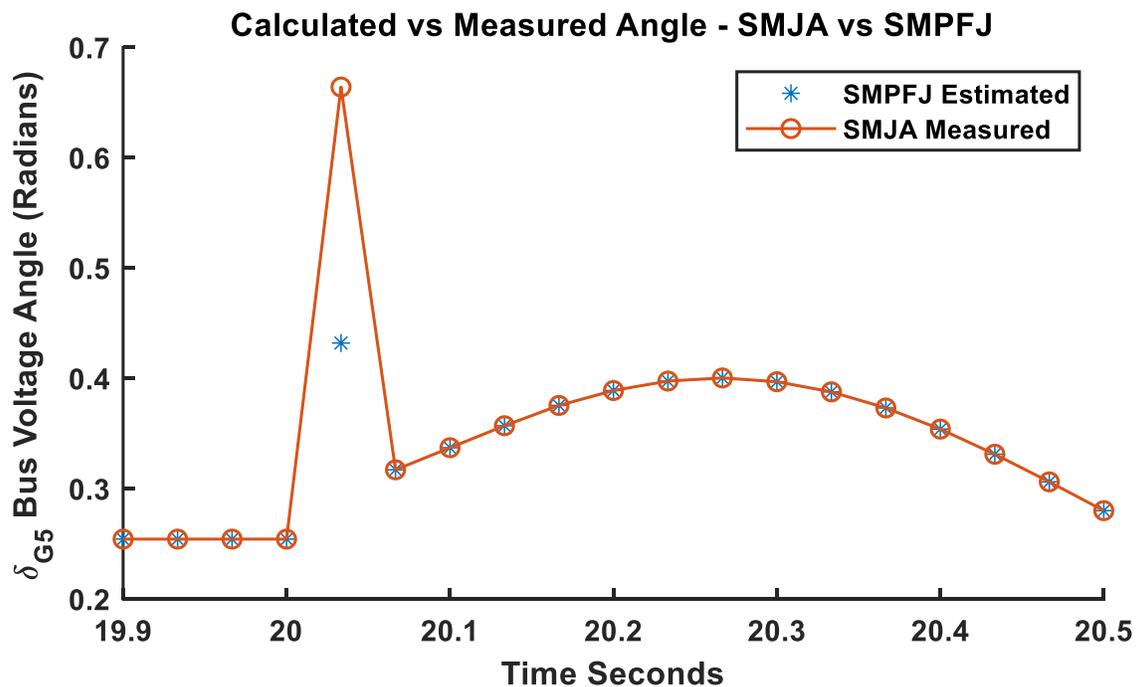


**Figure 5.11. Generator Bus Voltage Angle Divergence – Demonstration of SMJA and SMPFJ Divergence During Unstable Parameters**

Flags for voltage and angular instability can be used jointly to identify the impacted generator(s) and determine the nature of the unstable conditions. For all faults, unstable conditions were detected and flagged within one measurement of the event. For faults that were cleared, flags were not seen in the next measurement. With respect to the introduction of a constant fault, flags were persistent, once detected, until system collapse occurred.

The mean and median computational speed for the construction of the MSSM $Y_{Bus}$ and associated parameters, SMJA matrix with SVD analysis, and calculation of the SMPFJ matrix with analysis for divergence against the SMJA matrix are shown in **Table 5.2**. Note that these calculations were made with an i9-12900HK 2.50GHz processor. Coding in Matlab was implemented to capture the computation times of the individual processes. **Table 5.3** shows the accuracy of using the SMPFJ as a state estimator considering three contingencies: MSSM constructed from the most recent measurement, MSSM constructed with measurements delayed 12 cycles, MSSM constructed with measurements delayed 60 cycles. The loading is still derived from the present measurement. The MSSM with no delay would ideally be used for actual state estimation applications and comparisons, but the mean and median absolute errors using unideal or delayed data show the viability of the method to perform state estimation with excellent precision considering old or imperfect data.

**Table 5.2. Computational Efficiency- Case Study 5.4.1**

| Computed Process | Mean Computation Time (S) | Median Computation Time (S) |
|---|---|---|
| MSSM $Y_{Bus}$ Generation | $1.22 \times 10^{-5}$ | $1.15 \times 10^{-5}$ |
| SMJA Calculation and SVD Analysis | $2.61 \times 10^{-5}$ | $2.55 \times 10^{-5}$ |
| SMPFJ Calculation and Analysis | $1.81 \times 10^{-4}$ | $1.79 \times 10^{-4}$ |

**Table 5.3. SMPFJ State Estimation Accuracy: Case Study 5.4.1**

| Measurement – | MSSM Parameter | Absolute Mean Percent Error (%) | Absolute Median Percent Error (%) |
|---|---|---|---|
| Estimated Voltage Magnitude Error | No MSSM Delay | 0.0574 | $1.68 \times 10^{-8}$ |
| | 12 cycle MSSM Delay | 0.3864 | $6.46 \times 10^{-5}$ |
| | 60 cycle MSSM Delay | 0.6428 | $1.394 \times 10^{-4}$ |
| Estimated Bus Voltage Angle Error | No MSSM Delay | 0.0693 | $5.805 \times 10^{-8}$ |
| | 12 cycle MSSM Delay | 0.5060 | $1.228 \times 10^{-4}$ |
| | 60 cycle MSSM Delay | 0.7868 | $2.401 \times 10^{-4}$ |

These simulated case studies contain a high density of serious system events, which are expected to reflect error in the values derived from the SMPFJ matrix. Considering a delayed MSSM model will further introduce a source of error. The median error in **Table 5.3** is more appropriate when considering the expected accuracy for the majority of measurements, as state estimation performed with this methodology will yield noticeable error on the rare conditions that unstable parameters are present. Despite the abnormally frequent major events simulated, the state estimation accuracy is very high precision over the simulation. Case studies in *Section 5.5*, which are run on streamed utility PMU data, are a proper reflection of the expected accuracy when this methodology is applied to real system data containing less frequent major system events but having more short-term volatility of the signal. The results in **Table 5.2** show that the methodology is computationally efficient and only required 1/76th of a cycle to resolve all processes for the

simulated case. **Table 5.3** demonstrates that the accuracy of the method is acceptable as a state estimator, considering both present and delayed data in the implementation.

## 5.4.2   *Instability Due to Load*

Case Study 5.4.2 focuses on the detection of unstable load conditions. The initial fault from Case Study 5.4.1 was kept in the model, but all other faults were removed. A load at Bus 27 was incrementally increased until system instability occurred. The events are summarized in **Table 5.4**. The loads were deliberately incremented by large values to encourage system instability within a reasonable number of updates.

**Table 5.4. Sequence of Events for Case Study 5.4.2**

| Event | Location | Start Time of Event (S) | End Time of Event (S) |
|---|---|---|---|
| Fault | Bus 16 | 20 | 20.05 |
| Load Increase: 3000e6/800e6 MW/MVAR | Bus 27 | 200 | N/A (Continuous) |
| Load Increase: 4000e6/1000e6 MW/MVAR | Bus 27 | 240 | N/A (Continuous) |
| Load Increase: 6000e6/1500e6 MW/MVAR | Bus 27 | 300 | N/A (Continuous) |

**Figure 5.12** visualizes the lowest 10 sigmas relating to the SMJA matrix, with the zoom window in the sub-plot showing a higher resolution snapshot during the final load increase that caused system instability. It can be noted that a near-singularity is observed at the same measurement where the SMJA and SMPFJ show critical divergence.
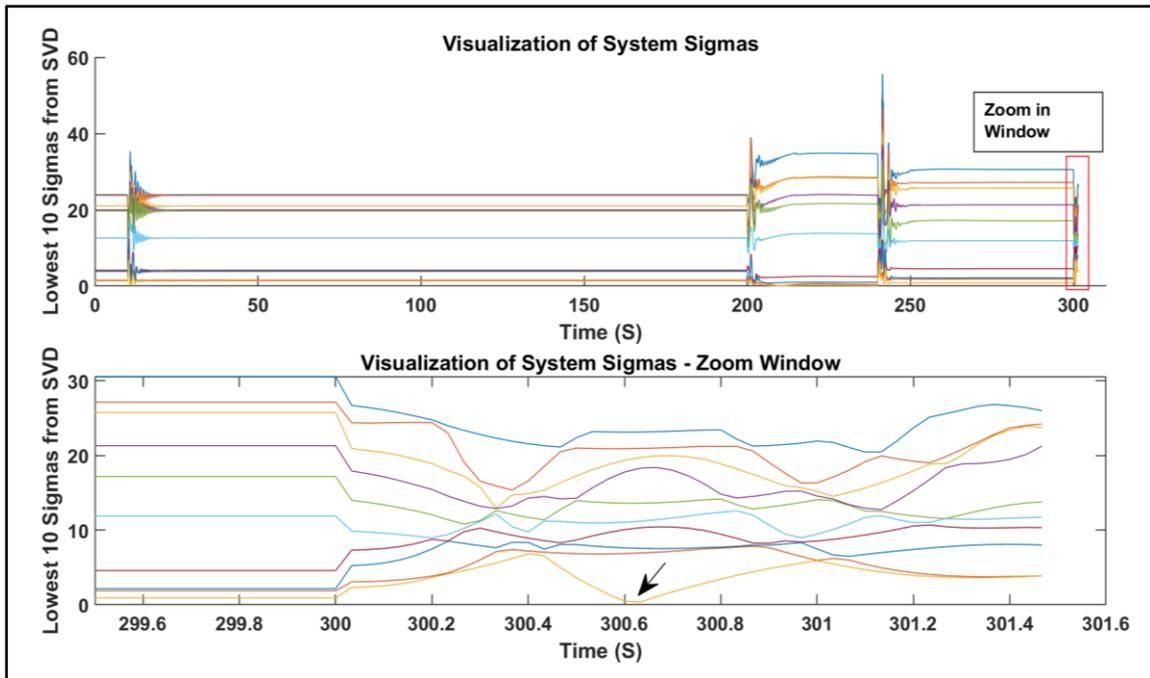
**Figure 5.12. Visualization of the Generators' Lowest 10 SVD Sigmas During Case Study 5.4.2 Faulted Conditions and Increasing Load (With Zoom Window)**

**Figure 5.13** shows the flags for voltage instability over the simulation window. **Figure 5.14** and **Figure 5.15** show higher resolution snapshots of the flags for voltage magnitude and angular instability, respectively, during the unstable load condition.
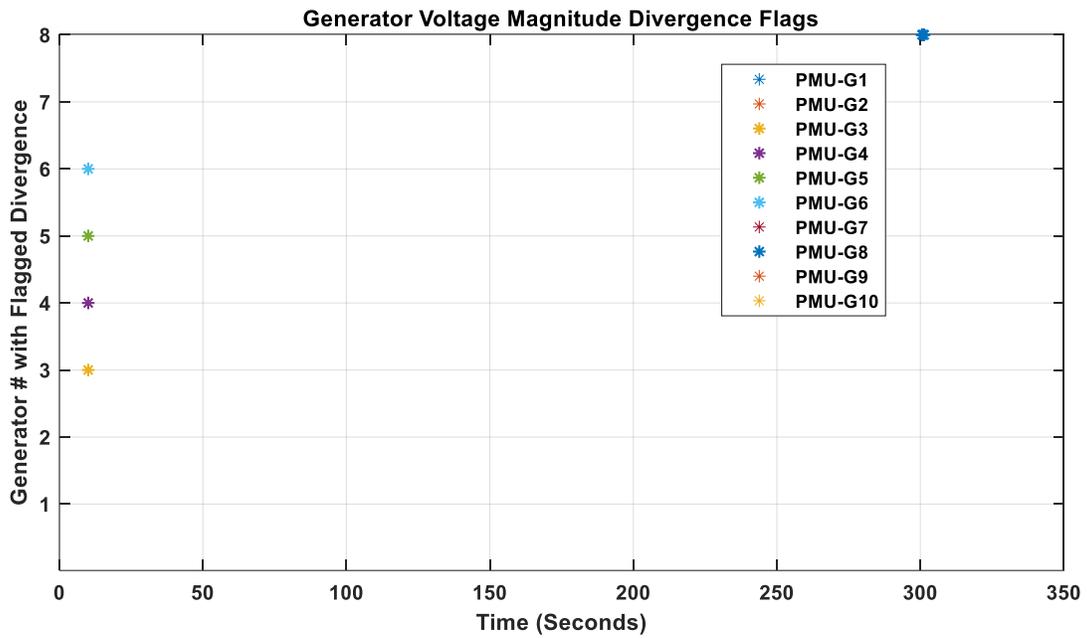
**Figure 5.13. Unstable Load – Flags for Generator Voltage Magnitude Divergence**
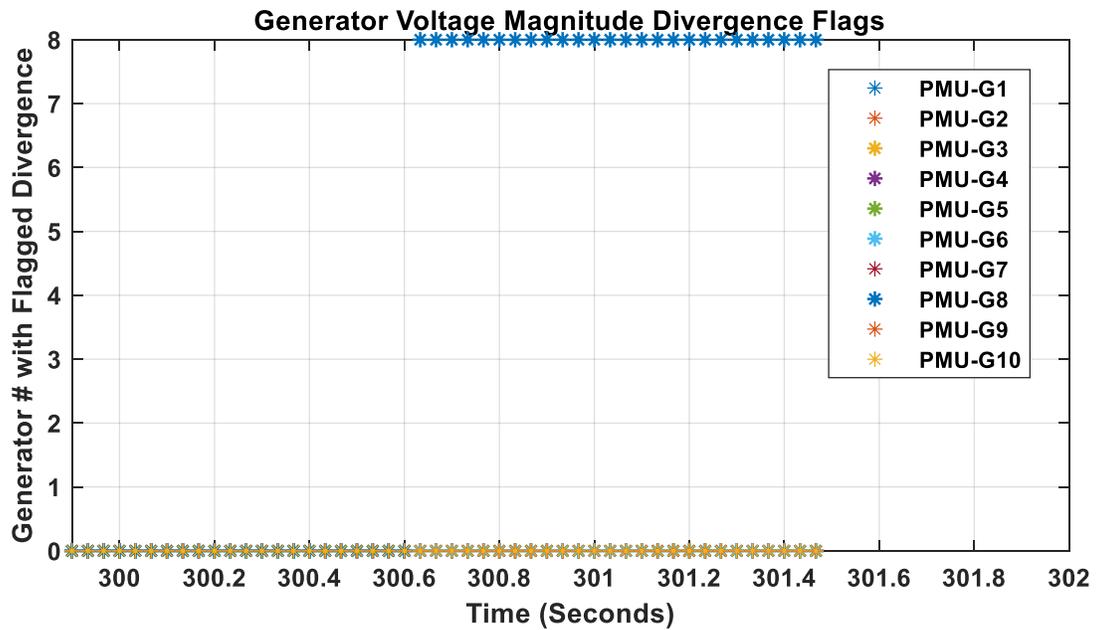


**Figure 5.14. Unstable Load – Generator Voltage Magnitude Divergence Flags During Instability**

116
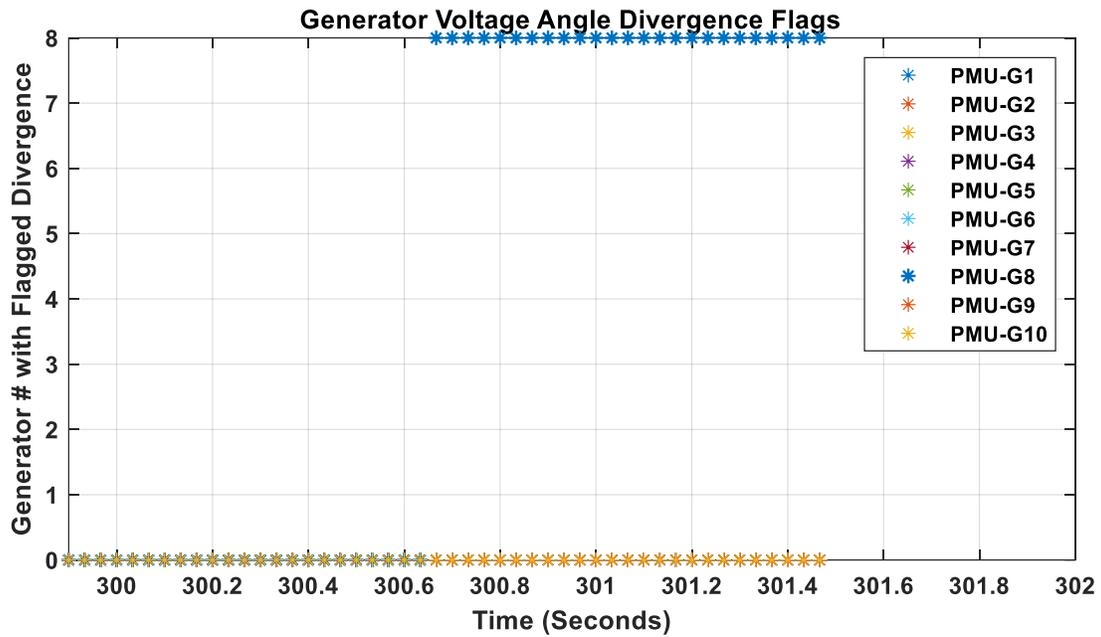
**Figure 5.15. Unstable Load – Generator Bus Voltage Angle Divergence Flags During Instability**

Both voltage and angular instability flags are detected at nearly the same point in time shortly after the unstable load event, with **Figure 5.14** showing a critical flag one measurement sooner in the voltage magnitude divergence. A disturbance in the sigmas can be detected at 300 seconds and correlated with the load increase. The resulting divergence in the SMJA and SMPFJ matrices can be directly correlated to this event.

**Figure 5.16** shows the development of angular instability, with the measured bus voltage angle for the PMU observing generator 8 diverging until angular instability occurs, consistent with the system weak point identification shown in the prior two figures.
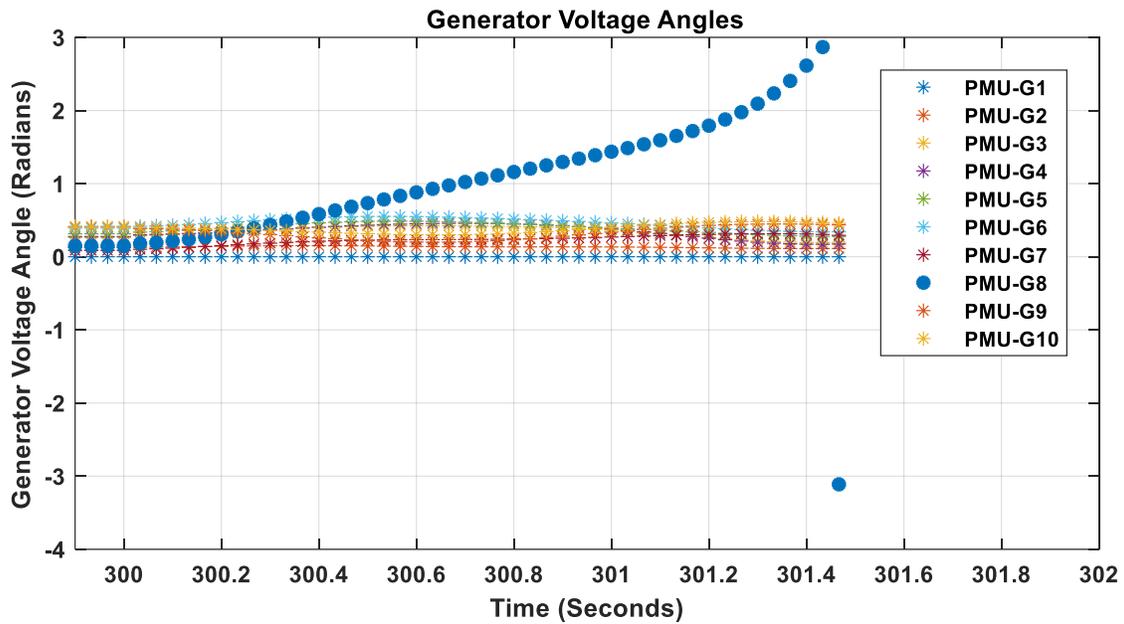
**Figure 5.16. Unstable Load – Generator Voltage Angle During Instability**

The computational speed and state estimation accuracy for the unstable load simulation data are shown in **Table 5.5** and **Table 5.6**.

**Table 5.5. Computational Efficiency- Case Study 5.4.2**

| Computed Process | Mean Computation Time (S) | Median Computation Time (S) |
|---|---|---|
| MSSM $Y_{Bus}$ Generation | $9.66 \times 10^{-6}$ | $9.40 \times 10^{-6}$ |
| SMJA Calculation and SVD Analysis | $2.31 \times 10^{-5}$ | $2.27 \times 10^{-5}$ |
| SMPFJ Calculation and Analysis | $1.59 \times 10^{-4}$ | $1.64 \times 10^{-4}$ |

**Table 5.6. SMPFJ State Estimation Accuracy: Case Study 5.4.2**

| Measurement – | MSSM Parameter | Absolute Mean Percent Error (%) | Absolute Median Percent Error (%) |
|---|---|---|---|
| Estimated Voltage Magnitude Error | No MSSM Delay | 0.0195 | $1.68 \times 10^{-7}$ |
| | 12 cycle MSSM Delay | 0.4050 | $9.09 \times 10^{-6}$ |
| | 60 cycle MSSM Delay | 0.7678 | $3.74 \times 10^{-5}$ |
| Estimated Bus Voltage Angle Error | No MSSM Delay | 0.0233 | $5.804 \times 10^{-8}$ |
| | 12 cycle MSSM Delay | 0.4777 | $1.110 \times 10^{-5}$ |
| | 60 cycle MSSM Delay | 0.8776 | $4.66 \times 10^{-5}$ |

### 5.4.3   *Mitigation of Instability Due to Load*

Case Study 5.4.3 explores a process to mitigate the unstable load conditions in *Section 5.4.2*. The methodology to detect and mitigate unstable load conditions is detailed in **Figure 5.2** and **Figure 5.3**. Considering the flags for bus voltage magnitude and phase angle, the command to drop the unstable load condition is issued at 300.633 seconds. Assuming a delay for the breaker to open, load shedding is implemented at 300.75 seconds. This also shows that a delay in implementing a mitigating action derived from this methodology can still result in a stable system. **Table 5.7** summarizes the sequence of events for the simulation.

**Table 5.7. Sequence of Events for Case Study 5.4.3**

| Event | Location | Start Time of Event (S) | End Time of Event (S) |
|---|---|---|---|
| Fault | Bus 16 | 20 | 20.05 |
| Load Increase: 3000e6/800e6 MW/MVAR | Bus 27 | 200 | N/A (Continuous) |
| Load Increase: 4000e6/1000e6 MW/MVAR | Bus 27 | 240 | N/A (Continuous) |
| Load Increase: 6000e6/1500e6 MW/MVAR | Bus 27 | 300 | (*)300.75 |
| Mitigation Action: Load Shedding | Bus 27 | (*)300.75 | N/A |

**Figure 5.17** visualizes the ten lowest system sigmas during the simulation. After load shedding occurs, it can be observed that the system sigmas are able to return to approximately the same values as the prior system state before the unstable load condition was introduced.
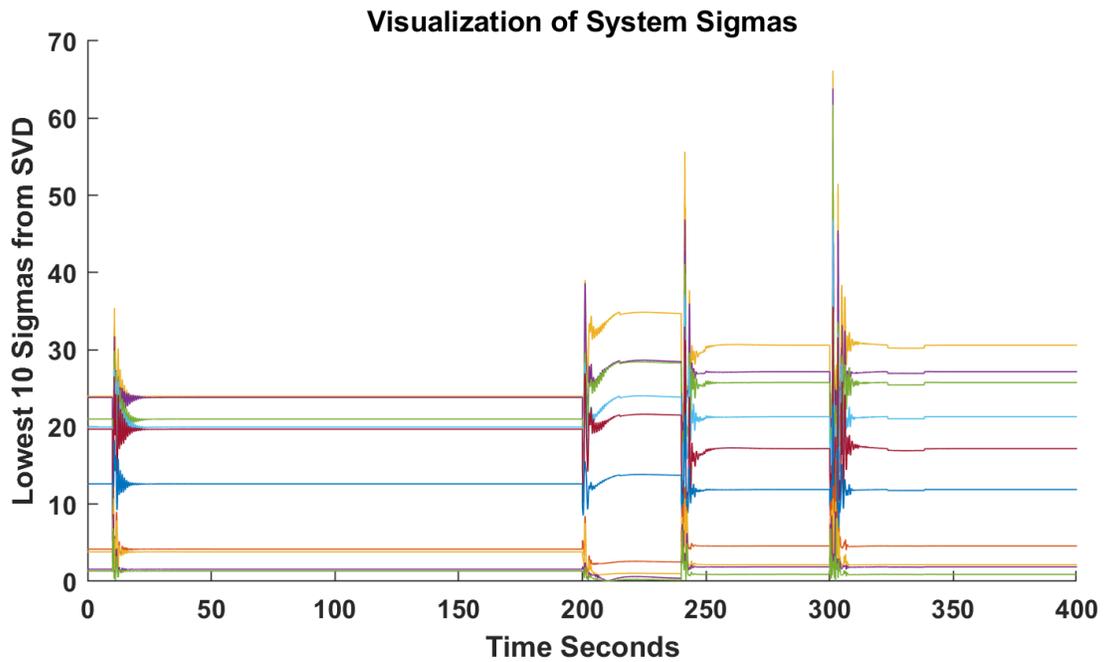
**Figure 5.17. Visualization of the Generators' Lowest 10 SVD Sigmas During Case Study 5.4.3 –
Unstable Load Mitigation**

**Figure 5.18** visualizes that the flags for angular instability clear shortly after the unstable load is removed from the model, allowing the system to reach a stable state. The measured bus voltage angle for the PMU at generator 8 can be seen in **Figure 5.19** converging to a stable solution after load shedding is implemented.
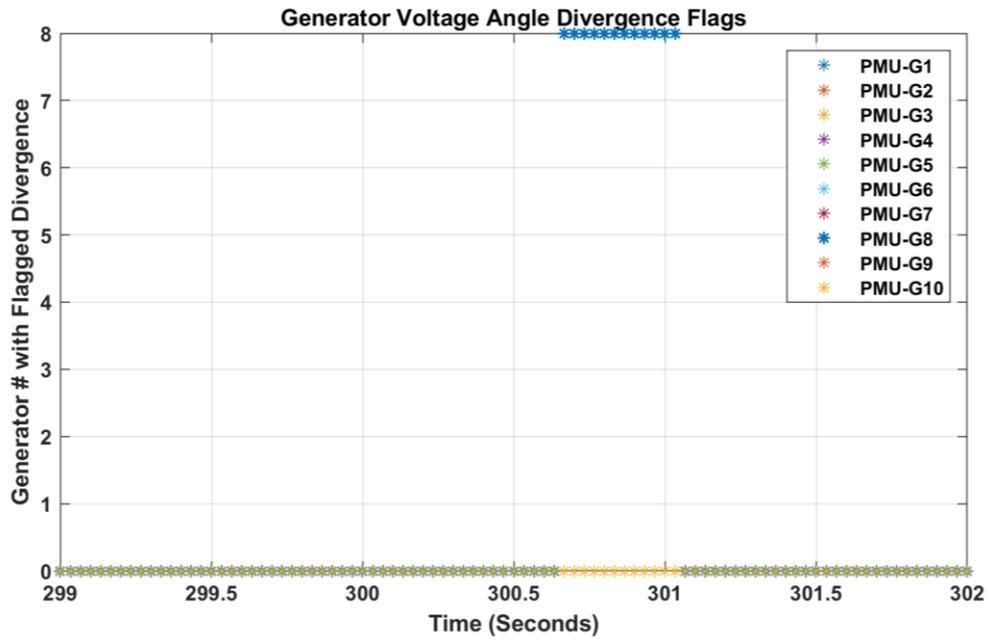
**Figure 5.18. Stable Case – Generator Voltage Angle Divergence Flags**
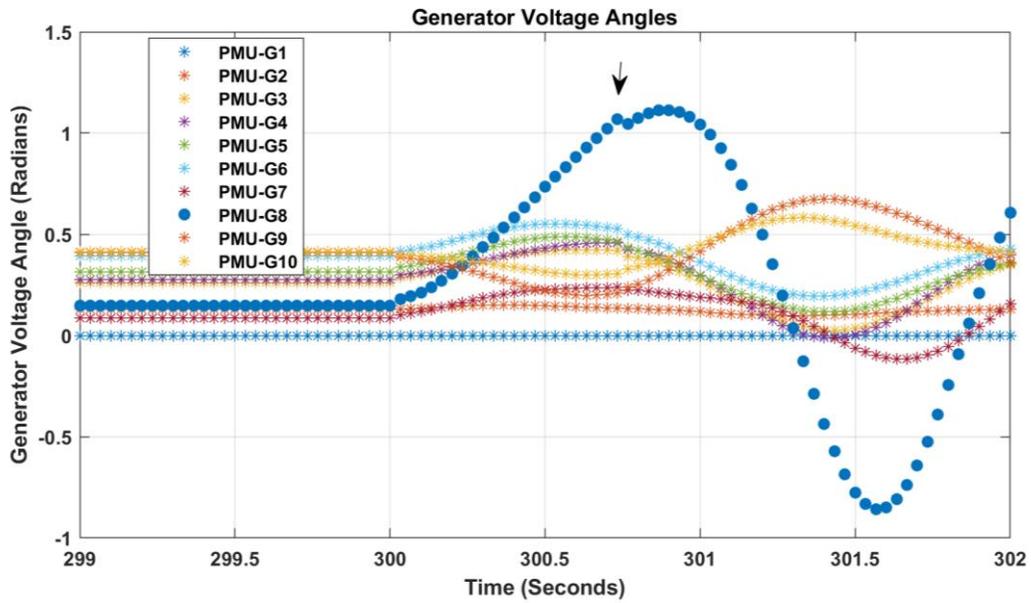


**Figure 5.19. Stable Case – Generator Voltage Before and After Load Shedding of Unstable Load Condition**

This case study demonstrates that the MSSM, SMJA, and SMPFJ can be utilized with singular value decomposition and comparison of the Jacobian variants to determine the cause of instability, identify system weak points, and implement appropriate actions to mitigate unstable conditions. The computational speed and state estimation accuracy for case study 5.4.3 data are shown in **Table 5.8** and **Table 5.9**.

**Table 5.8. Computational Efficiency- Case Study 5.4.3**

| Computed Process | Mean Computation Time (S) | Median Computation Time (S) |
|---|---|---|
| MSSM $Y_{Bus}$ Generation | $9.93 \times 10^{-6}$ | $9.50 \times 10^{-6}$ |
| SMJA Calculation and SVD Analysis | $2.38 \times 10^{-5}$ | $2.33 \times 10^{-5}$ |
| SMPFJ Calculation and Analysis | $1.59 \times 10^{-4}$ | $1.67 \times 10^{-4}$ |

**Table 5.9. SMPFJ State Estimation Accuracy: Case Study 5.4.3**

| Measurement | MSSM Parameter | Absolute Mean Percent Error (%) | Absolute Median Percent Error (%) |
|---|---|---|---|
| Estimated Voltage Magnitude Error | No MSSM Delay | 0.0036 | $1.68 \times 10^{-7}$ |
| | 12 cycle MSSM Delay | 0.4367 | $5.09 \times 10^{-5}$ |
| | 60 cycle MSSM Delay | 0.7528 | $2.40 \times 10^{-4}$ |
| Estimated Bus Voltage Angle Error | No MSSM Delay | 0.0030 | $5.860 \times 10^{-8}$ |
| | 12 cycle MSSM Delay | 0.5157 | $1.680 \times 10^{-5}$ |
| | 60 cycle MSSM Delay | 0.8653 | $7.10 \times 10^{-5}$ |

**5.5    SMJA and SMPFJ Case Study Validation Using Utility PMU Data**

Streamed PMU data from a large utility system was considered to assess the methodology's accuracy and computational efficiency with respect to real PMU data streams and data that originates from a larger system. Fifty PMUs from the transmission system were selected for analysis of approximate equivalent machine models. These fifty PMUs produced positive real power over the entire timeframe that the data stream was sampled, making them more appropriate mathematically to test performance of higher machine models than were simulated, shown in **Figure 5.20** with 100MVA as the base value. This dataset is considered computationally equivalent to 50 machines from a large system, showing the potential of this methodology to handle larger systems and more complex data. A sub-set of 20 PMUs from these 50 PMUs is processed to show the methodology's flexibility to perform analysis and state estimation when considering a reduced number of available measurements. These two datasets are considered in the first two studies.
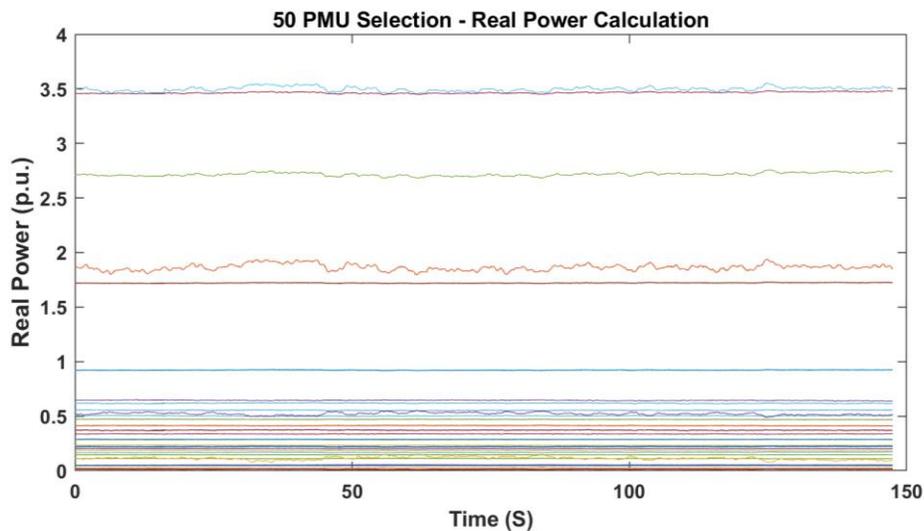


**Figure 5.20. Real Power Signals for Utility 50 PMU Dataset**

In order to demonstrate the algorithm's general flexibility and scalability, all available PMU measurements were considered in the final use case. Of the 124 total available PMU data streams, 112 PMU streams were selected for further processing, as 12 devices had a combination of sustained time synchronization issues or significant data dropouts. An algorithm to relate redundant measurements, such as two devices on each side of a transformer, is also presented, and a process to modify the MSSM construction is shown. Although the utility dataset does not include unstable parameters, the accuracy of calculated system state estimation is assessed considering the most recent and historical MSSM models. Computational efficiency is also assessed.

### 5.5.1   *20-Machine Equivalent Utility Dataset Analysis*

This first case study analyzes the 20 PMU measurement sub-set of the total 50 PMU measurements selected for positive real power delivery. This demonstrates the flexibility of this method to perform state estimation and stability analysis on a sub-set of available measurements, as well as scalability of the solution as more generators or buses are incorporated into the mathematical model. For simplicity, all PMUs will be referred to in *Sections 5.5.1* and *5.5.2* as monitoring or observing generator's 1 through 50.

For all calculations and measurements in this section, the resulting SMJA and SMJAC matrices were approximately identical, resulting in no flags for any PMU generator voltage magnitude or generator bus voltage angle. **Figure 5.21** shows the bus voltage magnitude estimate and measurement for generator 9, considering a 12-cycle delay in the MSSM. **Figure 5.22** shows the estimate and measurement for the bus voltage angle

at generator 9, considering the same delay in the MSSM. The computational speed and

state estimation accuracy for Section 5.5.1 data are shown in **Table 5.10** and **Table 5.11**.
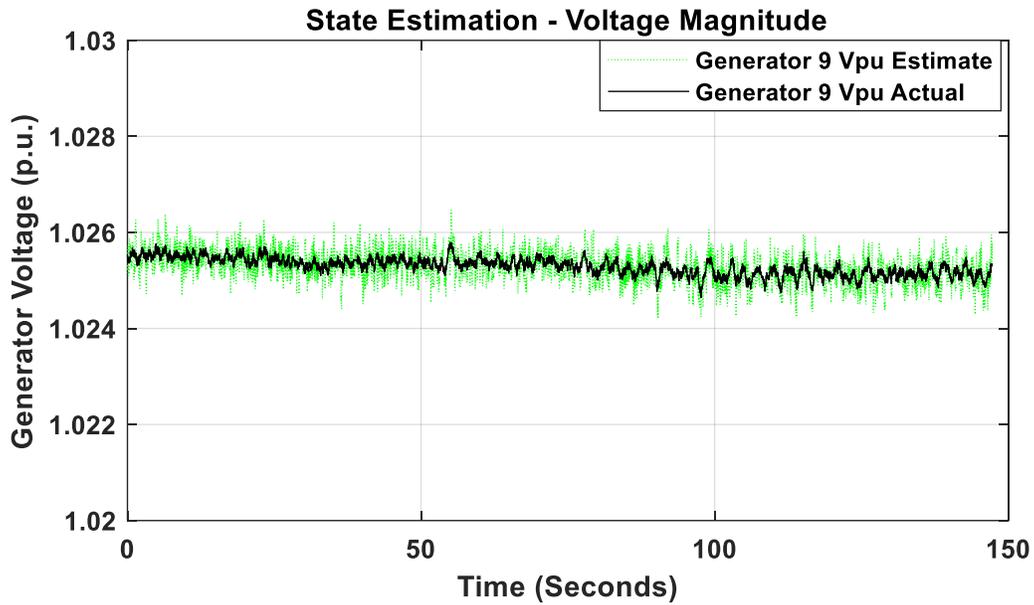


**Figure 5.21. Generator 9 PMU Bus Voltage Magnitude – Estimate vs. Measurement Considering 12-Cycle Delay**
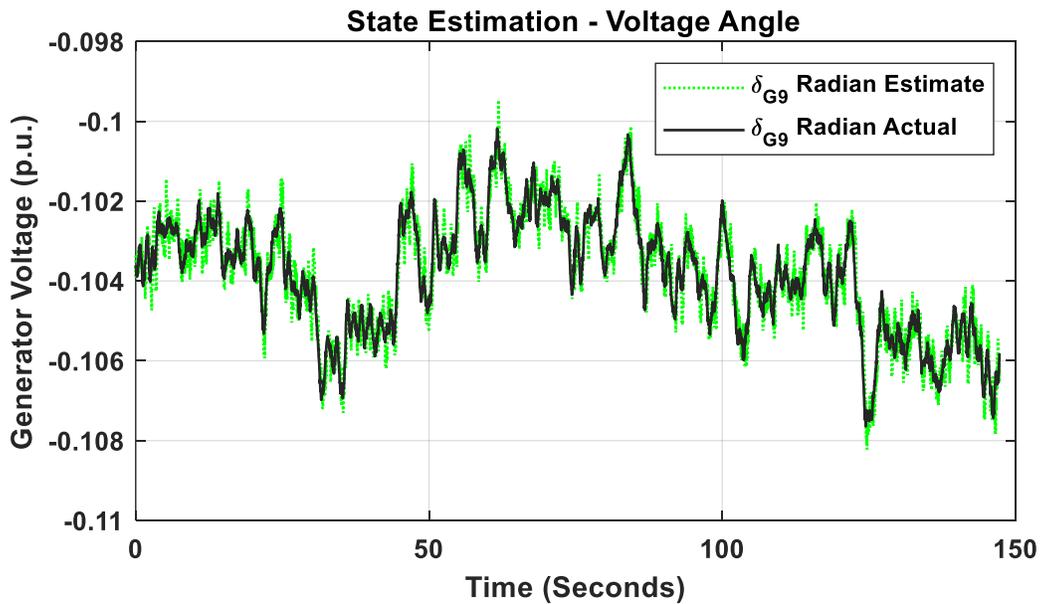


**Figure 5.22. Generator 9 PMU Bus Voltage Angle– Estimate vs. Measurement Considering 12-Cycle Delay**

**Table 5.10. Computational Efficiency- Case Study 5.5.1**

| Computed Process | Mean Computation Time (S) | Median Computation Time (S) |
|---|---|---|
| MSSM $Y_{Bus}$ Generation | $1.48 \times 10^{-5}$ | $1.40 \times 10^{-5}$ |
| SMJA Calculation and SVD Analysis | $3.51 \times 10^{-5}$ | $3.4 \times 10^{-5}$ |
| SMPFJ Calculation and Analysis | $2.36 \times 10^{-4}$ | $2.34 \times 10^{-4}$ |

**Table 5.11. SMPFJ State Estimation Accuracy: Case Study 5.5.1**

| Measurement | MSSM Parameter | Absolute Mean Percent Error (%) | Absolute Median Percent Error (%) |
|---|---|---|---|
| Estimated Voltage Magnitude Error | No MSSM Delay | $5.97 \times 10^{-6}$ | $5.92 \times 10^{-6}$ |
| | 12 cycle MSSM Delay | 0.0511 | 0.0425 |
| | 60 cycle MSSM Delay | 0.0722 | 0.0609 |
| Estimated Bus Voltage Angle Error | No MSSM Delay | $4.62 \times 10^{-6}$ | $4.56 \times 10^{-6}$ |
| | 12 cycle MSSM Delay | 0.0376 | 0.0313 |
| | 60 cycle MSSM Delay | 0.0648 | 0.0538 |

*5.5.2   50-Machine Equivalent Utility Dataset Analysis*

The analysis of this section considers the entire set of 50 PMUs selected from the utility data to test computational speed and accuracy of a 50-machine equivalent model. There was no divergence in the SMJA and SMPFJ to flag unstable conditions detected for any of the 50 PMU devices. The computational speed and state estimation accuracy for

*Section 5.5.2* data are shown in **Table 5.12** and **Table 5.13**. The lowest 10 sigmas generated by applying SVD to the SMJA is shown in **Figure 5.23** to demonstrate output for real system data. Conclusions from this data will be discussed at the end of *Section 5.5.3*.
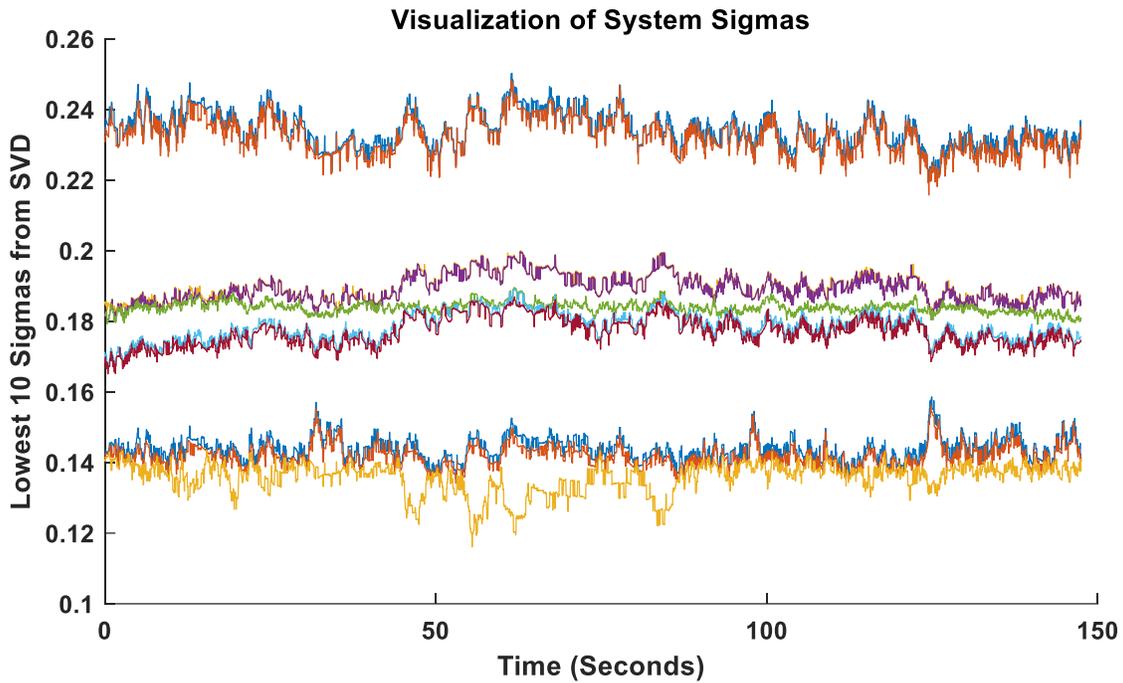


**Figure 5.23. Visualization of Lowest 10 Sigmas from SVD – 50-Machine Equivalent**

**Table 5.12. Computational Efficiency- Case Study 5.5.2**

| Computed Process | Mean Computation Time (S) | Median Computation Time (S) |
|---|---|---|
| MSSM $Y_{Bus}$ Generation | $3.25 \times 10^{-5}$ | $3.06 \times 10^{-5}$ |
| SMJA Calculation and SVD Analysis | $1.01 \times 10^{-4}$ | $9.72 \times 10^{-5}$ |
| SMPFJ Calculation and Analysis | $7.96 \times 10^{-4}$ | $7.52 \times 10^{-4}$ |

128

**Table 5.13. SMPFJ State Estimation Accuracy: Case Study 5.5.2**

| Measurement – | MSSM Parameter | Absolute Mean Percent Error (%) | Absolute Median Percent Error (%) |
|---|---|---|---|
| Estimated Voltage Magnitude Error | No MSSM Delay | $1.47 \times 10^{-4}$ | $2.13 \times 10^{-4}$ |
| | 12 cycle MSSM Delay | 0.0609 | 0.0505 |
| | 60 cycle MSSM Delay | 0.0953 | 0.0801 |
| Estimated Bus Voltage Angle Error | No MSSM Delay | $1.13 \times 10^{-4}$ | $1.57 \times 10^{-4}$ |
| | 12 cycle MSSM Delay | 0.0413 | 0.0343 |
| | 60 cycle MSSM Delay | 0.0737 | 0.0612 |

### 5.5.3   Analysis of 112 PMU Utility Dataset

The final case study of this chapter analyzes data from 112 utility PMU devices. The generation of the initial MSSM considers all PMUs to be bus PMUs modelled on individual branches. **Table 5.14** and **Table 5.15** show the computational speed and accuracy when considering each PMU as a bus PMU in the derivation.

**Table 5.14. Computational Efficiency- Case Study 5.5.3**

| Computed Process | Mean Computation Time (S) | Median Computation Time (S) |
|---|---|---|
| MSSM $Y_{Bus}$ Generation | $7.595 \times 10^{-5}$ | $7.470 \times 10^{-5}$ |
| SMJA Calculation and SVD Analysis | $2.349 \times 10^{-4}$ | $2.291 \times 10^{-4}$ |
| SMPFJ Calculation and Analysis | 0.0023 | 0.0022 |

**Table 5.15. SMPFJ State Estimation Accuracy: Case Study 5.5.3**

| Measurement | MSSM Parameter | Absolute Mean Percent Error (%) | Absolute Median Percent Error (%) |
|---|---|---|---|
| Estimated Voltage Magnitude Error | No MSSM Delay | $5.84 \times 10^{-4}$ | $2.43 \times 10^{-4}$ |
| | 12 cycle MSSM Delay | 0.0637 | 0.0455 |
| | 60 cycle MSSM Delay | 0.1040 | 0.0708 |
| Estimated Bus Voltage Angle Error | No MSSM Delay | $6.39 \times 10^{-4}$ | $3.07 \times 10^{-4}$ |
| | 12 cycle MSSM Delay | 0.0468 | 0.0351 |
| | 60 cycle MSSM Delay | 0.0865 | 0.0647 |

In utility data, PMUs that are effectively redundant are often installed at key points in the system. An example of this is often installation of PMUs on both sides of a critical transformer. If not provided, these redundant devices can be mathematically determined by analyzing the signals. Given real-world implementation, sources of error can be introduced by various factors, such as the classification and accuracy of instrument transformers used by devices supporting PMU functionality to measure the voltage and current magnitudes [80]. It is very possible and likely that two PMUs on opposite sides of a transmission line will see a difference in the current magnitude and phase angle if they are connected to CTs with about a 1% error. If PMUs utilize protection class CTs, the error can be higher, and will be assumed at about 5%. A key assumption of the described process is that redundant PMU devices should have a real power signal that is highly correlated. Regardless of the error for instrumentation, the behavior of the near power signal should

be highly correlated over time between two redundant devices. Assuming a sufficiently large window of "n" total reports (1000 or more should be sufficient), the discrete derivative of real power is calculated for each PMU (i) in **Equation 133**.

$$\Delta P_i = \sum_{k=2}^{N} P_i(k+1) - P_i(k) \tag{133}$$

To be conservative, assuming approximately a 5% error for each PMUs current value, a 10% error threshold is used for the mean real power values. For each PMU, a comparison is run to identify all PMUs that have a mean real power within a range of 10% of the target PMU and a standard deviation in the real power signal within 10% of the target bus's power signal standard deviation. If a list of "L" PMUs exists fitting these criteria, a least absolute value algorithm is applied to see which discrete derivative of power most closely matches the target signal. The output of **Equation 134** is the other system PMU meeting the aforementioned criteria with the least absolute value in the discrete real power derivative with respect to $PMU_i$.

$$\text{Pair}_i = LAV \left( \sum_{l=1}^{L} \sum_{k=1}^{N} |\Delta P_i(k) - \Delta P_l(k)| \right) \tag{134}$$

Lastly, in order for a pair of redundant PMUs to be determined, the match must be mutual, meaning that if a match is made for $PMU_1$ determining $PMU_2$ to be the optimal redundant pair, then for $PMU_2$, a match will need to be made to $PMU_1$ as the optimal redundant pair.

With respect to the SMJA and SMPFJ matrices, determining pairs can be used to decrease computational complexity if paired PMU measurements are nearly identical during an iteration. An example of redundant pairs is shown in **Figure 5.24** for the $PMU_2$ and $PMU_{56}$.
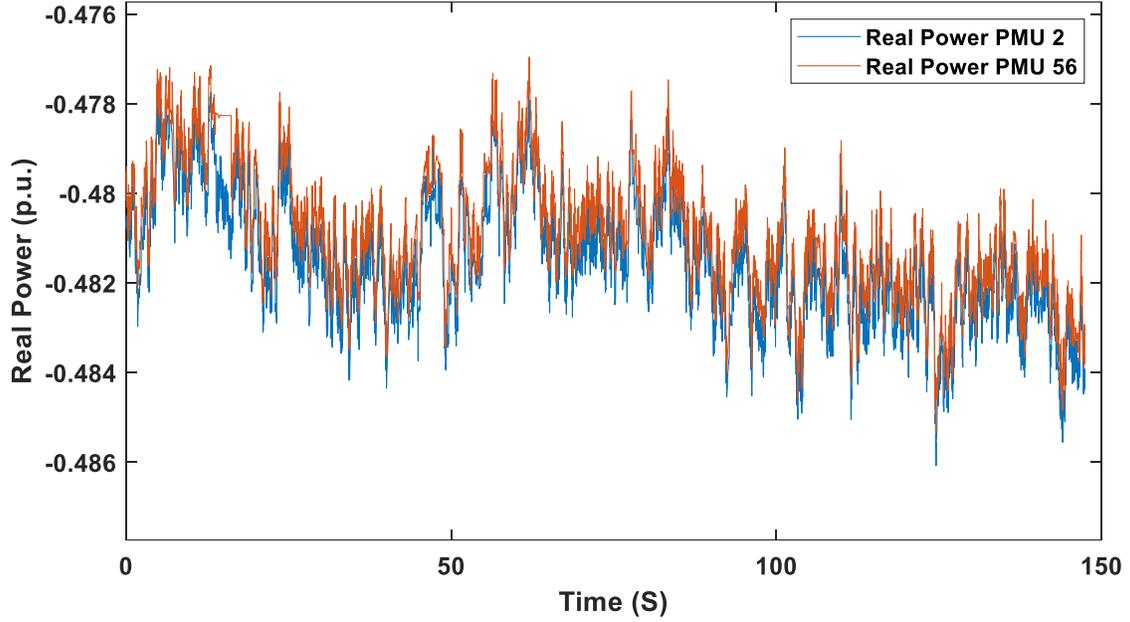
**Figure 5.24. Real Power with Respect to Identified Redundant Pairs – PMU 2 and PMU 56**

For the utility PMU dataset, 39 pairs were determined. In order to accommodate these pairs, the MSSM matrix can be modified per **Figure 5.25**, with all pairs being grouped together on their respective branches.
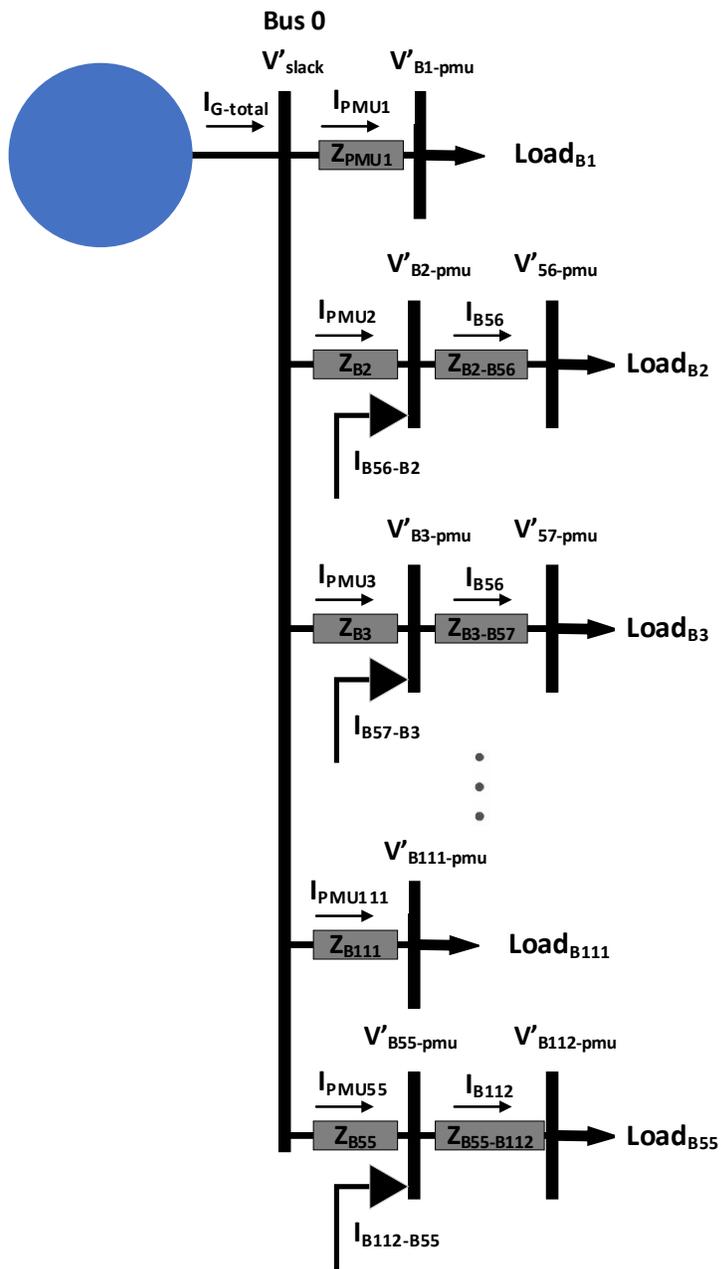
**Figure 5.25. Modified MSSM Configuration for Redundant Pairs in Utility Data**

Table 5.16 and Table 5.17 show the computational speed and accuracy considering the MSSM accounting for the redundant PMU pairs.

**Table 5.16. Computational Efficiency Considering Redundancy – 112 Bus Case**

| Computed Process | Mean Computation Time (S) | Median Computation Time (S) |
|---|---|---|
| MSSM $Y_{Bus}$ Generation | $1.503 \times 10^{-4}$ | $1.476 \times 10^{-4}$ |
| SMJA Calculation and SVD Analysis | $4.616 \times 10^{-4}$ | $4.560 \times 10^{-4}$ |
| SMPFJ Calculation and Analysis | 0.0030 | 0.0030 |

**Table 5.17. SMPFJ State Estimation Accuracy Considering Redundancy – 112 Bus**

| Measurement – | MSSM Parameter | Absolute Mean Percent Error (%) | Absolute Median Percent Error (%) |
|---|---|---|---|
| Estimated Voltage Magnitude Error | No MSSM Delay | $5.30 \times 10^{-4}$ | $1.59 \times 10^{-4}$ |
| Estimated Bus Voltage Angle Error | No MSSM Delay | $7.94 \times 10^{-4}$ | $2.19 \times 10^{-4}$ |

Table 5.15 and Table 5.17 demonstrate that the methodology can process and accurately estimate the system state of 112 PMUs from a large utility system without full observability. This also includes analysis for unstable parameters in the system, but all parameters were well within stable calculations during the analyzed timeframe. Considering the mean computation time, all calculations for 112 PMUs, without redundant pairing, require approximately 0.0026 seconds to fully converge and complete all analysis of the Jacobian matrix variants. Utilizing redundant pairs in the MSSM model is shown to increase computation time (0.0031 seconds) if applied as shown, but this implementation

does appear to marginally increase accuracy and could be leveraged in real system. *Section 5.4* demonstrated the methodologies robust capabilities to detect unstable parameters and accurately identify impacted system weak points. *Section 5.5* proves the viability of the methodology to scale to larger systems while maintaining real-time convergence. Given the computing power of the i9-12900HK 2.50GHz processor that executed the code in these studies, the methodology would be able to process data prior to the next report rate for a much larger system, not requiring an upgrade in processor or code efficiency until approximately 1000 PMU data streams. The accuracy of utilizing the SMPFJ matrix to perform state estimation is shown to be excellent, even when utilizing out of date MSSM model information. Despite the added complexity in the signals, the proposed method was able to perform more accurate state estimation for the utility PMU data than the simulated case studies. This can be attributed to the simulated data containing an abnormal frequency of major system events and unstable parameters.

CHAPTER SIX

CONCLUSION

The research and applications detailed in this dissertation are built to only require the existing PMU data streams, providing system operators with a comprehensive real-time situational awareness solution. The individual tools developed and evaluated as part of this research provide focused solutions to significant problems encountered by power system operators. PMUs provide high-resolution data at a report rate that significantly exceeds the convergence of state estimation and traditional stability monitoring. Considerable progress has been made to decrease computation times of these traditional solutions, but for applications requiring system topology integration, the full report rate of the PMU data is typically not consumed. Development of very robust and efficient model-based applications have been established to utilize PMU data in real-time, but these applications require nearly the full system to be observable. They can also require detailed alignment of external data sources to the PMU data. Although full system observability is ideal, it is costly to implement and maintain. Utilities will often strategically place PMUs at critical points in the system to gain targeted observability. These sub-systems contain valuable data that is often overlooked if the full report rate is not analyzed. It is vital that tools are developed to support successful real-world deployment.

Matrix Pencil Method and Prony Analysis were evaluated with respect to both simulated and utility PMU data for reliable detection of unstable or poorly damped oscillations of interest. These oscillations tend to be slow and are generally missed by other stability monitoring applications until the oscillation has already started to cause

136

stability issues in the system. MPM was determined to provide more robust and computationally efficient modal estimation capabilities for real-world implementation. A methodology utilizing the biorthogonal wavelet transform and power system analysis was able to detect and classify a set of system events, which is necessary to aid system operators in their understanding of the current system conditions. The application was initially built considering a larger window length for analysis. A modified implementation considering a shorter window length, 5 seconds, was also developed to decrease the time for solution convergence. Both applications were able to detect and classify faults, load switching, capacitor switching, and line outage, converging effectively in real-time. The application with a smaller window length and shorter solution time will likely be preferred for large scale implementation. The discrete Jacobian approximations in Chapter 2 provide the capability to immediately detect and flag unstable system conditions. The Jacobian variants derived in Chapter 5 augment these capabilities by providing the mathematical framework to monitor developing singularities in the Jacobian matrix variants, determine impacted system parameters, identify weak points in the system, and perform state estimation. Additionally, the novel method derived in Chapter 5 constitutes the most significant contribution of this dissertation, providing context to the real system that can be used to assist in mitigation of unstable system parameters.

As power systems continue to evolve, the availability of synchronized measurement technology will increase. It is likely that new assumptions regarding data availability and system observability can be integrated for applications built to support real-world systems. With improved assumptions and additional data availability, the

methodologies developed in this research could be augmented. For example, the MSSM could be modified to consider 80% system observability, with all major generators and loads observable. This would significantly increase the guaranteed effectiveness of the methodology and the MSSM construction could be optimized to consider new assumptions. New techniques to analyze the derived Jacobian variants could also be explored to assist in mitigation measures as unstable system parameters are detected.

The methodologies evaluated and developed as part of this research were successful at providing model-free solutions relevant to system operators for situational awareness. These methodologies are compatible with both simulated and utility PMU data, and they were developed to be capable of consuming the full report rate of all PMUs in a real-world system, either converging sub-cycle or within a reasonable timeframe to allow overlapping windows of analysis. The methodology developed in Chapter 5 was able to bridge the gap between a model-free and model-based methodology by substituting a synthetic mathematical model and deriving a novel way to detect system divergence. Ultimately, this methodology was able to mitigate instability in the real system, something that is often abstract or unclear in many model-free PMU applications. In the future, it may be possible to improve the Jacobian variant or MSSM construction to align with the true power flow Jacobian of a system more closely.

# REFERENCES

[1]     W. Xu, Z. Huang, X. Xie and C. Li, "Synchronized Waveforms – A Frontier of Data-Based Power System and Apparatus Monitoring, Protection, and Control," in *IEEE Transactions on Power Delivery*, vol. 37, no. 1, pp. 3-17, Feb. 2022, doi: 10.1109/TPWRD.2021.3072889.

[2]     R. Sodhi and M. I. Sharieff, "Phasor measurement unit placement framework for enhanced wide-area situational awareness," *IET Gener., Transm., Distrib.*, vol. 9, no. 2, pp. 172–182, Jan. 2015.

[3]     H. A. Ferrer and E. O. Schweitzer III, *Modern solutions for protection, control, and monitoring of electric power systems*, Quality Books, Inc., 2010.

[4]     K. E. Martin *et al*., "Exploring the IEEE Standard C37.118–2005 Synchrophasors for Power Systems," in *IEEE Transactions on Power Delivery*, vol. 23, no. 4, pp. 1805-1811, Oct. 2008, doi: 10.1109/TPWRD.2007.916092.

[5]     "IEEE Standard for Synchrophasors for Power Systems," in *IEEE Std C37.118-2005 (Revision of IEEE Std 1344-1995)*, vol., no., pp.1-65, 22 March 2006, doi: 10.1109/IEEESTD.2006.99376.

[6]     "IEEE Standard for Synchrophasor Measurements for Power Systems," in *IEEE Std C37.118.1-2011 (Revision of IEEE Std C37.118-2005)*, vol., no., pp.1-61, 28 Dec. 2011, doi: 10.1109/IEEESTD.2011.6111219.

[7]     OpenPDC Grid Protection Alliance, 2023. Available online at: https://github.com/GridProtectionAlliance/openPDC.

[8]     P. W. Sauer and M. A. Pai, "Power system steady-state stability and the load-flow Jacobian," in *IEEE Transactions on Power Systems*, vol. 5, no. 4, pp. 1374-1383, Nov. 1990, doi: 10.1109/59.99389.

[9]     E. Bompard, E. Carpaneto, G. Chicco, and R. Napoli, "A dynamic interpretation of the load-flow Jacobian singularity for voltage stability analysis," Int. J. Elect. Power Energy Syst., vol. 18, no. 6, pp. 385–395, Apr. 1996.

[10]    P. A. Lof, T. Smed, G. Andersson and D. J. Hill, "Fast calculation of a voltage stability index," in *IEEE Transactions on Power Systems*, vol. 7, no. 1, pp. 54-64, Feb. 1992, doi: 10.1109/59.141687.

[11]    B. Stott and O. Alsac, "Fast Decoupled Load Flow," in *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-93, no. 3, pp. 859-869, May 1974, doi: 10.1109/TPAS.1974.293985.

[12]    R. A. M. van Amerongen, "A general-purpose version of the fast decoupled load flow," in *IEEE Transactions on Power Systems*, vol. 4, no. 2, pp. 760-770, May 1989, doi: 10.1109/59.193851.

[13]    G. Liu, J. Ning, Z. Tashman, V. M. Venkatasubramanian and P. Trachian, "Oscillation Monitoring System using synchrophasors," *2012 IEEE Power and Energy Society General Meeting*, San Diego, CA, USA, 2012, pp. 1-8, doi: 10.1109/PESGM.2012.6345444.

[14]    Guoping Liu and V. Venkatasubramanian, "Oscillation monitoring from ambient PMU measurements by Frequency Domain Decomposition," *2008 IEEE International*

*Symposium on Circuits and Systems (ISCAS)*, Seattle, WA, 2008, pp. 2821-2824, doi: 10.1109/ISCAS.2008.4542044.

[15]     N. Zhou, Zhenyu Huang, F. Tuffner, J. Pierre and Shuangshuang Jin, "Automatic implementation of Prony analysis for electromechanical mode identification from phasor measurements," *IEEE PES General Meeting*, Minneapolis, MN, USA, 2010, pp. 1-8, doi: 10.1109/PES.2010.5590169.

[16]     H. Breulmann et al., "Analysis and damping of inter-area oscillations in the UCTE/CENTREL power system," in Proc. CIGRE Session, Paris, France, Aug. 2000, Paper 38-113.

[17]     L. L. Grant and M. L. Crow, "Comparison of Matrix Pencil and Prony methods for power system modal analysis of noisy signals," *2011 North American Power Symposium*, Boston, MA, USA, 2011, pp. 1-7, doi: 10.1109/NAPS.2011.6024892.

[18]     Guoping Liu, J. Quintero and V. M. Venkatasubramanian, "Oscillation monitoring system based on wide area synchrophasors in power systems," *2007 iREP Symposium - Bulk Power System Dynamics and Control - VII. Revitalizing Operational Reliability*, Charleston, SC, 2007, pp. 1-13, doi: 10.1109/IREP.2007.4410548.

[19]     A. Cagnano, E. De Tuglie and F. Torelli, "Estimation of power system dominant modes," *2009 IEEE Bucharest PowerTech*, Bucharest, Romania, 2009, pp. 1-7, doi: 10.1109/PTC.2009.5282042.

[20]     J. Khazaei et al., "Distributed Prony analysis for real-world PMU data," Elect. Power Syst. Res., vol. 133, no. 1, pp. 113–120, 2016.

[21]    Wu J.Y., "Extracting Damping Ratio Using Wavelets, Master of Engineering," Massachusetts Institute of Technology, 2001.

[22]    Kejun Mei, S. M. Rovnyak and Chee-Mun Ong, "Dynamic event detection using wavelet analysis," *2006 IEEE Power Engineering Society General Meeting*, Montreal, QC, Canada, 2006, pp. 7 pp.-, doi: 10.1109/PES.2006.1709003.

[23]    Hongshan Zhao, ZengQiang Mi and Hui Ren, "Modeling and analysis of power system events," *2006 IEEE Power Engineering Society General Meeting*, Montreal, QC, Canada, 2006, pp. 6 pp.-, doi: 10.1109/PES.2006.1709575.

[24]    Z. Y. He, S. B. Gao, X. Q. Chen, J. Zhang, Z. Bo, and Q. Qian "Study of a new method for power system transients classification based on wavelet entropy and neural network," *Int. J. Electr. Power Energy Syst.*, vol. 33, no. 3, pp. 402–410, 2011.

[25]    D. Nguyen, R. Barella, S. A. Wallace, X. Zhao and X. Liang, "Smart grid line event classification using supervised learning over PMU data streams," *2015 Sixth International Green and Sustainable Computing Conference (IGSC)*, Las Vegas, NV, USA, 2015, pp. 1-8, doi: 10.1109/IGCC.2015.7393695.

[26]    V. Torres et al., "Modeling and detection of high impedance faults," *Int. J. Elect. Power Energy Syst.*, vol. 61, pp. 163–172, 2014.

[27]    N. Taghipourbazargani, G. Dasarathy, L. Sankar and O. Kosut, "A Machine Learning Framework for Event Identification via Modal Analysis of PMU Data," in *IEEE Transactions on Power Systems*, vol. 38, no. 5, pp. 4165-4176, Sept. 2023, doi: 10.1109/TPWRS.2022.3212323.

[28]    D. -I. Kim, T. Y. Chun, S. -H. Yoon, G. Lee and Y. -J. Shin, "Wavelet-Based Event Detection Method Using PMU Data," in *IEEE Transactions on Smart Grid*, vol. 8, no. 3, pp. 1154-1162, May 2017, doi: 10.1109/TSG.2015.2478421.

[29]    V. Roy, S. S. Noureen, S. B. Bayne, A. Bilbao and M. Giesselmann, "Event Detection From PMU Generated Big Data using R Programming," *2018 IEEE Conference on Technologies for Sustainability (SusTech)*, Long Beach, CA, USA, 2018, pp. 1-6, doi: 10.1109/SusTech.2018.8671342.

[30]    S. -W. Sohn, A. J. Allen, S. Kulkarni, W. M. Grady and S. Santoso, "Event detection method for the PMUs synchrophasor data," *2012 IEEE Power Electronics and Machines in Wind Applications*, Denver, CO, USA, 2012, pp. 1-7, doi: 10.1109/PEMWA.2012.6316400.

[31]    M. Ghalei Monfared Zanjani, H. Kazemi Karegar, H. Ashrafi Niaki and M. Ghelei Monfared Zanjani, "Application of PMUs for high impedance fault detection of distribution network by considering effect of transformer vector group," *18th Electric Power Distribution Conference*, Kermanshah, Iran, 2013, pp. 1-7, doi: 10.1109/EPDC.2013.6565973.

[32]    S. Bruno, M. De Benedictis and M. La Scala, ""Taking the pulse" of Power Systems: Monitoring Oscillations by Wavelet Analysis and Wide Area Measurement System," *2006 IEEE PES Power Systems Conference and Exposition*, Atlanta, GA, USA, 2006, pp. 436-443, doi: 10.1109/PSCE.2006.296352.

[33]    I. Niazazari and H. Livani, "A PMU-data-driven disruptive event classification in distribution systems," *Elect. Power Syst. Res.*, vol. 157, pp. 251–260, Apr. 2018.

[34]    I. Niazazari and H. Livani, "Disruptive event classification using PMU data in distribution networks," *2017 IEEE Power & Energy Society General Meeting*, Chicago, IL, USA, 2017, pp. 1-5, doi: 10.1109/PESGM.2017.8274154.

[35]    M. Pavlovski, M. Alqudah, T. Dokic, A. A. Hai, M. Kezunovic and Z. Obradovic, "Hierarchical Convolutional Neural Networks for Event Classification on PMU Measurements," in *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-13, 2021, Art no. 2514813, doi: 10.1109/TIM.2021.3115583.

[36]    I. Niazazari *et al*., "PMU-data-driven Event Classification in Power Transmission Grids," *2021 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, Washington, DC, USA, 2021, pp. 1-5, doi: 10.1109/ISGT49243.2021.9372227.

[37]    Y. Liu *et al*., "Robust Event Classification Using Imperfect Real-World PMU Data," in *IEEE Internet of Things Journal*, vol. 10, no. 9, pp. 7429-7438, 1 May1, 2023, doi: 10.1109/JIOT.2022.3177686.

[38]    M. Cui, J. Wang, J. Tan, A. R. Florita and Y. Zhang, "A Novel Event Detection Method Using PMU Data with High Precision," in *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 454-466, Jan. 2019, doi: 10.1109/TPWRS.2018.2859323.

[39]    S. Pandey, A. K. Srivastava and B. G. Amidan, "A Real Time Event Detection, Classification and Localization Using Synchrophasor Data," in *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4421-4431, Nov. 2020, doi: 10.1109/TPWRS.2020.2986019.

[40]     S. S. Negi, N. Kishor, K. Uhlen and R. Negi, "Event Detection and Its Signal

Characterization in PMU Data Stream," in *IEEE Transactions on Industrial Informatics*,

vol. 13, no. 6, pp. 3108-3118, Dec. 2017, doi: 10.1109/TII.2017.2731366.

[41]     David Chan Tat Wai and Xia Yibin, "A novel technique for high impedance fault

identification," in *IEEE Transactions on Power Delivery*, vol. 13, no. 3, pp. 738-744, July

1998, doi: 10.1109/61.686968.

[42]     Shyh-Jier Huang and Cheng-Tao Hsieh, "High-impedance fault detection utilizing

a Morlet wavelet transform approach," in *IEEE Transactions on Power Delivery*, vol. 14,

no. 4, pp. 1401-1410, Oct. 1999, doi: 10.1109/61.796234.

[43]     T. M. Lai, L. A. Snider, E. Lo and D. Sutanto, "High-impedance fault detection

using discrete wavelet transform and frequency range and RMS conversion," in *IEEE

Transactions on Power Delivery*, vol. 20, no. 1, pp. 397-407, Jan. 2005, doi:

10.1109/TPWRD.2004.837836.

[44]     A.-R. Sedighi, M. -R. Haghifam, O. P. Malik and M. -H. Ghassemian, "High

impedance fault detection based on wavelet transform and statistical pattern recognition,"

in *IEEE Transactions on Power Delivery*, vol. 20, no. 4, pp. 2414-2421, Oct. 2005, doi:

10.1109/TPWRD.2005.852367.

[45]     M. Sarlak and S. M. Shahrtash, "High-Impedance Faulted Branch Identification

Using Magnetic-Field Signature Analysis," in *IEEE Transactions on Power Delivery*,

vol. 28, no. 1, pp. 67-74, Jan. 2013, doi: 10.1109/TPWRD.2012.2222056.

[46]     A. Ghaderi, H. A. Mohammadpour, H. L. Ginn and Y. -J. Shin, "High-Impedance

Fault Detection in the Distribution Network Using the Time-Frequency-Based

Algorithm," in *IEEE Transactions on Power Delivery*, vol. 30, no. 3, pp. 1260-1268, June 2015, doi: 10.1109/TPWRD.2014.2361207.

[47]    A. A. Girgis, W. Chang and E. B. Makram, "Analysis of high-impedance fault generated signals using a Kalman filtering approach," in *IEEE Transactions on Power Delivery*, vol. 5, no. 4, pp. 1714-1724, Oct. 1990, doi: 10.1109/61.103666.

[48]    A. N. Milioudis, G. T. Andreou and D. P. Labridis, "Enhanced Protection Scheme for Smart Grids Using Power Line Communications Techniques—Part II: Location of High Impedance Fault Position," in *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1631-1640, Dec. 2012, doi: 10.1109/TSG.2012.2208988.

[49]    P. M. Ashton, G. A. Taylor, M. R. Irving, I. Pisica, A. M. Carter and M. E. Bradley, "Novel application of detrended fluctuation analysis for state estimation using synchrophasor measurements," in IEEE Transactions on Power Systems, vol. 28, no. 2, pp. 1930-1938, May 2013, doi: 10.1109/TPWRS.2013.2248027.

[50]    S. G. Ghiocel *et al*., "Phasor-Measurement-Based State Estimation for Synchrophasor Data Quality Improvement and Power Transfer Interface Monitoring," in *IEEE Transactions on Power Systems*, vol. 29, no. 2, pp. 881-888, March 2014, doi: 10.1109/TPWRS.2013.2284098.

[51]    Chenxi Xu and A. Abur, "Robust linear state estimation with equality constraints," *2016 IEEE Power and Energy Society General Meeting (PESGM)*, Boston, MA, USA, 2016, pp. 1-5, doi: 10.1109/PESGM.2016.7741552.

[52]    X. Chen, K. J. Tseng and G. Amaratunga, "State estimation for distribution systems using micro-synchrophasors," *2015 IEEE PES Asia-Pacific Power and Energy*

*Engineering Conference (APPEEC)*, Brisbane, QLD, Australia, 2015, pp. 1-5, doi: 10.1109/APPEEC.2015.7381051.

[53]    A. S. Dobakhshari, M. Abdolmaleki, V. Terzija and S. Azizi, "Robust Hybrid Linear State Estimator Utilizing SCADA and PMU Measurements," in IEEE Transactions on Power Systems, vol. 36, no. 2, pp. 1264-1273, March 2021, doi: 10.1109/TPWRS.2020.3013677.

[54]    Y. Weng, R. Negi, C. Faloutsos and M. D. Ilić, "Robust Data-Driven State Estimation for Smart Grid," in *IEEE Transactions on Smart Grid*, vol. 8, no. 4, pp. 1956-1967, July 2017, doi: 10.1109/TSG.2015.2512925.

[55]    G. Valverde, S. Chakrabarti, E. Kyriakides and V. Terzija, "A Constrained Formulation for Hybrid State Estimation," in *IEEE Transactions on Power Systems*, vol. 26, no. 3, pp. 1102-1109, Aug. 2011, doi: 10.1109/TPWRS.2010.2079960.

[56]    B. Zargar, A. Angioni, F. Ponci and A. Monti, "Multiarea Parallel Data-Driven Three-Phase Distribution System State Estimation Using Synchrophasor Measurements," in *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 9, pp. 6186-6202, Sept. 2020, doi: 10.1109/TIM.2020.2967512.

[57]    M. Glavic and T. Van Cutsem, "Wide-Area Detection of Voltage Instability from Synchronized Phasor Measurements. Part I: Principle," in *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1408-1416, Aug. 2009, doi: 10.1109/TPWRS.2009.2023271.

[58]    M. Glavic and T. Van Cutsem, "Wide-Area Detection of Voltage Instability from Synchronized Phasor Measurements. Part II: Simulation Results," in *IEEE Transactions*

*on Power Systems*, vol. 24, no. 3, pp. 1417-1425, Aug. 2009, doi:

10.1109/TPWRS.2009.2023272.

[59]    H. -Y. Su and C. -W. Liu, "Estimating the Voltage Stability Margin Using PMU

Measurements," in *IEEE Transactions on Power Systems*, vol. 31, no. 4, pp. 3221-3229,

July 2016, doi: 10.1109/TPWRS.2015.2477426.

[60]    A. Reddy, K. Ekmen, V. Ajjarapu and U. Vaidya, "PMU based real-time short

term voltage stability monitoring - Analysis and implementation on a real-time test bed,"

2014 North American Power Symposium (NAPS), Pullman, WA, USA, 2014, pp. 1-6,

doi: 10.1109/NAPS.2014.6965485.

[61]    S. Dasgupta, M. Paramasivam, U. Vaidya and V. Ajjarapu, "Real-Time

Monitoring of Short-Term Voltage Stability Using PMU Data," in *IEEE Transactions on

Power Systems*, vol. 28, no. 4, pp. 3702-3711, Nov. 2013, doi:

10.1109/TPWRS.2013.2258946.

[62]    J. M. Lim and C. L. DeMarco, "Model-free voltage stability assessments via

singular value analysis of PMU data," *2013 IREP Symposium Bulk Power System

Dynamics and Control - IX Optimization, Security and Control of the Emerging Power

Grid*, Rethymno, Greece, 2013, pp. 1-10, doi: 10.1109/IREP.2013.6629399.

[63]    J. M. Lim and C. L. DeMarco, "Estimating power flow conditioning from phasor

measurement data," *2012 50th Annual Allerton Conference on Communication, Control,

and Computing (Allerton)*, Monticello, IL, USA, 2012, pp. 1338-1346, doi:

10.1109/Allerton.2012.6483373.

[64]    Y. Ge, A. J. Flueck, D. -K. Kim, J. -B. Ahn, J. -D. Lee and D. -Y. Kwon, "Power System Real-Time Event Detection and Associated Data Archival Reduction Based on Synchrophasors," in *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 2088-2097, July 2015, doi: 10.1109/TSG.2014.2383693.

[65]    A. Allen, M. Singh, E. Muljadi, and S. Santoso, ''PMU data event detection: A user guide for power engineers,'' Nat. Renew. Energy Lab., Golden, CO, USA, Tech. Rep. NREL/TP-5D00-61664, 2014.

[66]    T. Overbye, P. Sauer, C. DeMacro, B. Lesieutre, and M. Venkatasubramanian, "Using PMU data to increase situational awareness," *PSERC Final Project Rep.*, 2010.

[67]    C. Basu *et al*., "Understanding events for wide-area situational awareness," *ISGT 2014*, Washington, DC, USA, 2014, pp. 1-5, doi: 10.1109/ISGT.2014.6816408.

[68]    A. J. Allen, S. -W. Sohn, S. Santoso and W. M. Grady, "Algorithm for screening PMU data for power system events," *2012 3rd IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*, Berlin, Germany, 2012, pp. 1-6, doi: 10.1109/ISGTEurope.2012.6465867.

[69]    S. Kantra and E. B. Makram, "Development of the Decoupled Discrete-Time Jacobian Eigenvalue Approximation for Situational Awareness Utilizing openPDC," *Journal of Power and Energy Engineering*, 2016, 4, pp. 21-35.

[70]    S. Kantra and E. B. Makram, "Expansion of the Decoupled Discrete-Time Jacobian Eigenvalue Approximation for Model-Free Analysis of PMU Data," *Journal of Power and Energy Engineering*, 2017, 5, pp. 14-35.

[71]    S. Kantra, H. A. Abdelsalam and E. B. Makram, "Application of PMU to detect high impedance fault using statistical analysis," *2016 IEEE Power and Energy Society General Meeting (PESGM)*, Boston, MA, USA, 2016, pp. 1-5, doi: 10.1109/PESGM.2016.7741454.

[72]    P. Z. Peebles, Jr., *Probability, Random Variables, and Random Signal Principles*, 4th ed. New York: McGraw-Hill, 2001

[73]    Liu Fan, Sun Cai-xin and Zhang Xiao-xing, "Researching on Superposition of Multi-Harmonics Sources in Power Supply System Based on the Central Limit Theorem," *2005 IEEE/PES Transmission & Distribution Conference & Exposition: Asia and Pacific*, Dalian, 2005, pp. 1-4, doi: 10.1109/TDC.2005.1546793.

[74]    P. Kundur, *Power System Stability and Control*. New York: McGraw Hill, 1994.

[75]    S. Mallat, *A Wavelet Tour of Signal Processing*. San Diego, CA: Academic, 1998.

[76]    MATLAB/Simulink, 2023. Available online at: https://www.mathworks.com/.

[77]    T. K. Sarkar and O. Pereira, "Using the matrix pencil method to estimate the parameters of a sum of complex exponentials," in *IEEE Antennas and Propagation Magazine*, vol. 37, no. 1, pp. 48-55, Feb. 1995, doi: 10.1109/74.370583.

[78]    MathWorks File Exchange. "10-Machine New-England Power System IEEE Benchmark". https://www.mathworks.com/matlabcentral/fileexchange/54771-10-machine-new-england-power-system-ieee-benchmark.

[79]    W. O'Brien, E. Udren, K. Garg, D. Haes and B. Sridharan, "Catching falling conductors in midair — detecting and tripping broken distribution circuit conductors at

protection speeds," *2016 69th Annual Conference for Protective Relay Engineers (CPRE)*, 2016, pp. 1-11, doi: 10.1109/CPRE.2016.7914881.

[80]    T. G. Klimova, A. D. Guseynov, M. S. Malyutin, V. D. Uksekov and K. V. Latyshov, "Analysis of Phasor Measurement Units Operation Under Various Grid Conditions," *2021 4th International Youth Scientific and Technical Conference on Relay Protection and Automation (RPA)*, Moscow, Russian Federation, 2021, pp. 1-20, doi: 10.1109/RPA53216.2021.9628753.