Adaptive and Intelligent PMUs for Smarter Applications

Final Project Report

Power Systems Engineering Research Center

Empowering Minds to Engineer
the Future Electric Energy System

Adaptive and Intelligent PMU for Smarter Applications

Final Project Report

Project Team

Anurag K. Srivastava, Project Leader
HyoJong Lee, Graduate Student
Saugata Biswas, Graduate Student
Washington State University

A. P. Sakis Meliopoulos
George Cokkinides
Zhenyu Tan, Graduate Student
Rui Fan, Graduate Student
Georgia Institute of Technology

Peter W. Sauer
Karl Reinhard, Graduate Student
Kenta Kirihara, Graduate Student
Bogdan Pinte, Graduate Student
Michael Quinlan, Graduate Student
Yang Liu, Undergraduate Student
University of Illinois at Urbana-Champaign

PSERC Publication 16-01

January 2016

For information about this project, contact

Anurag K Srivastava, Project Leader
Associate Professor, School of Electrical Engineering and Computer Science
Director, Smart Grid Demonstration and Research Investigation Lab (SGDRIL)
Energy Systems Innovation Center (ESIC)
Washington State University
355 Spokane St
Pullman, WA 99164-2752
Phone: 509-335-2348
Fax: 509-335-3818
Email: asrivast@eecs.wsu.edu

Power Systems Engineering Research Center

The Power Systems Engineering Research Center (PSERC) is a multi-university Center conducting research on challenges facing the electric power industry and educating the next generation of power engineers. More information about PSERC can be found at the Center's website: http://www.pserc.org.

For additional information, contact:

Power Systems Engineering Research Center
Arizona State University
527 Engineering Research Center
Tempe, Arizona 85287-5706
Phone: 480-965-1643
Fax: 480-965-0745

Notice Concerning Copyright Material

Acknowledgements

This is the final report for the Power Systems Engineering Research Center (PSERC) research project titled “Adaptive and Intelligent PMU for Smarter Applications” (project S-57). We express our appreciation for the support provided by PSERC’s industry members and by the National Science Foundation under the Industry / University Cooperative Research Center program.

We wish to thank our industry advisors for their support and help: Evangelos Farantatos (EPRI), Paul Myrda (EPRI), Farrokh Habibiashrafi (Southern California Edison), Innocent Kamwa (Hydro-Québec), Xiaoming Feng (ABB), Reynaldo Nuqui (ABB), Qiang Zhang (ISO-New England), William Kamwa (AEP), Giuseppe Stanciulescu, (BC Hydro), and Jim Kleitsch (American Transmission Company, ATC).

Additionally, we are grateful to National Instruments and RTDS, Inc. for their support.

Executive Summary

Performance of Phasor Measurement Units (PMUs) varies with operating conditions such as dynamic system states, harmonics, off-nominal frequency and changes due to load changes and/or faults. Phasor based applications may utilize only part of the PMU measurements like frequency, voltage magnitude or angles. Accuracy of these specific measurements depends on estimation and filtering algorithms within PMU, which can be switched between different algorithms to adopt with specific applications and operating conditions for higher performance. The PMU can self-report critical data quality information such as estimation error and GPS status using user-defined bits to provide better decision support for operators.

Distributed applications using PMU data may require some of the computation to be done within PMU. Data management can be done in intelligent manner to minimize the computational and communication burden centrally as well as to enable enhanced applications. Additionally, user defined bits can be used in flexible and intelligent manner to realize enhanced capability of PMU for smarter applications. PMU data quality and interchangeability of PMU with changing applications is another important issue.

This project focuses on all the above aspects to develop 'adaptive' and 'intelligent' PMU for smarter applications and believes that 'one PMU does not fit all applications and all operating conditions'. In this project, we have developed several versions of advanced PMU with different features: the adaptive PMU, the standard PMU, virtual PMUs and the distribution PMU. In addition, improved testing methods to characterize the accuracy of these PMUs are also reported.

For the adaptive PMU algorithm, wavelet transform (WT) based phasor estimation is proposed as an alternative estimation technique for dynamic system condition like off-nominal frequency. The system frequency is updated by WT based estimation of target frequency. Also, multiple filtering techniques have been developed to provide options for different operating conditions. Switching techniques is proposed to identify system-operating conditions and choose suitable estimation algorithm within PMU. The standard PMU algorithm performs discrete Fourier transform (DFT) utilizing a variable time window so as the frequency changes the integration interval always contains an integer number of cycles. Thus, spectral leakage and other known errors of the Fourier transform for a signal of changing frequency are eliminated. We have also developed a virtual PMU using the DFT based Standard PMU algorithm. The performance of Standard PMU algorithm is evaluated with variable sampling rates and several different interpolation methods. Performance results are provided in the report for all these PMUs using PMU Performance Analyzer (PPA) and WinXFM platform as well as upgraded test beds. Distribution PMU is designed as a single-phase low cost PMU using National Instruments Platform.

Synchrophasor data quality is another important issue. PMU data can be dropped by communication system, have high noise, or have errors attributable to GPS signal errors.

We had access to industry provided field PMU data and developed techniques to identify defective synchrophasor data using statistical techniques.

This work is opening up various possibilities for more intelligent applications as well as more intelligent automation of power system functions. One of the applications is dynamic state estimation based protection, a.k.a. setting-less protection. Experimental results show that the setting-less protection method provides better performance than traditional protective schemes.

The report highlights challenges of using synchrophasor data collected during power system equilibrium state transitions to understand and characterize dynamic behavior. Interoperability and interchangeability is an important issue for intelligent PMUs. We propose an interoperability standard for intelligent PMUs in terms of power device model data exchange.

This project resulted in following publications:

[1] Pinte, Bogdan; Michael Quinlan, Andy Yoon, Karl Reinhard, and Peter W. Sauer. "A One-Phase, Distribution-Level Phasor Measurement Unit for Post-Event Analysis", *Proceedings, Power Engineering Conference Illinois*, Champaign, IL, February, 2014.

[2] Kirihara, Kenta; Karl E. Reinhard, Andy K. Yoon, and Peter W. Sauer. "Investigating Synchrophasor Data Quality Issues", *Proceedings, Power Engineering Conference Illinois*, Champaign, IL, February, 2014.

[3] Pinte, Bogdan; Michael Quinlan, and Karl Reinhard. "Low Voltage Micro-Phasor Measurement Unit (μPMU)", *Proceedings of the Power Engineering Conference,* Champaign, IL, February, 2015.

[4] Kirihara, Kenta; Karl E. Reinhard, Yang Liu, and Peter W. Sauer. "Synchrophasor Visualizer", *Proceedings of the Power Engineering Conference*, Champaign, IL, February, 2015.

[5] Biswas, S.; A. K. Srivastava, J. Park, and J. Castaneda. "A New Tool for Testing of Phasor Measurement Units: PMU Performance Analyzer", special issue on *Sensors and Data analytics for Smart Grid Infrastructure, IET Generation, Transmission & Distribution,* August 2014.

[6] Biswas, S.; and A. K. Srivastava. "Synchrophasor Device Testing And Related Standards", *Handbook of Smart Grid Development*, Book chapter, Wiley, 2015.

[7] Biswas, S.; and A. Srivastava. "PMU performance analyzer (PPA)", Patent pending, USPTO: PCT/US2014/061412, October 2014.

[8] Tushar, Ghosh; S. Biswas A. K. Srivastava. "A Comparative Study of Model and Measurement based Voltage Stability Approaches", *North American Power Symposium*, Charlotte, NC, October, 2015.

[9] Biswas, S.; Tushar, and A. Srivastava. "Performance Analysis of a New Synchrophasor Based Real Time Voltage Stability Monitoring (RT-VSM) Tool", *North American Power Symposium*, Pullman, WA, September 2014.

[10] Meliopoulos, A.P. Sakis; G. J. Cokkinides, Z. Tan, S. Choi, Y. Lee and P. Myrda. "Setting-Less Protection: Feasibility Study," *46th Hawaii International Conference System Sciences (HICSS)*, pp. 2345-2353, January 7-10, 2013.

[11] Meliopoulos, A.P. Sakis; L. Sun, R. Fan and P. Myrda. "Update on Object-oriented DSE Based Protection," *Fault and Disturbance Conference*, Atlanta, GA, April, 2014.

[12] Meliopoulos, A.P. Sakis. "Distributed Dynamic State Estimator Enables Seamless DSA," *Proceedings of the IEEE-PES 2014 General Meeting*, Washington, D.C., July 27-31, 2014.

[13] Liu, Yu; A.P. Sakis Meliopoulos, Rui Fan, and Liangyi Sun. "Dynamic State Estimation Based Protection of Microgrid Circuits," *Proceedings of the IEEE-PES 2015 General Meeting*, Denver, CO, July 26-30, 2015.

Student Theses:

[1] Kirihara, Kenta. "*Big Data in Power Systems: A Statistical Approach on Synchrophasor Application,"* University of Illinois at Urbana-Champaign, M.S. in Electrical and Computer Engineering, July 2015.

[2] Pinte, Bogdan. "*Development and Implementation of an Open-Box Distribution-Level Phasor Measurement Unit,"* University of Illinois at Urbana-Champaign, M.S. in Electrical and Computer Engineering, May 2015.

[3] Quinlan, Michael. "*Development and Deployment of a Residential Voltage Level Phasor Measurement Unit*," University of Illinois at Urbana-Champaign, M.S. in Electrical and Computer Engineering, May 2015.

[4] Saugata Biswas. "*Synchrophasor Based Voltage Stability Monitoring and Control of Power Systems*", Washington State University, Ph.D., September 2014.

Table of Contents

List of Figures

List of Tables

Part 1: Adaptive PMU, Self-Reporting Error and Application of User-Defined Bits

1. Introduction

This part relate to work done at Washington State University.
Phasor Measurement Unit (PMU) enables power system monitoring and decision support in real time. PMU estimates phasor data using filtered sampling waveform from Potential Transformer (PT) and Current Transformer (CT) with synchronized time signal by Global Positioning System (GPS). One of the first PMU based on Discrete Fourier Transform (DFT) was introduced in early 1980 [1-1]. DFT based PMUs have been broadly studied by researchers and integrated in power system. Theoretically, standard DFT based phasor estimator algorithm provides lower Total Vector Errors (TVEs) at target frequency under steady state condition. However, power system operating conditions with high noise, harmonics, off-nominal frequency and transients leads to high total vector error using conventional DFT based phasor estimator. The conventional DFT based phasor estimator has following issues; 1) Unmatched sampling frequency rate with system [1-2], 2) Off-nominal frequency [1.3].

To minimize TVE, researchers have explored different DFT methods. Resampling method is introduced as a superior method. In this method, sampling rate is adjusted by estimated frequency. Resampled input signal is applied into phasor estimation with updated sampling parameters [1-4]. Smart DFT (SDFT) based phasor estimator is proposed in [1-5]. This method expends conventional DFT based Phasor estimator to deal with frequency deviation and harmonics. Using Total Least Squares estimation with Propagator Method, the Subspace based Phasor Estimator is proposed in [1-6]. The algorithm is designed for dynamic condition to detect damped signal and system oscillations. This reference also proposed compensation module to minimize TVE.

Even though improved DFT based phasor estimator for dynamic condition and noise was developed to satisfy IEEE C37.118.2005 standard, the DFT based phasor estimation requires higher resolution to detect target frequency, when the system is under dynamic condition or off-nominal frequency. Theoretically, DFT based method uses same resolution for every frequencies. It is challenging to find accurate frequency even for DFT with adjustable windows function to match sampling rate with system frequency.

However, Wavelet Transform as an alternative estimation technique contains time and frequency domain information. It is also able to adjust window size using the scaling function at target frequency. In this project, Wavelet Transform (WT) based phasor estimator is proposed to deal with dynamic condition and to develop adaptive PMU. In this project, WT based PMUs has been proposed to perform better in dynamic condition. The new algorithm needs to be validated under different operating condition of power system by PMU performance Analyzer (PPA) developed at Washington State University.

To switch algorithms, the switching algorithm is proposed using energy function to detect transient condition. Once it detects dynamic or transient condition, WT based phasor estimator is activated to estimate phasors.

The proposed PMU is able to switch from one phasor estimation algorithm to other based on system condition. Users can choose switching option, self-reporting, PMU cyber status, and result from built-in applications using User defined bits. The User Defined bits are transferred via C37.118.2005 protocol or customized protocol. The PMU can report self-estimated estimation error or GPS status using User Defined bits. Users can enable/disable these bits as needed. For User Defined bits, there are three possible options explored in this project: estimation error, and GPS or communications failure status.

The estimation error is calculated using regression analysis and estimated phasor. It calculates maximum estimation error for each reporting period. For normal condition, it uses estimation by DFT method and energy function approach supported by regression analysis to determine estimation error. Once the WT based estimation is enabled by system condition, the estimation error is determined using estimation by WT based estimation. The time signal from GPS signal is important for PMU to minimize phasor estimation error. The cyber status (GPS signal status) is reported using User Defined bits.

2. Discrete Fourier Transform (DFT) and Wavelet Transform (WT) Based PMU

2.1. Conventional DFT based Phasor Estimator

A model of continuous input signal $x(t)$ at the target system frequency f_0 with changing amplitude and angle (θ) in time as well as harmonics is represented by eq. (1.1)

$$x(t) = Amplitude \times \cos(2\pi f_0 t + \theta) \quad (1.1)$$

The sampled input signal $x(n)$ is represented by eq. (1.2)

$$x(n) = Amplitude \times \cos(2\pi f_0 n\Delta t + \theta) \quad (1.2)$$

Where n = a sampling point in window, and Δt = time interval between two sampling points, $\Delta t = 1/(f_0 N)$, N = number of sampling points in the window.

Input signal x(t) and x(n) are shown in Figure 1.1 with following parameters.
N = number of sampling points = 16 points
Angle = 180 degree
Amplitude = 100 V
Frequency = 60Hz

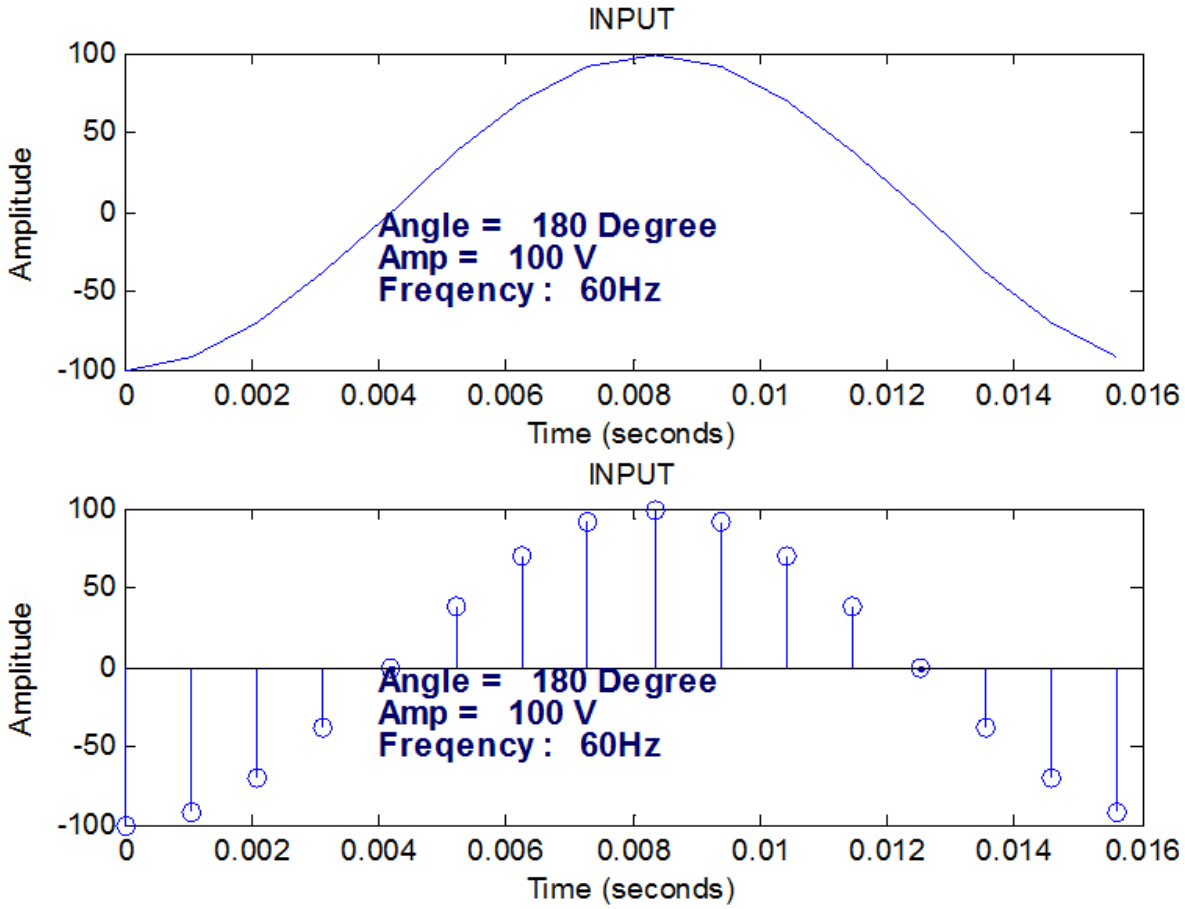


Figure 1.1: Input Signal of x(t) and x(n) at 60Hz

Conventional DFT based phasor estimator uses either non-recursive updates or recursive updates [1-4] as given in eq. (1.3) and (1.4).

$$X^K = \frac{\sqrt{2}}{K}\sum_{n=0}^{K-1} x(n+1)\left[\cos\left(\frac{2\pi n}{K}\right) - j\sin\left(\frac{2\pi n}{K}\right)\right] \tag{1.3}$$

$$X^K = X^{K-1} + \frac{\sqrt{2}}{K}\sum_{n=0}^{K-1} x(n+1)\left[\cos\left(\frac{2\pi (n+1)}{K}\right) - j\sin\left(\frac{2\pi (n+1)}{K}\right)\right] \tag{1.4}$$

The DFT based phasor estimator requires several cycles of sampling data to provide lower TVE. Typically, many of DFT based phasor estimator use several cycles for estimation.

Magnitude and angle is represented as given by eq. (1.5) and (1.6)

$$Mag = \sqrt{real(X^K)^2 + Imag(X^K)^2} \tag{1.5}$$

$$Ang = \tan^{-1}\frac{Imag(X^K)}{Real(X^K)} \tag{1.6}$$

For more detail of conventional DFT based phasor estimation, see reference [1-4].

2.2. Wavelet Transform based Phasor Estimator

In wavelet transform based phasor estimator, it convolutes with time varying continuous input signal. The discrete input signal is expressed as given by eq (1.7).

$$x(n) = Amplitude \times \cos(2\pi f n \Delta T + \theta), n = 0, 1, 2, 3, ... \quad (1.7)$$

where f = test frequency, ΔT = 1/frequency, and θ= angle

The key of good WT based phasor estimator is to find proper reference wavelet as mother wavelet. In this project, SGDRIL proposes modified sinusoidal wavelet as mother wavelet for phasor estimator as given by eq. (1.8) and (1.9).

$$\psi(t) = \sin(t) \cdot e^{(\sigma + j\omega_o)t} u(t) \quad (1.8)$$

$$\psi(\omega) = \frac{K}{(-\sigma + j(\omega - \omega_0)^2)} \quad (1.9)$$

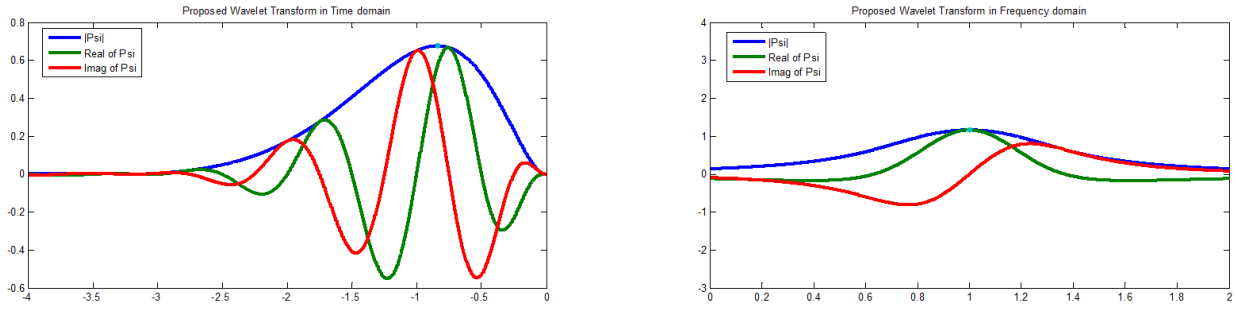


Figure 1.2: Proposed Reference Wavelet in Time and Frequency Domain

Using reference wavelet transform, the wavelet transform is convoluted with discrete input signal as in eq. (1.10) [1-7].

$$W(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \psi^* \left(\frac{t - b}{a} \right) dt \quad (1.10)$$

where a = scale factor, b = location.

$$W(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \psi^* \left(\frac{t - b}{a} \right) dt \quad (1.11)$$

Discrete Wavelet Transform of eq. (1.12) as eq. (1.13)

$$\mathrm{W(a,b)} = \frac{\Delta \mathrm{T}}{\sqrt{a}} \sum_{n=0}^{b} x(n)\, \psi^{*}\left(\frac{t-b}{a}\Delta T\right) \tag{1.12}$$

$$\mathrm{W(a,b)} = \frac{\Delta \mathrm{T}}{\sqrt{a}} \sum_{n=0}^{b} Amplitude \times \cos(2\pi f n \Delta T + \theta) \cdot \sin(t) \cdot e^{(\sigma + j\omega_o)\Delta t} \tag{1.13}$$

Discrete Wavelet Transform is decomposed as following given by eq. (1.13) [1-8].

$$\mathrm{W(a,b)} = \frac{\Delta \mathrm{T}}{\sqrt{a}} \sum_{n=0}^{b} Amplitude \times \cos(2\pi f b \Delta T + \theta + 2\pi a l \Delta T) \cdot \sin(t) \cdot e^{(\sigma + j\omega_o)\Delta t} \tag{1.14}$$

$$\mathrm{W(a,b)} = \frac{\Delta \mathrm{T}}{\sqrt{a}} \sum_{n=0}^{b} Amplitude \times \cos(2\pi f b \Delta T + \theta + 2\pi a l \Delta T) \cdot \sin(t) \cdot e^{(\sigma)\Delta t} \times [\cos(2\pi f) + j sin(2\pi f)] \tag{1.15}$$

For simplicity, this wavelet can represent as cosine and sine function to avoid complex number calculation.

$$\mathrm{W(a,b)} = \frac{\Delta \mathrm{T}}{\sqrt{a}} \sum_{n=0}^{b} Amplitude \times \cos(2\pi f b \Delta T + \theta + 2\pi a l \Delta T) \cdot \sin(t) \cdot e^{(\sigma)\Delta t} \times \cos(2\pi f) \tag{1.16}$$

$$\mathrm{W(a,b)} = \frac{\Delta \mathrm{T}}{\sqrt{a}} \sum_{n=0}^{b} Amplitude \times \cos(2\pi f b \Delta T + \theta + 2\pi a l \Delta T) \cdot \sin(t) \cdot e^{(\sigma)\Delta t} \times sin(2\pi f) \tag{1.17}$$

Using weight factor (U), the phasor information is calculated as in eq. (1.18). The weight factor is defined in [1-8].

$$\begin{aligned} [\mathrm{W}_{\text{real part}}] &= [\mathrm{U}][\mathrm{X}_{\text{real}}] \\ [\mathrm{W}_{\text{Imag part}}] &= [\mathrm{U}][\mathrm{X}_{\text{imag}}] \end{aligned} \tag{1.18}$$

Given phasor calculation, frequency information is updated as in eq. (1.19).

$$\Delta f = \frac{\mathrm{x}_{\text{real}} \times x'_{real} + x_{imag} \times x'_{imag}}{\left(x^2_{real} + x^2_{imag}\right)^2} \tag{1.19}$$

By calculated variation of frequency, the estimated frequency is updated by using eq. (1.20).

$$\text{updated frequency} = \text{previous freqency} + \Delta f \tag{1.20}$$

Once the wavelet transform is performed to meet tolerance (ε) setting, the magnitude and angle are estimated as eq. (1.21) and (1.22).

$$\text{Magnitude} = \sqrt{X_{real}^2 + X_{imag}^2} \tag{1.21}$$

$$\text{Angle} = \text{actan}\left(\frac{\text{X}_{\text{imag}}}{\text{X}_{\text{real}}}\right) \tag{1.22}$$

The overall flowchart of WT based phasor estimator is shown in Figure 1.3.

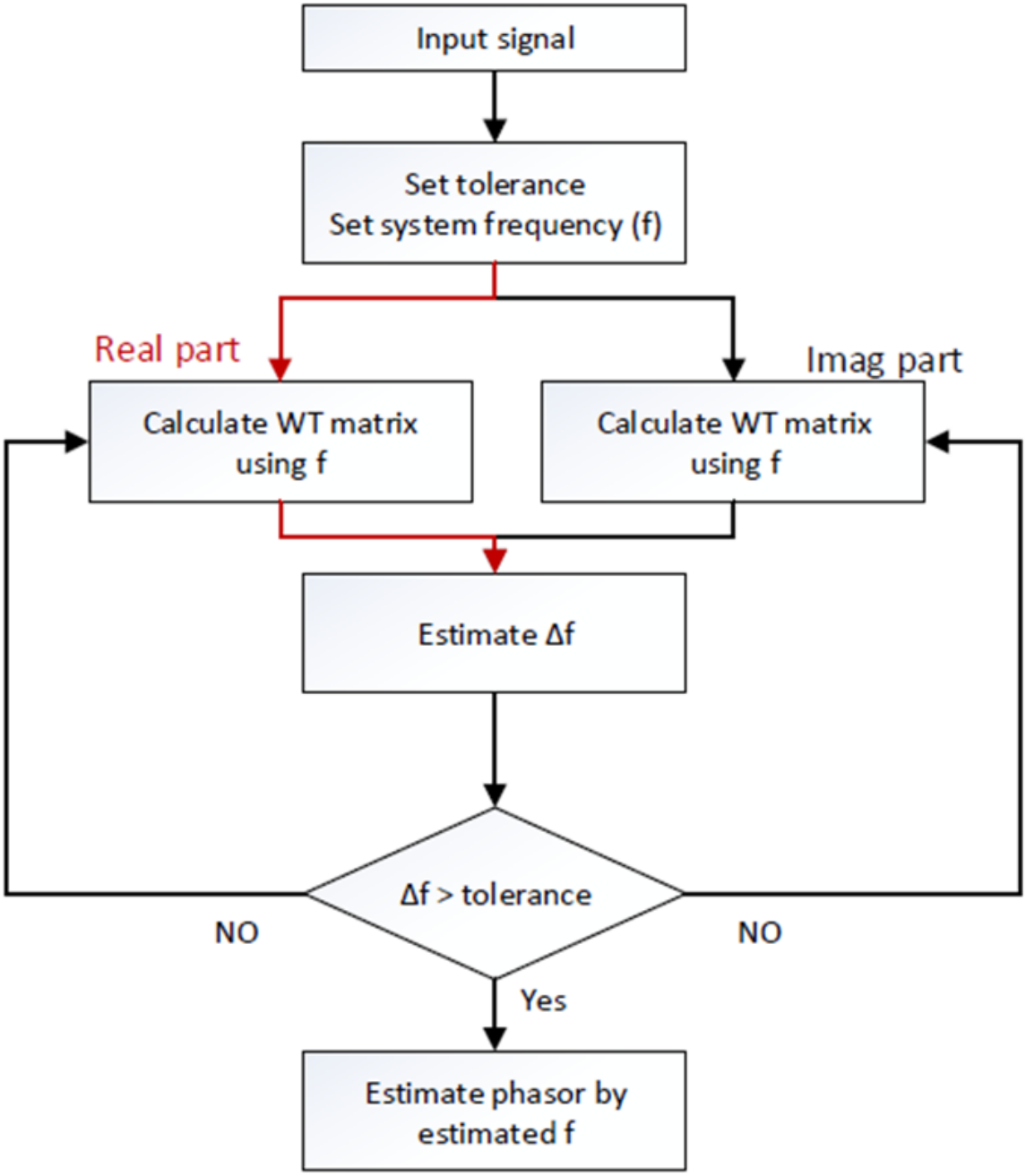


Figure 1.3: Flowchart of WT based Phasor Estimator

2.3. Simple Frequency Estimator (Zero Crossing Method)

To generate ref. wavelet, it requires initial frequency as initial parameters. It does not need to be accurate for initial frequency. However, it needs to be fast for reducing computation time. By these criteria, the Zero Crossing Method has been used to estimate initial frequency as eq. (1.23).

$$\text{Freqency} = \frac{1}{\Delta T} = \frac{1}{|first\ zero\ crossing\ point - second\ zero\ crossing\ point|} \tag{1.23}$$

The simple zero crossing method is shown in Figure 1.4.

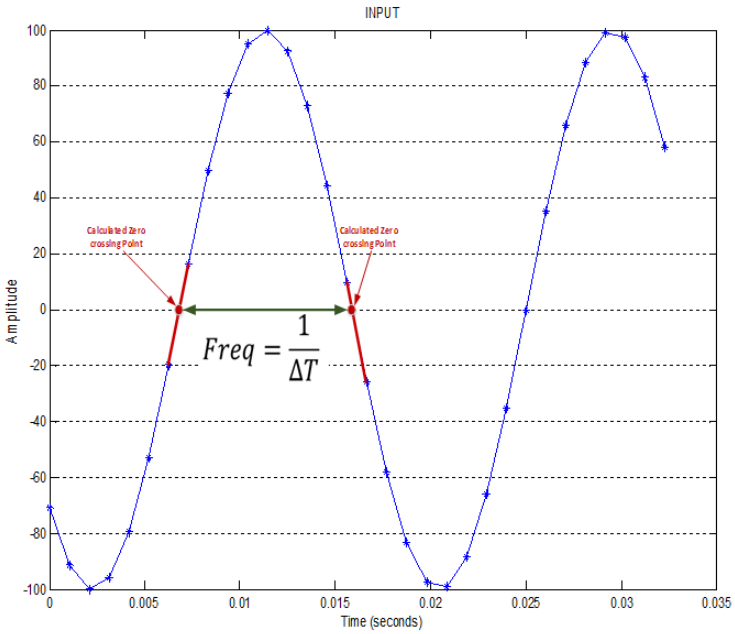


Figure 1.4: Initial Frequency Estimator using Zero Crossing Method

This zero crossing method requires only half cycles to estimate initial frequency. The method needs to reconstruct signal for eliminating noise and harmonic by DFT theory [1-10].

2.4. Simulation Results of DFT and WT based Phasor Estimator

Commonly, phasor estimations are designed by DFT for most commercial PMUs. The DFT based algorithm requires at least one cycle of window size to perform. The windows size offers trades off between accuracy and computation time. Minimum requirement of DFT based Phasor is one cycle, although several cycles are required for frequency estimation and to meet standard TVE requirements. In this section, DFT and WT based Phasor estimations are compared at normal condition.

For comparison between DFT based Phasor Estimator and WT based Phasor Estimator, both algorithms uses one cycle of window size. First, the algorithms have been tested by nominal frequency (60Hz) with/without Gaussian white noise (1dBW) as shown in Figures 1.5, 1.6, 1.7 and 1.8.

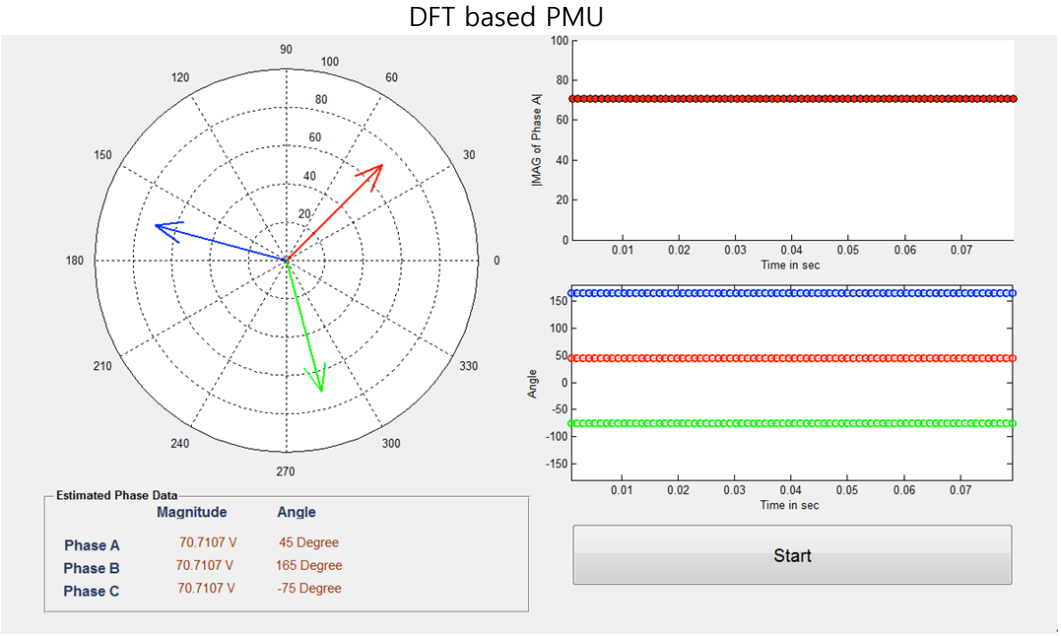


Figure 1.5: DFT based Phasor Estimator without Gaussian Noise

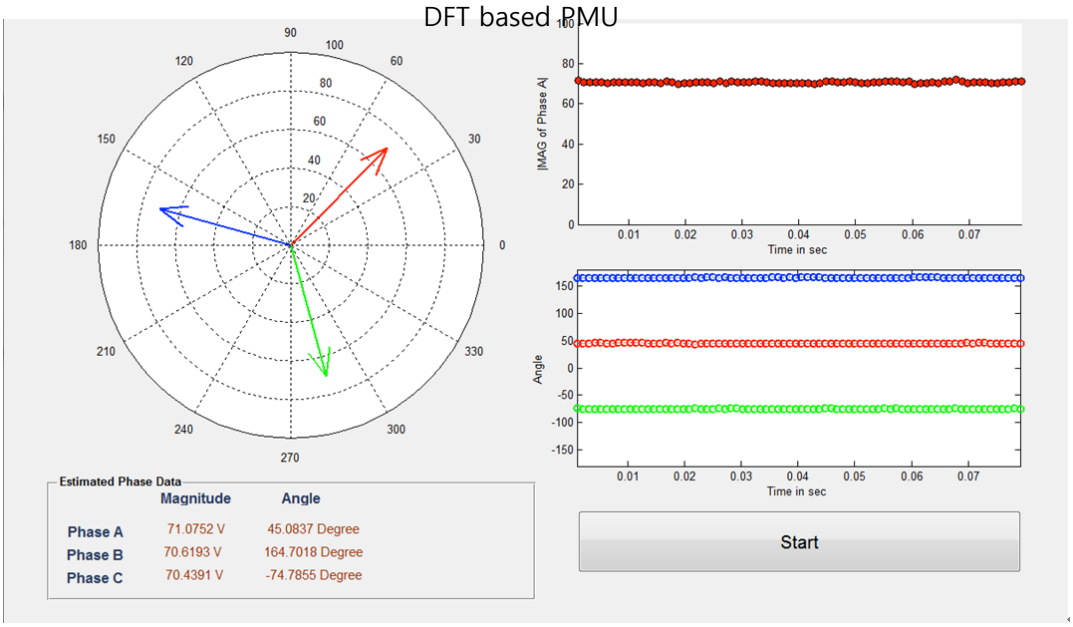


Figure 1.6: DFT based Phasor Estimator with Gaussian Noise

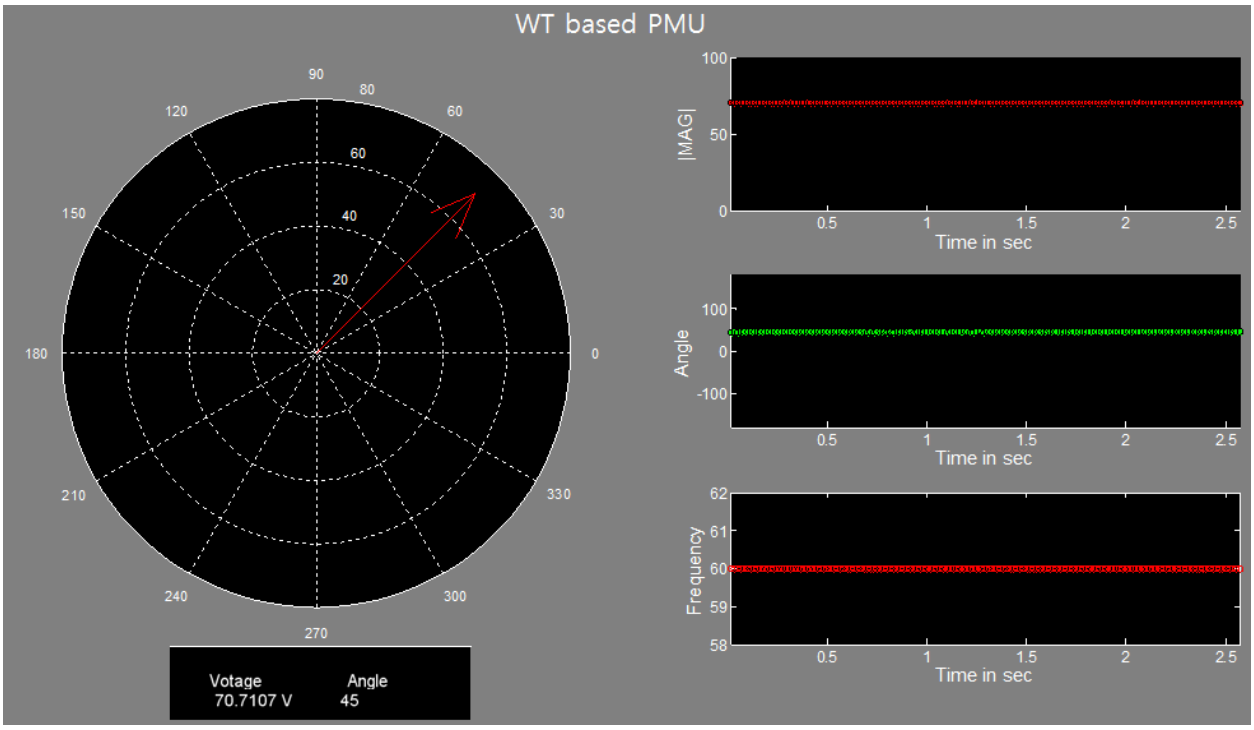


Figure 1.7: WT based Phasor Estimator without Gaussian Noise

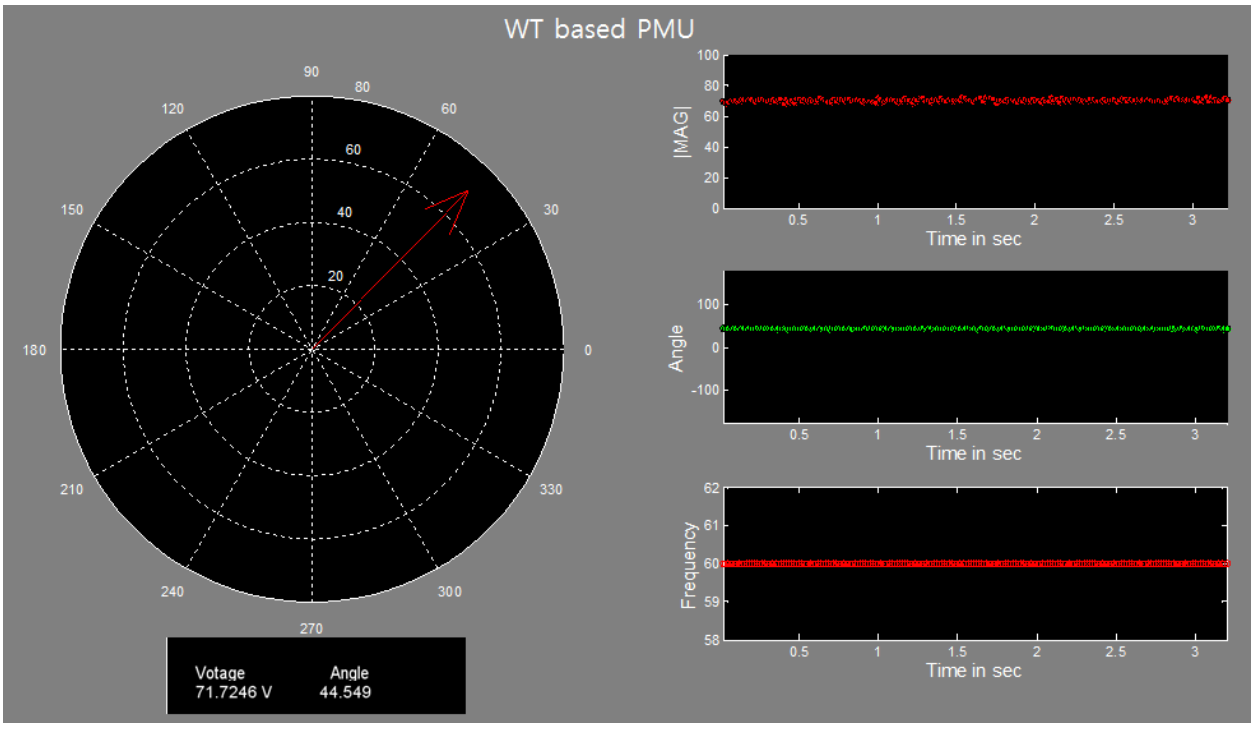


Figure 1.8: WT based Phasor Estimator with Gaussian Noise

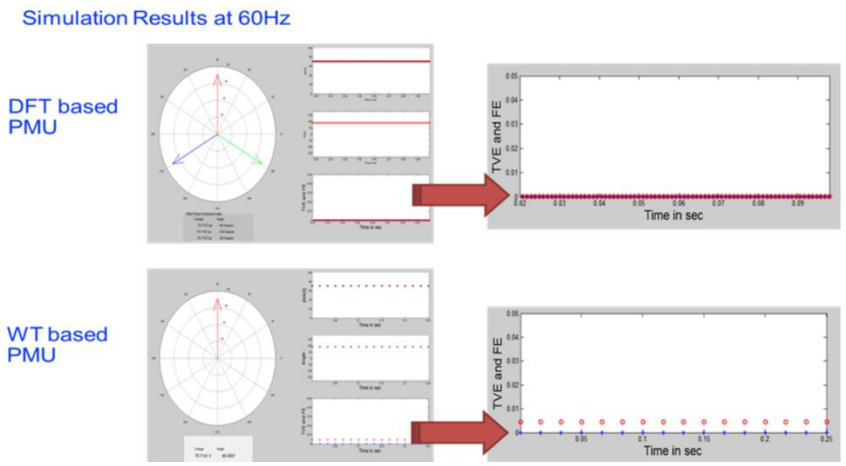


Figure 1.9: The Comparison of DFT/WT based PMU at 60Hz

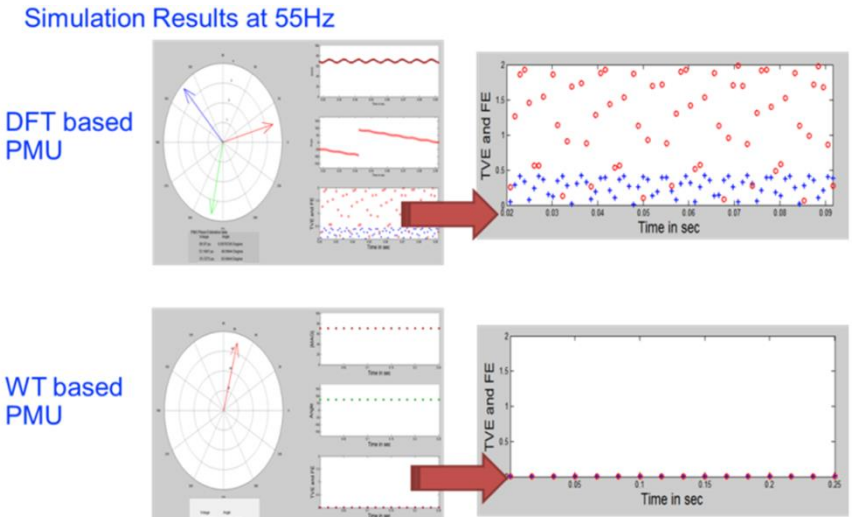


Figure 1.10: The Comparison of DFT/WT based PMU at 55Hz

2. Switching Algorithms

Architecture of adaptive PMU requires alternative phasor estimation algorithms and a switching algorithm to switch to the most suiTable algorithm depending on operating condition intelligently. Goal is to have minimum TVE possible for all operating conditions. Alternative algorithm using wavelet transform in dynamic condition is possible as a solution. Progress was made to design a switching algorithm to switch to a suiTable phasor/frequency algorithm in this time period. For this algorithm, signal processing is being used to determine mean square error and indicate either steady state condition or dynamic condition.

2.1. Dynamic Operating Condition Detector

Average power transmitted per cycle is constant under steady state condition with permitted noise. Once the system condition is changing to dynamic condition, the average power transmitted will

also be changing dynamically. By changing energy per each cycle, simple Energy Theory is applied to determine power system condition as eq. (1.24) [1-9].

$$\text{Energy} = \int_{-\infty}^{\infty} |x(t)|^2 dt \tag{1.24}$$

where x(t) = continuous input signal

In dynamic condition detector, one cycle of energy has been used to calculate transferred energy as in eq. (1.25).

$$\text{Energy}_{\text{per cycle}} = \sum_{0}^{N} |x(n\Delta T)|^2 \tag{1.25}$$

where $x(n\Delta T)$ = discrete input signal and N is number of sampling points per cycle.

The algorithm has been simulated in Matlab as shown in Figure 1.11

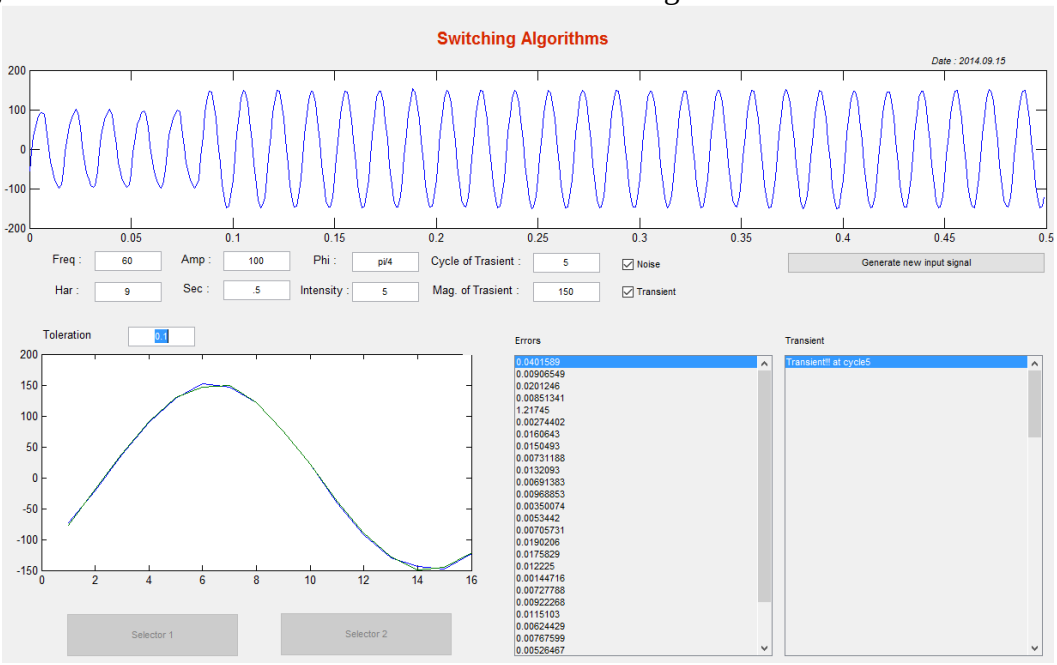


Figure 1.11: Algorithm to Switch between Steady State and Dynamic Conditions

2.2. Application of the Switching Algorithm

In this project, a dynamic condition detector is implemented for switching algorithm. The dynamic condition detector is based on using energy theorem as discussed in section 2.1. Once dynamic conditions have been detected, it will switch to WT based estimator as given in Figure 1.12.

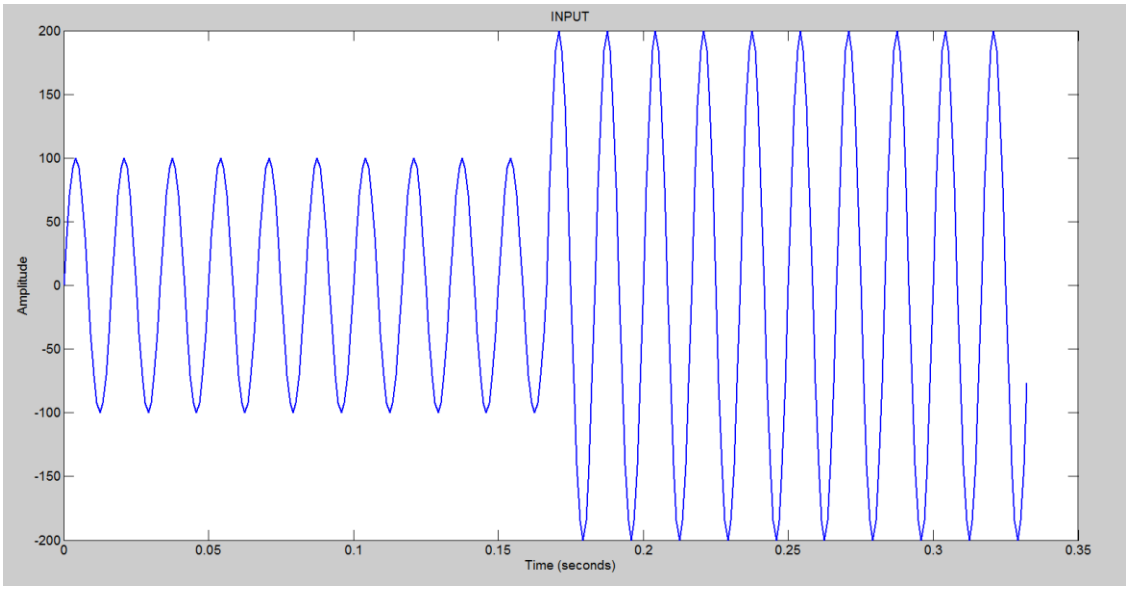


Figure 1.12: Sinusoidal Input Signal with Transient at 10 Cycle

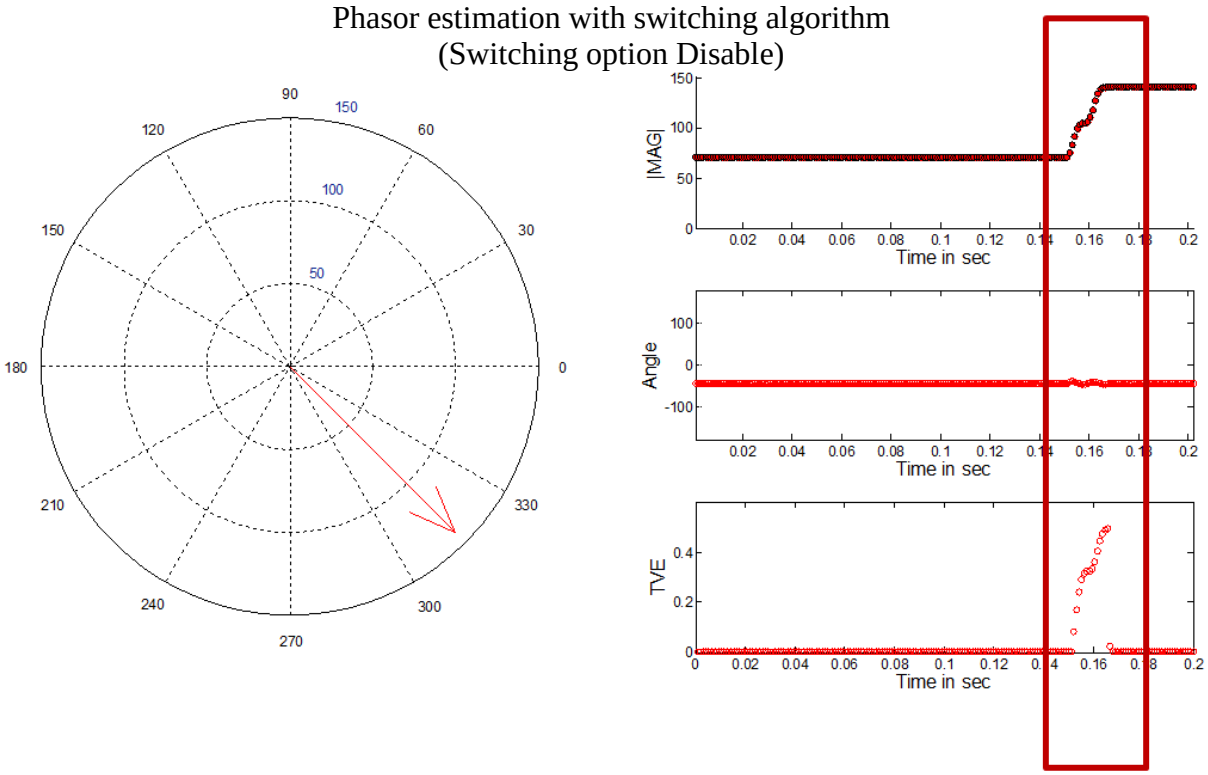


Figure 1.13: Phasor Estimation without Switching Algorithm and DFT during Transient

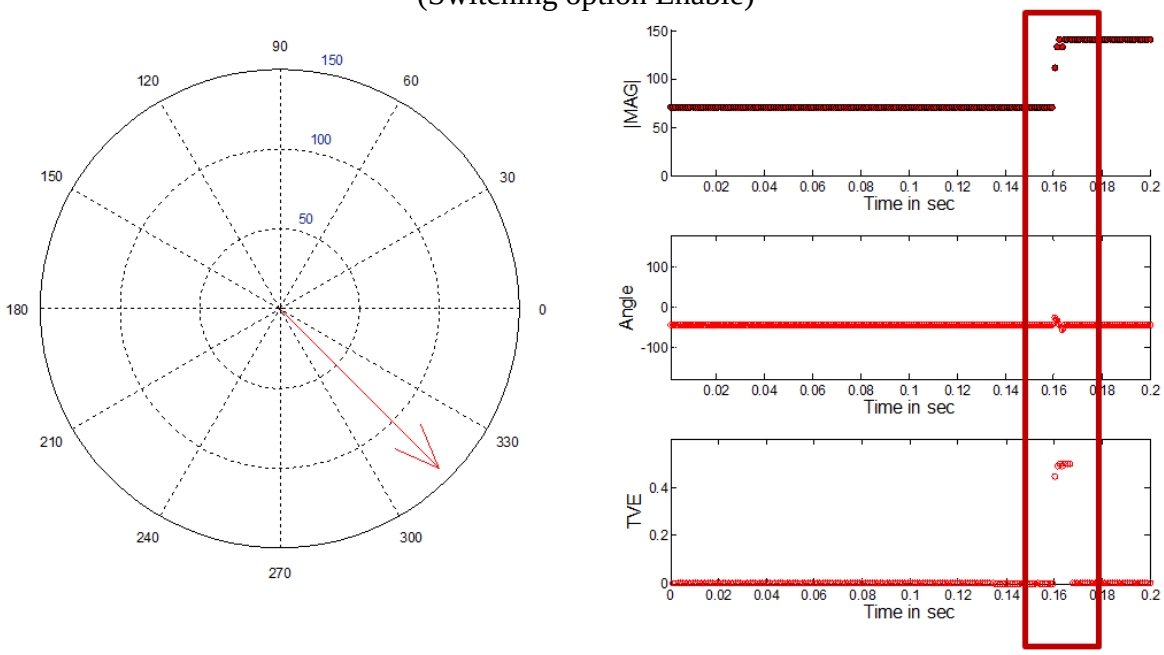


Figure 1.14: Phasor Estimation with Switching from DFT to WT at 10 cycles

As given by simulation result in Figure 1.12 – 1.14, WT based algorithm also has minor TVE error during transients. However, the response time and accuracy has been improved by switching algorithm with WT based estimator.

3. Develop library of phasor estimation algorithms and perform testing of phasor estimation

3.1. Library of Test condition for the proposed PMU

The criteria for PMU performance testing are provided by IEEE Standard C37.118.2011 [1-12]. The standard consists of two parts: C37.118.1 and C37.118.2. C37.118.1 indicates PMU performance testing requirements. C37.118.2 introduces communication protocol requirements for PMU. To perform PMU performance testing, it requires reference PMU. Signal generator has been selected as reference PMU to test proposed PMU. In Smart Grid Demonstration and Research Lab (SGDRIL), WSU, the test conditions are created in RTDS following IEEE Test Suite Specification (TSS) standard. The CT/PT signal is fed by GTAO card in RTDS. Test bed setup ensures that reference PMU and proposed PMU get same signals in same time as given by Figure 1.15.

Figure 1.15: Test Bed at SGDRIL, Washington State University

As shown in Figure 1.16, simple two-bus system is created in RTDS/RSCAD for performing testing using PPA. The test systems are created by this simple two bus system based on testing condition.

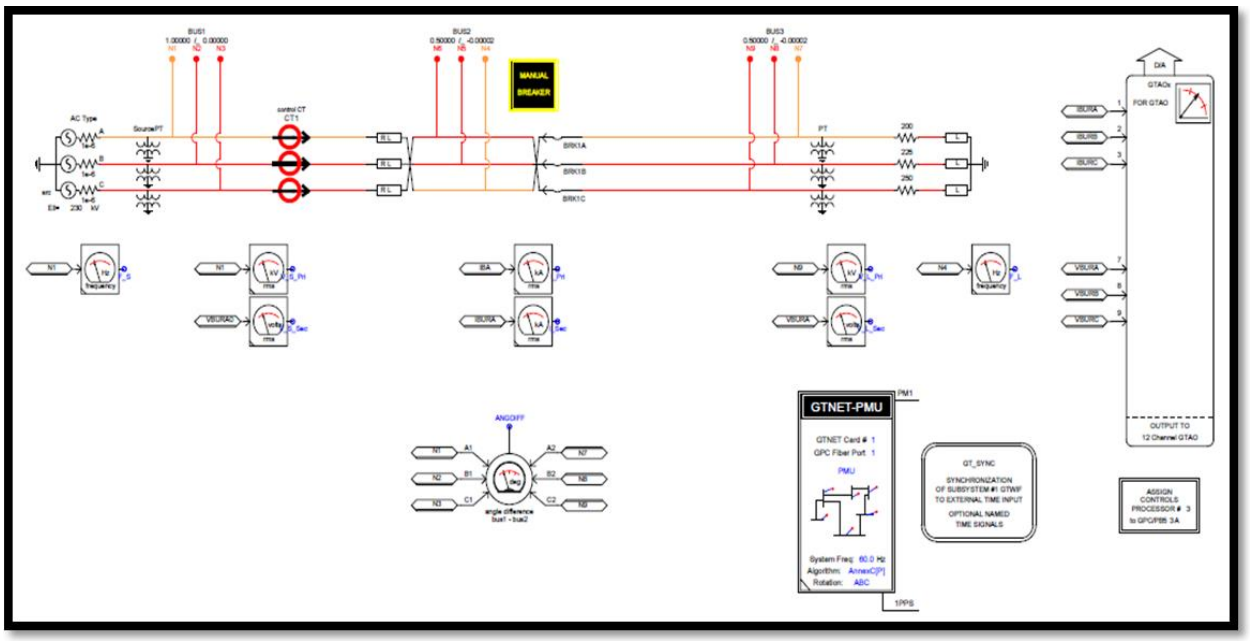


Figure 1.16: Base Test Case System used by PPA

Table 1.1: Library of PMU Testing Part 1: Phasor Estimation

Main Category of selecting Phasor estimation	
Phasor Estimation	Wavelet Transform Based
	DFT Based

Table 1.2: Library of PMU Testing Part 2: Test System Condition

Main Category of PMU Testing	**Quantities Changed during Testing**	**System Conditions during Testing**	**PMU Performance Evaluation Parameters**
Steady State Tests	Voltage & Current Magnitude	System is balanced	TVE, FE, RFE
		System is at off-nominal frequency	TVE, FE, RFE
		System has harmonics	TVE, FE, RFE
		System is at off-nominal frequency and has harmonics	TVE, FE, RFE
	Voltage & Current Angle	System is balanced	TVE, FE, RFE
		System has harmonics	TVE, FE, RFE
	Frequency	System is balanced	TVE, FE, RFE
		System has harmonics	TVE, FE, RFE
Dynamic Tests	Voltage Magnitude Step	System is balanced, at nominal frequency, without harmonics	Response Time, Delay Time, % Peak Overshoot
	Voltage Angle Step	System is balanced, at nominal frequency, without harmonics	Response Time, Delay Time, % Peak Overshoot
	Frequency Step	System is balanced, at nominal frequency, without harmonics	Frequency Response Time, ROCOF Response Time, Delay Time, % Peak Overshoot
	Frequency Ramp	System is balanced, at nominal frequency, without harmonics	FE, RFE
	Amplitude, Phase & Frequency Modulation	System is balanced, at nominal frequency, without harmonics	TVE, FE, RFE

Table 1.1 and Table 1.2 are created following IEEE TSS and IEEE C37.118.1 standard. The Tables show the library of phasor estimation algorithm and test system condition for performing tests. Each phasor estimation algorithm needs to be tested under steady state tests and dynamic tests. Based on the Tables, testing for proposed PMU with switching algorithm is performed under varying system condition. This test is important stage for the proposed PMU. The results are collected and analyzed for switching algorithm. The pre-setting configuration file for switching algorithm is set to provide less TVE for all system condition.

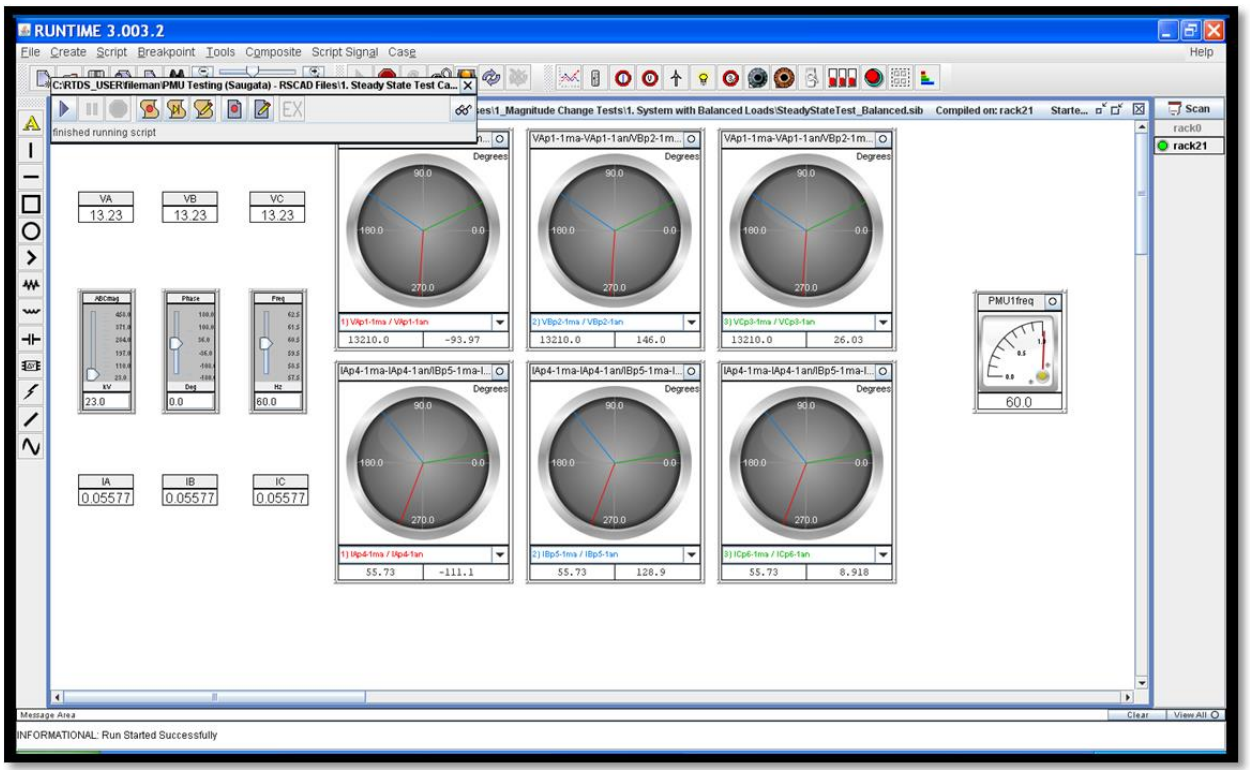


Figure 1.17: Real Time Display Window of RSCAD for Reference Signal Generator

3.2. Simulation Results

Simulation results by Matlab are shown in Figures 1.18 – 1.24.

3.2.1 Steady State Tests

Voltage Magnitude Test

Test condition: Voltage Magnitude Change Test at 60Hz

Voltage Magnitude in pu	Total Vector Error by DFT	Total Vector Error by WT
0.1	1.69E-08	1.69E-08
0.2	1.69E-08	1.69E-08
0.3	1.59E-08	1.59E-08
0.4	1.69E-08	1.69E-08
0.5	1.91E-08	1.91E-08
0.6	1.59E-08	1.59E-08
0.7	1.93E-08	0
0.8	1.69E-08	1.69E-08
0.9	1.50E-08	1.50E-08
1.0	1.91E-08	1.91E-08
1.1	1.73E-08	1.73E-08
1.2	1.59E-08	1.59E-08
1.3	0	2.07E-08
1.4	1.93E-08	0
1.5	1.80E-08	1.80E-08
1.6	1.69E-08	1.69E-08
1.7	1.59E-08	1.59E-08
1.8	1.50E-08	1.50E-08
1.9	2.01E-08	2.01E-08
2.0	1.91E-08	1.91E-08

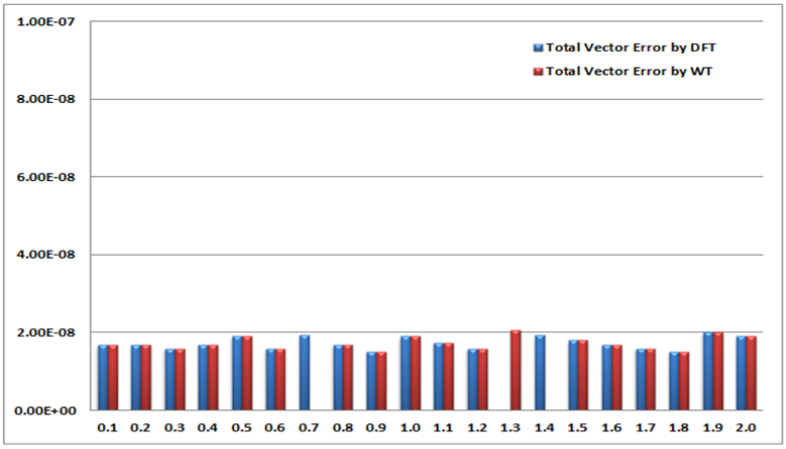


Figure 1.18: Total Vector Error (S.S. Voltage Magnitude at 60Hz)

Test condition: Voltage Magnitude Change Test at 55Hz (left y axis for DFT and right y axis for WT)

Voltage Magnitude in pu	Total Vector Error by DFT	Total Vector Error by WT
0.1	4.47E-02	1.69E-06
0.2	4.47E-02	1.69E-06
0.3	4.47E-02	0.00E+00
0.4	4.47E-02	1.69E-06
0.5	4.47E-02	0.00E+00
0.6	4.47E-02	0.00E+00
0.7	4.47E-02	0
0.8	4.47E-02	1.69E-06
0.9	4.47E-02	0.00E+00
1.0	4.47E-02	0.00E+00
1.1	4.47E-02	0.00E+00
1.2	4.47E-02	0.00E+00
1.3	0.044656919	2.07E-06
1.4	4.47E-02	0
1.5	4.47E-02	0.00E+00
1.6	4.47E-02	1.69E-06
1.7	4.47E-02	0.00E+00
1.8	4.47E-02	0.00E+00
1.9	4.47E-02	0.00E+00
2.0	4.47E-02	0.00E+00

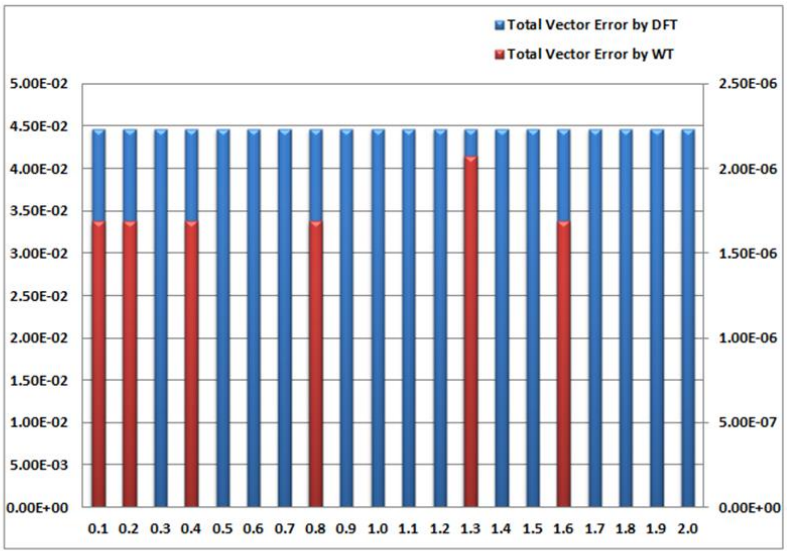


Figure 1.19: Total Vector Error (S.S. Voltage Magnitude at 60Hz)

Test condition: Voltage Magnitude Change Test at 60Hz with 3rd Harmonics

Voltage Magnitude in pu	Total Vector Error by DFT	Total Vector Error by WT
0.1	0.0E+00	1.00E-03
0.2	0.0E+00	1.00E-03
0.3	1.6E-08	1.00E-03
0.4	0.0E+00	1.00E-03
0.5	0.0E+00	1.00E-03
0.6	1.6E-08	1.00E-03
0.7	0.0E+00	1.00E-03
0.8	0.0E+00	1.00E-03
0.9	0.0E+00	1.00E-03
1.0	0.0E+00	1.00E-03
1.1	0.0E+00	1.00E-03
1.2	1.6E-08	1.00E-03
1.3	0.0E+00	1.00E-03
1.4	0.0E+00	1.00E-03
1.5	0.0E+00	1.00E-03
1.6	0.0E+00	1.00E-03
1.7	0.0E+00	1.00E-03
1.8	0.0E+00	1.00E-03
1.9	0.0E+00	1.00E-03
2.0	0.0E+00	1.00E-03

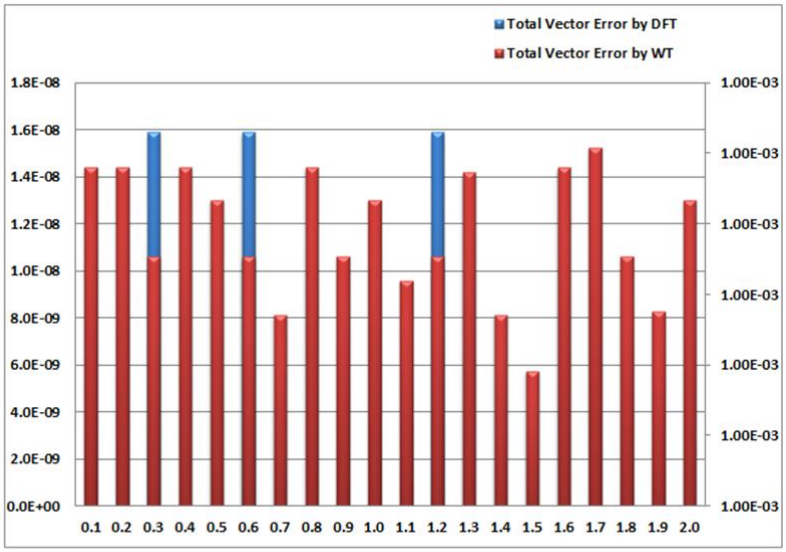


Figure 1.20: Total Vector Error (S.S. Voltage Magnitude at 60Hz with 3rd Harmonic)

Test condition: Voltage Magnitude Change Test at 55Hz with 3rd Harmonics

Voltage Magnitude in pu	Total Vector Error by DFT	Total Vector Error by WT
0.1	4.5E-02	1.00E-03
0.2	4.5E-02	1.00E-03
0.3	4.5E-02	1.00E-03
0.4	4.5E-02	1.00E-03
0.5	4.5E-02	1.00E-03
0.6	4.5E-02	1.00E-03
0.7	4.5E-02	1.00E-03
0.8	4.5E-02	1.00E-03
0.9	4.5E-02	1.00E-03
1.0	4.5E-02	1.00E-03
1.1	4.5E-02	1.00E-03
1.2	4.5E-02	1.00E-03
1.3	4.5E-02	1.00E-03
1.4	4.5E-02	1.00E-03
1.5	4.5E-02	1.00E-03
1.6	4.5E-02	1.00E-03
1.7	4.5E-02	1.00E-03
1.8	4.5E-02	1.00E-03
1.9	4.5E-02	1.00E-03
2.0	4.5E-02	1.00E-03

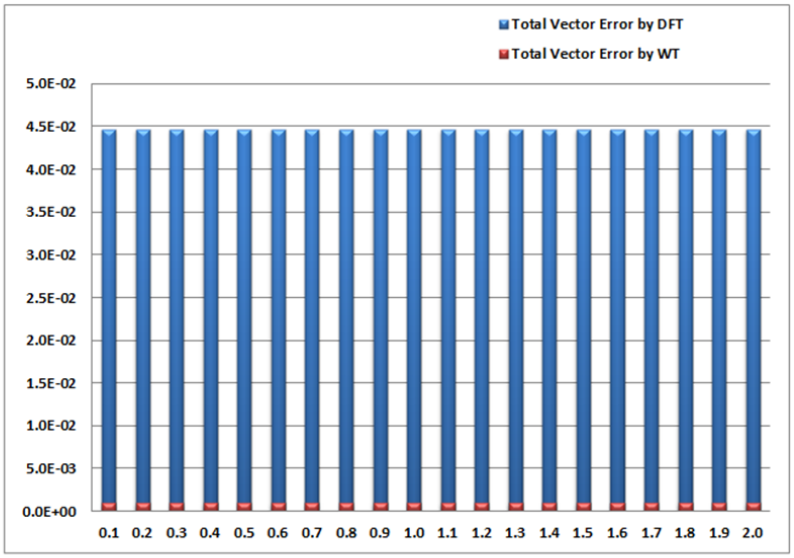


Figure 1.21: Total Vector Error (S.S. Voltage Magnitude at 55Hz with 3rd Harmonic)

Voltage Angle Test

Test condition: Voltage Angle Change Test at 60Hz

Voltage Angle in degree	Total Vector Error by DFT	Total Vector Error by WT
10.0	0.00E+00	0.00E+00
20.0	0.00E+00	0.00E+00
30.0	0.00E+00	0.00E+00
40.0	0.00E+00	0.00E+00
50.0	0.00E+00	0.00E+00
60.0	1.91E-08	1.91E-08
70.0	0.00E+00	0.00E+00
80.0	1.91E-08	1.91E-08
90.0	0.00E+00	0.00E+00
100.0	0.00E+00	0.00E+00
110.0	0.00E+00	0.00E+00
120.0	0.00E+00	0.00E+00
130.0	0.00E+00	0.00E+00
140.0	0.00E+00	0.00E+00
150.0	0.00E+00	0.00E+00
160.0	0.00E+00	0.00E+00
170.0	0.00E+00	0.00E+00
180.0	0.00E+00	0.00E+00
190.0	1.91E-08	1.91E-08
200.0	0.00E+00	0.00E+00

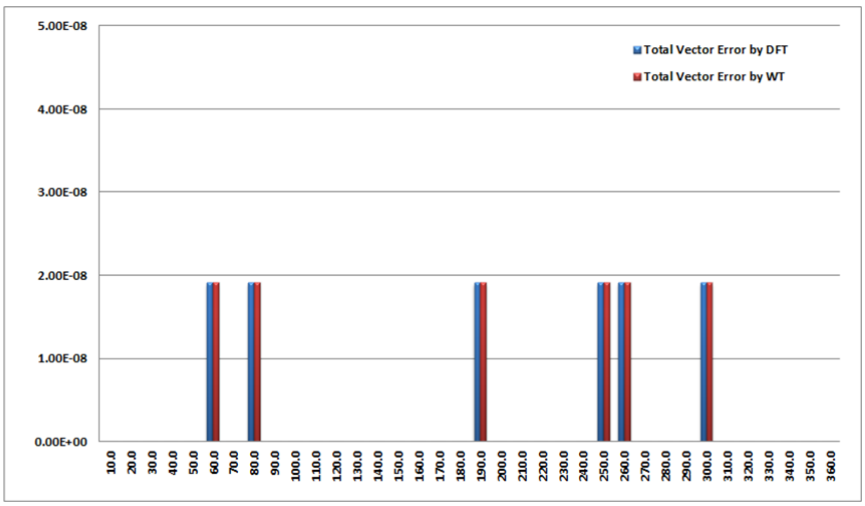


Figure 1.22: Total Vector Error (S.S. Voltage Angle at 60Hz)

Test condition: Voltage Angle Change Test at 55Hz

Voltage Angle in degree	Total Vector Error by DFT	Total Vector Error by WT
10.0	5.20E-02	0.00E+00
20.0	5.83E-02	0.00E+00
30.0	6.29E-02	1.91E-06
40.0	6.57E-02	0.00E+00
50.0	6.64E-02	0.00E+00
60.0	6.51E-02	0.00E+00
70.0	6.22E-02	0.00E+00
80.0	5.77E-02	0.00E+00
90.0	5.22E-02	0.00E+00
100.0	4.58E-02	0.00E+00
110.0	3.91E-02	1.91E-06
120.0	3.25E-02	0.00E+00
130.0	2.69E-02	0.00E+00
140.0	2.40E-02	0.00E+00
150.0	2.51E-02	0.00E+00
160.0	3.00E-02	0.00E+00
170.0	3.70E-02	0.00E+00
180.0	4.47E-02	1.91E-06
190.0	5.20E-02	1.91E-06
200.0	5.83E-02	1.91E-06

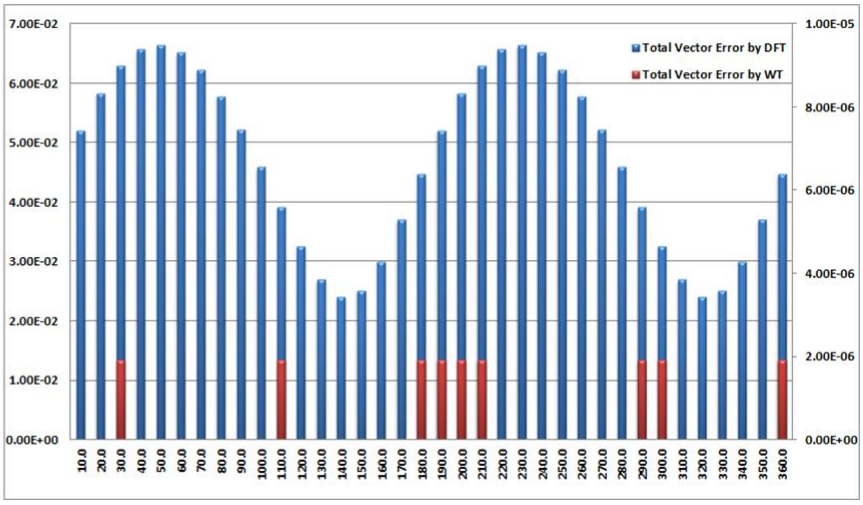


Figure 1.23: Total Vector Error (S.S. Voltage Angle at 55Hz)

Frequency Test

Frequency Change Test from 55Hz to 65Hz

Voltage Magnitude in pu	Total Vector Error by DFT	Total Vector Error by WT
55.0	4.47E-02	0.00E+00
55.1	4.37E-02	0.00E+00
55.2	4.28E-02	1.91E-06
55.3	4.18E-02	0.00E+00
55.4	4.09E-02	0.00E+00
55.5	3.99E-02	0.00E+00
55.6	3.90E-02	0.00E+00
55.7	3.80E-02	0.00E+00
55.8	3.71E-02	0.00E+00
55.9	3.61E-02	1.91E-06
56.0	3.52E-02	0.00E+00
56.1	3.42E-02	0.00E+00
56.2	3.33E-02	0.00E+00
56.3	3.23E-02	0.00E+00
56.4	3.14E-02	0.00E+00
56.5	3.05E-02	0.00E+00
56.6	2.95E-02	0.00E+00
56.7	2.86E-02	0.00E+00
56.8	2.77E-02	0.00E+00
56.9	2.68E-02	0.00E+00
57.0	2.58E-02	0.00E+00
.		
.		
.		

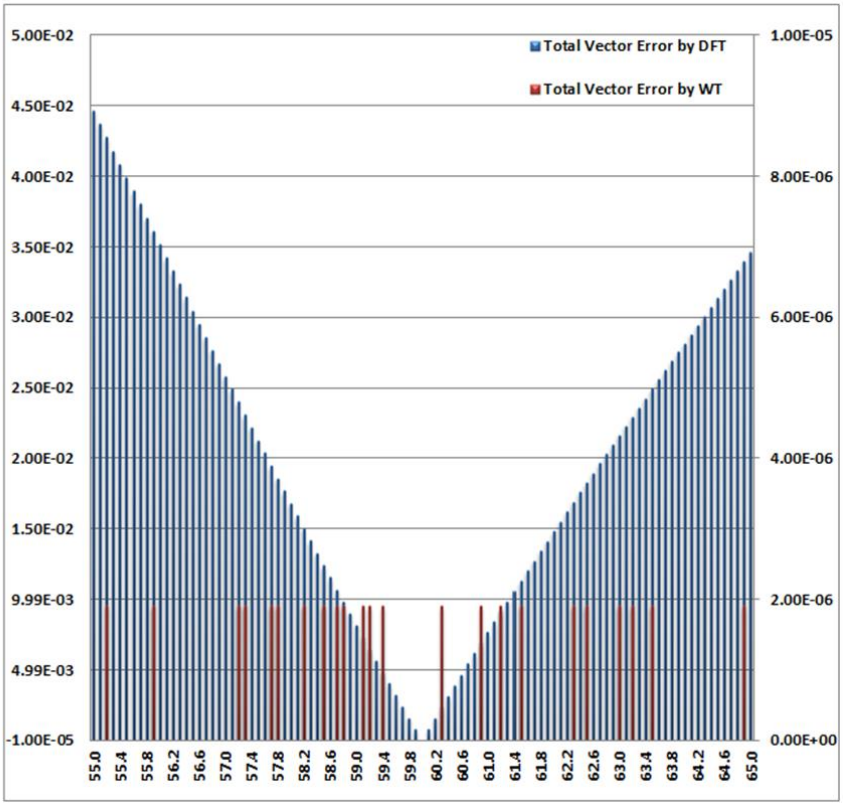


Figure 1.24: Total Vector Error (S.S. Frequency from 55 to 65Hz)

3.2.2 Dynamic Response Test

Delay time

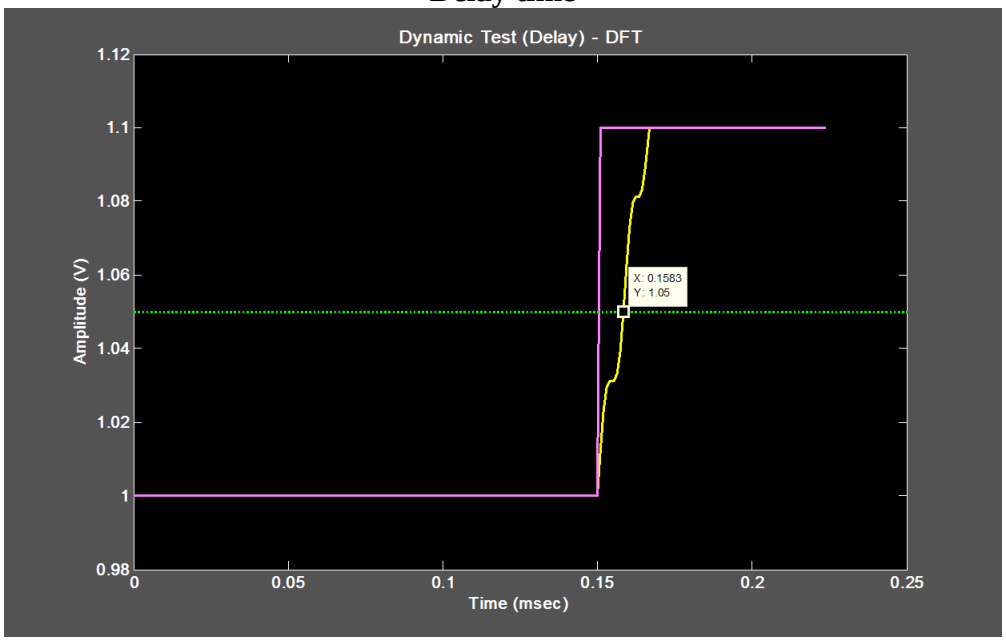


Figure 1.25: Delay Time for DFT based PMU

For delay testing, the amplitude is increased from 1pu to 1.1pu at 0.15msec. According to IEEE STD. C37.118 2011, delay time is determined based on a value in the middle of stepped parameter. In this case, 1.05pu is the middle of stepped parameter.

$$\text{Delay time} = 0.1583 \text{ msec} - 0.15\text{msec} = 0.0083 \text{ msec} = 8.3 \text{ µsec}$$

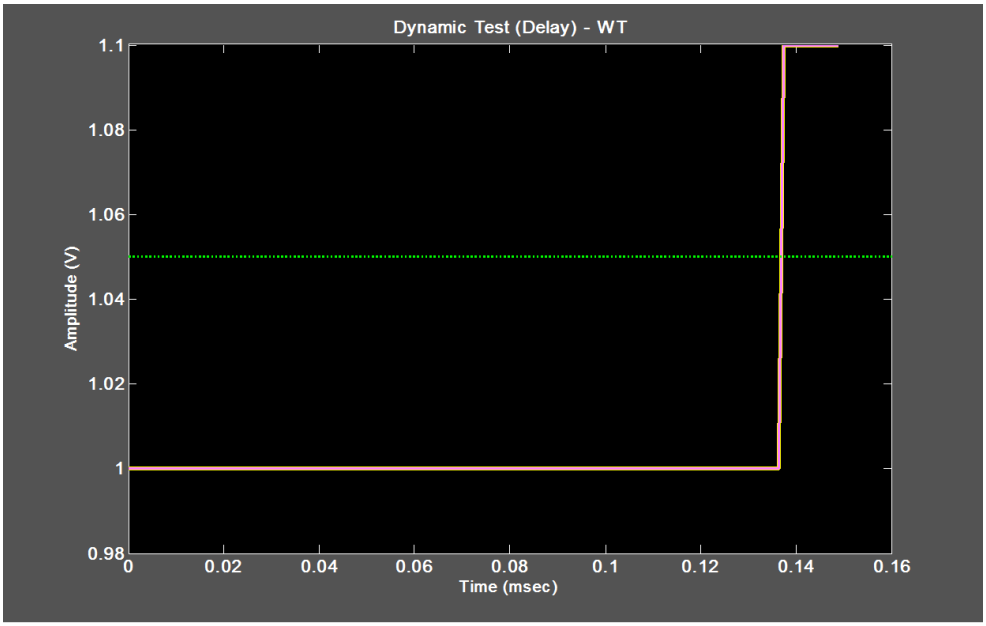


Figure 1.26: Delay Time for WT based PMU

For delay testing, the amplitude is increased from 1pu to 1.1pu at 0.15msec. In this case, 1.05pu is the middle of stepped parameter.

$$\text{Delay time} = 0.1365 \text{ msec} - 0.1365\text{msec} = 0 \text{ msec}$$

Theoretically, WT based PMU response immediately on dynamic response. Once the input data is filtered by analog filtering module, delay time can appear.
Only limited numbers of test cases are shown in this report.

4. Utilizing User Defined Bits for Estimation Error and GPS Status

4.1. Estimation Errors

The adaptive PMU can check self-estimation error and report the estimated error via communication channel (C37.118 or Adaptive PMU protocol) for data quality. One can reconfigure weighting factor of trust on PMU based power system application based on the self-reported PMU data quality. To determine Estimation error, linear regression method is applied that draws trend line as given by eq. (1.26).

$$y = ax + b \tag{1.26}$$

where a = slope of trend line and b = intercept point of y.

The slope and intercept point of trend line is computed following eq. (1.27) and eq. (1.28).

$$\text{a} = \frac{\sum_{i=0}^{n}(x_i + y_i) - \frac{\sum_{i=0}^{n} x_i \sum_{i=0}^{n} y_i}{n}}{\left(\sum_{i=0}^{n} x_i\right)^2 - \frac{\left(\sum_{i=0}^{n} x_i\right)^2}{n}} \tag{1.27}$$

$$\text{b} = \frac{\sum_{i=0}^{n} y_i - a\sum_{i=0}^{n} x_i}{n} \tag{1.28}$$

After finding trend line, outlier is calculated as in eq. (1.29).

$$\text{Maximum Outlier} = \sqrt{\left(\frac{\max(y_{trend} - y_{meas})}{b}\right)^2} \tag{1.29}$$

The simulation result for estimation error is shown in Figure 1.27.

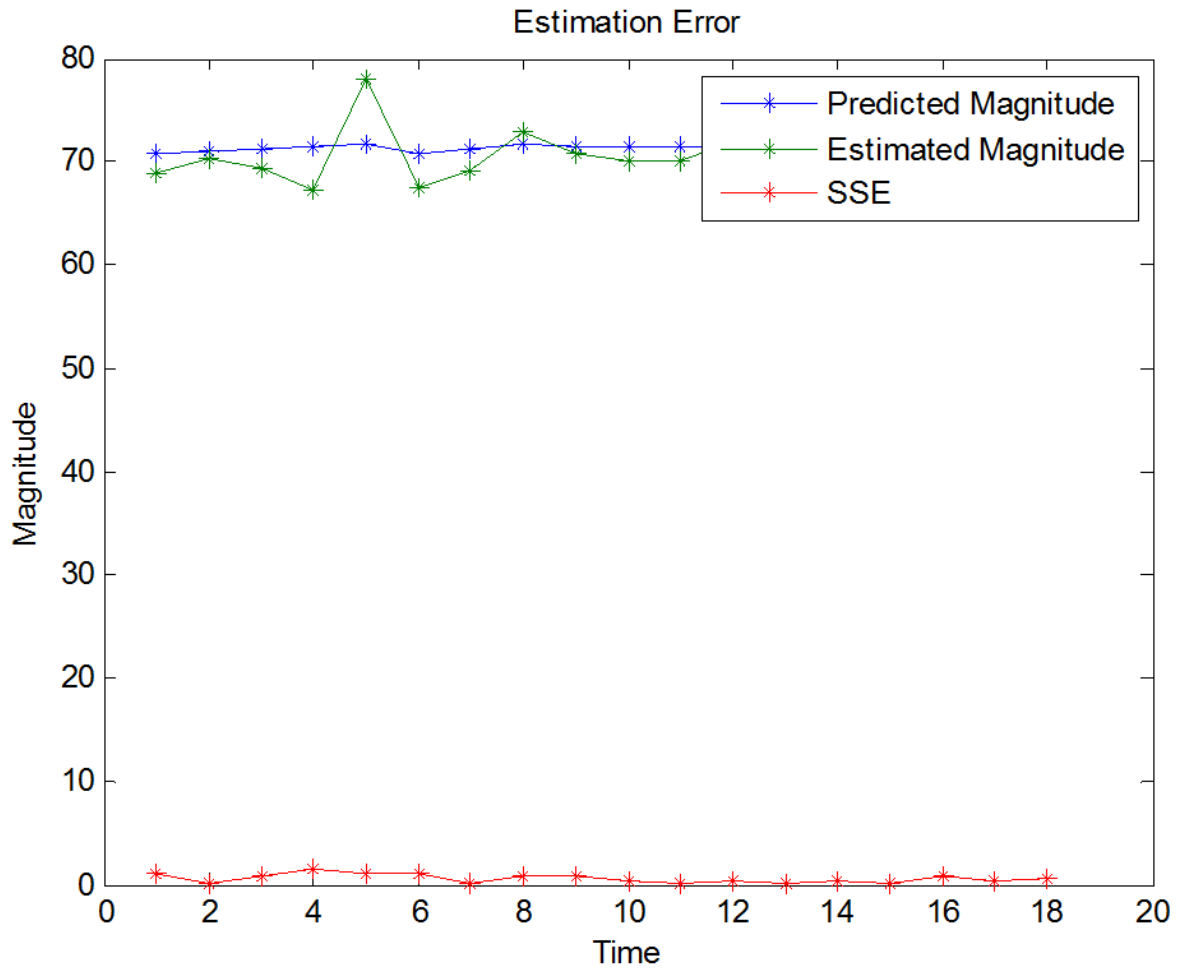


Figure 1.27: Estimation Errors by Linear Regression Method

For this linear regression method, linear equation has been applied to find trend in phasor data. The estimated errors are calculated by modified Sum-of-Squares Error (SSE) method as eq. (1.29) [1-13]. Other possibility for estimation error, quadratic equation based regression method can be used for estimation error under dynamic condition. This work still needs to improve in future.

4.2. PMU GPS status

According to IEEE C37.118 standard, PMU needs to use high accurate timestamp from time signal source. To meet this requirement, PMUs may use 1pps signal from GPS or IRIG-B signal.

NI PMU is used to check GPS status as shown in Figure 1.28.

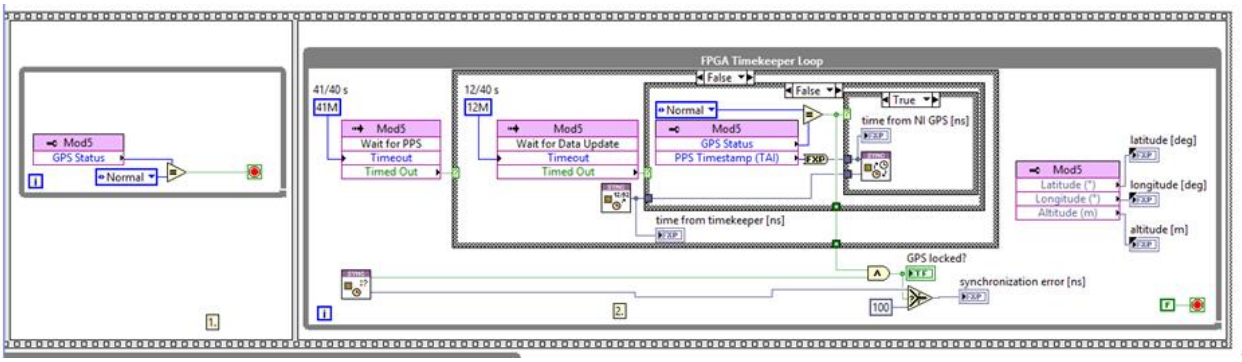


Figure 1.28: NI PMU GPS Block from NI

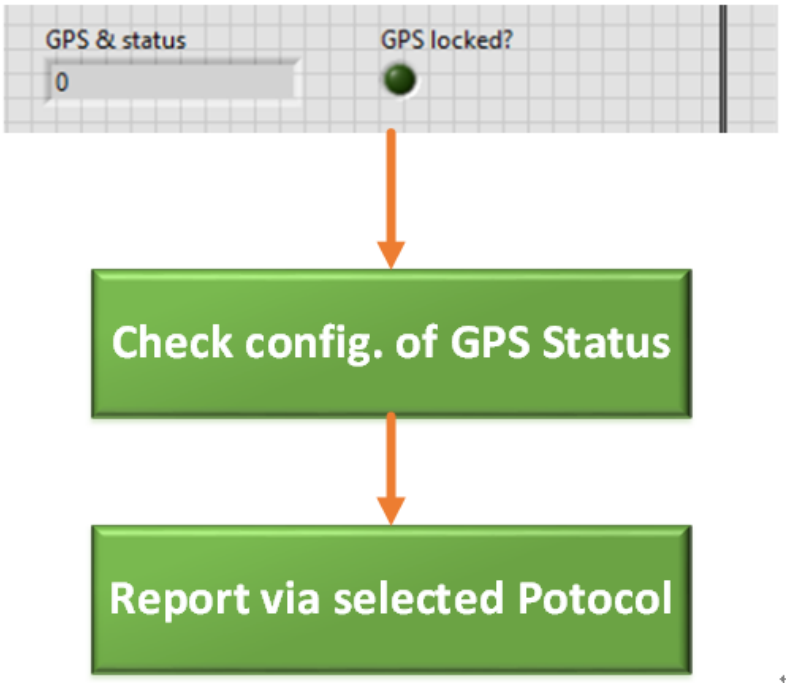


Figure 1.29: Flowchart of Reporting GPS Status

The loss of GPS signal is possible to generate bad data to affect analytical power system applications. To indicate and avoid the error by GPS, adaptive PMU can report GPS status via reserved bits in C37.118.2005 communication protocol or adaptive PMU protocol.

5. Conclusions

Phasor Measurement Units (PMUs) are critical component for real time situational awareness and decision support in power system. Better data quality of PMU is very important to make this possible. Most of the commercial PMUs use Discrete Fourier Transform (DFT) method to estimate phasor and frequency using sampled values. Simple DFT method may have an issue under dynamic system condition.

New algorithms based on wavelet transform (WT) have been developed as part of this project to reduce estimation errors in various power system conditions. WT based phasor estimation can determine system frequency using wavelet. Using estimated frequency and weight matrix, WT estimates phasor. WT estimates phasor with less than 0.1% TVE error for dynamic condition using very short window period in the range 0f 0.5 to 2 cycles. Therefore, WT based PMU helps to make faster control decision. Estimation algorithms can be switched between WT and additional estimation algorithms based on system operating condition to develop adaptive PMU.

Proposed estimation algorithm can be implemented in hardware device and need to be tested under various system conditions. The library of test scenarios has been developed using PMU performance analyzer to test this PMU. For self-reporting PMU, estimation error and GPS status can be reported using user-defined bits.

References

[1-1] De La Ree, J.; Centeno, V.; Thorp, J., and Phadke, A. "Synchronized phasor measurement applications in power systems," *Smart Grid, IEEE Transactions on*, vol. 1, no. 1, pp. 20–27, June 2010.

[1-2] Akke, M.; Thorp, J.S. "Sample Value Adjustment Improves Phasor Estimation at Off-Nominal Frequencies," *Power Delivery, IEEE Transactions on*, vol. 25, no. 4, pp. 2255-2263, October 2010.

[1-3] Funaki, T.; Tanaka, S. "Error estimation and correction of DFT in synchronized phasor measurement," *Transmission and Distribution Conference and Exhibition, Asia Pacific, IEEE/PES*, vol. 1, pp. 448-453, October 6-10, 2002.

[1-4] Phadke, A.G.; and J.S. Thorp. "*Synchronized Phasor Measurement and their Applications*", Springer, 2008.

[1-5] Jun-Zhe, Y; Chih-Wen, Liu. "A precise calculation of power system frequency and phasor," *Power Delivery, IEEE Transactions on*, vol. 15, no. 2, pp. 494-499, April 2000.

[1-6] Banerjee, P.; Srivastava, S.C. "A Subspace-Based Dynamic Phasor Estimator for Synchrophasor Application," *Instrumentation and Measurement, IEEE Transactions on*, vol. 61, no. 9, pp. 2436-2445, September 2012.

[1-7] Chaari, O.; Meunier, M. "A recursive wavelet transform analysis of earth fault currents in Petersen-coil-protected power distribution networks," *Time-Frequency and Time-Scale Analysis, Proceedings of the IEEE-SP International Symposium on*, pp. 162-165, 25-28 October 1994.

[1-8] Ren, J.; Kezunovic, M. "A wavelet method for power system frequency and harmonic estimation," *North American Power Symposium (NAPS)*, pp. 1-6, 26-28 September 2010.

[1-9] Proakis, J.; and Manolakis, D. *Digital Signal Processing: Principles, Algorithms and Applications (3rd Edition)*, Prentice Hall, October 5, 1995.

[1-10] Duric, M.B.; Durisic, Z.R. "Frequency measurement in power networks in the presence of harmonics using fourier and zero crossing technique," *Power Tech*, pp. 1-6, Russia, June 27-30, 2005.

[1-11] IEEE Standard for Synchrophasor Measurements for Power Systems," IEEE Std C37.118.1-2011 (Revision of IEEE Std C37.118-2005), pp. 1-61, December 28, 2011.

[1-12] Biswas, S.S.; Jeong Hun Kim, A. K. Srivastava. "Development of a smart grid test bed and applications in PMU and PDC testing," *North American Power Symposium (NAPS)*, pp. 1-6, September 9-11, 2012.

[1-13] Kutner, M. H.; C. J. Nachtsheim, and J. Neter. (2004), "Applied Linear Regression Models", 4th ed., p. 25, McGraw-Hill/Irwin, Boston, 2004.

Part 2: Standard and Virtual PMU, Interoperability and Application for Setting-Less Protection

1. Introduction

Work done at Georgia Tech is reported in this part.
The value of phasor measurement units (PMUs) comes from their ability to synchronize the voltage and current phasor measurements. The PMU performance changes with operating conditions and whether a PMU can accurately track the primary signal of power systems under the dynamic condition becomes crucial. Typically, PMU uses data samples taken from the waveform, and apply the Discrete Fourier Transform (DFT) to compute the phasor. However, under some dynamic conditions, the traditional DFT leads to error because of the spectral leakage, which causes blunder of the oscillation analysis [2-1]-[2-2].

The accuracy of estimated frequency, phasor magnitude or angles heavily depend on the estimation algorithm. In this report, a "Standard PMU" is proposed to provide an accurate computation of the phasors under continuously changing frequency of the system and under the presence of harmonics. The approach towards this PMU is to use a variable time window size that depends on the frequency of the waveforms and perform an accurate discrete Fourier for variable frequency. Spectral leakage and other known errors of the Fourier transform for a signal of changing frequency are eliminated. Georgia Tech has developed a virtual PMU program with implementation of the Standard PMU algorithm. In the Virtual PMU, this issue is addressed using fractional sample correction. This method accurately evaluates the Fourier integral by taking into account the contributions of the fractional end intervals that occur whenever the waveform period is not an integer multiple of the sampling rate. The fractional sample correction method has been implemented using trapezoidal and quadratic approximations. The integral of a quadratic equation is computed exactly over any arbitrary duration interval, given only three equally spaced samples. The performance of this Standard PMU algorithm is evaluated with variable sampling rates (1 to 10 ks/s) and different approximation methods. Georgia Tech also sets up a laboratory to test IEDs with PMU functionality and the hardware devices are specified in this report.

One of the applications is dynamic state estimation based protection, aka setting-less protection [2-3]. We have developed a protective program to receive real-time data through the merging unit and performs dynamic state estimation using the dynamic model of the protection zone. The dynamic state estimation simply determines how well the measured data fit the dynamic model of the protection zone. When the fit is within the accuracy of the meters by which the measurements are taking, the dynamic state estimator provides the true operating condition of the protection zone. Discrepancies indicate an internal abnormality. The scheme takes decisions based on the operating conditions of the protection zone. This scheme does not require any coordination with other protection functions. Experiment results show this setting-less protection method provides better performance than traditional schemes.

Interoperability/interchangeability is an important issue for the intelligent PMU [2-4]. The key to interoperability/ interchangeability is the ability to characterize the temporal and spatial definition of the data that are coming out of a PMU. For example if the data from the PMU are to be used in a specific application, such as state estimation, it is important that the data contain information of

(a) what time were taken, and (b) where in the network the data were measured, for example, generator A, phase A current or breaker X, phase B, etc. IEC61850 is supposed to enable this type of data through their SCL. We have proposed the State and Control Algebraic Quadratic Companion Form (SCAQCF) object, which is compatible with the IEC 61850, to support the interoperability of PMU.

It is also important to note that the type of information needed for full interoperability cannot be effective unless all devices/PMUs and other IEDs comply with the requirements. Note that for knowledge of the network condition, it is necessary that all devices provide information so that the network condition can be reconstructed.

2. Develop Standard PMU

The objective of the Standard PMU algorithm is to compute the phasor from a sequence of uniformly sampled power system waveform (voltage or current) as accurately as possible, and efficiently enough so that the computations can be performed in real time with the signal acquisition. In this section, the principle of standard PMU algorithm is first illustrated. Then the fractional sample correction method and frequency tracking are introduced. The performance of the Standard PMU algorithm is evaluated with variable sampling rate, and finally the virtual PMU are introduced.

2.1. Principle of the Standard PMU Algorithm

The phasor of a continuous time waveform *v(t)* is defined as [5]

$$\overrightarrow{V_k} = \frac{\omega}{2\pi}\left(\int_0^T \cos(\omega t) v(t) dt - j \int_0^T \sin(\omega t) v(t) dt \right)$$

where T is the period of the fundamental frequency of the waveform *v(t)*, and ω is the fundamental frequency of v(t). The analog current and voltage inputs are filtered by anti-aliasing filter to avoid aliasing errors. Then the signals are sampled by the A/D converter and the above equation is approximated as follows:

$$\vec{V} = \frac{\omega \Delta t}{2\pi}\left(\sum_{k=k_1}^{k_2} a_k \cos\left(\omega k \Delta t\right) v_k - j \sum_{k=k_1}^{k_2} a_k \sin\left(\omega k \Delta t\right) v_k \right)$$

where Δt is the A/D converter sampling interval and a_k are the waveform samples. The above expression yields the exact phasor of a sampled waveform only if the waveform period is an integer multiple of the sampling rate, so that the summation index range spans an integer number of fundamental frequency periods. Since practical data acquisition systems operate with a fixed sampling rate while the power system frequency varies, in general the waveform period is not an integer multiple of the sampling rate. This results in a significant error in the phasor computation. Regarding this issue, the Standard PMU algorithm is proposed to solve this problem.

The standard PMU algorithm is based on the observation that power system frequency changes slowly. The standard PMU monitors the power frequency, selects the time window as an integer number of power periods. For example, if the PMU reports 30 phasors per second as shown in Figure 2.1, then the time window is 2 cycles of the power periods. Similarly, if the PMU reports 60 phasors per second as shown in Figure 2.1, then the time window is 1 cycles of the power periods. The Standard PMU algorithm performs a discrete Fourier transformation over the selected time window.

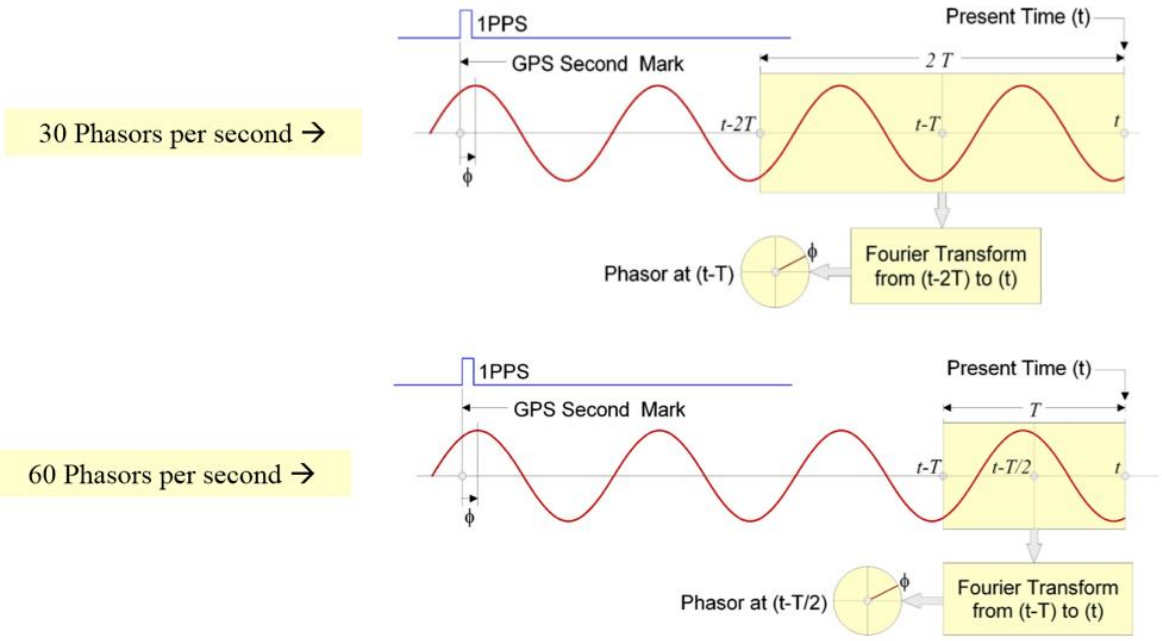


Figure 2.1: Time Window of the Standard PMU Algorithm

However, because the integration window is varying while the sampling rate is constant, the number of sampling intervals within the integration window is not an integer. This problem is solved by introducing the fractional sample correction method.

2.2. Fractional Sample Correction

The Standard PMU algorithm accurately evaluates the Fourier integral by taking into account the contribution of the fractional sampling interval occurring at the end of the integration window. For example, consider a function illustrated in Figure 2.2. A single period of the function to be integrated spans 17.5 sampling intervals. Computing the integral over a single cycle using proper integration methods requires the summation of the 17 areas located between successive samples, plus the yellow area that spans a fraction of a sampling interval. Traditional method only counts the 17 samples, which introduces error in the phasor computation because of omission of yellow fractional area. With the fractional sample correction method, the omitted yellow fractional area can be computed from the sampled values bracketing the sampling interval, by using proper approximation methods. The approximation methods can be either trapezoidal integration or quadratic integration method. This procedure results in a modified expression for the discrete Fourier transform that takes into account fractional samples.

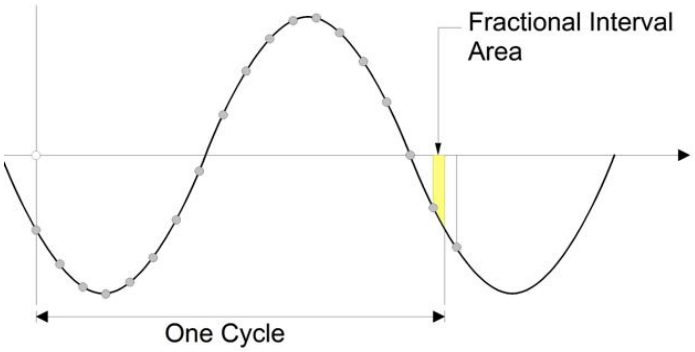


Figure 2.2: Fractional Sample Correction Examples

2.3. Fractional Sample Trapezoidal Correction

Refer to Figure 2.3 and consider the sampled function f_k for which we are evaluating the integral using the Trapezoidal integration method.

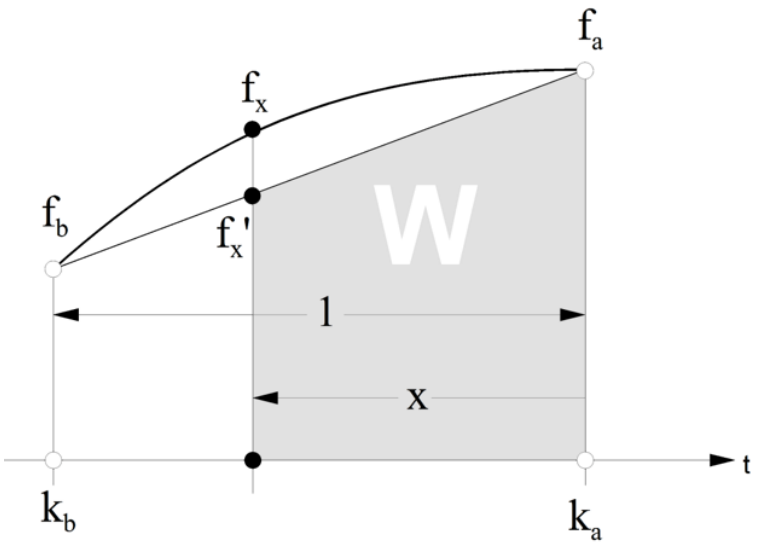


Figure 2.3: Fractional Sample Trapezoidal Integration

The area of the end trapezoidal is computed by first linearly interpolating the waveform at the end point x and then computing the area W as

$$W = (f_a + f_x)\frac{x}{2}$$

where $f_x = (1 - x)f_a + xf_b$

Substitute the f_x into W we can get:

$$W = Af_a + Bf_b$$

where $A = x - \frac{x^2}{2}$ and $B = \frac{x^2}{2}$

If the trapezoidal integration is applied on three consecutive samples shown in Figure 2.4, the area W+Y are computed as

$$W + Y = Af_a + Bf_b + \frac{1}{2}(f_a + f_c)$$

Notice that sample f_a also contributes to the next trapezoid. Therefore, a value of $\frac{1}{2}f_a$ is contributed to the total value. Thus the coefficient of A is set to

$$A = \frac{1}{2} + x - \frac{x^2}{2}$$

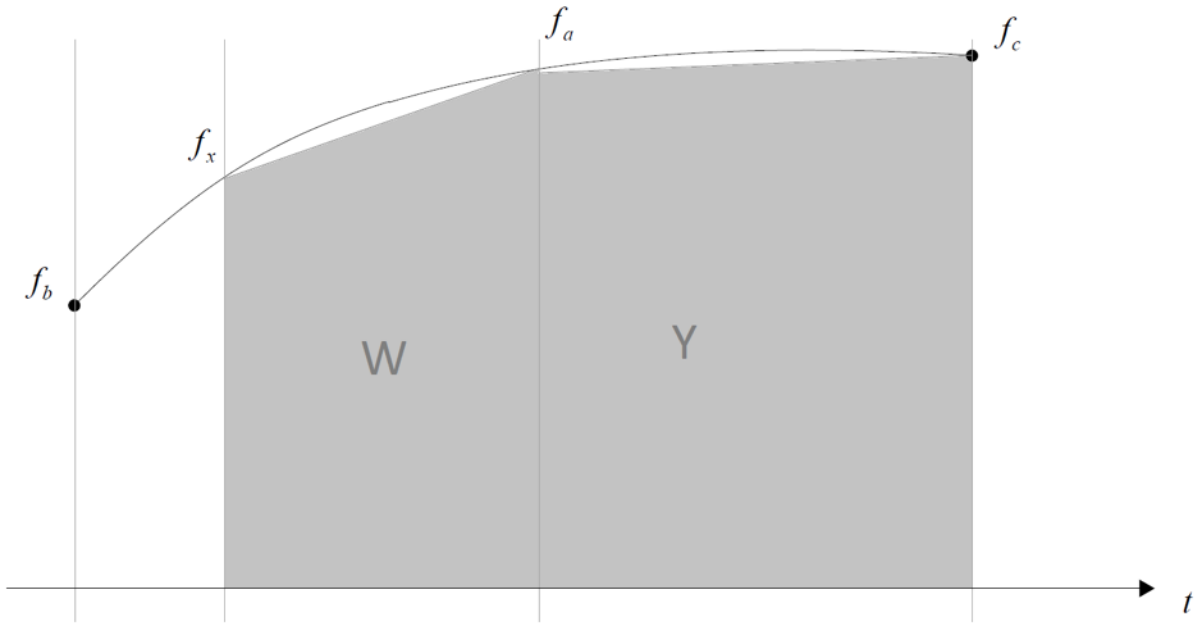


Figure 2.4: Fractional Sample Trapezoidal Integration on Three Consecutive Samples

Using the above results, a recursive algorithm for the computation of a discrete Fourier transform (DFT) of a uniformly sampled waveform f_k is developed by considering two snapshots k-1 and k of consecutive time instances differing by one sample period. The integration time window length is T_{k-1} in the first instant and T_k in the second instant, where $T_k = 2\pi/\omega_{BK}$ and ω_{BK} is the waveform tracked base frequency at time instant k.

Figure 2.5 illustrated the weighted samples that must be summed in each of the time instances to yield the corresponding phasor values:

$$\overrightarrow{V_k} = \sum_{k=k_1}^{k_2} a_k \cos(\omega k \Delta t) v_k - i \sum_{k=k_1}^{k_2} a_k \sin(\omega k \Delta t) v_k$$

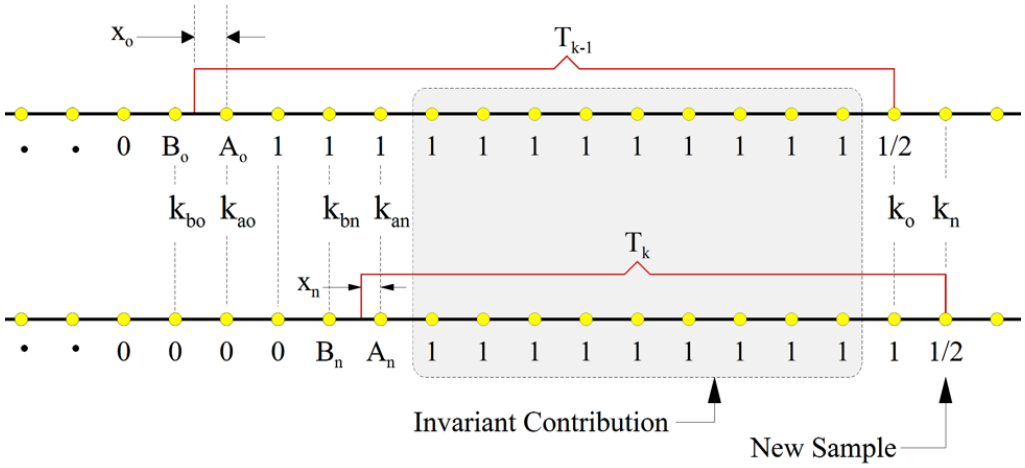


Figure 2.5: Two consecutive snapshots of the integration window

The Figure shows the factors a_k to be multiplied by the $v_k \cos(\omega k \Delta t)$ and $v_k \sin(\omega k \Delta t)$ terms. In the following treatment both these terms are referred as f_k. The number of terms (n) in these summed sequences, and the values of x_o and x_n depend of the waveform base frequency (ω) as follows:

$$T = 2\pi / \omega,$$
$$n = \text{int}\{T / \Delta t\} + 2, \textit{ and}$$
$$x = T / \Delta t - \text{int}\{T / \Delta t\}$$

The frequency ω is periodically estimated (typically every cycle) and thus the above values are adjusted accordingly. (The frequency estimation is presented in the next section).
Recall that from the trapezoidal integration of the fractional time interval:

$$A_0 = \frac{1}{2} + x_0 - \frac{x_0^2}{2} \quad \textit{and} \quad B_0 = \frac{x_0^2}{2}$$
$$A_n = \frac{1}{2} + x_n - \frac{x_n^2}{2} \quad \textit{and} \quad B_n = \frac{x_n^2}{2}$$

where x_o, x_n are the fraction of the first interval contained in the time window of the *old* and *new* instances respectively. Note that the samples enclosed in the gray rectangle have the same coefficients and thus the sum of these terms is the same for both instances. Thus, the recurrence formula is as follows:

$$V_{kn} = V_{k0} + \frac{f_{ka0}}{2} + \frac{f_{kb0}}{2} - A_0 f_{ka0} - B_0 f_{kb0} + A_n f_{kan} + B_n f_{kbn} + S$$

where f_k represents the $\cos(\omega k\Delta t)v_k$ $\sin(\omega k\Delta t)v_k$ terms, and

$$S = -\sum_{k=k_{a0}+1}^{k_{an}} f_k \quad if\ k_{ao} < k_{an}$$

$$S = \sum_{k=k_{a0}+1}^{k_{an}} f_k \quad if\ k_{ao} > k_{an}$$

$$S = 0 \quad if\ k_{ao} = k_{an}$$

Note that if the sampled waveform base frequency does not change between the two consecutive time instants, then $k_{an} = k_{a0} + 1$ and the term simply $S = -f_{kan}$.

2.4. Fractional Sample Quadratic Correction

The fractional sample correction method can also use the quadratic integration. The quadratic fractional sample correction terms are computed by observing that the integral of a quadratic equation can be computed exactly over any arbitrary interval, given only three equally spaced samples. Consider the plot of a quadratic function illustrated in Figure 2.6.

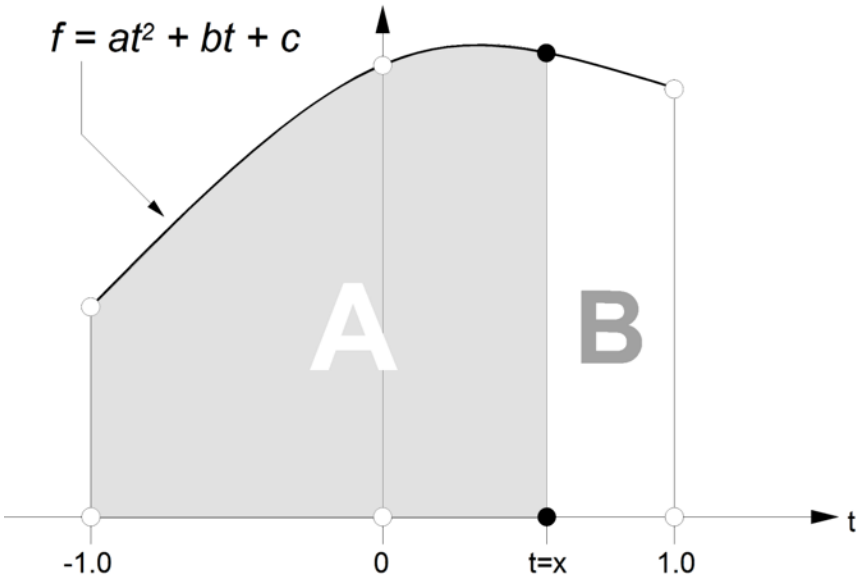


Figure 2.6: Fractional Sample Quadratic Integration

Let $f_{1,2,3}$ be three consecutive samples of a uniformly sampled sequence taken at $t = -1$, $t = 0$ and $t = 1$. Assuming the function f is given by a quadratic equation with coefficient a, b, c, yields:

$$f_1 = f(-1) = a - b + c$$

$$f_2 = f(0) = c$$

$$f_3 = f(1) = a + b + c$$

Solving for the coefficients a, b, c yields:

$$a = \frac{f_1}{2} - f_2 + \frac{f_3}{2}$$
$$b = \frac{f_3}{2} - \frac{f_1}{2}$$
$$c = f_2$$

The area is computed by integrating the above from -1 to x:

$$A = \int_{-1}^{x} \left(at^2 + bt + c\right) dt$$
$$= f_1\left(\frac{x^3}{6} - \frac{x^2}{4} + \frac{5}{12}\right) + f_2\left(-\frac{x^3}{3} + x + \frac{2}{3}\right) + f_3\left(\frac{x^3}{6} + \frac{x^2}{4} - \frac{1}{12}\right)$$

The remaining area B can be deduced by symmetry, by exchanging f_1 and f_3, and replacing x with $-x$, which yields:

$$B = f_1\left(-\frac{x^3}{6} + \frac{x^2}{4} - \frac{1}{12}\right) + f_2\left(\frac{x^3}{3} - x + \frac{2}{3}\right) + f_3\left(-\frac{x^3}{6} - \frac{x^2}{4} + \frac{5}{12}\right)$$

Applying the above quadratic approximation to the discrete time phasor computation formula, gives an equation of the general form:

$$\vec{V} = \sum_{k=k_1}^{k_2} a_k \cos\left(\omega k \Delta t\right) v_k - i \sum_{k=k_1}^{k_2} a_k \sin\left(\omega k \Delta t\right) v_k$$

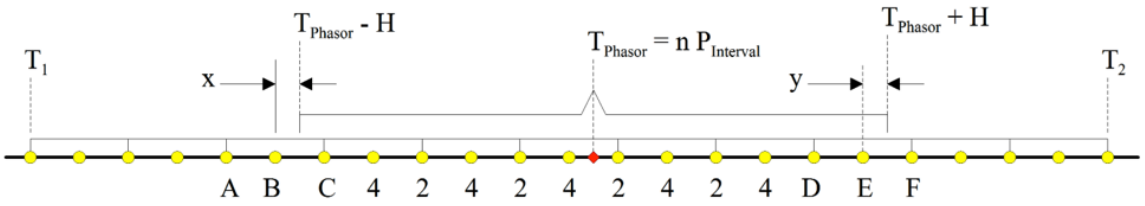


Figure 2.7: Quadratic Integration Time Interval

The time widow T is selected to match the sampled waveform period, where in general the time window end points do not coincide with a sampling instant, as shown in Figure 2.7. Assuming that the integration period starts at $(k_1 + 1 + x)\Delta t$, and ends at $k_2 - 1 + y)\Delta t$, where x and y be real numbers in the interval (-1, 1). Then the coefficients a_k are selected as

for $k = k_1$ $\quad a_k = x^3/6 + x^2/4 - 1/12$
for $k = k_1+1$ $\quad a_k = x^3/3 - x + 2/3$
for $k = k_1+2$ $\quad a_k = -x^3/6 + x^2/4 + 1/4$

for $k = k_1+3$ $a_k = 4/3$
for $k = k_1+4$ $a_k = 1/3$
for $k = k_1+5$ $a_k = 4/3$
…
…
for $k = k_2 – 4$ $a_k = 1/3$
for $k = k_2 – 3$ $a_k = 4/3$
for $k = k_2 – 2$ $a_k = y^3/6 – y^2/4 + 3/4$
for $k = k_2 – 1$ $a_k = –y^3/3 + y + 2/3$
for $k = k_2$ $a_k = y^3/6 + y^2/4 – 1/12$

2.5. Frequency Tracking

Application of the proposed fractional sample correction method requires knowledge of the fundamental frequency of the sampled waveform. Therefore, the frequency tracking algorithm must be included in the implementation. Because the frequency of power system voltage and current waveforms varies slowly, a simple phase rotation based method is sufficient to track the frequency of system, as shown in Figure 2.8.

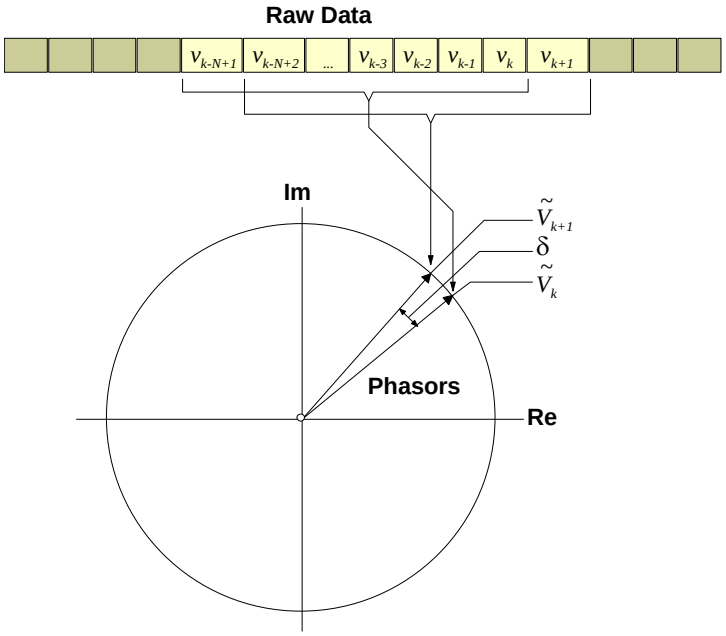


Figure 2.8: Quadratic Integration Time Interval

Specifically, the sampled waveform base frequency is estimated using the rate of change of the computed phasor phase angle. The estimation is typically performed once every cycle of the nominal system frequency. The frequency is computed as

$$\hat{f} = f_{base} + \frac{\delta}{2\pi\Delta t}$$

or

$$\hat{\omega} = w_{base} + \frac{\delta}{\Delta t}$$

2.6. Performance Evaluation

In order to evaluate the performance of the fractional sample correction method, the phase angle error for a 60 Hz waveform was computed as a function of sampling rate. The phase angle error of the quadratic integration method is plotted in Figure 2.9 (green curve) for sampling rates ranging from 1 to 10 kHz. For comparison purposes, the plot also includes the results of the discrete time phasor computation without any fractional sample correction (red curve), and phasor computation with fractional sample correction based on trapezoidal integration (blue curve).

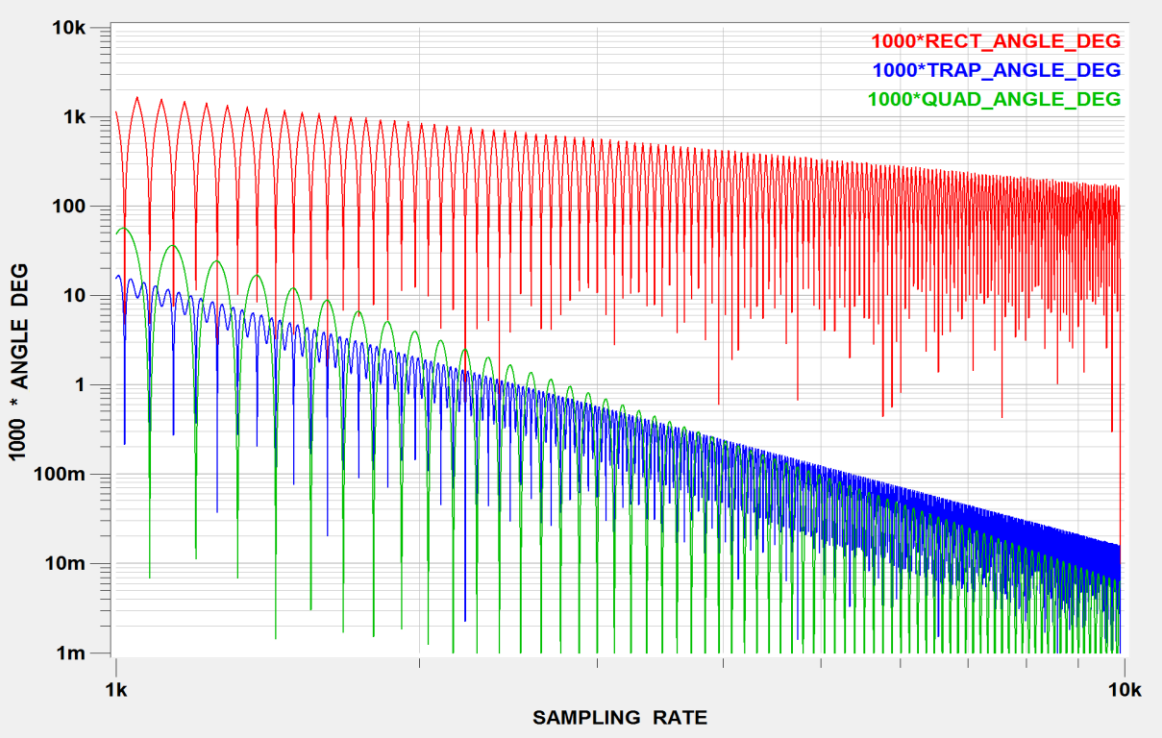


Figure 2.9: Performance Evaluation of Fractional Sample Correction Method

Note that above a sampling rate of 4 kHz the quadratic fractional sample correction yields the lowest phase error (below 0.0001 degrees). Table 3 presents the phase error of the three method at sampling rate of 8 kHz, the quadratic integration method provides the best performance while the non-correction method has the largest phase error.

Table 2.1: Phase error at sampling rate of 8 kHz

Correction Method	Phase Error (Degrees)
None	0.2
Trapezoidal	0.00004
Quadratic	0.00002

3. The Virtual PMU

In performing PMU accuracy evaluation in an absolute sense and under various conditions (such as amplitude and frequency transients, waveform distortion etc.) it is imperative that the output of a PMU under test is compared to the output of a "Standard PMU". Since the existing standards do not presently address this issue, a Standard PMU was designed and implemented using National Instruments data acquisition hardware and custom software. The Standard PMU software was implemented within the WinXFM program. Since the PMU capability was added within the data acquisition system as an object oriented software component we refer to it as a "Virtual PMU".

The virtual PMU is used as a benchmark against which the accuracy of the PMU under test is evaluated. Therefore, it is necessary that the virtual PMU provides accuracy that exceeds the accuracy required by the standards (Total Vector Error below 1%), and remains accurate during typical power system voltage and current amplitude and frequency transients and waveform distortion.

The developed virtual PMU has been integrated within the WinXFM program. Figure 2.10 shows a snapshot of the virtual PMU user interface.

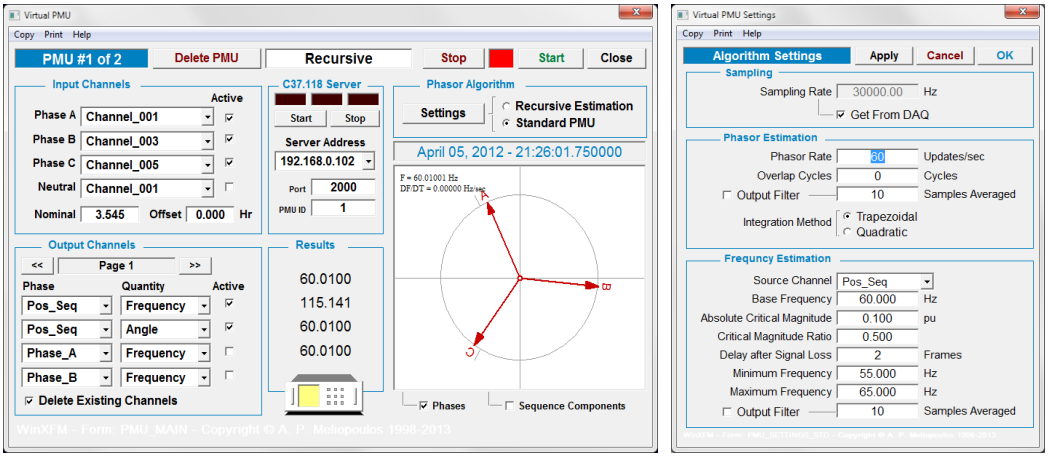


Figure 2.10: Virtual PMU User Interface

4. Testing and Characterization of IEDs with PMU Functionality

Testing and Characterization of IEDs with PMU functionality
The performance metrics of the PMU accuracy are:

- Magnitude Accuracy
- Time or Phase Angle Accuracy
- Frequency Accuracy
- Communication Compliance to IEEE Standard C37.118

The evaluation of these metrics is achieved using a system which consists of a multichannel waveform generation system, a high accuracy reference data acquisition system and a custom

software (program WinXFM) which controls both signal generation and data acquisition and facilitates acquired data analysis. The overall system diagram is illustrated in Figure 2.11.

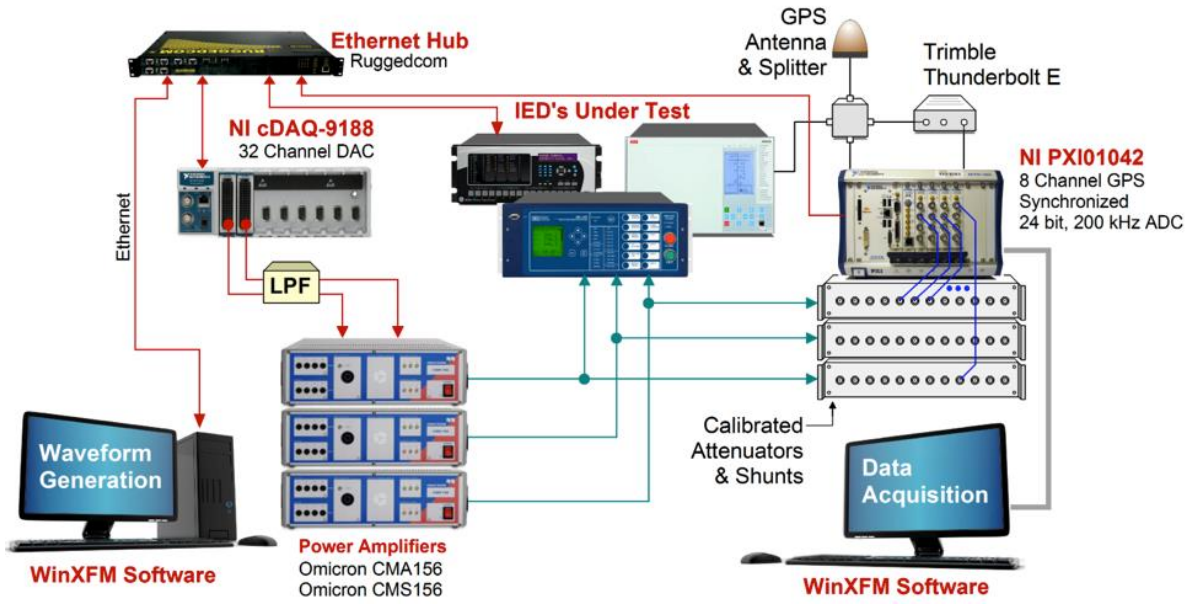


Figure 2.11: Virtual PMU User Interface

1. Waveform Generation. The waveform generation is based on a National instrument 32 channel D/A converter module (NI cDAQ-9188). Personal computer executing WinXFM sends waveform in streaming form to the converter module. The waveform data can either be playback of COMTRADE files or user defined mathematical expressions. The outputs of the D/A converters are amplified by a bank of Omicron amplifiers to standard levels for PMU inputs (120 Volts and 5 Amperes).

The output of the amplifiers are directly connected to the PMU under test and to the reference data acquisition system. WinXFM receives the computed phasors from the PMU under test via C37.118, and compare to the phasors computed by the "Standard PMU" in the same program. This approach allows accurate characterization of the PMU performance metrics, without being affected by delays and frequency response deviations introduced by the waveform generation system components (network delays, digital to analog converter latencies, and amplifiers). The measurement accuracy of the test system is only determined by the accuracy of the reference data acquisition system, described next.

Data Acquisition Hardware. The reference data acquisition system consists of a National Instrument PXI chassis. The PXI chassis contains 8 high accuracy data acquisition channels. Each data acquisition channel uses a dedicated 24 bit analog to digital converter (ADC) with a maximum sampling rate of 200 kHz. The ADC data acquisition clock is synchronized to a 10 MHz oven controlled crystal oscillator (OCXO) disciplined to the 1PPS clock from a GPS receiver (Trimble Thunderbolt E). This system provides sampling synchronized to the UTC time with accuracy better than 100 nanoseconds. The synchronization of the A/D converter sampling clock was verified by tracing the sample clock signal, the 10 MHz reference clock signal and the UTC 1-PPS signal on an oscilloscope. The oscilloscope was set to trigger on the 1-PPS signal and accumulate the history

of the traces over a period of 5 minutes. The results are illustrated in Figure 2.12. Note that the sample clock rising edge stays within 100 nanoseconds of the 1-PPS rising edge.

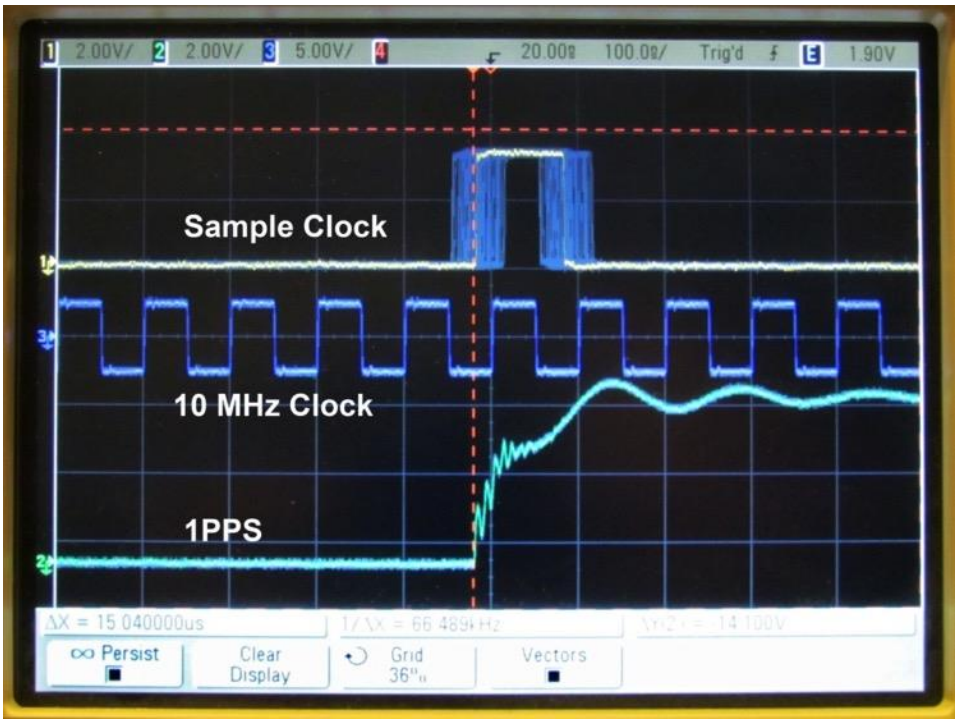


Figure 2.12: Verification of A/D Converter Sample Clock Synchronization

Data Analysis. The data analysis is performed within WinXFM embedded in Windows based National Instrument PXI chassis. Note that the same chassis also houses the A/D converter system. During the testing process, the WinXFM program performs the following tasks: (a) It computes phasors from the data acquired from the reference data acquisition system, and (b) it receives the phasors computed by the PMU under test (via a 37.118 synchrophasor stream, and (c) it compares the reference and test device phasor streams and presents the results in graphical form.

The phasor comparison is accomplished as follows. Both computed reference phasors and phasors from device under test are transferred to WinXFM program channels. WinXFM program channels can be displayed in plot format for comparison. Furthermore, the WinXFM program can generate "computed channels" defined by mathematical expressions in terms of other channels. For example, a phase error channel can be defined as the difference of the phase angle of a reference phasor minus the phase angle of the corresponding phasor computed by the device under test. Similarly frequency and magnitude error channels can be defined. The computed error channels are continuously plotted in real time for a variety of test waveforms, which may include frequency variations and amplitude transients, as well as GPS clock performance issues (such as poor satellite signal reception).

Several groups of tests are conducted:

1) Steady state frequency – where the frequency is constant, and measure a) timing error, b) magnitude error, c) frequency error

2) Frequency ramp – where the waveform is constant phasor with frequency ramping in constant rate, measure the a) timing error, b) magnitude error, c) frequency error

3) Voltage magnitude – the waveform is of variable magnitude for the voltage channels, measure the a) timing error, b) magnitude error, c) frequency error

4) Voltage magnitude step change – the test is waveform of step change in voltage channels, measure the a) timing error, b) magnitude error, c) frequency error

5) Current magnitude – the waveform is of variable magnitude for the current channels, measure the a) timing error, b) magnitude error, c) frequency error

6) Current magnitude step change -- the test is waveform of step change in current channels, measure the a) timing error, b) magnitude error, c) frequency error

5. Utilizing User Defined Bits for Estimation Error and Enhanced Applications

5.1. Setting-Less Protection

The objectives of this project is to develop an adaptive and intelligent PMU for smarter applications. One of the applications is dynamic state estimation based protection, aka setting-less protection. We have developed a protective program to receive real-time data through the merging unit and performs dynamic state estimation-based protection of the protection zone.

In general, the dynamic state estimation simply determines how well the measured data fit the dynamic model of the protection zone. When the fit is within the accuracy of the meters by which the measurements are taking, the dynamic state estimator provides the true operating condition of the protection zone. Discrepancies indicate an internal abnormality. The scheme takes decisions based on the operating conditions of the protection zone. This scheme does not require any coordination with other protection functions.

5.2. Principle of Setting-less Protection

The setting-less protection is an extension of current differential protection, a scheme that does not need coordination with any other protection function. The differential protection function and its extension to dynamic state estimation based protection are illustrated in Figure 2.13. In current differential protection the electric currents at all terminals of a protection zone are measured and their weighted sum must be equal to zero. Thus differential protection monitors the sum of the currents and as long as it is zero or near zero no action is taking. Note that the sum of the currents being equal to zero is dictated by Kirchoff's current law.

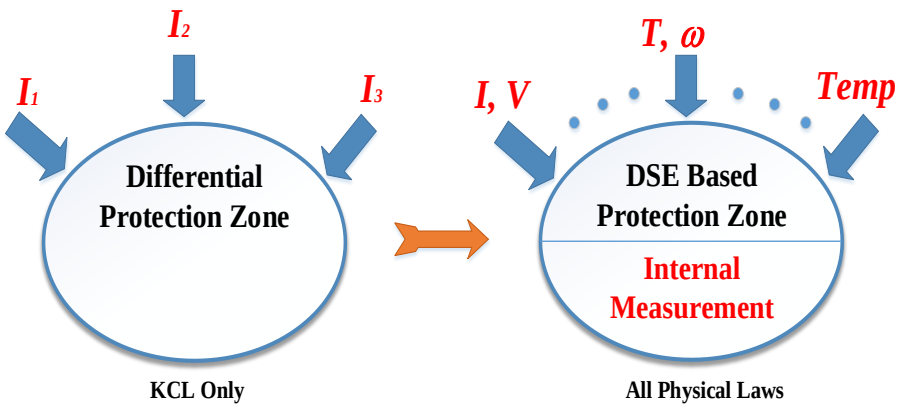


Figure 2.13: Conceptual Illustration of Setting-less Protection

In the setting-less protection, we measure not only currents but also voltages at the terminals of the protection zone and possibly speed and torque in case of rotating machinery, or other internal measurements including thermal measurements. The dynamic model of the device (physical laws such as KCL, KVL, motion laws, thermodynamic laws, etc.) dictates the relationship among all these measurements. The best way to determine whether these relationships are satisfied by the measurements is to perform a dynamic state estimation using the measurement and the dynamic model of the protection zone. This problem has been formulated and reported in earlier papers [2-6].

First the dynamical model of the protection zone is a set of differential and algebraic equations. Our method utilizes a quadratization procedure which reduces any higher order nonlinearities to no more than second order by the introduction of additional variables, if necessary (if model is linear or quadratic this process is not needed). This transformation does not change the model. We refer to this model as the quadratized dynamic model of the protection zone (QDM). The form of the QDM is given below.

Protection zone model:

$$i(t) = Y_{eqx1}\mathbf{x}(t) + D_{eqxd1}\frac{d\mathbf{x}(t)}{dt} + C_{eqc1}$$

$$0 = Y_{eqx2}\mathbf{x}(t) + D_{eqxd2}\frac{d\mathbf{x}(t)}{dt} + C_{eqc2}$$

$$0 = Y_{eqx3}\mathbf{x}(t) + \left\{ \begin{array}{c} \vdots \\ \mathbf{x}(t)^T \left\langle F^i_{eqxx3} \right\rangle \mathbf{x}(t) \\ \vdots \end{array} \right\} + C_{eqc3}$$

Measurements:

$$\mathbf{h}(\mathbf{x}) = Y_{mx}\mathbf{x}(t) + \left\{ \begin{array}{c} \vdots \\ \mathbf{x}(t)^T \left\langle F^i_{mx} \right\rangle \mathbf{x}(t) \\ \vdots \end{array} \right\} + D_{mx}\frac{d\mathbf{x}(t)}{dt} + C_m$$

where ***x*** is the state variables, ***h*** are the measurements. The protection zone model has three sets of equations. The first set of equations are external equations. The left sides of the external

equations are terminal currents. The second and third sets of equations are internal equations of the zone. The second set is linear equations and the third set is nonlinear equations. The measurements are just the mathematical expression for each measurement.

We have developed three approaches to solve the dynamic state estimation problem: Unconstraint weight least square method, constraint weight least square method and the extended Kalman filter method. For the purpose of illustration, we only discuss about the unconstraint weight least square method here.

In this method, the QDM is integrated (in our approach we use the Quadratic Integration Method) to convert the dynamic model of the protection zone into an Algebraic equivalent. The AQCF is shown below:

$$\begin{Bmatrix} i(t) \\ 0 \\ 0 \\ i(t_m) \\ 0 \\ 0 \end{Bmatrix} = Y_{eqx}\mathbf{x}(t,t_m) + \begin{Bmatrix} \vdots \\ \mathbf{x}(t,t_m)^T \left\langle F_{eqx}^i \right\rangle \mathbf{x}(t,t_m) \\ \vdots \end{Bmatrix} + N_{eqx}\mathbf{x}(t-h) + M_{eq}i(t-h) + K_{eq}$$

The measurements are also converted into an algebraic companion form which has the following syntax:

$$\mathbf{h(x)} = Y_{eqx}\mathbf{x}(t,t_m) + \begin{Bmatrix} \vdots \\ \mathbf{x}(t,t_m)^T \left\langle F_{eqx}^i \right\rangle \mathbf{x}(t,t_m) \\ \vdots \end{Bmatrix} + N_{eqx}\mathbf{x}(t-h) + M_{eq}i(t-h) + K_{eq}$$

The internal equality parts of the AQCF are created as constraints in this method. The estimation problem is stated as follows:

$$Minimize \quad J = \sum_{i=1}^{n} \left(\frac{h_i(x,u) - z_i}{\sigma_i} \right)^2 = \sum_{i=1}^{n} s_i^2 = \eta^T W \eta$$

where $s_i = \frac{\eta_i}{\sigma_i}$, $W = diag\left\{ ..., \frac{1}{\sigma_i^2}, ... \right\}$; σ_i is the accuracy of the meter by which the corresponding measurement *z* was obtained; *W* is the diagonal matrix whose non-zero entries are the inverse of the variance of the measurement error.

The solution is given with the following iterative algorithm:

$$\mathbf{x}^{\nu+1} = \mathbf{x}^{\nu} - (H^T W H)^{-1} H^T W (\mathbf{y} - \mathbf{z})$$

where *H* is the Jacobian matrix:

$$H = \frac{\partial \mathbf{y}(\mathbf{x})}{\partial \mathbf{x}} = Y_{m,x} + \left\{ \begin{array}{c} \vdots \\ \mathbf{x}^T F^i_{m,x} + F^i_{m,x}\mathbf{x} \\ \vdots \end{array} \right\}$$

Once the solution is obtained, the chi-square test is applied to all the measurements. The chi-square test quantifies the goodness of fit between the model and measurements by providing the probability that the measurements are consistent with the dynamic model of the component. The chi-square is calculated as follows:

$$\xi = \sum_i \left(\frac{h_i(x) - z_i}{\sigma_i}\right)^2$$

Based on the chi-square, the confidence level (or probability) that the measurements and the model fit together within the accuracy of the meters is computed with:

$$Pr[\chi^2 \geq \xi] = 1 - Pr[\chi^2 \leq \xi] = 1 - Pr(\xi, v)$$

where v is the degree of freedom, which is the difference between the number of measurements and states.

It is obvious that confidence level around 1.0 (small chi-square value) infers the measurements are highly consistent with the dynamic model that we formulated, which means there is no internal abnormality. On the other side, confidence level around 0.0 (big chi-square value) infers the measurements do not fit with the dynamic model predicted values, which means our dynamic model cannot represent the current actual component. It is clear some internal abnormalities have changed the structure of the model, so the protective relay should take action (trip or diagnose) and protect the component from further damage. The decision to trip or to diagnose depends on whether the internal abnormality is permanent and the severity of it, which can be represented by the value of chi-square.

5.3. Estimation Based Protection Program

Based on the principle of setting-less protection, Georgia Tech has developed an estimation based protection (EBP) program. The user interface is shown in Figure 2.14.

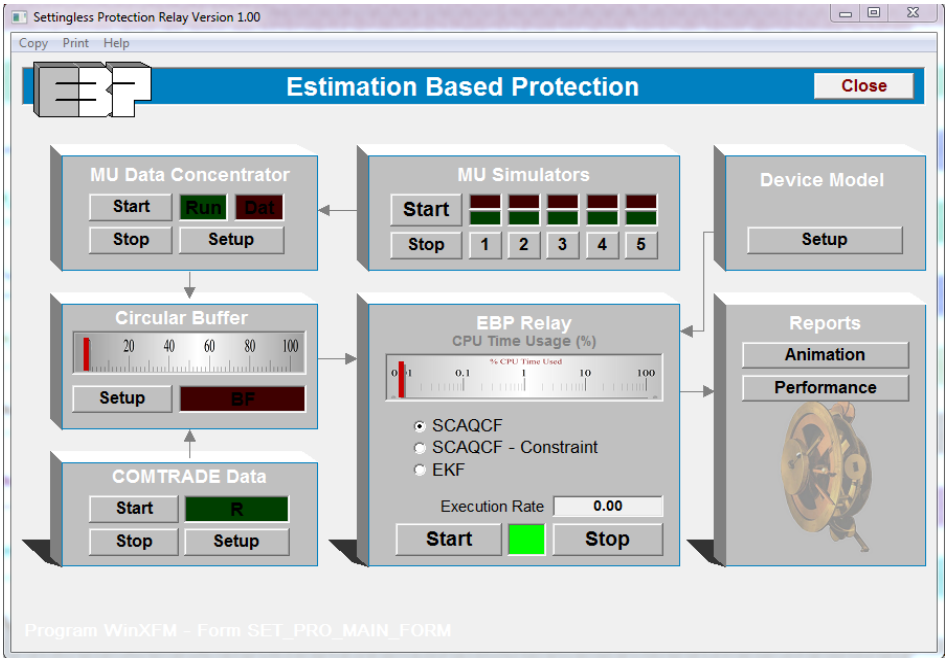


Figure 2.14: Use Interface of Estimation based Protection Program

The EBP program has two kinds of input data. One is the measurement model of protection zone in AQCF syntax, the other one is the real-time measurements data, which comes from either merging units (process bus) or COMTRADE files. There are three options to do the dynamic state estimations: unconstraint weight least square method, constraint weight least square method and the extended Kalman filter method. A Chi-square test is applied in this program and the confidence level is calculated to determine if a trip action is needed [2-7].

5.4. Setting-less Protection Numerical Example: Transform Protection

For the purpose of comparing the performance of the DSE based protection method with legacy protection functions, an event involves a 5% internal ground fault near the neutral of the transformer is present. The traditional protection function is the differential method which has two kinds of settings. One is the minimum pickup current $I_{min} = 0.1I_N$; the other one is differential ratio threshold $K_{th} = I_{op}/I_{res} = 20\%$. The test system used for numerical test is shown in Figure 2.15. A 750kVA 7.98kV/0.277kV single-phase saturable transformer is the protection zone of interest. It is designated as T1 in the Figure 2.15. The detain model of this single-phase transformer is provided in Appendix A.

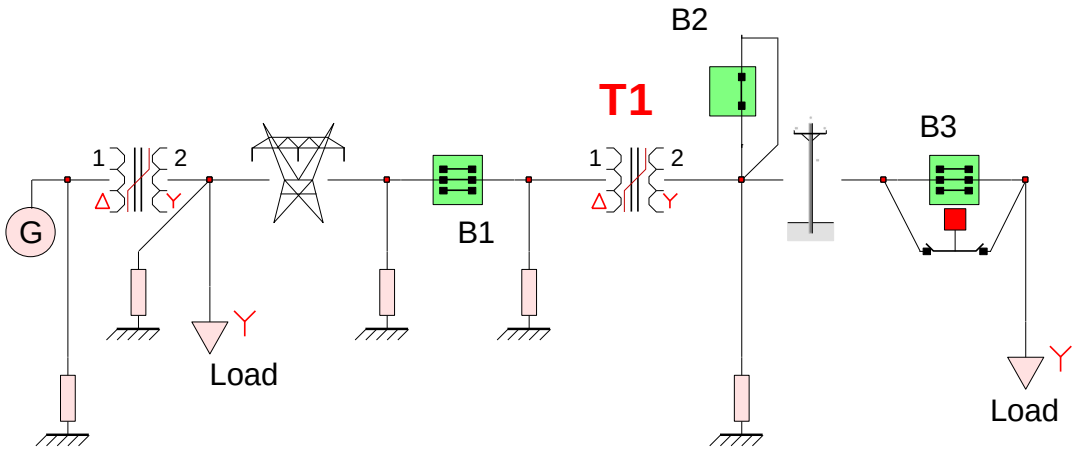


Figure 2.15: Transformer Testing System

Initially, the breakers B1 and B3 are closed while B2 is open at the beginning. At time t = 3.5s, the breaker B2 is suddenly closed and a 5% turn-ground fault near the neutral terminal of transformer happens to the phase A of secondary windings. To protect the transformer against this turn-ground fault, both the differential protection and proposed setting-less protection scheme are implemented. The results of two protection schemes are discussed as following:

Method One: Setting-less Protection

The results of proposed setting-less protection are shown in Figure 2.16. The residual and Chi-square are very small while the confidence level is 100% during normal operation, indicating that transformer is in a good health condition. When the 5% turn-ground internal fault happens at time t=3.5s, the residual and Chi-square decrease rapidly and the confidence level drops to zero. The zero confidence level indicates abnormalities inside the transformer and protection actions should be taken. It is noticed that confidence level is oscillating during the fault period since the internal fault is too small. An integral function is applied to smooth the waveform and a trip decision is taken to protect the transformer, as shown in the Figure 2.16.

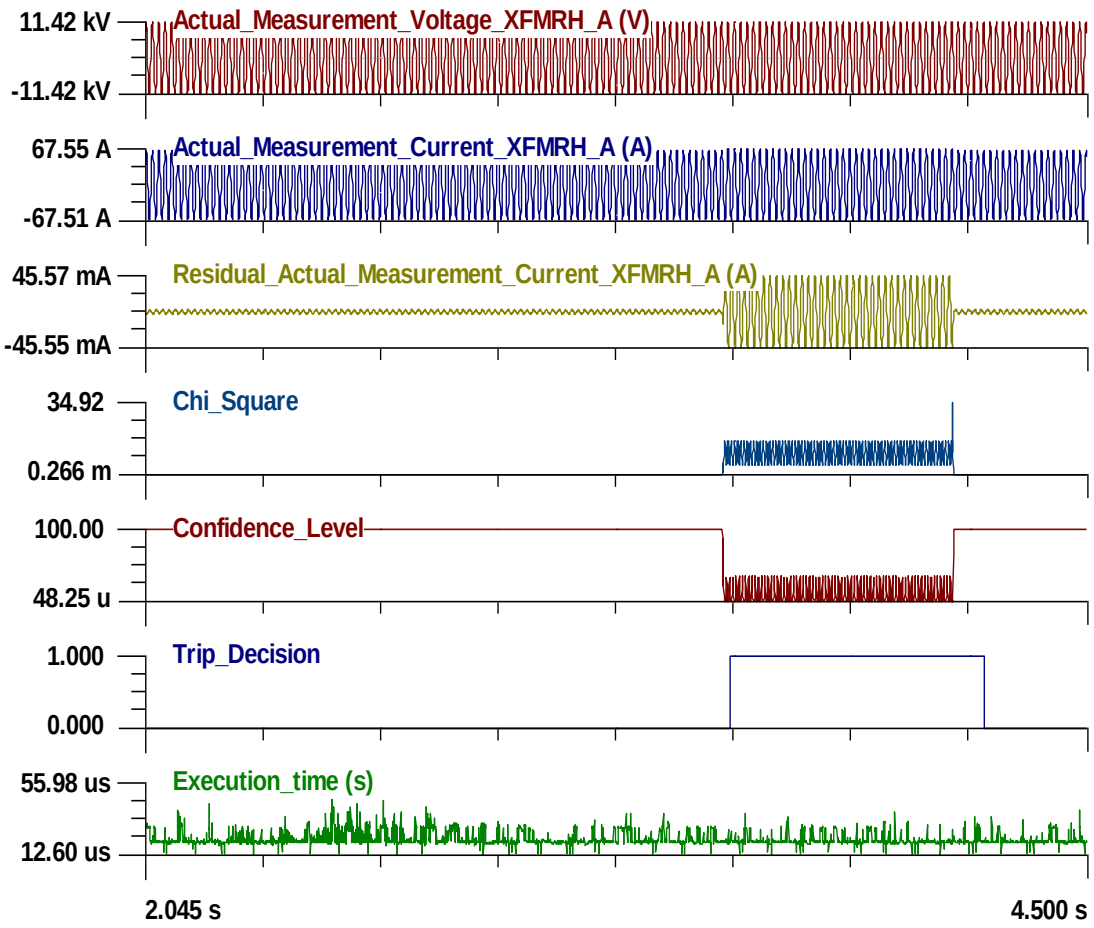


Figure 2.16: Setting-less Protection for Turn-ground Fault

Method Two: Traditional Differential Protection

The results for the differential protection are shown in Figure 2.17. At the beginning the differential protection scheme detects almost zero operating current when the transformer is operating normally. At time t = 3.5s, the breaker B2 is suddenly closed and an operating current occurs at the primary side of the transformer. However, the operating current is very small and the maximum differential index K is about 6.2%. The detected differential index K is far smaller than the threshold. The differential scheme fails to alert the relay about this internal fault. As a consequence, the transformer is not tripped and the damages are inevitable.

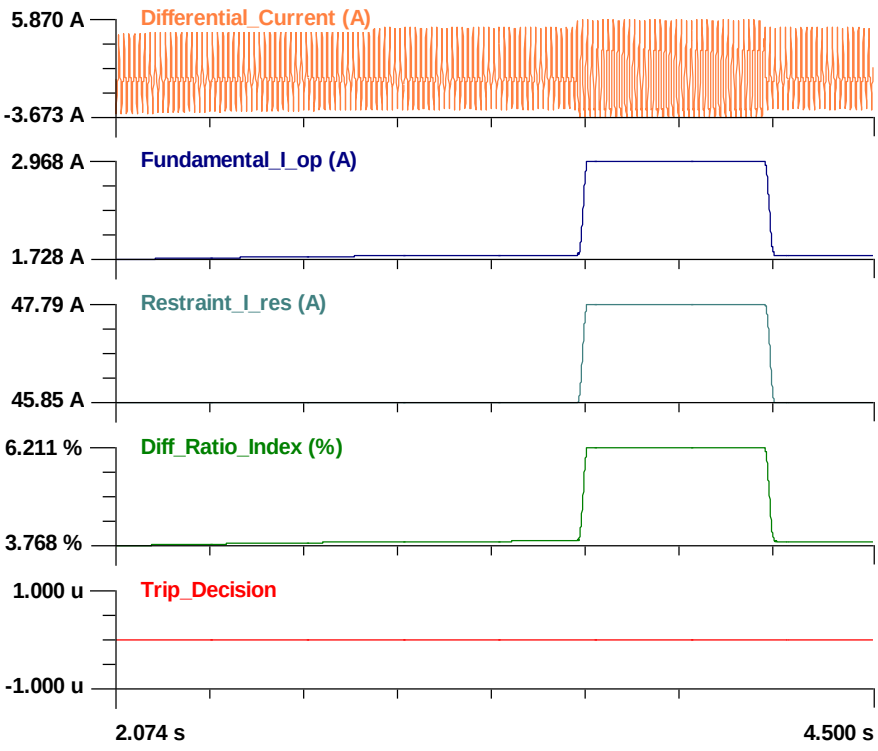


Figure 2.17: Differential Protection for Turn-ground Fault

In conclusion, the setting-less protection method provides better protections than traditional methods regarding the internal turn-ground fault. The setting-less protection method provides dependable and secure protection for transformer with additional benefits, such as fast computing speed, simple settings, and object-oriented nature with the implementation of standard AQCF format.

5.5. Set up Requirement for Inter-changeable PMU for Different Applications

Interoperability is an important issue for the intelligent PMU. Georgia Tech proposed a standard to support the interoperability of PMU for different application. The standard is based on the State and Control Algebraic Quadratic Companion Form (SCAQCF) object and it is compatible with the IEC 61850-6 approach [8]. IEC61850 is supposed to enable this object through its SCL. The measurement object is defined below:

$$\begin{bmatrix} \mathbf{z}(t) \\ 0 \\ z(t_m) \\ 0 \end{bmatrix} = Y_{m,x}\mathbf{x} + \begin{Bmatrix} \vdots \\ \mathbf{x}^T F^i_{m,x}\mathbf{x} \\ \vdots \end{Bmatrix} + Y_{m,u}\mathbf{u} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T F^i_{m,u}\mathbf{u} \\ \vdots \end{Bmatrix} + \begin{Bmatrix} \vdots \\ \mathbf{u}^T F^i_{m,ux}\mathbf{x} \\ \vdots \end{Bmatrix} + C_m$$

$$C_m = N_{m,x}\mathbf{x}(t-h) + N_{m,u}\mathbf{u}(t-h) + M_m i(t-h) + K_m$$

Measurement Definition: *NodeName, FromTo, Type, Quantity* (compatible with IEC 61850 protocol)
Measurement noise error: *dMetererScale, dMeterSigmaPU*

where:

$\mathbf{z}$: measurement variables at both time *t* and time t_m

$\mathbf{x}$: external and internal state variables of the measurement model, $\mathbf{x} = [\mathbf{x}(t), \mathbf{x}(t_m)]$

$\mathbf{u}$: control variables of the measurement model, i.e. transformer tap, etc. $\mathbf{u} = [\mathbf{u}(t), \mathbf{u}(t_m)]$

$Y_{m,x}$: matrix defining the linear part for state variables,

$F_{m,x}$: matrices defining the quadratic part for state variables,

$Y_{m,u}$: matrix defining the linear part for control variables,

$F_{m,u}$: matrices defining the quadratic part for control variables,

$F_{m,ux}$: matrices defining the quadratic part for the product of state and control variables,

C_m : history dependent vector of the measurement model,

$N_{m,x}$: matrix defining the last integration step state variables part,

$N_{m,u}$: matrix defining the last integration step control variables part,

M_m : matrix defining the last integration step through variables part,

K_m : constant vector of the measurement model,

NodeName: the node name at which the measurement is taken

FromTo:the terminal and phase where the measurement is taken from

Type: whether the measurement is a actual one, or a derived, virtual or pseudo one

Quantity: the type of measurement, such as voltage, current and so on

dMetererScale: the scale that meters use (in metric units),

dMeterSigmaPU: the standard deviation for the measurements (in per. unit),

Given the above SCAQCF object, PMUs and other IEDs can communicate or operate with all the necessary data information interoperably. For example if a measurement is given with the following SCAQCF object:

$$z_1(t) = x_1(t) - x_2(t)$$

Measurement type: Actual measurement

Measured Quantity: Voltage measurement

NodeName: ATL_TXMR

FromTo: Phase A, Phase N

dMeterSigmaPU: 0.01.

dMetererScale: 115,000

With this SCAQCF object the PMUs and other IEDs could know all the facts about this specific measurement:

(1) It is an actual measurement; (2) It is a voltage measurement; (3) The location of measurement is at node ATL_TXMR; (4) It is a Phase A-N measurement; (5) the standard deviation of this measurement is 0.01 pu and (6) The scale of measuring meter is 115,000. In summary, it is an actual phase A-N voltage measurement at node ATL_TXMR with a measuring meter scale of 115,000V and a standard deviation of 0.01 pu.

When exchanging a PMU or other IED of one manufacturer to another using the SCAQCF object, all the applications using the data from this device should continue to properly operate irrespectively of what algorithms the new PMU or IED uses. In other words, interoperability is achieved with SCAQCF object.

The SCAQCF object also enables the automated creation of additional measurements: (1) derived measurements; (2) virtual measurements, and (3) pseudo measurements. Derived measurements are the measurements, which can be derived from the actual measurements; pseudo measurements are the measurements which are normally not measured, like the voltage or current at the neutral terminal; virtual measurements present physical laws of a device with internal equations.

It is important to note that the type of information needed for full interoperability cannot be effective unless all devices/PMUs and other IEDs comply with above SCAQCF object. For knowledge of the network condition, it is also necessary that all devices provide above SCAQCF information so that the network condition can be reconstructed.

6. Conclusions

The value of phasor measurement units (PMUs) comes from their ability to synchronize the voltage and current phasor measurements. We have found that the PMU performance changes with operating conditions and whether a PMU can accurately track the frequency of the primary signals. We have also found variability among various vendors of PMUs. Specifically, while PMUs of various vendors perform almost similarly under steady state or near steady state conditions, under transient conditions and off nominal frequency the differences are apparent. Our investigation indicates that this is the result of different computational algorithms that each vendor uses for the computation of the phasors. In order to provide a standard for all vendors, the concept of the "Standard PMU" has been developed. The approach used for the Standard PMU is based on a variable time window size that depends on the frequency of the waveforms. The window is an exact integer number of cycles of the fundamental frequency at the time. Subsequently an accurate discrete Fourier transform is performed over this time window. Because the sampling frequency is fixed and tied to the GPS cock, the time window contains fractional samples at the two ends of the window. The Fourier transform is computed over the time window by approximating the signal with a quadratic function between samples. The approach eliminates spectral leakage and other known errors of the Fourier transform. The algorithm has been streamlined and has been used as the standard for testing and characterizing PMUs. We recommend that all manufacturers use the "Standard PMU" algorithm. It requires a few more computations but its stability and performance especially during transients and harmonic distortion are superior to the current algorithms.

The Standard PMU algorithm is especially useful when GPS synchronized sampled value data are coming from merging units. In this case it becomes an integral part of the process bus of any substation equipped with merging units and process bus.

The performance of PMUs and its impact on various applications has been investigated and the findings are as follows. A highly accurate PMU is critical to the performance of the state estimation application, the protection function of the system as well as for the dynamic monitoring of the system. The impact on state estimation and protection is discussed in the report. We also focused on advanced applications using advanced PMUs using not only phasors but also GPS synchronized sampled values. One such application is the setting-less protection or dynamic state estimation based protection. A protective relay has been developed which receives GPS synchronized sampled values form merging units and performs dynamic state estimation using the dynamic model of the protection zone. The dynamic state estimation simply determines how well the measured data fit the dynamic model of the protection zone. When the fit is within the accuracy of the meters by which the measurements are taking, the dynamic state estimator provides the true operating condition of the protection zone. Discrepancies indicate an internal abnormality. The scheme takes decisions based on the operating conditions of the protection zone. This scheme does not require any coordination with other protection functions. Numerical experiments show this setting-less protection method provides better performance than traditional schemes. The performance of this method, i.e. its accuracy, selectivity and speed depends on the accuracy of the GPS measurements. GPS synchronized measurements using merging units provide the highest accuracy data acquisition as compared to other technologies. The conclusion is that use of high accuracy GPS synchronized measurements is a critical enabler of new approaches for protection and control of the power system.

The final conclusion is that for PMUs to be an enabler of the digital substation, they should be interoperable and interchangeable. The key to interoperability/ interchangeability is the ability to characterize the temporal and spatial definition of the data that are coming out of a PMU. For example if the data from the PMU are to be used in a specific application, such as state estimation, it is important that the data contain information of (a) what time were taken, and (b) where in the network the data were measured, for example, generator A, phase A current or breaker X, phase B, etc. IEC61850 is supposed to enable this type of data through their SCL. We have proposed the State and Control Algebraic Quadratic Companion Form (SCAQCF) object, which is compatible with the IEC 61850, to support the interoperability of PMU. The proposed standard, in addition to meeting the interoperability requirements of IEC 61850 but also automates the exchange of the mathematical model of various components of the physical substation.

In summary, the utility of PMUs in a digital substation is dependent upon the accuracy of the data (time and magnitude) and its ability to meet interoperability standards. The report identifies the critical requirements for PMU technology to support advanced applications.

References

[2-1] Liu, H.; T. Bi, and Q. Yang. "The impact of digital filter on the PMU dynamic performance." *Zhongguo Dianji Gongcheng Xuebao*, vol. 32. no. 19, 2012.

[2-2] Romano, P.; and M. Paolone. "Enhanced interpolated-DFT for synchrophasor estimation in FPGAs: Theory, implementation, and validation of a PMU prototype." *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no.12, pp. 2824-2836, 2014.

[2-3] Meliopoulos, A.P. Sakis; G.J. Cokkinides, Z. Tan, S. Choi, Y. Lee and P. Myrda. "Setting-Less Protection: Feasibility Study," *46th Hawaii International Conference System Sciences (HICSS)*, pp.2345-2353, January 7-10, 2013.

[2-4] Moraes, R.; et al. "PMU interoperability, steady-state and dynamic performance tests." *IEEE Transactions on Smart Grid*, vol. 3, no.4 pp. 1660-1669, 2014.

[2-5] Zivanovic, R. and C. Cairns. "Implementation of PMU technology in state estimation: an overview." *IEEE 4th AFRICON*, vol. 2, pp. 1006-1011, 1996.

[2-6] Farantatos, E.; G.K. Stefopoulos, G.J. Cokkinides and A.P. Sakis Meliopoulos. "PMU-based dynamic state estimation for electric power systems," *IEEE 2009 Power & Energy Society General Meeting*, pp.1-8, July 26-30, 2009.

[2-7] Moore, David S. *Chi-Square Tests*. No. Mimeograph. Department of Statistics, Purdue University at West LaFayette, Indiana, 1976; Available at: http://www.stat.purdue.edu/docs/research/tech-reports/1-527/tr-469.pdf.

[2-8] Meliopoulos, A.P. Sakis; L. Sun, R. Fan and P. Myrda. "Update on Object-oriented DSE Based Protection," *Fault and Disturbance Conference*, Atlanta, GA, April 2014.

Part 3: Distribution PMUs, Synchrophasor Measurements during Dynamic Events, and PMU Data Quality

1. Introduction

University of Illinois contribution to this project pursued three lines of research:

1. The team developed a single-phase phasor measurement unit using off-the-shelf National Instruments hardware and LabVIEW. The principle objective was to build an "open box PMU" with fully known functional design and options to quickly implement design improvements. The effort evolved to focus upon PMUs designed for installation on distributions systems, which is an emerging application. Bogdan Pinte and Michael Quinlan were largely responsible for this effort.

2. The team investigated the efficacy of PMU generated synchrophasor data for understanding power system transitions between markedly different operating states. This investigation involved developing a MATLAB simulation of a synchronous machine dynamic model undergoing transition between distinct operating states – and then examining the differences between the exact dynamic response and the synchrophasor data generated during the transition. Karl Reinhard pursued this effort under Professor Sauer's guidance.

3. The team investigated statistical and signal processing analytic approaches and then developed software tools for screening synchrophasor data generated by multiple PMUs to identify unusual system events and potentially defective data. A supporting effort developed a software tool to simulate an online use-case. Kenta Kirihara was largely responsible for this effort.

2. Distribution Level Phasor Measurement Units (PMUs)

Phasor Measurement Units are most commonly installed in transmission system (110 kV and above) substations. These units are increasingly being built as integral sub-components of relays and other protective devices, which are typically very expensive. The business case for installing PMUs in the distribution system is predicated upon useful applications and low PMU cost.

As number of PMUs increases, so does the need to ensure PMU performance standardization. In 2012, Texas A&M and Xpert Power associates tested PMUs commercially available in the United States against IEEE Standard C37.118.1-2011 requirements. As shown in Table 3.1, no tested PMU fully complied with existing IEEE standards [3-1]. Most of the devices performed to standard in the steady state voltage magnitude, phase angle, and frequency variation tests. Notably, across the board the tested PMUs failed to meet dynamic performance standards.

Table 3.1: Conformance Test Result Summary

CONFORMANCE TEST RESULT SUMMARY

PMU	Class	Steady State Test									Dynamic State Test								
		Magnitude Variation			Phase Angle Variation			Frequency Variation			Measurement Bandwidth			Frequency Ramp			Step Change		
		TVE	FE	RFE	TVE	FE	RFE	TVE	FE	RFE	TVE	FE	RFE	TVE	FE	RFE	RT	DT	MO
PMU A	P	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	**F**	**F**	**F**
	M	S	S	S	S	S	S	**F**	S	S	S	**F**	S	**F**	**F**	**F**	S	**F**	**F**
PMU A-1	P	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	**F**	S	**F**
	M	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	S	S	**F**
PMU B	P	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	S	**F**	S
	M	S	S	S	S	S	S	S	S	S	**F**	**F**	S	**F**	**F**	**F**	S	**F**	S
PMU C	P	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	S	S	S
	M	S	S	S	S	S	S	S	S	S	S	S	S	**F**	**F**	**F**	S	S	S
PMU D	P	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	**F**	**F**	**F**
	M	S	S	S	S	S	S	S	S	S	**F**	**F**	S	**F**	**F**	**F**	S	**F**	**F**
PMU E	P	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	**F**	S	**F**
	M	S	S	S	S	S	S	**F**	S	**F**	**F**	**F**	S	S	**F**	**F**	S	S	**F**
PMU F	P	S	S	S	S	S	S	**F**	S	S	S	**F**	S	**F**	**F**	**F**	S	S	S
	M	S	S	S	S	S	S	**F**	S	S	**F**	**F**	S	**F**	**F**	**F**	S	S	S
PMU G	P	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	**F**	S	**F**
	M	S	S	S	S	S	S	S	S	S	S	**F**	S	S	**F**	**F**	S	S	**F**
PMU H	P	S	**F**	S	S	**F**	S	S	**F**	S	S	S	S	S	**F**	**F**	S	S	S
	M	S	**F**	S	S	**F**	S	S	**F**	S	S	S	S	S	**F**	**F**	S	S	S

S: satisfied; F: failure; P: class P; M: class M; TVE: total vector error; FE: frequency error; RFE: rate of change of frequency error; RT: response time; DT: delay time; MO: maximum over/under shoot.

Note : Conformance testing performed by Texas A&M University and XpertPower Associates in 2012 found that PMUs being marketed in the U.S. did not completely meet IEEE Std. C37.118.1-2011 specifications [3-1]

To date, PMU installations have occurred almost exclusively at the transmission level. As PMU technology matures, potential applications at the power distribution level are being proposed as a result of efforts like the Smart Grid Initiative [3-2]. As the distribution system takes on new dynamic characteristics, due in part to the renewable and distributed energy sources, it also requires closer monitoring. Transmission level PMUs alone cannot provide the visibility required for monitoring. Further, distribution-level PMUs open the door to measuring distribution system performance – and then providing near real time information to enable both evaluating and improving distribution system performance.

We undertook our effort to design and build a single-phase PMU (μPMU) to understand the internal workings of an IEEE standard-compliant device – and to investigate new ways to meet or surpass the standards. We sought to build an open box PMU to facilitate understanding and enable innovative approaches (commercially available PMUs are typically closed box to protect proprietary design and performance information).

Additionally, we set out to design a PMU costing less than $1000 to open the door for much wider PMU installation – with an eye specifically to distribution system application. We used our PMU to investigate anomalies in distribution system voltage and phase. We also sought to assemble synchrophasor data streams in a phasor data concentrator (PDC) using the internet.

Figure 3.1: Distribution level μPMU Prototype

This project was started as part of the Synchrophasor Data Quality (SDQ) Initiative. Specific aims were to build an open-box, secondary distribution level (120 V), low-cost phasor measurement unit. This was motivated in part because secondary distribution level PMUs are inherently cheaper in part because it involves a single low-voltage.

Synchrophasor data quality error sources can be categorized in many ways including data processing, digital signal processing, equipment specifications, installation, measurement, network failure, PMU configuration, and PMU standards. The hope was that a device built from scratch would give us an understanding of data quality issues caused by the hardware and software – and spanning the range of the commonly recognized error sources.

Our distribution level μPMU (Figure 3.1) has a 10 kilo-Samples per second (kS/s) sampling rage and provides the flexibility to reprogram the system to higher frequencies. It has a user selectable output rate of 10 Hz or 20 Hz. An important feature is its battery backup that allows the device to operate during periods of grid instability up to one hour. The most interesting grid behavior happens during such periods and having that data available helps validate system models as well as understand causes of grid instabilities. Finally, our parts cost was approximately $350 per μPMU.

2.1. Hardware

The block diagram of our device is shown in Figure 3.2. It uses a National Instruments myRIO-1900 as the main piece of hardware. It has a Xilinx Z-7010 field-programmable gate array (FPGA). Such a processor is desired because it provides accurate software loop timing needed to collect, process, analyze and report the synchrophasor data. It also receives the GPS second pulse and Coordinated Universal Time (UTC) used for synchronization and time stamping data.

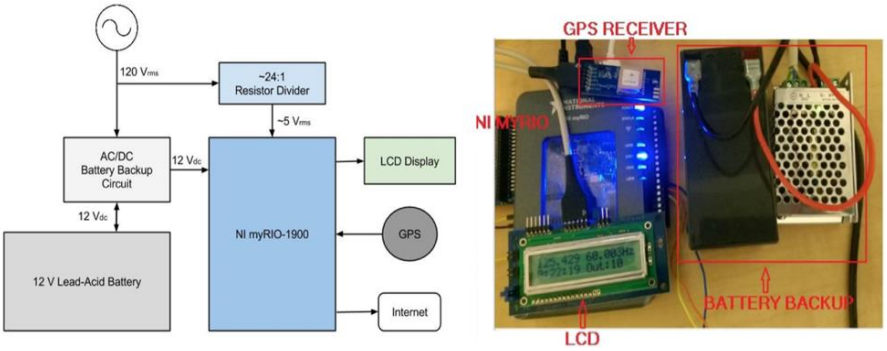


Figure 3.2: µPMU Block Diagram (left), PMU Components (right)

The wall voltage must be stepped down in order to allow the ± 10 V analog inputs to sample it. This is done through a 24:1 resistor divider circuit. We chose a resistor divider rather than a transformer in order to optimize the cost and size of our µPMU. The ±10 V analog input has an absolute accuracy of ±200 mV, according to the data sheet, which corresponds to a 2% maximum voltage error.

An AC/DC converter powers the device and also keeps a 12 V battery charged to provide backup power during periods of grid fluctuations. An LCD (liquid-crystal display) updated every second displays the latest synchrophasor to the user. There are two options for storing synchrophasors. This can be done locally on a flash drive, or sent to an openPDC server via C37.118.2 communication standard for remote storage. A phasor data concentrator is a server that aggregates synchrophasors from multiple locations, allowing data archival over long periods of time. The openPDC is just an open source phasor data concentrator that complies with the synchrophasor communication standards.

2.2. Software

In order to understand the difficulties encountered trying to produce an accurate synchrophasor, one must understand the software side of it. We use NI's LabVIEW software to program the myRIO-1900. We chose to implement the µPMU software in LabVIEW due to its intuitive and graphical programming nature. It is not only appropriate for developers who are not expert programmers, but also helpful in reinforcing the Open-box µPMU's concept.

A representative LabVIEW block diagram implementing the µPMU is shown in Figure 3.3. Using LabVIEW, the PMU uses an FPGA and a real time processor to produce a synchrophasor. The FPGA is programed to sample the distribution level voltage at a rate of 10 kS/s and store the data in a DMA FIFO. The µPMU's function can be very easily modified allowing the modifications to improve existing code function – or adding new functionality. Rapid prototyping enables integrating new tools to the PMU – for instance capturing dynamic transitions between equilibrium states (Section 3) or evaluating synchrophasor data quality (Section 5.4).

Software on the myRIO samples the wall voltage (120V) at 10 kS/s. In order to produce 10 phasors per second, each second data must be divided into 10 fragments, or synchrophasors. Each fragment, therefore, consists of 1000 points. The voltage RMS, frequency, phase, and timestamp

are then calculated for each fragment, as discussed in the paragraphs below. The reason why we wait for a whole second worth of data before calculating synchrophasor parameters is because the myRIO's 40 MHz clock is only accurate to within 1 μs; over time, this adds up and translates to either having an extra sample point or losing a sample point. The extra sample point will interfere with the synchrophasor calculation if not accounted for. In order to account for the lacking, or extra, point in our samples, we calculate exactly how many sample points have been taken during the last second. This, of course, introduces a one second latency between the time when data has been taken and the synchrophasor generated has been sent to the user.

According to the IEEE C37.118 synchrophasor standard, phase is defined as the deviation in degrees from a standard cosine wave. In LabVIEW, phase is calculated using the Extract Single Tone Information VI. LabVIEW Virtual Instruments (VIs) are prebuilt programs that are available for users to simplify their work. Frequency is also calculated using the same prebuilt VI. The VI has the time signal (1000 data points) as the input, along with several other optional inputs. Detected frequency and phase in degrees are among the several outputs. The VI calculates these parameters by extracting the wave's fundamental frequency through complex mathematical derivations. Voltage RMS is also calculated using a prebuilt RMS VI. It does so by using the standard RMS formula, where V_i stands for the voltage value of the i^{th}sample.

$$V_{rms} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} V_i^2}$$

2.3 PMU Placement in the Distribution System

PMU placement plays a very important role in the deployment process of distribution level phasor measurement units. There have been many graph theory approaches to placement of PMUs within the transmission system. However, these do not necessarily apply to the distribution system. These transmission level placement algorithms heavily rely on the meaning of synchrophasors obtained from the transmission system. Synchrophasor data interpretation, especially phase interpretation, has different challenges within the distribution system. For example, there could be phase shifting transformers and capacitor banks between different distribution level PMUs that will affect phase measurements. If these intermediate devices are not taken into account, it becomes impossible to give meaning to voltage phase. Even if the distribution layout between PMUs is known, it might still be very difficult to give meaning to voltage phase. Another aspect applying to location placement is the three-phase nature of the distribution system. Compared synchrophasor data streams must be taken from the same phase, as any phase imbalance affects voltage magnitude and voltage phase interpretation.

PMU location placement within the distribution grid requires investigation research to understand the unique factors relevant to placement – e.g. the most useful data and impediments to collecting and using data. Moreover, synchrophasor data interpretation may differ in a distribution system compared to transmission synchrophasor data.

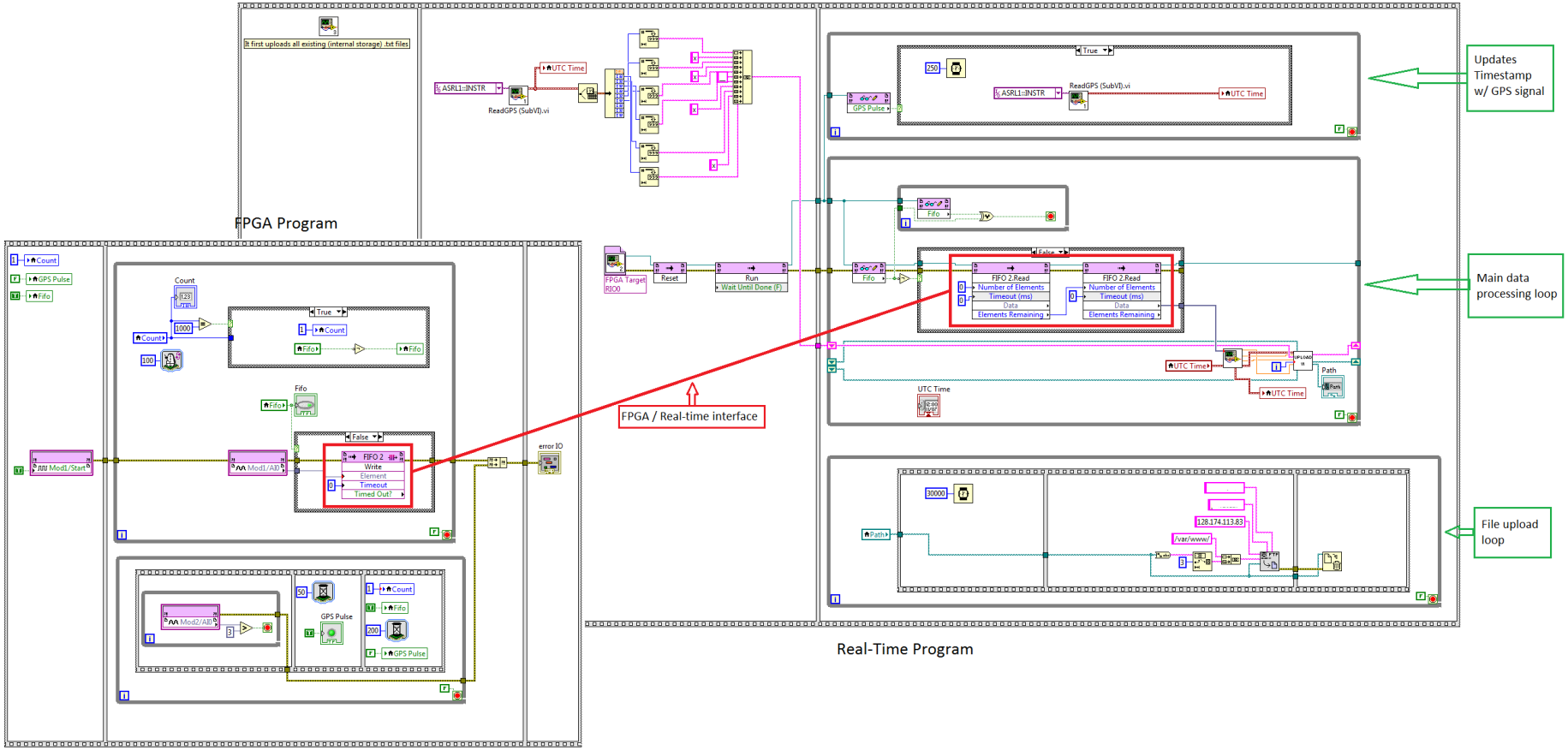
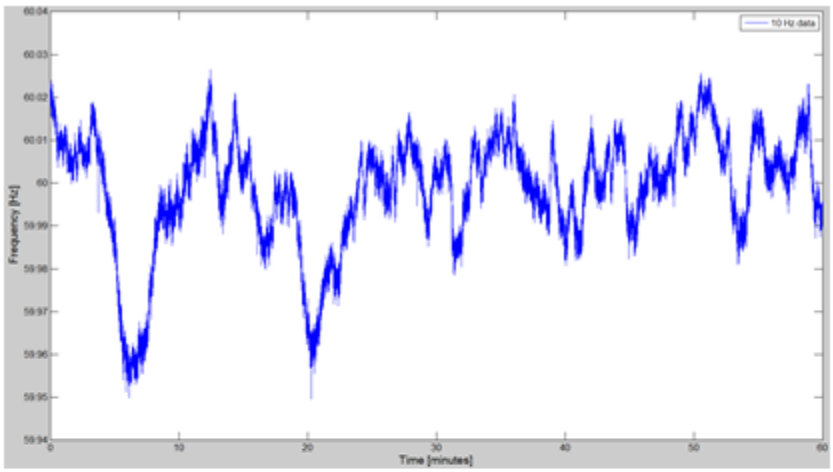
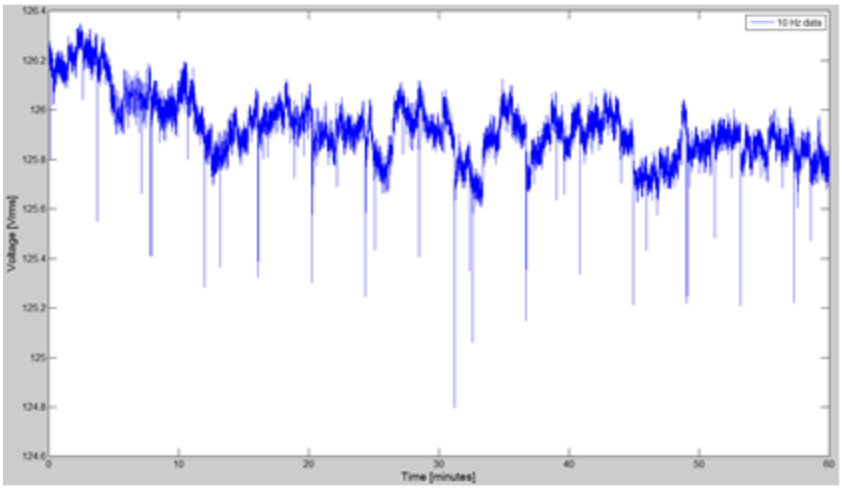


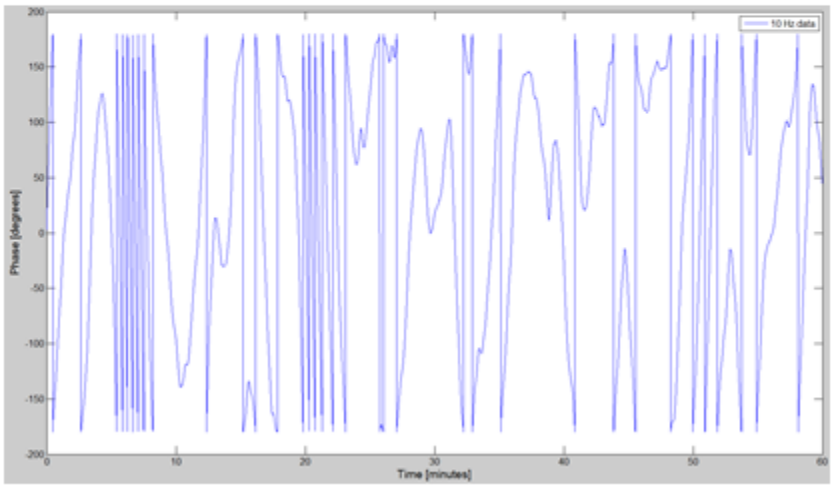
Figure 3.3: LabVIEW µPMU Programming Block Diagram



a. Frequency data



b. RMS voltage data



c. Phase data

Figure 3.4: One Hour Synchrophasor Data Sample

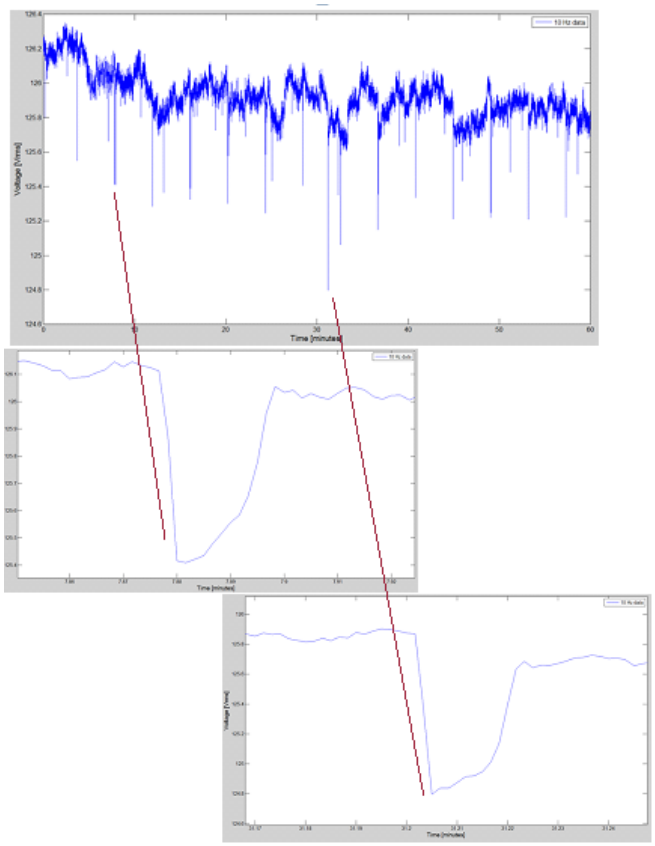


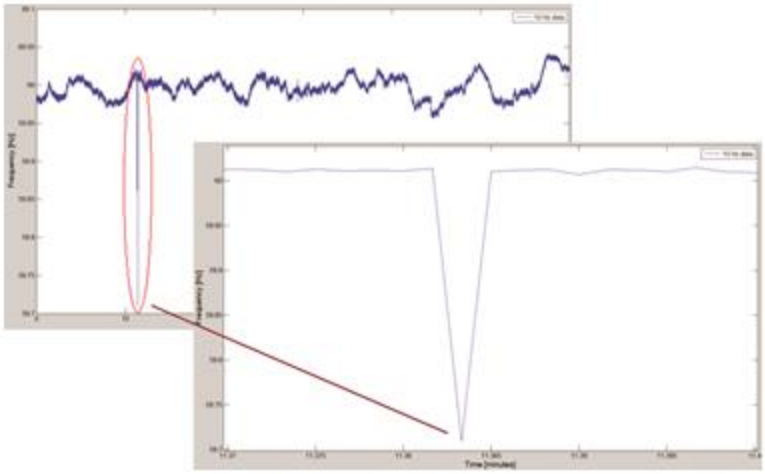
Figure 3.5: One Hour Synchrophasor Data Sample – Recurring Voltage Dips

2.4. Sample Hour-Long Synchrophasor Data

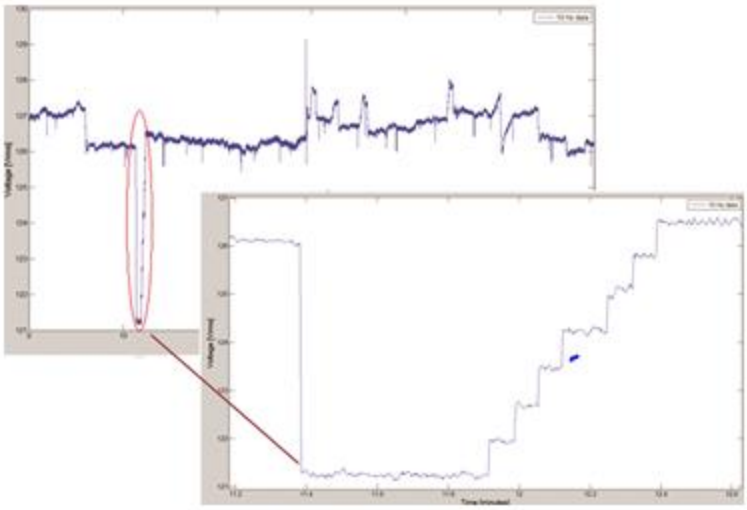
Figure 3.4 shows one hour of synchrophasor voltage data. This data was reported at 10 Hz; therefore, each graph consists of 36000 data points. Frequency remains within 40 mHz of nominal 60 Hz frequency. Voltage RMS data is more erratic with data values outside of the allowed 5% deviation from 120V. The most common events are sudden, recurring, short voltage dips as seen in Figure 3.5. These dips are recurring behavior, with each dip approximately 0.6 V and lasting for approximately one second. Other noticeable events are steps or drops in voltage, representing transformers changing their tap ratios to ensure distribution level voltage regulation. Voltage phase behavior, however, does not yet provide much meaning on its own.

2.5 Unusual Event Detection

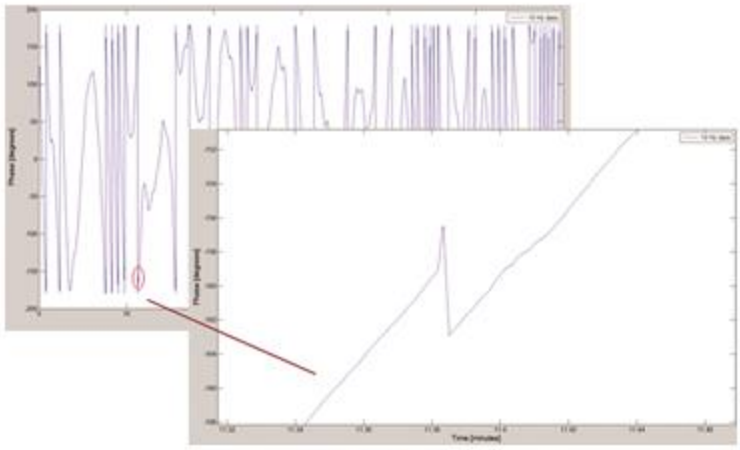
A major benefit of a distribution level PMU is its ability to record unusual events that otherwise would not be observed. Such an event was captured and its impact seen in voltage frequency, root mean square, and phase, as shown in Figure 3.6. At the beginning of the event, frequency dropped by 300 mHz for a tenth of a second. The voltage dropped suddenly by 5 V RMS, and then recovered in steps over the course of one minute campus. The cause of the unusual behavior was investigated but not identified.



a. Frequency data



b. RMS voltage data



c. Phase data

Figure 3.6: Unusual μPMU Recorded Distribution System Event

3. Observing Transient Power System Behavior via Phasor Data

3.1. Introduction

This investigation seeks to investigate the utility of using phasor data to capture transient system behavior between significantly different equilibrium states. In both system dynamic behavior analysis and real-time control, there are a number of issues to be investigated regarding the utility of synchrophasor data produced by PMUs. By design, the PMU produces phasor values (magnitude and phase) at the system's nominal frequency (50 or 60 Hz). A typical PMU component diagram is shown in Figure 3.7.

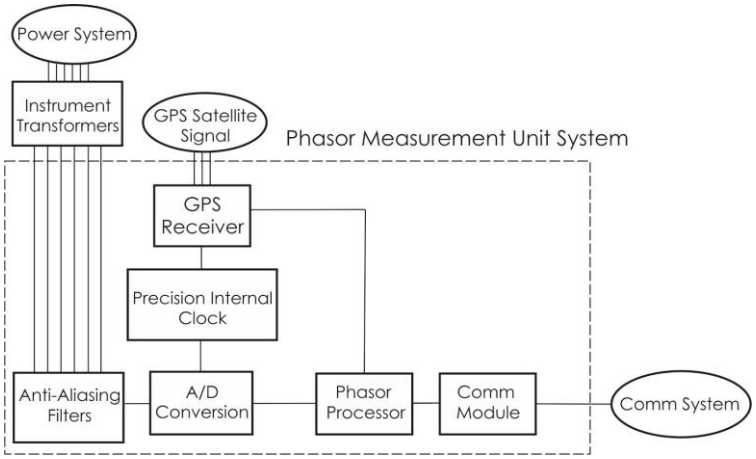


Figure 3.7: Phasor Measurement Unit Component Diagram

First, the data produced by the PMU when the system is not operating at or near the fundamental frequency is not a "phasor;" rather, it is a number generated by an algorithm set to process measurements at the system's operating frequency. That being said, the data produced under dynamic conditions may still be useful if interpreted properly.

Second, the sampled voltage and current values are altered from the point of measurement by bandwidth limited CT/PT sensors and the digital signal processing that produces phasors. Potentially, these "filters" eliminate useful information occurring at other than the fundamental frequency. This is particularly important under dynamic conditions which may have high frequency content discarded during filtering processes.

Continuous 3-phase current is represented:

$$I_a = \sqrt{2}\sqrt{I_d^2 + I_q^2}\cos\left(\omega_s t + \delta + (\tan^{-1}\frac{I_q}{I_d}) - \frac{\pi}{2}\right)$$

$$I_b = \sqrt{2}\sqrt{I_d^2 + I_q^2}\cos\left(\omega_s t + \delta + (\tan^{-1}\frac{I_q}{I_d}) - \frac{\pi}{2} - \frac{2\pi}{3}\right)$$

$$I_c = \sqrt{2}\sqrt{I_d^2 + I_q^2}\cos\left(\omega_s t + \delta + (\tan^{-1}\frac{I_q}{I_d}) - \frac{\pi}{2} + \frac{2\pi}{3}\right)$$

Three-phase voltage data is similarly represented. The PMU receives the current and voltage signals via instrument transformers stepping down from high voltage to instrumentation voltages. Following anti-aliasing filtering and digitization, digital signal processing produces voltage and current phasor data.

Applying the discrete Fourier transformation to the sampled current data produces the discrete current representation:

$$\bar{I}_{\underset{DFT}{a}} @\frac{\sqrt{2}}{N}\sum_{k=1}^{N} I_{ak} e^{-jk\frac{2\pi}{N}} \quad \bar{I}_{\underset{DFT}{POS}} @\frac{1}{3}\left(\bar{I}_{\underset{DFT}{a}} + \bar{I}_{\underset{DFT}{b}} e^{j\frac{2\pi}{3}} + \bar{I}_{\underset{DFT}{c}} e^{-j\frac{2\pi}{3}}\right)$$

The d-q reference frame current and voltage phasors are represented respectfully by:

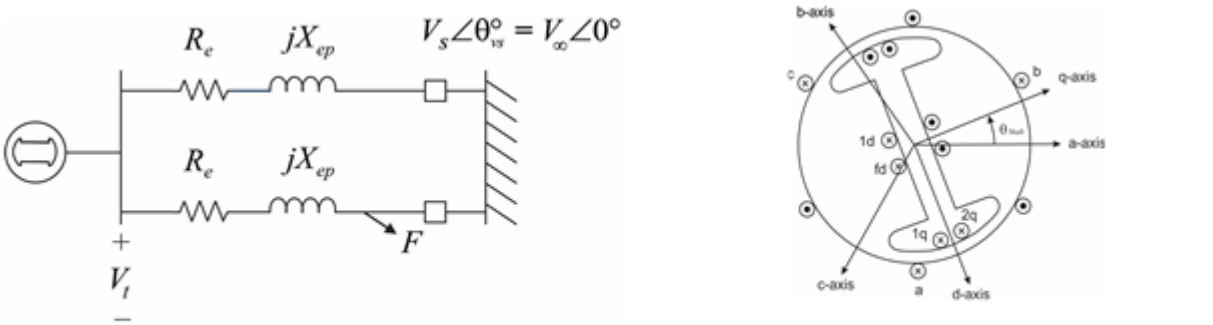
$$\bar{I}_{DQ} @\left(I_D + jI_Q\right) @\left(I_d + jI_q\right) e^{j(\delta - \pi/2)}$$

$$\bar{V}_{DQ} @\left(V_D + jV_Q\right) @\left(V_d + jV_q\right) e^{j(\delta - \pi/2)}$$

Understanding differences between simulation-based phasors and measurement-based phasors is fundamentally important to interpreting synchrophasor data produced during transitions between equilibrium states.

3.2. Synchronous Machine Connected to an Infinite Bus

Dynamic system behavior, including both fast and slow dynamic states, can be very accurately modeled using the Multi-Time-Scale Synchronous Machine model connected by a transmission line to an infinite bus. The synchronous machine and simulation model are shown in Figure 3.8. The synchronous machine-infinite bus system is comprised of 14 dynamic state variables and six algebraic equations [3-3]. The variables and equations are shown in Figures 3.9 and 3.10.



a. Synchronous Machine connected by a pair of three phase transmission lines to an infinite bus.

b. Synchronous machine schematic showing the abc and dq reference frames

Figure 3.8: Dynamic System Model – Synchronous Machine Connected to an Infinite Bus

Synchronous Machine

3 Fast Dynamic States $\psi_{de}, \psi_{qe}, \psi_{oe}$

$$\psi_{de} = \psi_d + \psi_{ed} \qquad X_{de} = X_d + X_{ep} \qquad \psi_{ed} = X_{ep}(-I_d)$$

$$R_{se} = R_s + R_e \qquad \omega_s = 2\pi f \qquad \varepsilon = \frac{1}{\omega_s}$$

$$\psi_{qe} = \psi_q + \psi_{eq} \qquad X_{qe} = X_q + X_{ep} \qquad \psi_{eq} = X_{ep}(-I_q)$$

(1) (5.42) $$\varepsilon \frac{d\psi_{de}}{dt} = R_{se} I_d + \left(1 + \frac{\varepsilon}{T_s}\omega_t\right)\psi_{qe} + V_s \sin(\delta - \theta_{vs})$$

(2) (5.43) $$\varepsilon \frac{d\psi_{qe}}{dt} = R_{se} I_q - \left(1 + \frac{\varepsilon}{T_s}\omega_t\right)\psi_{de} + V_s \cos(\delta - \theta_{vs})$$

(3) (5.44) $$\varepsilon \frac{d\psi_{oe}}{dt} = R_{se} I_o$$

11 not so fast Dynamic States $E'_q, \psi_{1d}, E'_d, \psi_{2q}, \delta, \omega_t, E_{fd}, R_f, V_R, T_m, P_{SV}$

(4) (5.45) $$T'_{do}\frac{dE'_q}{dt} = -E'_q - (X_d - X'_d)\bullet\left[I_d - \frac{X'_d - X''_d}{(X'_d - X_{ls})^2}\left(\psi_{1d} + (X'_d - X_{ls})I_d - E'_q\right)\right] + E_{fd}$$

(5) (5.46) $$T''_{do}\frac{d\psi_{1d}}{dt} = -\psi_{1d} + E'_q - (X'_d - X_{ls})I_d$$

(6) (5.47) $$T'_{qo}\frac{dE'_d}{dt} = -E'_d + (X_q - X'_q)\bullet\left[I_q - \frac{X'_q - X''_q}{(X'_q - X_{ls})^2}\left(\psi_{2q} + (X'_q - X_{ls})I_q + E'_d\right)\right]$$

(7) (5.48) $$T''_{qo}\frac{d\psi_{2q}}{dt} = -\psi_{2q} - E'_d - (X'_q - X_{ls})I_q$$

$$T_s = \sqrt{\frac{2H}{\omega_s}} \qquad \varepsilon = \frac{1}{\omega_s}$$

(8) (5.49) $$T_s\frac{d\delta}{dt} = \omega_t$$

$$\omega_t = T_s(\omega - \omega_s)$$

(9) (5.50) $$T_s\frac{d\omega_t}{dt} = T_M - (\psi_{de}I_q - \psi_{qe}I_d) - T_{FW}$$

$$\omega = \frac{\omega_t}{T_s} + \omega_s$$

Excitation System

(10) (5.54) $$T_E\frac{dE_{fd}}{dt} = -\left(K_E + S_E(E_{fd})\right)E_{fd} + V_R$$ **Main Exciter**

(11) (5.55) $$T_F\frac{dR_f}{dt} = -R_f + \frac{K_F}{T_F}E_{fd}$$ **Stabilizing Transformer**

(12) (5.56) $$T_A\frac{dV_R}{dt} = -V_R + K_A R_f - \frac{K_A K_F}{T_F}E_{fd} + K_A(V_{ref} - V_t)$$ **Pilot Exciter**

(5.57) $$V_R^{\min} \le V_R \le V_R^{\max}$$

Figure 3.9: Synchronous Machine Dynamic State Equations [3-3]

Mechanical Power System

(13) (5.63) $T_{CH}\frac{dT_M}{dt} = -T_M + P_{SV}$

(14) (5.64) $T_{SV}\frac{dP_{SV}}{dt} = -P_{SV} + P_C - \frac{\varepsilon}{R_D T_s}\omega_t$

(5.65) $0 \le P_{SV} \le P_{SV}^{\max}$

Single Stage Turbine Model

Governor Model *

- P_{SV} = Steam Valve Setting ($\propto$ Pressure)
- P_C = Power Change Setting
- P_C = 0 when Circuit is Open
- R_D = Speed Droop Regulation Quantity
- $\varepsilon = 2\pi(60)$ is retained to preserve governor function

6 Algebraic States $I_d, I_q, I_o, V_d, V_q, V_t$

(15) (5.51.1) $I_d = \frac{1}{X''_{de}}\left(-\psi_{de} + \frac{(X''_d - X_{ls})}{(X'_d - X_{ls})}E'_q + \frac{(X'_d - X''_d)}{(X'_d - X_{ls})}\psi_{1d}\right)$

(16) (5.52.1) $I_q = \frac{1}{X''_{qe}}\left(-\psi_{qe} - \frac{(X''_q - X_{ls})}{(X'_q - X_{ls})}E'_d + \frac{(X'_q - X''_q)}{(X'_q - X_{ls})}\psi_{2q}\right)$

(17) (5.53) $\psi_{oe} = -X_{oe} I_o$

(18) (5.58) $V_t = \sqrt{V_d^2 + V_q^2}$

(19) (5.59.1) $V_d = R_e I_d - \left(1 + \frac{\varepsilon}{T_s}\omega_t\right)X_{ep} I_q + \varepsilon X_{ep}\frac{dI_d}{dt} + V_s \sin(\delta - \theta_{vs})$

(20) (5.60.1) $V_q = R_e I_q + \left(1 + \frac{\varepsilon}{T_s}\omega_t\right)X_{ep} I_d - \varepsilon X_{ep}\frac{dI_q}{dt} + V_s \cos(\delta - \theta_{vs})$

$R_{se} = R_s + R_e$

$\psi_{de} = \psi_d + \psi_{ed}$
$X_{de} = X_d + X_{ep}$
$\psi_{ed} = X_{ep}(-I_d)$

$\psi_{qe} = \psi_q + \psi_{eq}$
$X_{qe} = X_q + X_{ep}$
$\psi_{eq} = X_{ep}(-I_q)$

(5.51.3) $\frac{dI_d}{dt} = \frac{1}{X''_{de}}\left[-\frac{1}{\varepsilon}\left(R_{se} I_d + \left(1 + \frac{\varepsilon}{T_s}\omega_t\right)\psi_{qe} + V_s \sin(\delta - \theta_{vs})\right) + \ldots\right.$

$\frac{(X''_d - X_{ls})}{(X'_d - X_{ls})}\left(\frac{1}{T'_{do}}\right)\left(-E'_q - (X_d - X'_d)\cdot\left[I_d - \frac{X'_d - X''_d}{(X'_d - X_{ls})^2}\left(\psi_{1d} + (X'_d - X_{ls})I_d - E'_q\right)\right] + E_{fd}\right) + \ldots$

$\left.\frac{(X'_d - X''_d)}{(X'_d - X_{ls})}\left(\frac{1}{T''_{do}}\right)\left(-\psi_{1d} + E'_q - (X'_d - X_{ls})I_d\right)a\right]$

(5.52.3) $\frac{dI_q}{dt} = \frac{1}{X''_{qe}} - \left[\left(R_{se} I_q - \left(1 + \frac{\varepsilon}{T_s}\omega_t\right)\psi_{de} + V_s \cos(\delta - \theta_{vs})\right) - \ldots\right.$

$\frac{(X''_q - X_{ls})}{(X'_q - X_{ls})}\left(\frac{1}{T'_{qo}}\right)\left(-E'_d + (X_q - X'_q)\cdot\left[I_q - \frac{X'_q - X''_q}{(X'_q - X_{ls})^2}\left(\psi_{2q} + (X'_q - X_{ls})I_q + E'_d\right)\right]\right) + \ldots$

$\left.\frac{(X'_q - X''_q)}{(X'_q - X_{ls})}\left(\frac{1}{T''_{qo}}\right)\left(-\psi_{2q} - E'_d - (X'_q - X_{ls})I_q\right)\right]$

Figure 3.10: Synchronous Machine Dynamic and Algebraic Equations [3-3]

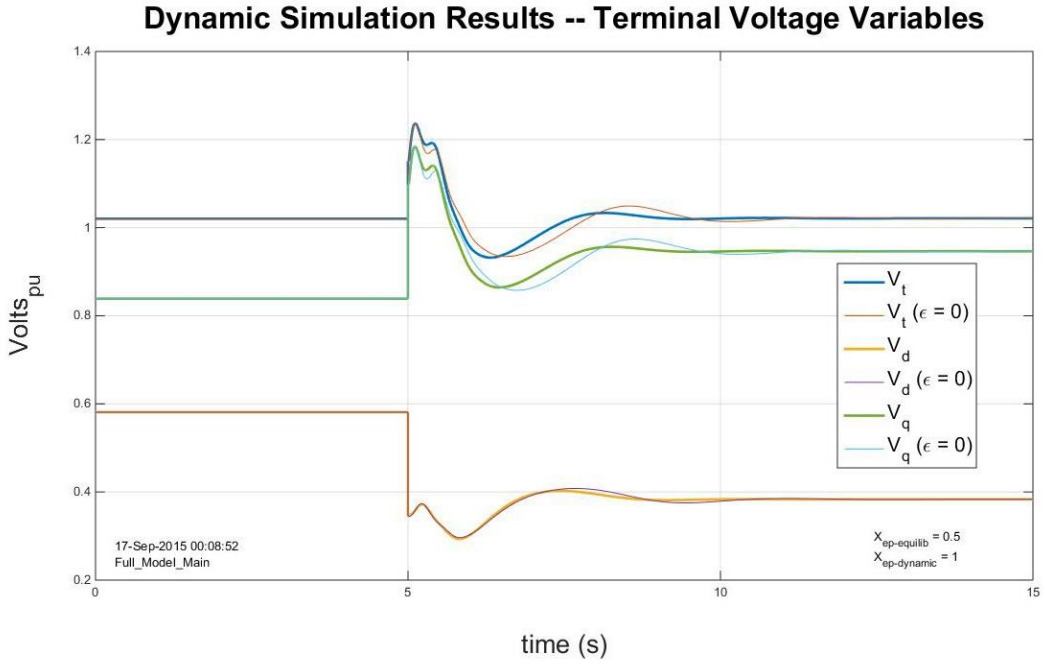


Figure 3.11: Terminal Voltages Following Transmission Line Reactance Doubling at t = 5sec

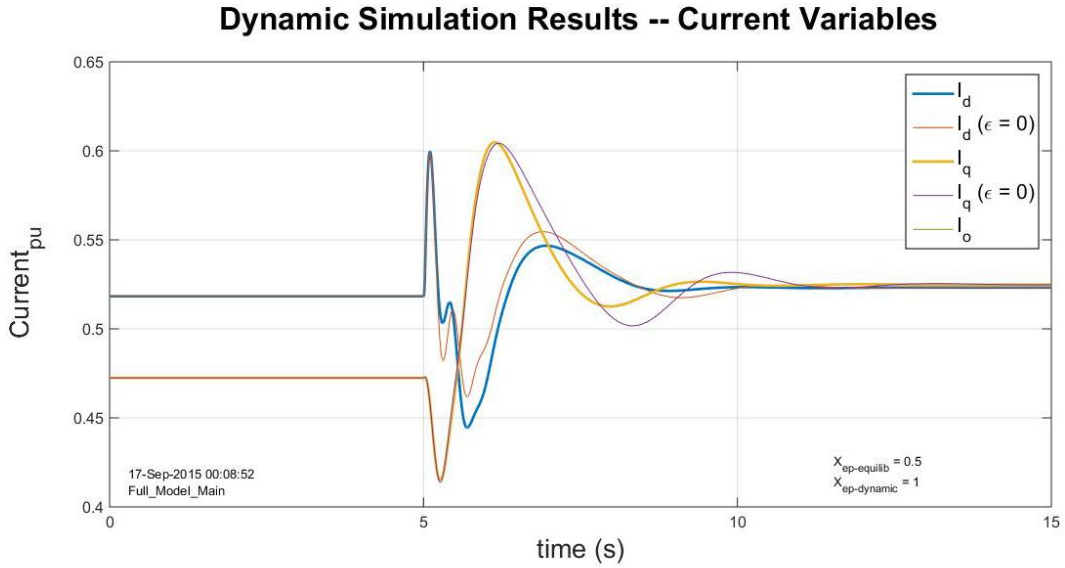


Figure 3.12: Terminal Currents following Transmission Line Reactance Doubling at t = 5sec

3.3. Simulation Results

We wrote a MATLAB simulation using the ode15s differential equation solver, which proved suitable for solving this stiff differential-algebraic equation system. The system depicted in Figures 3.8 - 3.10 operates in equilibrium until t = 5 seconds, when one of two identical transmission lines

connecting the generator to the infinite bus is switched out – doubling the transmission line reactance from .5 to 1.0 pu. Simulation results for the terminal voltages and currents are shown in Figures 3.11 and 3.12 as the system transitions between the two equilibrium states.

Each Figure shows two superimposed simulation results. The terminal characteristics for the full dynamic model (voltages and currents with $\varepsilon \neq 0$) includes the fast transient states modeled by equations (5.42), (5.43), and (5.44). The corresponding terminal characteristics disregarding the fast transients (setting $\varepsilon = 0$ in 5.42, 5.43, and 5.44) are shown for comparison (note that $\varepsilon \neq 0$ in other equations– in particular equation (5.64) that models mechanical governor behavior). The differences between the full and reduced model simulations are very similar, but nonetheless distinguishable.

The simulation shows the theoretical d-q reference frame current and voltage phasors. Of future interest is investigating the degree to which PMUs accurately capture dynamic state changes. Of specific interest is determining the difference between full-model simulated dynamic behavior and physically measured dynamic behavior captured in synchrophasor data.

4. PMU Data Quality—High-Performance Data Clustering

To identify the effects of PMU data quality, two distinct approaches were investigated. The first effort was to develop methods for finding event signatures in PMU data. The second effort developed a software tool to simulate an online use-case, known as Synchrophasor Visualizer (SPV). Both efforts ultimately feed to a promising data quality improvement remedy.

When humans are asked to detect data quality issues, it is a fairly simple task. This is due to the inherent ability of the human mind to classify and group information. When a pattern or signature is atypical, the human mind is capable of identifying it as an anomaly. However, the human brain is not capable of sifting through terabytes of synchrophasor data collected at speeds to pace data collection rates. Therefore the only practical solution is to develop and implement big data search algorithms to develop and search for anomalies and associated signatures.

One such big data analysis tool is Situational Awareness and Alerting Report (SitAAR) developed by Brett Amidan at Pacific Northwest National Laboratory (PNNL). Using k-means clustering methods, and other preprocessing to clean the data, the tool allows signatures to be clustered together and outliers to be found [3-4]. In other words, the tool allows a separation of system events by comparing them to different system operational modes. In detail, SitAAR divides the time series into one-minute intervals and adds a quadratic model to the interval. The one-minute interval, which equates to 1800 PMU data points (assuming a 30 Hz-reporting rate) allows for robust analysis. Essentially, atypical signatures, when considered from a pattern recognized in a quadratic regression, should be identified as atypical.

We established partnerships with American Transmission Company (ATC) and the Statistics Department, Pacific Northwest National Laboratory laying the ground work for “discovery” analysis of synchrophasor data being measured on ATC’s system. UIUC completed a Non-Disclosure Agreement with ATC to obtain a one year archive (2013) from ATC’s 100+ PMUs. In

coordination with PNNL, the team adapted PNNL's R-project based Synchrophasor Data Statistical Analysis Tool (SitAAR) to analyze ATC's archived synchrophasor data [3-5].

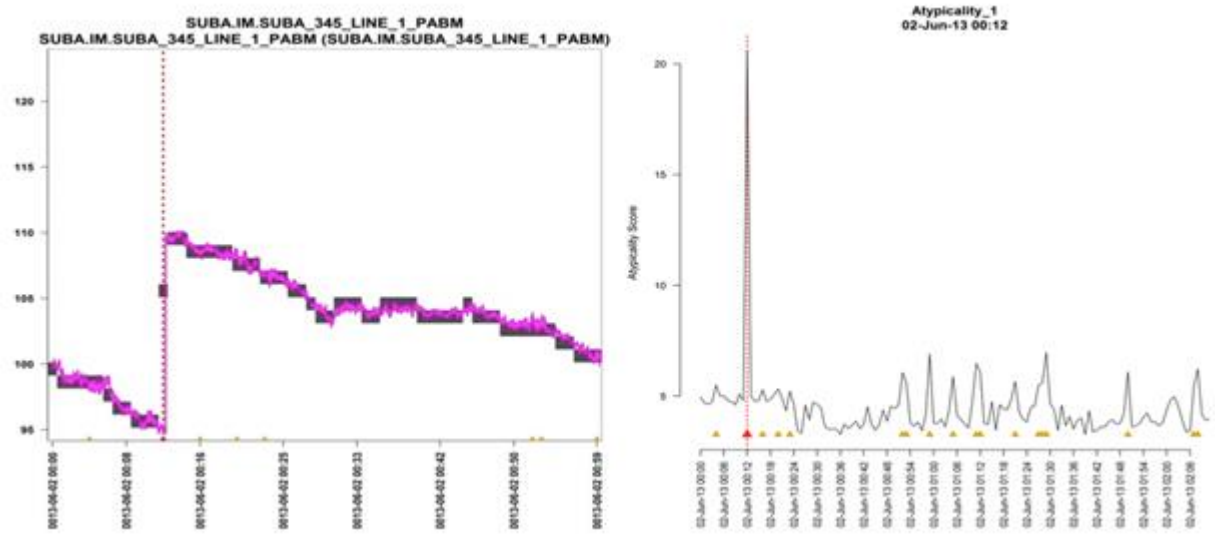


Figure 3.13: Voltage Phase Data with an Identified Signature (left) and A-typicality Score (right)

As of October 2013, ATC system had 92% of the PMUs delivering 99.9% reliable data with a 99.9% success rate [3-6]. Using this data and averaging all of the PMUs, the expected value of reliable data per PMU is 1600 data points per minute, which is sufficient for an accurate quadratic regression. This allows SitAAR to recognize the pattern of the interval such as gradual change or sudden changes in the reported value, in quadratic model parameters. The parameters of each model are used to signify a point on an n-dimensional space to determine which cluster the interval belongs to.

Figure 3.13 shows one of the signatures that SitAAR captured in the ATC data. In this instance, a line trip was found via current magnitude data. The event was captured due to the interval's high "a-typicality" score.

Similarly, SitAAR identified several signatures in the system as shown in Figures 3.14 to 3.16. In Figures 3.14 and 3.15, the same event was captured from different data streams: current magnitude and frequency. Since current flow is interrupted on a line during a line trip, when the line is restored, the current flow is also restored. This sudden change in behavior has been detected through the current flow changing across the system and system frequency. Figure 3.16 shows the voltage magnitude data used to identify a third signature of a line tripping.

From SitAAR pattern recognition (also known as *signature detection use case*) finding signatures is most reliable from voltage and current data streams. We found that frequency data streams were significantly less useful for identifying system event signatures.

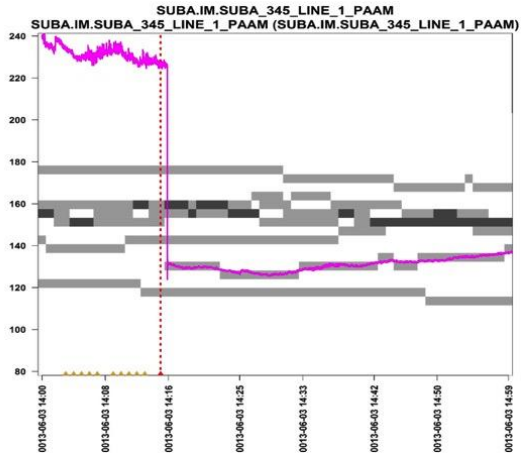


Figure 3.14: Current Magnitude Change Detected in *second* Line Trip Signature

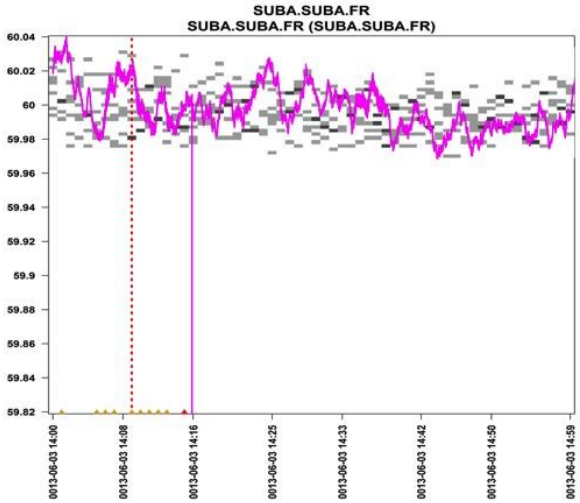


Figure 3.15: SitAAR Detecting Frequency Disturbance in the *second* Line Trip

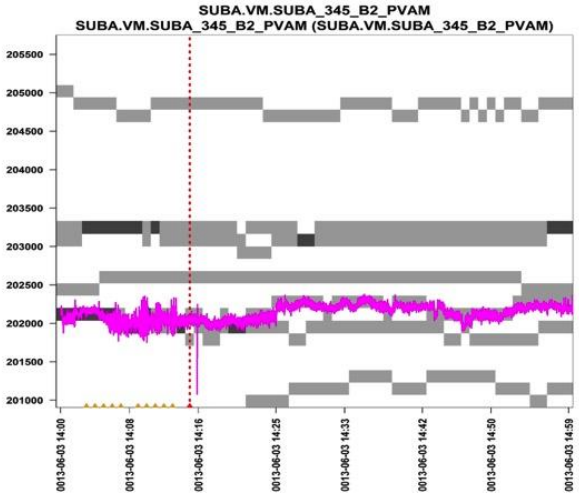


Figure 3.16: Voltage Magnitude Capturing a *third* Line Trip

When used off-line, SitAAR's k-means clustering technique is very effective. However, extending SitAAR to dynamic use cases is problematic. The principal limitation to real-time application is computational speed. For the 20GB data set used in this investigation, SitAAR's computation time was approximately 12 hours on an Intel I-7 processor on the Windows 7 operating system. To process 20 TB (the approximate size of ATC's 2013 data archive) on the same computer would require approximately 18 months.

Accepting an 18 month processing time is not worth the wait – and is likely optimistic. Second, the RAM necessary to keep the clustered data will be quickly depleted requiring slower virtual RAM. The consequent memory swapping will further impact CPU overhead, and thus, add to processing time.

Using the exact same method online, as in clustering streaming data, would again lead to the same issues regarding physical hardware. The k-means centroid would be fairly static, but the overhead necessary to keep track of these centroid may be computational expensive. For a practical use case, such as online usage or large-scale analysis of the whole archive data, a more efficient and dynamic algorithm for signature detection is necessary.

Another SitAAR drawback is that distinguishing between system event signatures and data quality signatures is difficult. Accurate signature categorization is a must to leveraging the installed PMUs to full advantage.

5. Synchrophasor Visualizer

Online usage, otherwise referred to as "real-time" usage, is one of two synchrophasor technology goals. In October 2014, North American Synchro Phasor Initiative (NASPI) noted that WECC was the only participating entity (of 13) to have successfully implemented all of the objective synchrophasor applications through summer 2014 [3-7]. Many of the projects are still in research stage. A fundamental challenge is the lack of engineers and other technically capable persons working on the synchrophasor task.

To address this challenge and to simulate an environment where statistical methods can be implemented, a baseline is proposed. Synchrophasor Visualizer (SPV) is a Java-based application that includes essential synchrophasor network functionality – specifically aiming to simulate real-time synchrophasor environment and to use it as a testbed for testing synchrophasor data processing schemes [3-8].

5.1. SPV Overview

Figure 3.17 shows the functional design of the SPV. The synchrophasor data stream can originate from either a live stream or from an existing synchrophasor data archive. Once the data is loaded into the application, the data is utilized in two ways – data visualization and event detection. The two sections work independently and are displayed upon user control.

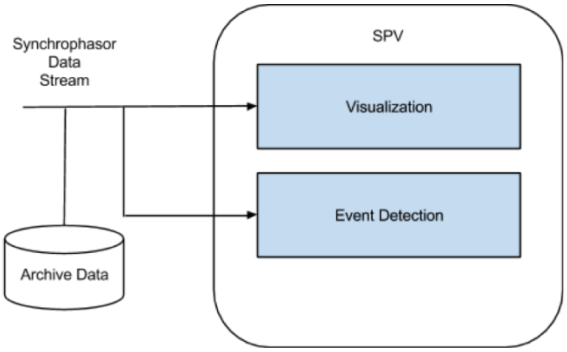


Figure 3.17: Synchrophasor Visualization Application Functional Design

SPV has four different processes: 1) Data retrieval, 2) Data processing, 3) Data visualization, and 4) User control.

The concept is easily implemented using modular programming. Creating independent processing sections allows for multiple tasks to be processed independently of each other. However, increasing the number of parallel processes brings about challenges when handling synchrophasor data streams. SPV leverages the fundamental parallel processing concept that query tasks benefit from having more threads; whereas, computation tasks benefit having fewer threads. SPV adheres of these 2 principles using Java programming.

5.2. SPV Data Retrieval

In the data retrieval block of the SPV, two switchable modes of operation are available: one streaming phasor data concentrator (PDC) and one playing back database synchrophasor archive. The user selects the data streaming source.

5.3. SPV Data Processing

SPV's Data Processing module performs two roles: 1) receiving streaming data and 2) processing data for the visualization module. These two distinct computations are executed in parallel. The first computation is a Fast Fourier Transform (FFT) and the second uses variation of a 1-D moving distribution for anomaly detection. Both tasks are core CPU intensive.

5.4. Fast Fourier Transform

FFT is an algorithm in which time-series data is represented in a Fourier domain. From a Big Data perspective, the transformation from time-series domain to Fourier domain is similar to quadratic regression. The algorithm is computationally inexpensive and therefore well suited for real-time data screening computations. The FFT output is useful for detecting power system oscillations. Since oscillations are fairly uncommon in a power system, except for the wind farms case [3-9], this feature will not be used as heavily.

In SPV, 3000 data points of synchrophasor voltage magnitude for one phase are collected and an FFT is performed after zero-padding. The advantage of using this many points is that it allows a

detection of slower frequency oscillation by having many data points to reference. If an oscillation is detected, a signal is sent to the visualization process.

5.5. Moving Distribution-based Anomaly Detection (MDAD)

The core synchrophasor data processing function lies in moving distribution-based anomaly detection (MDAD). This process is a 1-D version of k-means clustering detection through the combination of rule-based statistical methods. First, the latest 3000 points (100 seconds of synchrophasor voltage data) are collected. The mean and the standard deviation of the data points are calculated. The calculation is updated as each new data point enters the system. The MDAD process triggers an alarm if the four latest data points are more than four standard deviations from the mean.

Figure 3.17 shows 3 distribution computations in a synchrophasor data stream containing an anomalous event. In each sub-Figure, the right plot shows the original time-series data and the left shows the distribution in the 100 second time interval. In a stable state (Figure 3-17a), the voltage magnitude distribution is pseudo-normal. However in an unstable state, the voltage magnitude distribution is not normal, and the distribution begins to a form a multi-modal distribution (Figures 3-17 b and c). Once a new stable state is reached, the distribution of the measurement will return to being pseudo normal. Thus, the moving distribution-based anomaly detection captures this shift from one stable point to another. This algorithm's benefit is its computational efficiency.

MDAD involves low-level computation operations that can be used in simple programs. For instance, the java application computes one hour of ATC synchrophasor data in an Excel spreadsheet in less than 30 seconds.

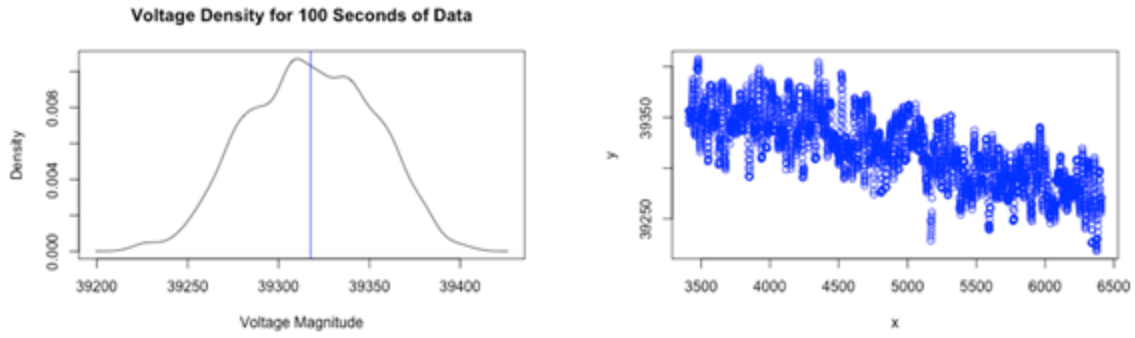
We tested algorithm's effectiveness using 15 anomalies in our test data set identified using SitAAR, which proved to be very effective in identifying events. We found that MDAD successfully detected each SitAAR identified anomaly; Table 3-2 summarizes the anomaly types. MDAD is effective for flagging outlier and system events.

Table 3.2: Effectiveness of MDAD

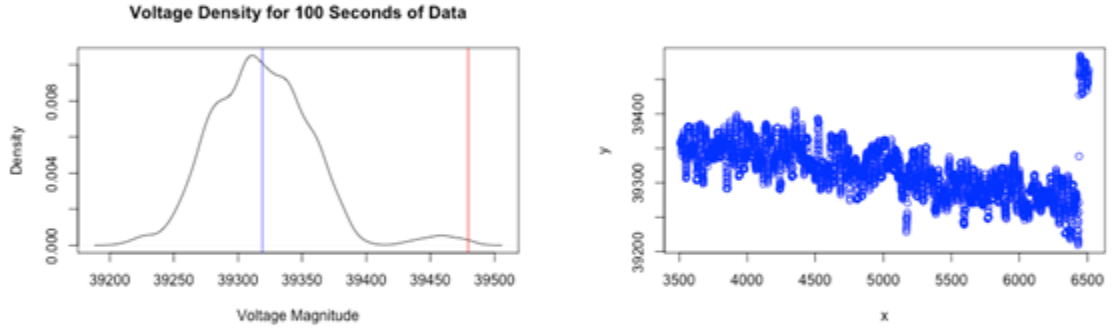
Substation	Time	Measurement Type	Suspected Cause	Result
SUBA	0013-06-03 12:59	PA-IM	Load Change	Effective
SUBA	0013-06-03 12:59	PB-IM	Load Change	Effective
SUBA	0013-06-03 12:59	PC-IM	Load Change	Effective
SUBA	0013-06-03 14:16	FR	Unknown	Effective
SUBA	0013-06-03 14:16	PA-IM	Line Trip	Effective
SUBA	0013-06-03 14:16	PB-IM	Line Trip	Effective
SUBA	0013-06-03 14:16	PC-IM	Line Trip	Effective
SUBA	0013-06-03 14:16	PA-VM	Line Trip	Effective
SUBA	0013-06-03 14:16	PB-VM	Line Trip	Effective
SUBA	0013-06-03 14:16	PC-VM	Line Trip	Effective
SUBA	0013-06-04 02:16	PA-VM	Line Trip	Effective
SUBA	0013-06-04 02:16	PB-VM	Line Trip	Effective
SUBA	0013-06-04 02:16	PC-VM	Line Trip	Effective
SUBAY	0013-10-31 06:08	PA-VM	Double Cap. Switching	Effective
SUBBJ	0013-10-31 21:08	PA-VM	Cap. Bank Switching I/O	Effective

5.6. SPV Visualization

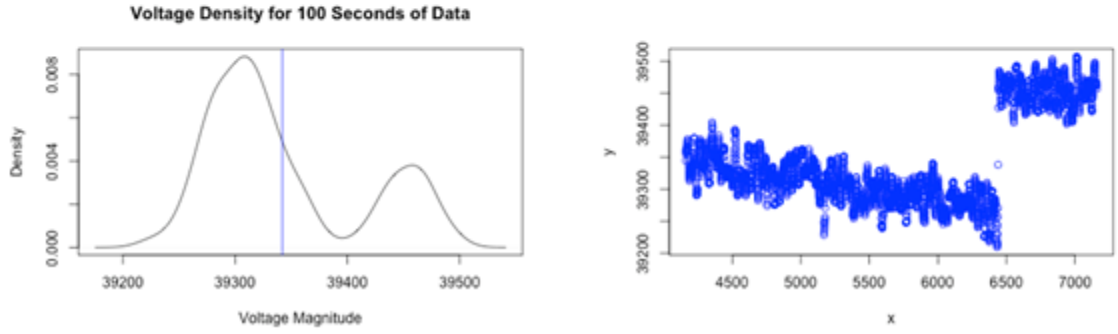
Figure 3.18 shows the SPV Visualization view. PMU locations, indicated by circles, are posted on a Google Map Application Programming Interface (API). Selecting a PMU displays its real-time voltage magnitude and phase angle measurements. When the wide-area geographic awareness situational information is needed, the user chooses either voltage magnitude (Figure 3.19) or angle depicted as a gradient (heat) map (Figure 3.20). Voltage magnitudes are per unit values. When a system event (anomaly) occurs, the event is displayed (Figure 3.21).



a. Voltage Data Distribution (left) and raw data (right) before transience.



b. Distribution (left) and raw data (right) during transience. The red line indicates fault detected.



c. Distribution (left) and raw data (right) after transience.

Figure 3.18: Sequence of 3 MDAD Computations showing Voltage Magnitude Distribution before, during, after an Anomalous Transmission System Event

The visualization is mapped and refreshed at 10 Hz. There are two reasons for this refresh rate. First, operators disliked faster refresh rates [3-10]. Second, the 10 Hz refresh rate reduces computational demands allowing more data threads to be included in the analysis.

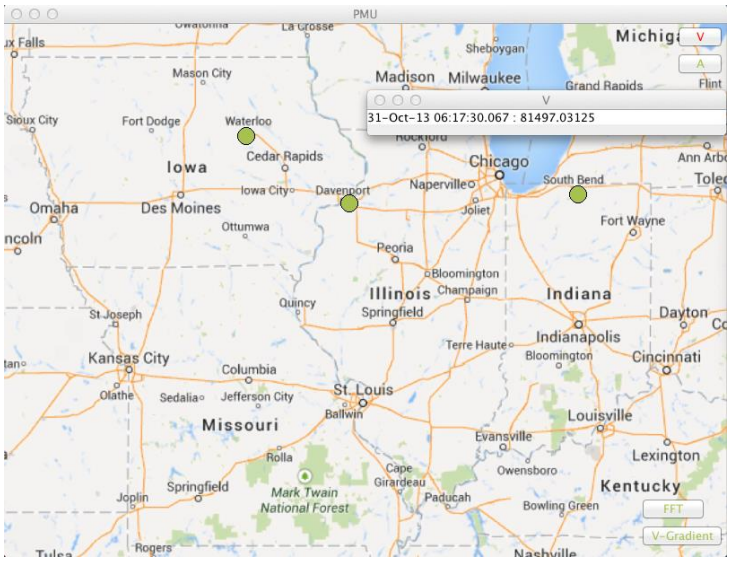


Figure 3.19: SPV Default Page – Voltage Magnitude View

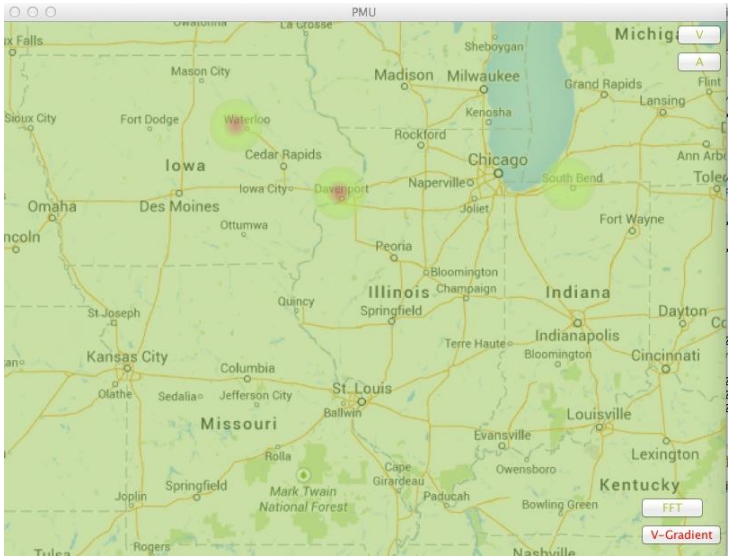


Figure 3.20: SPV Angle Gradient View

Note: the disturbances at Waterloo and Davenport, IA

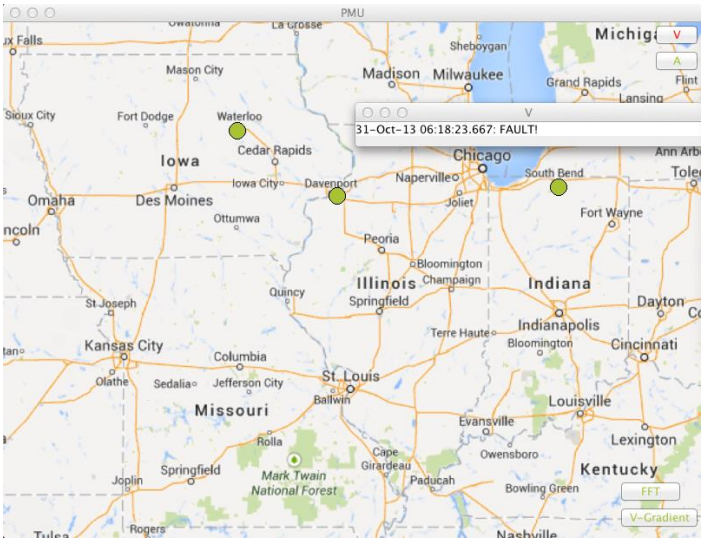


Figure 3.21: SPV Showing a Detected Aault

5.7. Discoveries from SPV

Through SPV's development, several important points regarding data quality were noted. Small missing data packets (reporting rate is reduced as low as 90%) has little detrimental minimal effect on SPV's functionality. In practical terms, sparse data availability has minimal effect. However, data loss in larger blocks is problematic. Similar issues will arise when data quality is jeopardized either by malicious intent or PMU failure.

Summarizing information visually does make synchrophasor data more useful. At the same time, this visual display of information amplifies the presence of faulty PMU data.

6. Data Quality Improvement

With regards to data quality, several cases must be considered:

Case 1: Data is correct and available.
Case 2: Data is incorrect and available.
Case 3: Data is unavailable.

Case 1 is the ideal. Case 2 is when data is faulty due to measurement, networking, or vulnerability issues. Case 3 is a special case of case 2.

When considering cases 2 and 3, several actions can be taken. One action is to omit the faulty data for analysis; another is to correct the data; and the last action is to recover data (for case 3). Depending on the situation, different combinations of the method can be used. From a data usage standpoint, however, the three methods are fundamentally equal, and can be derived from data correlation.

6.1. Pair-Wise Correlation in the PMU Data

When events are captured by synchrophasor data, it is rare for a PMU to be alone in recording a disturbance an event. This is due to the inherent correlation that exists in the synchrophasor data between PMU installation locations. Figures 3.22 thru 3.24 show the pair-wise correlation of the voltage magnitude, angle, and frequency metrics for 6 PMUs in a geographic region.

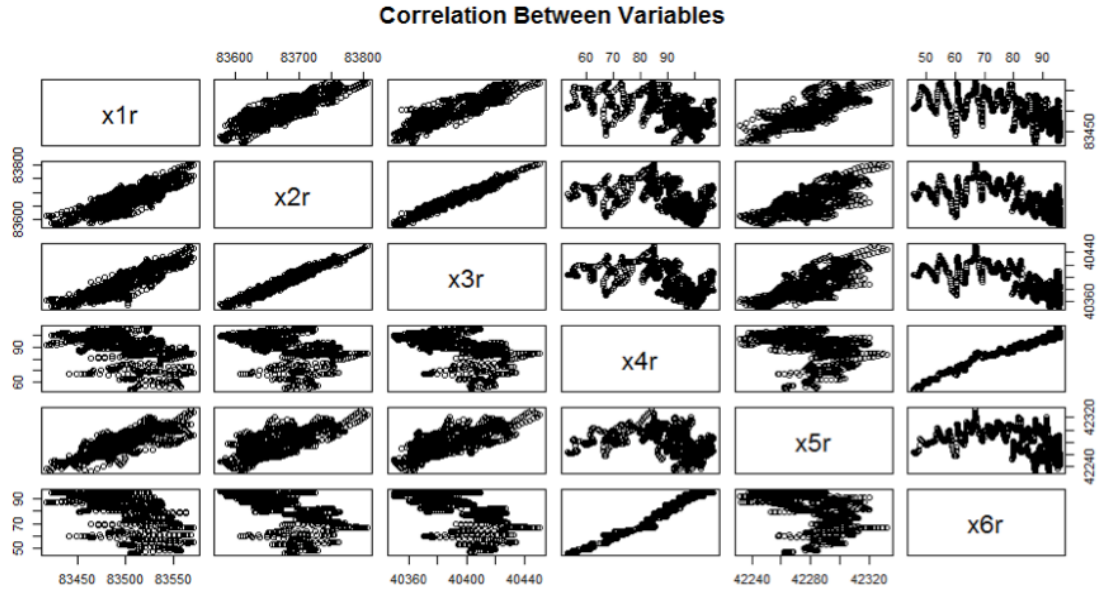


Figure 3.22: Pair-wise Correlation of Synchrophasor Data (Voltage Magnitude)

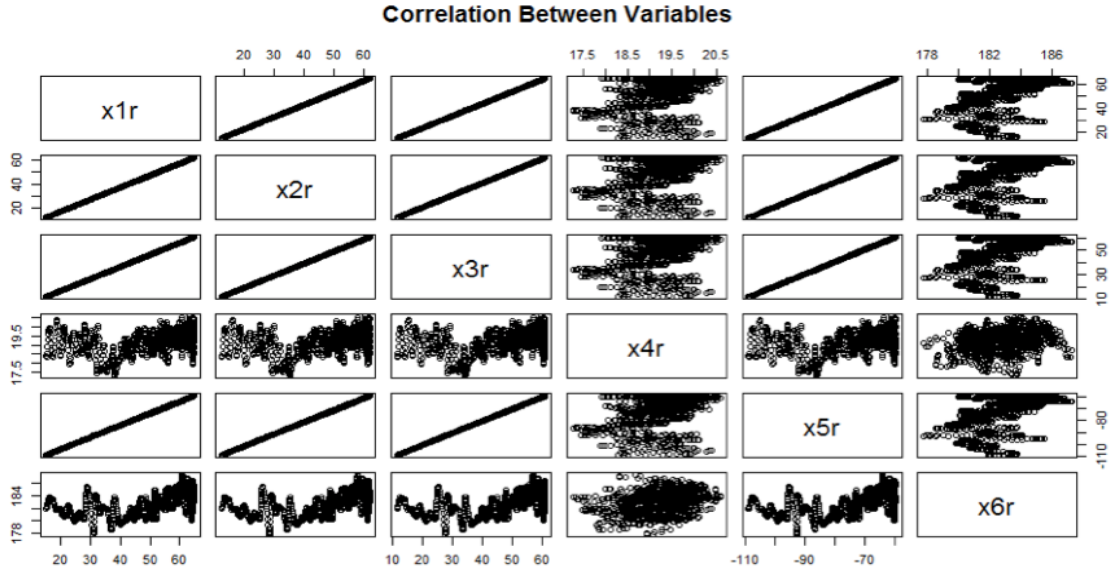


Figure 3.23: Pair-wise Correlation of Synchrophasor Data (Voltage Angle)

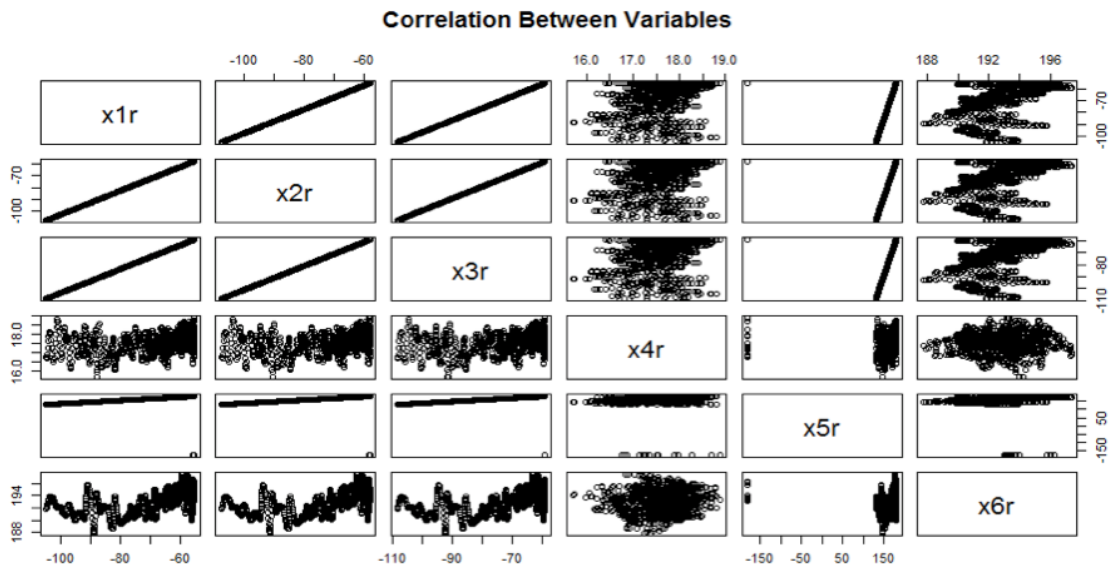


Figure 3.24: Pair-wise Correlation of Synchrophasor Data (Frequency)

The pair-wise correlation listed in Figures 3.22-3.24 show an important aspect in synchrophasor data. Those with high correlation show a linear relationship between the data sets. For example from this specific sample dataset, it is apparent that PMU locations x1r to x3r and x5r are highly correlated, and others are not. Conceptually, these PMUs observe the same location in the power system and are semi-redundant.

In a paper presented by Wang, Chow, et al., method of data recovery is discussed using a singular value decomposition and matrix re-creation [3-11]. For ease of discussion, this will be referred to as the RPI method for data re-creation in this report. The pair-wise correlations from the previous Figures show that the use of multivariate linear models is another valid method for re-creating data. Using a combination of MDAD and correlation models, an alternate approach to correcting and re-creating missing data is proposed.

6.2. Data Re-creation from Multivariate Model

A multivariate statistical model is used as one data re-creation method. The concepts behind the RPI method and the multivariate model are similar. The RPI method essentially uses matrix transformation to solve for a system of differential equations, and recover the missing data based on present surrounding data. The multivariate model approach is very similar to the RPI method, with difference being the way the relationship between the data is described. Consider the case where data from one or more PMUs is lost due to a communications system fault. Geographically close PMUs are still available and assumed to be correct. In a multivariate model, the missing PMU parameters are expressed as a linear model of other PMU parameters.
The specific steps taken to re-create the missing data are as follows:

1. Multivariate linear model for each parameter is created based on pre-data loss relationship.
2. A model is reduced using the Akaike Information Criterion (AIC).

3. Multivariate model is utilized to re-create missing data.

To demonstrate this method, a non-defective synchrophasor data set from seven locations was selected. Masking and removing certain sections of data, the residual data was used to evaluate the above method. Figures 3.25-3.27 show re-created missing data.

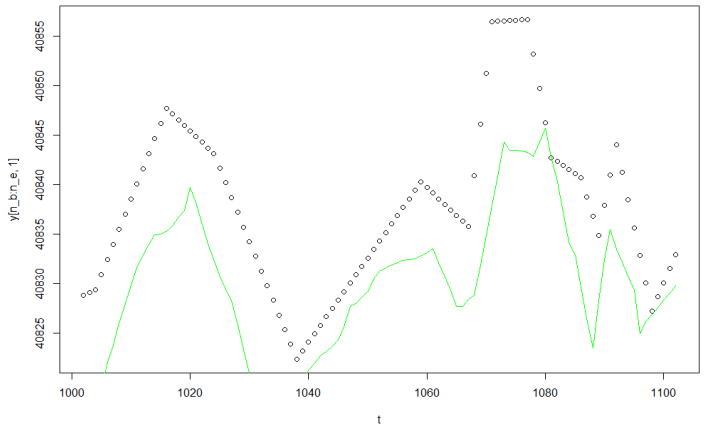


Figure 3.25: Voltage Magnitude Data Re-creation (line indicates re-created data, points indicate simulated missing data) with 0.01957% average residual error

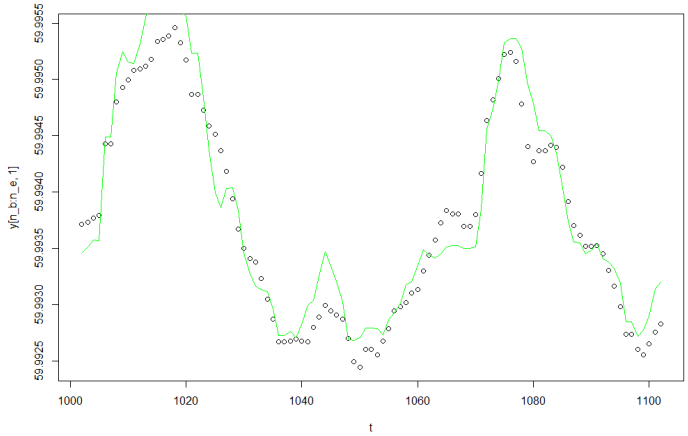


Figure 3.26: Frequency Data Re-creation (line indicates re-created data, points indicate simulated missing data) with 0.000327% average residual error

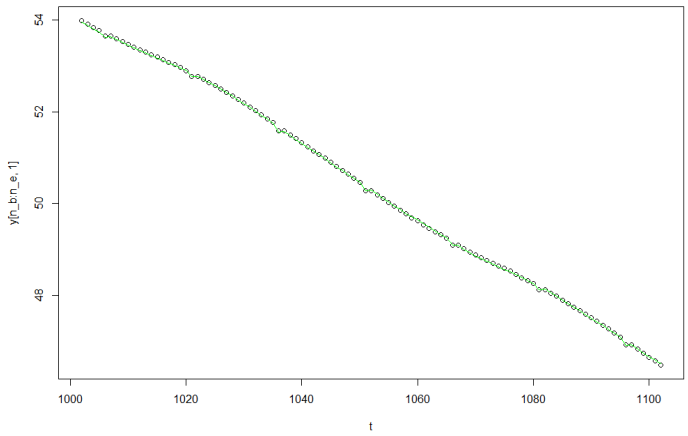


Figure 3.27: Phase Data Re-creation (line indicates re-created data, points indicate simulated missing data) with 0.0232% average residual error

As shown in these Figures, the data with relatively high correlation yield a lower residual error. The question still remains whether the residual error is more accurate than the state estimation or RPI method. Unlike the RPI method, data can be processed linearly and in real-time. Since the statistical model must be created before the re-creation of the data, there is an initial cost to the computation. However, every consecutive loss in data can be recovered with the same amount of time as the previous ones since it is a one-way function. When recovering larger amounts of data, the linear computation computational effort is significantly less than the RPI method.

6.3. PMU Security and Data Correction

Using the combination of the statistical methods presented (correlation, MDAD, and multivariate model), it is possible to detect and correct faulty or spoofed PMU measurement data.

Consider the following case. A PMU on a particular system has been injected with significant false data to attack the system. This can be either spoofing the GPS as presented in [3-12] or simply modifying measurement with some other form of cyber-attack to the PMU. Such action is taken to jeopardize the validation of security based on PMU measurements [3-13] or simply to disturb wide-area situational awareness. Regardless of the intention, PMU location is assumed to be under a cyber-attack, and is assumed to be inaccurate.

Under this situation, the pair-wise correlation and MDAD can be observed to determine whether the data has been tampered with. When MDAD observes an anomaly, but only in one PMU, it can be a flag that the data may have been tampered with. The multivariate model can be used to correct this data.

7. Design Computational Block within PMU for Distributed Application

When considering computational block within PMU for distributed application, the most critical algorithms that must be leveraged are the algorithms presented in section 3. To organize the functional design of such PMU, refer to Figure 3.28.

The synchrophasor packet and the measurements themselves are discussed in section 2. The source of errors in the synchrophasor data, found through the efforts described in 2, can be accounted for and either remedied or removed in order to process. In this section, the specific measures that must be taken for the pattern recognition will be discussed.

The primary motive for the pattern recognition filter is to apply (or adapt) system level-developed alerting signatures to distributed applications. To develop such filter, techniques for big data analysis will be utilized.

The most critical aspect of the pattern recognition filter is the database of "pre-found" signatures. In machine learning, these are referred to as the results obtained from the training data. Archived PMU data will be used to find these signatures by using a combination of methods presented in section 5. Specifically, the steps taken are as below.

1. Reduce PMU data using MDAD.
2. Create clusters using different metrics in the data.
3. Link the signatures to system events.

Using these steps, the list of labeled signatures in the PMU data as well as the best method for identifying data will be researched. The development of the pattern recognition filter will assess all of these factors to answer these questions.

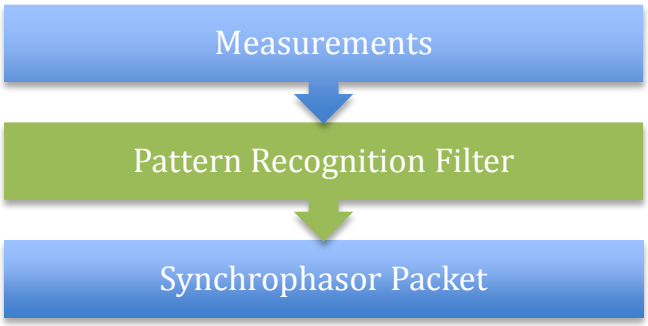


Figure 3.28: Proposed Computational Block Design within a Distributed Application PMU

8. Conclusions

Distribution Level PMU

Our PMU research was built upon the premise that there is no better way to understand PMUs from measurement to synchrophasor data stream than to implement one. Many commercial PMUs do not fully comply with the IEEE Standard C37.118 – and the reasons are not clear due to their "closed box" nature. In designing and implementing our PMU, we investigated the standard in great detail – and came to much clearer understanding of the requirements and the technical requirements to implement them. We developed an "open box" PMU that is fully transparent in the processing that produces a phasor measurement.

We were able to generate nearly-compliant synchrophasor data – and with additional effort are confident that we would have achieved our goal of fully complying with C37.118. We were both pleased and excited to observe distribution level power system events – events that otherwise would have passed unnoticed. We see the promise that distribution level PMUs have for improving awareness of system performance – and improving and ultimately optimizing system performance. With our PMU costing on the order of $350, we demonstrated that this technology's cost can be significantly reduced – opening the door to dramatically expanded field installations.

Meeting IEEE Standard C37.118 synchrophasor data standards is non-trivial – requiring the integration and synchronization of hardware and software to deliver compliant synchrophasor data. Not surprisingly, we encountered multiple hardware and software coding implementation challenges. We found phase measurements to be particularly challenging due to the very precise timing constraints inherent to complying with performance standards. Some of the challenges we encountered were no doubt the result of implementing our PMU on National Instruments FPGA-based hardware – the desired prototyping flexibility afforded by the reprogrammable FPGA also adds another layer of complexity. We encountered complications passing synchrophasor data from the PMU to the synchrophasor data concentrator server outside the laboratory setting – passing data through multiple firewalls can be a daunting task.

We envision our next efforts to be placing our four prototype μPMUs on the distribution system and concentrating the synchrophasor data streams via openPDC. This will open the door to collecting and analyzing data; we are particularly interested in leveraging our uninterruptible power supply feature to collect during power failure and restoration events.

Observing Transient Power System Behavior via Phasor Data

We developed a MATLAB simulation for the Multi-Time-Scale Synchronous Machine Model connected to an infinite bus. The simulation solves 14 dynamic state and 6 algebraic variables comprising the differential algebraic equation system. The simulation solves both the full dynamic model and the reduced dynamic model commonly used in commercial programs providing dynamic machine modeling. We observed that the difference between the 2 models is small, but observable.

Future work will include applying PMU algorithms to the terminal variable data to compute synchrophasor data during dynamic disturbances. This will enable us to investigate the efficacy of

using synchrophasor data captured during equilibrium state changes to accurately characterize dynamic system performance.

Synchrophasor Data Quality Applications

Our efforts investigating approaches to distill useful information from the synchrophasor data stream produced useful results. We completed a ground- breaking Non-Disclosure Agreement with American Transmission Corporation (ATC) to use its 2013 synchrophasor data archive (including 100+ PMUs). We found that PNNL's SitAAR data clustering (statistical) approach was very effective identifying power system events from magnitude and phase synchrophasor data – but notably less effective using frequency data. We were able to identify distinct event signatures. However, the computational effort to use SitAAR limits its utility as a real-time event screening tool.

Drawing upon our SitAAR results, we implemented computationally fast FFT methods to screen synchrophasor data – and found that these were as effective identifying our 15 unusual power system test events in real time. This prompted us to develop the Synchrophasor Visualization (SPV) application for visualizing a power system's state using synchrophasor data and displaying the result geospatially on a Google Map Application Programming Interface (API). We found that SPV delivered accurate results with small data packet flow interruptions – limiting false alarms from data loss during transmission.

We envision a next step in this line of research is to embed this and similar application modules directly into PMUs – providing a capability to rapidly turn streaming data into useful, actionable information. These applications could also be incorporated into Phasor Data Concentrators (PDCs) to provide timely, actionable information from multiple synchrophasor data streams.

From a data quality and availability perspective, we were able to estimate system state at PMU's location from synchrophasor data generated at other PMUs. We accomplished this using pair-wise correlation and moving distribution-based anomaly detection (MDAD). These techniques can be used either to recreate lost data or to validate "suspicious (equipment failure or possibly malicious intent)" data streams.

References

[3-1] Kezunovic, M. "Verifying interoperability and application performance of PMUs and PMU-enabled IEDs at the device and system level," *North American Synchrophasor Initiative Working Group Meeting*, Denver, CO, June 5, 2012.

[3-2] Sexauer, J.; P. Javanbakht, and S. Mohagheghi. "Phasor measurement units for the distribution grid: Necessity and benefits", *Innovative Smart Grid Technologies (ISGT), 2013 IEEE PES*, pp.1,6, 24-27, February 2013. DOI: 10.1109/ISGT.2013.6497828.

[3-3] Sauer, Peter W. and Pai, M.A. *Power System Dynamics and Stability*. Prentice Hall, 1998.

[3-4] Brett, A. "Baselining studies and analyses," *Presented at 2013 Transmission Reliability Program Peer Review*, Washington, D.C., June 27, 2013 [Online]. Available: http://energy.gov/sites/prod/files/2013/07/f2/2013TRR-2Amidan.pdf.

[3-5] Kirihara, K.; K. E. Reinhard, A. K. Yoon, P. W. Sauer. "Investigating Synchrophasor Data Quality Issues," in *Power and Energy Conference at Illinois (PECI), 2014*, pp. 1, 4, February 28-March 1, 2014.

[3-6] Kleitsch, J. "Smart grid investment grant update," *Presented at October 2013, North American Synchro Phasor Initiative Work Group Meeting*, Rosemont, IL, October 22-24, 2013. [Online]. Available: https://www.naspi.org/File.aspx?fileID=1177.

[3-7] North American Synchro Phasor Initiative, "Recovery act smart grid investments synchrophasor project status," June 2014 [Online]. Available: https://www.smartgrid.gov/sites/default/files/synchrophasor project status 061014-1.

[3-8] Kirihara, K.; K. E. Reinhard, Y. Liu; P. W. Sauer. "Synchrophasor visualizer," in *Power and Energy Conference at Illinois (PECI), 2015 IEEE* , pp. 1, 4, February 20-21, 2015.

[3-9] Wan, Y. "Synchronized phasor data for analyzing wind power plant dynamic behavior and model validation," U.S. Department of Energy, Oakridge, TN, 2013. [Online]. Available: http://www.nrel.gov/docs/fy13osti/57342.pdf.

[3-10] Obradovich, J. "NASPI phasor tools visualization workshop: A quick summary," Atlanta, 2012. [Online]. Available: http://www.nerc.com/pa/rrm/hp/2012%20Human%20Performance%20Conference/NASPI%20Viz%20Workshop%20Summary_NERC%20HP_JHObradovich.pdf.

[3-11] Wang, M.; J. H. Chow, et al. "A low-rank matrix approach for the analysis of large amounts of power system synchrophasor data," *System Sciences (HICSS), 2015 48th Hawaii International Conference on*, pp. 2637, 2644, January 5-8, 2015.

[3-12] Jiang, X.; J. Zhang, B. J. Harding, J.J. Makela, A. D. Dominguez-Garcia. "Spoofing GPS receiver clock offset of phasor measurement units," *Power Systems, IEEE Transactions on*, vol. 28, no. 3, pp. 3253-3262, August 2013.

[3-13] Makarov, Y. V.; et al. "PMU-based wide-area security assessment: Concept, method, and implementation," *Smart Grid, IEEE Transactions on*, vol. 3, no. 3, pp. 1325-1332, September 2012.

Appendix A: Power Transform AQCF Model

To perform the numerical case of transformer with setting-less protection method, a high-fidelity model of transformer is required. In this appendix, the transformer model is present in the algebraic quadratic companion form (AQCF) format. First, the compact mode of single-phase transformer is described, then the quadratic model is derived and finally the AQCF model of transformer is present with quadratic integration method.

The overall physical model of a single phase, saturable core variable tap transformer is illustrated in Figure A.1. Core losses are also considered in the model, expressed by the conductance g_c. The "numerical stabilizers" g_{s1}, g_{s2}, g_{s3} *and* g_{s4} are introduced to eliminate possible numerical problems. The compact model is based on the circuit analysis of the equivalent circuit of Figure A.1.

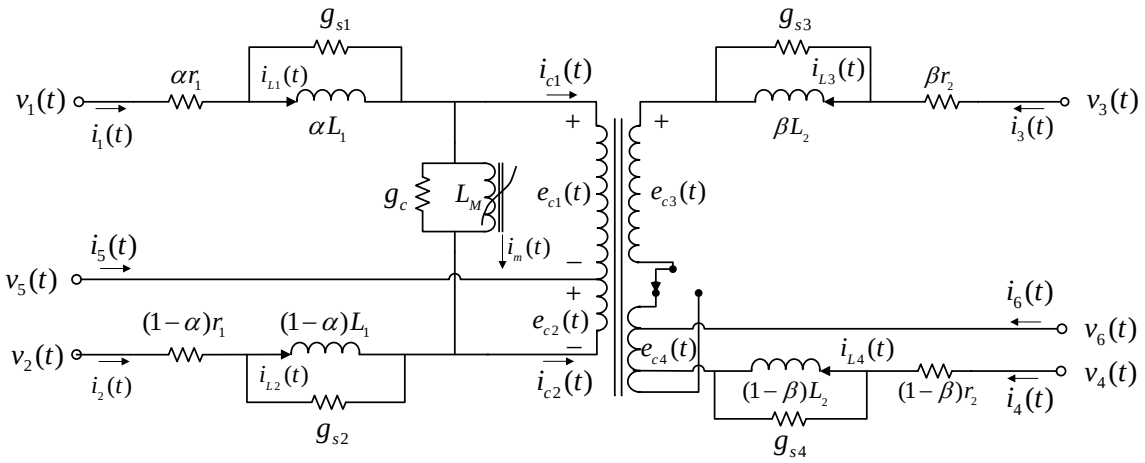


Figure A.1: Single Phase Transformer Equivalent Circuit

Time Domain Model – Compact Form

The model with nonlinear magnetizing inductor in the compact form is described by equations (A1.1) through (A1.18)

$$i_1(t) = i_{L1}(t) + g_{s1}\alpha L_1 \frac{di_{L1}(t)}{dt} \tag{A1.1}$$

$$i_2(t) = i_{L2}(t) + g_{s2}(1-\alpha)L_1 \frac{di_{L2}(t)}{dt} \tag{A1.2}$$

$$i_3(t) = i_{L3}(t) + g_{s3}\beta L_2 \frac{di_{L3}(t)}{dt} \tag{A1.3}$$

$$i_4(t) = i_{L4}(t) + g_{s4}(1-\beta)L_2 \frac{di_{L4}(t)}{dt} \tag{A1.4}$$

$$i_5(t) = -i_1(t) - i_2(t) \tag{A1.5}$$

$$i_6(t) = -i_3(t) - i_4(t) \tag{A1.6}$$

$$0 = v_1(t) - v_5(t) - \alpha r_1 i_1(t) - \alpha L_1 \frac{di_{L1}(t)}{dt} - e_{c1}(t) \quad \text{(A1.7)}$$

$$0 = v_2(t) - v_5(t) - (1-\alpha) r_1 i_2(t) - (1-\alpha) L_1 \frac{di_{L2}(t)}{dt} + e_{c2}(t) \quad \text{(A1.8)}$$

$$0 = v_3(t) - v_6(t) - \beta r_2 i_1(t) - \beta L_2 \frac{di_{L3}(t)}{dt} - e_{c3}(t) \quad \text{(A1.9)}$$

$$0 = v_4(t) - v_6(t) - (1-\beta) r_2 i_2(t) - (1-\beta) L_2 \frac{di_{L4}(t)}{dt} + e_{c4}(t) \quad \text{(A1.10)}$$

$$0 = \alpha N_1 i_{c1}(t) - (1-\alpha) N_1 i_{c2}(t) + \beta N_2 i_3(t) - (1-\beta) N_2 i_4(t) \quad \text{(A1.11)}$$

$$0 = e_{c1}(t) - \alpha N_1 \frac{d\Phi(t)}{dt} \quad \text{(A1.12)}$$

$$0 = e_{c2}(t) - (1-\alpha) N_1 \frac{d\Phi(t)}{dt} \quad \text{(A1.13)}$$

$$0 = e_{c3}(t) - \beta N_2 \frac{d\Phi(t)}{dt} \quad \text{(A1.14)}$$

$$0 = e_{c4}(t) - (1-\beta) N_2 \frac{d\Phi(t)}{dt} \quad \text{(A1.15)}$$

$$0 = -i_1(t) + i_{c1}(t) + i_m(t) + g_c\left(e_{c1}(t) + e_{c2}(t)\right) \quad \text{(A1.16)}$$

$$0 = -i_2(t) + i_{c2}(t) - i_m(t) - g_c\left(e_{c1}(t) + e_{c2}(t)\right) \quad \text{(A1.17)}$$

$$0 = i_m(t) - i_0 \left| \frac{\Phi(t)}{\Phi_0} \right|^n sign(\Phi(t)) \quad \text{(A1.18)}$$

There are 18 states variables

$$X = \begin{bmatrix} v_1(t), v_2(t), v_3(t), v_4(t), v_5(t), v_6(t), \\ i_{L1}(t), i_{L2}(t), i_{L3}(t), i_{L4}(t), \emptyset(t) \\ e_{c1}(t), e_{c2}(t), e_{c3}(t), e_{c4}(t), i_{c2}(t), i_{c2}(t), i_m(t) \end{bmatrix}$$

And 6 though variables

$$I = [i_1(t), i_2(t), i_3(t), i_4(t), i_5(t), i_6(t)]$$

Time Domain Model – Quadratic Form

The model is quadratized by introducing additional internal state variables, so that the nth exponent is replaced by equations of at most quadratic degree. Since the exact degree of nonlinearity is not known until the user specifies it, the model performs automatic quadratization of the equations. A special procedure is used, so that the model is quadratized using the minimum number of additional internal states, while also maintaining the scarcity of the resulting equations. The methodology is based on expressing the exponent in binary form. The binary representation provides all the

information about the number of new variables and equations that need to be introduced and about the form of the equations (products of new variables). The procedure is described later. Following this procedure the model can be converted into the standard quadratized form:

$$i_1(t) = i_{L1}(t) + g_{s1}\alpha L_1 z_1(t) \tag{A2.1}$$

$$i_2(t) = i_{L2}(t) + g_{s2}(1-\alpha) L_1 z_2(t) \tag{A2.2}$$

$$i_3(t) = i_{L3}(t) + g_{s3}\beta L_2 z_3(t) \tag{A2.3}$$

$$i_4(t) = i_{L4}(t) + g_{s4}(1-\beta) L_2 z_4(t) \tag{A2.4}$$

$$i_5(t) = -i_1(t) - i_2(t) \tag{A2.5}$$

$$i_6(t) = -i_3(t) - i_4(t) \tag{A2.6}$$

$$0 = z_1(t) - \frac{di_{L1}(t)}{dt} \tag{A2.7}$$

$$0 = z_2(t) - \frac{di_{L2}(t)}{dt} \tag{A2.8}$$

$$0 = z_3(t) - \frac{di_{L3}(t)}{dt} \tag{A2.9}$$

$$0 = z_4(t) - \frac{di_{L4}(t)}{dt} \tag{A2.10}$$

$$0 = z_5(t) - \frac{d\Phi(t)}{dt} \tag{A2.11}$$

$$0 = v_1(t) - v_5(t) - \alpha r_1 i_1(t) - \alpha L_1 z_1(t) - e_{c1}(t) \tag{A2.12}$$

$$0 = v_2(t) - v_5(t) - (1-\alpha) r_1 i_2(t) - (1-\alpha) L_1 z_2(t) + e_{c2}(t) \tag{A2.13}$$

$$0 = v_3(t) - v_6(t) - \beta r_2 i_1(t) - \beta L_2 z_3(t) - e_{c3}(t) \tag{A2.14}$$

$$0 = v_4(t) - v_6(t) - (1-\beta) r_2 i_2(t) - (1-\beta) L_2 z_4(t) + e_{c4}(t) \tag{A2.15}$$

$$0 = \alpha N_1 i_{c1}(t) - (1-\alpha) N_1 i_{c2}(t) + \beta N_2 i_3(t) - (1-\beta) N_2 i_4(t) \tag{A2.16}$$

$$0 = e_{c1}(t) - \alpha N_1 z_5(t) \tag{A2.17}$$

$$0 = e_{c2}(t) - (1-\alpha) N_1 z_5(t) \tag{A2.18}$$

$$0 = e_{c3}(t) - \beta N_2 z_5(t) \tag{A2.19}$$

$$0 = e_{c4}(t) - (1-\beta) N_2 z_5(t) \tag{A2.20}$$

$$0 = -i_1(t) + i_{c1}(t) + i_m(t) + g_c\left(e_{c1}(t) + e_{c2}(t)\right) \tag{A2.21}$$

$$0 = -i_2(t) + i_{c2}(t) - i_m(t) - g_c\left(e_{c1}(t) + e_{c2}(t)\right) \tag{A2.22}$$

$$0 = i_m(t) - i_0 y_m(t) sign(\Phi(t))^{n+1} \quad \text{(A2.23)}$$

$$0 = y_1(t) - \frac{\Phi(t)^2}{\Phi_0^2} \quad \text{(A2.24)}$$

$$0 = y_2(t) - y_2(t)^2 \quad \text{(A2.25)}$$

……

……

$$0 = y_{m1}(t) - y_{m1-1}(t)^2 \quad \text{(A2.23+m1)}$$

$$0 = y_{m1+1}(t) - y_{i1}(t) \cdot y_{j1}(t) \quad \text{(A2.23+m1+1)}$$

$$0 = y_{m1+2}(t) - y_{m1+1}(t) \cdot y_{j2}(t) \quad \text{(A2.23+m1+2)}$$

……

……

$$\begin{cases} 0 = y_m(t) - y_{m-1}(t) \cdot y_{jm2}(t), \ \textit{if n even} \\ 0 = y_m(t) - y_{m-1}(t) \cdot \dfrac{\Phi(t)}{\Phi_0}, \ \textit{if n odd} \end{cases} \quad \text{(A2.23+m)}$$

There are 23+m states

$$X(t) = \begin{bmatrix} v_1(t), v_2(t), v_3(t), v_4(t), v_5(t), v_6(t), \\ i_{L1}(t), i_{L2}(t), i_{L3}(t), i_{L4}(t), \emptyset(t), \\ z_1(t), z_2(t), z_3(t), z_4(t), z_5(t), \\ e_{c1}(t), e_{c2}(t), e_{c3}(t), e_{c4}(t), i_{c1}(t), i_{c2}(t), \\ i_m(t), y_1(t), y_2(t), \cdots\cdots, y_m(t) \end{bmatrix}$$

And 6 though variables

$$I = [i_1(t), i_2(t), i_3(t), i_4(t), i_5(t), i_6(t)]$$

Based on the above formulation, the number of additional internal states and equations m is computed as follows:

$m = m_1 + m_2$
where
$m_1 = \text{int}(\log_2(n))$

$m_2 =$ (# of ones in the binary representation of n) – 1

The sets of indices i and j in the last set of equations (M4.23+m1+1 to M4.23+m) are provided by positions of ones in the binary representation of n. The values of these indices are equal to the values of the power of 2 corresponding to that position, meaning that the right most positions is indexed 0 and the left most indexed $\text{int}(\log_2(n))$. The variable y_0 is by definition equal to λ, so it is not used and appears as λ in the equations. If the symbol y_0 was used instead of λ then only the first case in equation (M4.23+m1+m2) would have been necessary, since this would include the second when the index becomes 0. However, since λ is physically related to flux, for the time being the use of this symbol is preferred instead.

Time Domain Model –AQCF Model

The differential equations in above model are integrated with the quadratic integration method and the equations that are algebraic are sufficed to be written at times t and t_m. The AQCF model yields the following model.

At Time t

$$i_1(t) = i_{L1}(t) + g_{s1}\alpha L_1 z_1(t) \tag{A3.1}$$

$$i_2(t) = i_{L2}(t) + g_{s2}(1-\alpha) L_1 z_2(t) \tag{A3.2}$$

$$i_3(t) = i_{L3}(t) + g_{s3}\beta L_2 z_3(t) \tag{A3.3}$$

$$i_4(t) = i_{L4}(t) + g_{s4}(1-\beta) L_2 z_4(t) \tag{A3.4}$$

$$i_5(t) = -i_1(t) - i_2(t) \tag{A3.5}$$

$$i_6(t) = -i_3(t) - i_4(t) \tag{A3.6}$$

$$0 = i_{L1}(t) - \frac{h}{6} z_1(t) - \frac{2h}{3} z_1(t_m) - i_{L1}(t-h) - \frac{h}{6} z_1(t-h) \tag{A3.7}$$

$$0 = i_{L2}(t) - \frac{h}{6} z_2(t) - \frac{2h}{3} z_2(t_m) - i_{L2}(t-h) - \frac{h}{6} z_2(t-h) \tag{A3.8}$$

$$0 = i_{L3}(t) - \frac{h}{6} z_3(t) - \frac{2h}{3} z_3(t_m) - i_{L3}(t-h) - \frac{h}{6} z_3(t-h) \tag{A3.9}$$

$$0 = i_{L4}(t) - \frac{h}{6} z_4(t) - \frac{2h}{3} z_4(t_m) - i_{L4}(t-h) - \frac{h}{6} z_4(t-h) \tag{A3.10}$$

$$0 = \Phi(t) - \frac{h}{6} z_5(t) - \frac{2h}{3} z_5(t_m) - \Phi(t-h) - \frac{h}{6} z_5(t-h) \tag{A3.11}$$

$$0 = v_1(t) - v_5(t) - \alpha r_1 i_1(t) - \alpha L_1 z_1(t) - e_{c1}(t) \tag{A3.12}$$

$$0 = v_2(t) - v_5(t) - (1-\alpha) r_1 i_2(t) - (1-\alpha) L_1 z_2(t) + e_{c2}(t) \tag{A3.13}$$

$$0 = v_3(t) - v_6(t) - \beta r_2 i_1(t) - \beta L_2 z_3(t) - e_{c3}(t) \tag{A3.14}$$

$$0 = v_4(t) - v_6(t) - (1-\beta) r_2 i_2(t) - (1-\beta) L_2 z_4(t) + e_{c4}(t) \tag{A3.15}$$

$$0 = \alpha N_1 i_{c1}(t) - (1-\alpha) N_1 i_{c2}(t) + \beta N_2 i_3(t) - (1-\beta) N_2 i_4(t) \tag{A3.16}$$

$$0 = e_{c1}(t) - \alpha N_1 z_5(t) \tag{A3.17}$$

$$0 = e_{c2}(t) - (1-\alpha) N_1 z_5(t) \tag{A3.18}$$

$$0 = e_{c3}(t) - \beta N_2 z_5(t) \tag{A3.19}$$

$$0 = e_{c4}(t) - (1-\beta) N_2 z_5(t) \tag{A3.20}$$

$$0 = -i_1(t) + i_{c1}(t) + i_m(t) + g_c\left(e_{c1}(t) + e_{c2}(t)\right) \tag{A3.21}$$

$$0 = -i_2(t) + i_{c2}(t) - i_m(t) - g_c\left(e_{c1}(t) + e_{c2}(t)\right) \tag{A3.22}$$

$$0 = i_m(t) - i_0 y_m(t) sign(\Phi(t))^{n+1} \tag{A3.23}$$

$$0 = y_1(t) - \frac{\Phi(t)^2}{\Phi_0^2} \tag{A3.24}$$

$$0 = y_2(t) - y_2(t)^2 \tag{A3.25}$$

……

……

$$0 = y_{m1}(t) - y_{m1-1}(t)^2 \tag{A3.23+m1}$$

$$0 = y_{m1+1}(t) - y_{i1}(t) \cdot y_{j1}(t) \tag{A3.23+m1+1}$$

$$0 = y_{m1+2}(t) - y_{m1+1}(t) \cdot y_{j2}(t) \tag{A3.23+m1+2}$$

……

……

$$\begin{cases} 0 = y_m(t) - y_{m-1}(t) \cdot y_{jm2}(t), \ \textit{if n even} \\ 0 = y_m(t) - y_{m-1}(t) \cdot \dfrac{\Phi(t)}{\Phi_0}, \ \textit{if n odd} \end{cases} \tag{A3.23+m}$$

At Time t_m

$$i_1(t_m) = i_{L1}(t_m) + g_{s1}\alpha L_1 z_1(t_m) \quad \text{(A3m.1)}$$

$$i_2(t_m) = i_{L2}(t_m) + g_{s2}(1-\alpha) L_1 z_2(t_m) \quad \text{(A3m.2)}$$

$$i_3(t_m) = i_{L3}(t_m) + g_{s3}\beta L_2 z_3(t_m) \quad \text{(A3m.3)}$$

$$i_4(t_m) = i_{L4}(t_m) + g_{s4}(1-\beta) L_2 z_4(t_m) \quad \text{(A3m.4)}$$

$$i_5(t_m) = -i_1(t_m) - i_2(t_m) \quad \text{(A3m.5)}$$

$$i_6(t_m) = -i_3(t_m) - i_4(t_m) \quad \text{(A3m.6)}$$

$$0 = i_{L1}(t_m) + \frac{h}{24} z_1(t) - \frac{h}{3} z_1(t_m) - i_{L1}(t-h) - \frac{5h}{24} z_1(t-h) \quad \text{(A3m.7)}$$

$$0 = i_{L2}(t_m) + \frac{h}{24} z_2(t) - \frac{h}{3} z_2(t_m) - i_{L2}(t-h) - \frac{5h}{24} z_2(t-h) \quad \text{(A3m.8)}$$

$$0 = i_{L3}(t_m) + \frac{h}{24} z_3(t) - \frac{h}{3} z_3(t_m) - i_{L3}(t-h) - \frac{5h}{24} z_3(t-h) \quad \text{(A3m.9)}$$

$$0 = i_{L4}(t_m) + \frac{h}{24} z_4(t) - \frac{h}{3} z_4(t_m) - i_{L4}(t-h) - \frac{5h}{24} z_4(t-h) \quad \text{(A3m.10)}$$

$$0 = \Phi(t_m) + \frac{h}{24} z_5(t) - \frac{h}{3} z_5(t_m) - \Phi(t-h) - \frac{5h}{24} z_5(t-h) \quad \text{(A3m.11)}$$

$$0 = v_1(t_m) - v_5(t_m) - \alpha r_1 i_1(t_m) - \alpha L_1 z_1(t_m) - e_{c1}(t_m) \quad \text{(A3m.12)}$$

$$0 = v_2(t_m) - v_5(t_m) - (1-\alpha) r_1 i_2(t_m) - (1-\alpha) L_1 z_2(t_m) + e_{c2}(t_m) \quad \text{(A3m.13)}$$

$$0 = v_3(t_m) - v_6(t_m) - \beta r_2 i_1(t_m) - \beta L_2 z_3(t_m) - e_{c3}(t_m) \quad \text{(A3m.14)}$$

$$0 = v_4(t_m) - v_6(t_m) - (1-\beta) r_2 i_2(t_m) - (1-\beta) L_2 z_4(t_m) + e_{c4}(t_m) \quad \text{(5m.15)}$$

$$0 = \alpha N_1 i_{c1}(t_m) - (1-\alpha) N_1 i_{c2}(t_m) + \beta N_2 i_3(t_m) - (1-\beta) N_2 i_4(t_m) \quad \text{(5m.16)}$$

$$0 = e_{c1}(t_m) - \alpha N_1 z_5(t_m) \quad \text{(5m.17)}$$

$$0 = e_{c2}(t_m) - (1-\alpha) N_1 z_5(t_m) \quad \text{(5m.18)}$$

$$0 = e_{c3}(t_m) - \beta N_2 z_5(t_m) \quad \text{(5m.19)}$$

$$0 = e_{c4}(t_m) - (1-\beta) N_2 z_5(t_m) \quad \text{(5m.20)}$$

$$0 = -i_1(t_m) + i_{c1}(t_m) + i_m(t_m) + g_c\left(e_{c1}(t_m) + e_{c2}(t_m)\right) \quad \text{(5m.21)}$$

$$0 = -i_2(t_m) + i_{c2}(t_m) - i_m(t_m) - g_c\left(e_{c1}(t_m) + e_{c2}(t_m)\right) \tag{5m.22}$$

$$0 = i_m(t_m) - i_0 y_m(t_m) sign(\Phi(t_m))^{n+1} \tag{5m.23}$$

$$0 = y_1(t_m) - \frac{\Phi(t_m)^2}{\Phi_0^2} \tag{5m.24}$$

$$0 = y_2(t_m) - y_2(t_m)^2 \tag{5m.25}$$

……

……

$$0 = y_{m1}(t_m) - y_{m1-1}(t_m)^2 \tag{5m.23+m1}$$

$$0 = y_{m1+1}(t_m) - y_{i1}(t_m) \cdot y_{j1}(t_m) \tag{5m.23+m1+1}$$

$$0 = y_{m1+2}(t_m) - y_{m1+1}(t_m) \cdot y_{j2}(t_m) \tag{5m.23+m1+2}$$

……

……

$$\begin{cases} 0 = y_m(t_m) - y_{m-1}(t_m) \cdot y_{jm2}(t_m), \ \textit{if n even} \\ 0 = y_m(t_m) - y_{m-1}(t_m) \cdot \dfrac{\Phi(t_m)}{\Phi_0}, \ \textit{if n odd} \end{cases} \tag{5m.23+m}$$

There are 46+2m ‘states’

$$X = [X(t) \quad X(t_m)]$$

$$X(t) = \begin{bmatrix} v_1(t), v_2(t), v_3(t), v_4(t), v_5(t), v_6(t), \\ i_{L1}(t), i_{L2}(t), i_{L3}(t), i_{L4}(t), \lambda(t), \\ z_1(t), z_2(t), z_3(t), z_4(t), z_5(t), \\ e_{c1}(t), e_{c2}(t), e_{c3}(t), e_{c4}(t), i_{c1}(t), i_{c2}(t), \\ i_m(t), y_1(t), y_2(t), \cdots\cdots, y_m(t) \end{bmatrix}$$

And 12 ‘through variables’

$$I = \begin{bmatrix} i_1(t), i_2(t), i_3(t), i_4(t), i_5(t), i_6(t) \\ i_1(t_m), i_2(t_m), i_3(t_m), i_4(t_m), i_5(t_m), i_6(t_m) \end{bmatrix}$$