The Frequency Monitor Network (FNET) 
Design and Situation Awareness Algorithm Development

By

Jian Zuo

Dissertation submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Approved by:

Dr. Yilu Liu (Chair)  
Dr. Richard W. Conners  
Dr. Jih-Sheng (Jason) Lai  
Dr. Ahmad Safaai-Jazi  
Dr. Tao Lin

Mar 26, 2008  
Blacksburg, Virginia U.S.A.

Keywords: Wide Area Measurement and Control (WAMC), Recursive Discrete Fourier Transfer (RDFT), FNET, Multiple Units based Frequency Deviation Detection (MUFDD), Swing Equation, Single Machine Infinite Bus (SMIB), Time Difference of Arrival (TDOA), Event location estimation

Copyright 2008, Jian Zuo
Wide Area Measurements (WAMs) have been widely used in the energy management system (EMS) of power system for monitoring, operation and control. In recent years, the advent of synchronized Phasor Measurements Unit (PMU) has added another dimension to the field of wide-area measurement. However, the high cost of the PMU, which includes the manufacture and deployment fee, is a hurdle to the wide use of the PMU in power systems. Unlike traditional PMUs, the frequency monitoring network (FNET) developed by the Virginia Tech Power IT lab is an Internet–based, GPS–synchronized, wide-area frequency monitoring network deployed at the distribution level, providing a low-cost and easily deployable WAMs solution. In this dissertation, the research work can be categorized into two parts: FNET Design and Situation Awareness Algorithm Development.

In the first part, the author first analyzes the Recursive Discrete Fourier Transfer (RDFT) method, which is a core frequency estimation algorithm proposed by Dr A.G. Phadke and J.S. Thorp and modified by J. Chen for FNET. This algorithm is proved to be theoretically accurate. However due to the limitation of hardware, the sampling clock induces inaccuracy via a residue problem. After further investigation of the residue problem, a variable computation windows size method is developed to overcome this hardware limitation and provide accurate frequency estimation.

Following the improvement on the frequency estimation algorithm design, the author proposes new function design, called the “Automatic Geographic Information System (GIS) Report”, for FNET. Unlike the PMU, the Frequency Disturbance Record (FDR), which is the frequency measurement device installed on distribution level for FNET, may be frequently relocated. Therefore, to track the current locations of FDRs is important for further application based on FNET data. An automatic Geographic Information System (GIS) Report function enables FNET to automatically track the position of each FDR.
The second part of this research work focuses on Situation Awareness Algorithm developments, which are primarily designed for distribution-level WAMs, but are certainly not limited to that application. First, a Multiple Units-based Frequency Deviation Detection (MUFDD) algorithm is developed. This algorithm provides a method for monitoring power systems and detecting disturbances based on frequency deviation in real-time. It has been proven to be accurate and efficient for detecting system disturbances with low-cost distribution-level measurements of FNET. Then an oscillation-based power system event location estimation algorithm is developed. It is the first time the concept that the frequency oscillation is related to the distance from the measurement to the disturbance source is used to estimate the event location based on distribution-level measurements. Finally, after the introduction of work integrating FNET with SuperPDC, which is the phasor data platform developed by Tennessee Valley Authority (TVA), the Virtual Phasor Reference Angle (VPRA) calculation algorithm is developed. The VPRA is critical for future phasor applications.
Acknowledgement

First, I would like to express my sincere appreciation to Prof. Yilu Liu, my academic advisor, for her consistent encouragement, patience and support throughout my research activities at Virginia Tech. Her selfless help during my early days in this new country and new environment are my most unforgettable memories. Not to mention that her lavishly sharing her knowledge and creativity helped me break through many obstacles in my academic career.

Much appreciation also goes to Prof. Richard W. Conners, Prof. Jih-Sheng (Jason) Lai, Prof. Ahmad Safaai-Jazi and Prof. Tao Lin for their valuable comments and suggestions on my research work and serving as my committee members. The research work would not have been possible without their guidance and assistance.

I would like to express my gratitude and appreciation to all the professors, friends and students in Power Lab. Many thanks go to faculty members, Prof. Virgilio Centeno, Prof. Arun G. Phade, Prof. James S. Thorp and my colleagues, Dr. Li Zhang, Dr. Zhan Zhong, Dr. Shu-Jen Steven Tsai, Dr. Chunchun Xu, Dr. Mark Baldwin, Dr. R. Matthew Gardner, Dr. Hengxu Zhang, Dr. Dawei Fan, Dr. Wei Li, Dr. Kyung Soo Kook, Dr. Chunjuan Jia, Dr. Macros Angel Donolo, Dr. Qing Sheng Zhao, Dr. Jin Tao, Mr. Andrew Jex Arana, Mr. Emanuel Ernesto Bernabeu, Mr. Keith McKenzie, Ms. Jinyuan Dong, Mr. Josh Wang, Mr. Lei Wang, Mr. Joshua Wang, Mr. Jason Bank, Mr. Jon Burget, Mr. Kevin Khan, Mr. Tao Xia, Mrs. Ming Zhou, Mrs. Vivian Liang, Mr. Yicheng Zhang, Ms. Shengnan Shao and Mr. Chong Pan, for their academic inspiration, support and friendship. Many thanks also go to friends in Blacksburg for their generous help, brotherhood and friendship. Special thanks go to Dr. Wei Li, Dr. R. Matthew Gardner and Ms. Jingyuan Dong for their suggestions on my dissertation and helps on proofreading.

Part of this work was supported by Tennessee Valley Authority (TVA), the Electric Power Research Institute (EPRI), and the National Science Foundation (NSF). I would like to thank friends at TVA, Mr. Ritchie J. Carrol, Mrs. Sandra C. Affare, Mr. Paul Trachian, Mr. Bruce E. Rogers and Mrs. Lisa Beard, Mr. and Mrs. Arts, Mr. and Mrs. Mehul, Mr. Darrel, Mr. Pinal Patel, Mr. Tad Beaty, Ms. Keran McMickle, and Mrs. Sudha Kashyap for their support and friendship during my stay at TVA. My experience of
working at TVA will be very important to my future career life. I also want to thank Dr. Guorui Zhang from EPRI for his guidance.

I would like to express gratitude to my wife, Su Luo, for her love, support and companionship during the stressful time pursuing my degree. As both of us have been pursuing degrees, this hectic life would not have been easy without her companionship and love. Her loving support has been and always will be my most precious treasure in the world. Moreover, how lucky I am to have such wonderful parents and parents-in-law who always encourage us to pursue our dream with their selflessness love and support.
To Luo, Su
and our Parents
# Table of Contents

List of Figures ........................................................................................................................................ ix

List of Tables ........................................................................................................................................... xi

Chapter 1  Introduction ......................................................................................................................... 1

1.1 Review of Synchronized Wide Area Measurement (WAMs) .................................................. 1
1.2 Synchronized Wide Area Measurement (WAMs) Applications ................................................. 3
1.3 Introduction of Virginia Tech Synchronized Frequency Monitor Network ................................. 4
1.4 Organization of Study ..................................................................................................................... 9

Chapter 2  FNET Frequency Estimation Algorithm .......................................................................... 11

2.1 Overview ........................................................................................................................................ 11
2.2 Frequency Estimation in Power System ....................................................................................... 12
2.3 Residue problem and Variable Computation Window Solution ................................................. 18
   2.3.1 Residue problem .................................................................................................................. 19
   2.3.2 Variable Computation Windows ....................................................................................... 29
2.4 Testing and conclusion ................................................................................................................... 35

Chapter 3  Time Stamp Calibration and Automatic GIS Info Report Design ................................. 39

3.1 Introduction .................................................................................................................................... 39
3.2 FDR time stamp calibration .......................................................................................................... 40
3.3 Automatic GIS Information Report Design .................................................................................. 43
3.4 Conclusion ....................................................................................................................................... 46

Chapter 4  Power System Dynamic Disturbance Detection Based on FNET ................................. 48

4.1 Power System Dynamics and Operation States ........................................................................... 48
4.2 Frequency Deviation and Power System Disturbance ................................................................ 49
4.3 Multiple Units-based Frequency Deviation Detection (MUFDD) Algorithm .......................... 52
4.4 System Frequency Dynamics during Disturbance ....................................................................... 58
4.5 On-line Event Database ............................................................................................................... 62
4.6 Conclusion ....................................................................................................................................... 64

Chapter 5  Event Location Estimation Based on Frequency Oscillation ........................................ 65

5.1 Steady Frequency and Frequency Dynamics during Disturbance .............................................. 65
5.2 Single Machine Infinite Bus (SMIB) and Frequency Deviation Dynamic .............................. 65
5.3 Two-Area Model and Frequency Deviation Dynamic ............................................................... 69
# Table of Content

5.4 Electromechanical Wave Propagation Based Disturbance Location Estimation.. 75  
5.5 Using Frequency Oscillation to Improve Event Location Estimation .......... 78  
5.6 Weight Functions .................................................................................. 83  
5.7 Testing and Case Analysis ...................................................................... 84  
5.8 Conclusion ............................................................................................ 87  

Chapter 6  

FNET and Phasor Application Based on SuperPDC ....................................... 89  
6.1 Introduction of SuperPDC ....................................................................... 89  
6.2 SuperPDC Structure and Integration with FNET ...................................... 90  
6.3 Virtual Phasor Reference Angle Application Base on SuperPDC .......... 95  
6.4 Proposed Future Applications based on the SuperPDC ......................... 100  
6.5 Conclusion .............................................................................................. 101  

Chapter 7  

Conclusion and Future Work .......................................................................... 102  
7.1 Conclusion ............................................................................................... 102  
7.2 Contribution ............................................................................................. 104  
7.3 Future Work ............................................................................................. 106  

References ........................................................................................................ 107  

Appendix A. Time stamp function module of GPS receiver module ............... 112  
Appendix B. Multiple-Unit based Frequency Deviation Detection (MUFDD) .... 114  
Appendix C. Oscillation based Disturbance Location Estimation (ODLE) .......... 120  
Appendix D. Virtual Phasor Reference Angle (VPRA) Calculation .......... 124
List of Figures

Figure 1.1 Wide Area Measurement Development Chronicle ................................................. 3
Figure 1.2 The Frequency Monitor Network ........................................................................ 6
Figure 1.3 Deployment of FNET in US .................................................................................. 7
Figure 1.4 Data Streamer Table of 5-second Average Real-time Frequency ....................... 8
Figure 2.1 Waveform Re-sampling ..................................................................................... 17
Figure 2.2 Diagram of Sampling System of FDR ................................................................. 19
Figure 2.3 MIOS Block Diagram ......................................................................................... 21
Figure 2.4 DASM Block Diagram ....................................................................................... 22
Figure 2.5 Flowchart of Sampling Clock Generation ........................................................... 23
Figure 2.6 Sampling Time Shift Caused by Residue Problem ............................................. 24
Figure 2.7 Testing Result of Losing Sampling Point ............................................................ 28
Figure 2.8 Angle Discontinuity Caused by Residue Problem ............................................. 28
Figure 2.9 Sampling Clock Waiting Time ........................................................................... 30
Figure 2.10 Different Sampling Intervals Due to Residue Correction ............................... 31
Figure 2.11 Impact of Different Sampling Interval ............................................................. 32
Figure 2.12 Frequency Dips ............................................................................................... 32
Figure 2.13 Flowchart of Variable Computation Window Size Implementation ................. 34
Figure 2.14 Testing Environment Setup ............................................................................. 35
Figure 2.15 Voltage Angle Comparisons with Fix and Variable Computation Window .... 36
Figure 2.16 Frequency Comparisons with Fixed and Variable Computation Window ....... 37
Figure 3.1 10-Second Difference between FDR and PMU .................................................. 40
Figure 3.2 Flowchart of Tagging Time stamp with Measured Data by FDR ....................... 41
Figure 3.3 Flowchart of Tagging Time Stamp without Buffer by FDR ................................. 42
Figure 3.4 Sub-second Time Difference between FDR and PMUs [31] ............................ 43
Figure 3.5 Flowchart of Encapsulate the GIS Information into Data Stream ..................... 45
Figure 3.6 Data Stream with GIS Information .................................................................. 46
Figure 4.1 Time Frame of the Power System Dynamic Phenomena [23] ........................... 48
Figure 4.2 Power System Operation States ......................................................................... 49
Figure 4.3 Typical Frequency Excursion during Frequency Drop Disturbance ................. 54
Figure 4.4 Frequency Raw Data of a Single Unit (EUS) ..................................................... 55
Figure 4.5 Comparing $\frac{df}{dt}$ based on Single Unit and Multiple Units ......................... 56
Figure 4.6 Detail of the System Frequency Dynamics and $\frac{df}{dt}$ at Disturbance ............. 56
Figure 4.7 WECC System Frequency $\frac{df}{dt}$ and Disturbance Dynamic ......................... 58
Figure 4.8 System Frequency Pattern of Disturbance ................................................. 59
Figure 4.9 Local Low Frequency Oscillations with Disturbance .................................. 60
Figure 4.10 Electromechanical-wave Propagation in Disturbance ................................. 61
Figure 4.11 DC Transmission Line Trip ......................................................................... 61
Figure 4.12 System Frequency in LA Blackout on Sep 12, 2005 ..................................... 62
Figure 4.13 Flow Chart of On-line Event Database ....................................................... 63
Figure 5.1 Single Machine Infinite Bus System ............................................................... 66
Figure 5.2 Lossless Transmission Voltage Profiles with Different Load Condition ......... 67
Figure 5.3 Angle Variations along Transmission Line ..................................................... 69
Figure 5.4 Two-Area Model ........................................................................................... 70
Figure 5.5 Transfer Function Relating Speed and Power ................................................. 70
Figure 5.6 Equivalent System Transfer Function Diagram of Two-area Model ............... 72
Figure 5.7 Simplified System Transfer Function Diagram ............................................. 72
Figure 5.8 Block Diagram of System Transfer Function ................................................ 73
Figure 5.9 Typical Frequency Drop of Generator Trip in EUS [63] ................................. 77
Figure 5.10 Comparisons of FDR Measurement and PSS/E Simulation ...................... 77
Figure 5.11 Frequency Waveforms in Bruce Unit Tripping ........................................... 79
Figure 5.12 Locations of Bruce, Toronto, RPI and VT .................................................. 79
Figure 5.13 Curve-fitting of Frequency Waveform ....................................................... 81
Figure 5.14 Difference between curve fitting and original data ..................................... 81
Figure 5.15 First Order Deviation of Curve Fitting ....................................................... 82
Figure 5.16 Location Estimation for Bruce Case ............................................................ 84
Figure 5.17 Frequency Perturbation of Vogtle Plant Case .............................................. 85
Figure 5.18 Location Estimation for Vogtle Plant Case ............................................... 85
Figure 5.19 Case 19 –Lacygne and FDR Position ......................................................... 87
Figure 6.1 The Placement of PMUs (deployed and proposed) ......................................... 90
Figure 6.2 SuperPDC Structure .................................................................................... 91
Figure 6.3 Data Structure of Interface and Different Protocol ...................................... 93
Figure 6.4 The Virtual Reference Angle at Browns Ferry Bus ..................................... 95
Figure 6.5 Phasor Angle Unwrapping with Starting Point ............................................ 96
Figure 6.6 Unwrapping Start from Point B ................................................................. 96
Figure 6.7 Unwrapping Problem in Previous Algorithm ............................................. 98
Figure 6.8 Virtual Phasor Reference Angle with SuperPDC .......................................... 100
List of Tables

Table 1.1 Comparing PMU and FDR ................................................................. 5
Table 3.1 GPS Message Format.................................................................. 44
Table 4.1 Threshold for Different Interconnections.............................. 54
Table 4.2 Disturbances Detected by Trigger (June to Sep 2005).......... 58
Table 5.1 Disturbance Location Estimation Comparison ....................... 86
Chapter 1

Introduction

1.1 Review of Synchronized Wide Area Measurement (WAMs)

Deregulation, competition and an increase in the complexity of today’s electric power system interconnections have exacerbated power system stability issues. The problems encountered by today’s power system operation are system-wide disturbances, which are not able to be covered by existing protection and control system. As power grids get even more heavily loaded by transient bulk power transfers, the system becomes more vulnerable and even minor equipment failures can result in cascading trips and eventually system-wide blackouts. For example, the August 14, 2003 blackout, which was the largest blackout in North American history, was caused by cascade tripping. It affected an estimated 10 million people in the province of Ontario (about one-third of the population of Canada) and 40 million people in eight U.S. states (about one-seventh of the population of the U.S.). Outage-related financial losses were estimated at $6 billion USD [1-3].

To ensure system stability in a heavily loaded system, all or most installed components should remain in service and the right control action must be taken quickly if some of them fail to function properly and cause a disturbance. To meet this requirement, a real-time monitoring system is needed. A wide-area measurement system provides power system operators with real-time knowledge of various instability issues and events when they occur. A typical wide-area measurement system or WAMS system is built upon a reliable communication system connecting power stations, network control centers and sub stations. The GPS satellite system is used for synchronization and timing accuracy, and a number of Phasor Measurement Units or PMUs stream the synchronized
real-time data through the communication link to data concentrator.

As the major component, the Phasor Measurement Unit (PMU) was introduced in mid-1980s. Since then, the subject of wide-area measurements in a power system using PMUs has been receiving considerable attention from researchers in the fields. The first PMU development was supported at Virginia Tech. One of the milestone leaps for PMU development was the recursive algorithm for calculating symmetrical components of voltages and currents, known as the Symmetrical Components of Discrete Fourier Transform (SCDFT)[4], as one of the major outcomes of the development of the Symmetrical Component Distance Relay (SCDR) for protecting high-voltage transmission lines between the 70s and 80s. This led to the next step of development: a synchronized system clock for sampling voltage and current at different locations across a power system network. The Global Positioning System (GPS) provided a perfect solution for this problem. Although the precision of synchronization was not very good in the early years of this system, at present it is possible to achieve synchronization accuracies of 1 μs, which is coordinate to 0.021° in a 60 Hz system.

Based upon the prototype PMU developed at Virginia Tech, Macrodyne Co. began to manufacture commercial PMUs. Several new features were added to their units, including an internal GPS receiver, with a 16-bits sigma-delta analog-to-digital converter for each analog input channel, and several modem interfaces for remote access to the PMU [4]. A data concentrator was also developed to collect data from many PMUs, align the data by the time stamps and provide phasor outputs for application software as needed. In the meantime, an IEEE standard 1344 [5] for PMUs that defines the output data format of PMUs was also developed. Over the years, the total number of PMUs installed world-wide has reached beyond hundreds. As more and more phasor applications emerge,
it is expected this number will increase even faster. Figure 1.1 shows a chronicle of the WAMs development.

![Figure 1.1 Wide Area Measurement Development Chronicle](image)

1.2 Synchronized Wide Area Measurement (WAMs) Applications

As the Wide Area Measurements (WAMs) have been developed, more and more research is focus on the applications of WAMs in Power Systems. An example of this is integrating WAMs with a state estimator. With the direct measurement of synchronized positive sequence voltages and currents from the power grid, the “state estimation” can be shifted to “state measurement” [6-8]. Provided with assured synchronization and a high rate of measurement data possible, one can for the first time consider monitoring the dynamic phenomena in real-time.

Besides the real-time monitoring of power system dynamics, some research has been done on using WAMs in feed-back control. The adaptive relay is the key component of such research [9-11]. However, some of the concerns of using the PMU as the feedback of the controllers are data transmission latency and data availability [12-14].
Model validation is another application of WAMs [7]. In a power system, operators rely on system models to know how power flows will change either as a result of manual or relay-initiated changes to topology, and they can make economical changes at low customer cost. However, if the system model is not accurate enough, such actions may violate the system stability criteria, such as loading a transmission line over the stability limit. Therefore using WAMs to validate a system model and improve model accuracy are very crucial for the economic optimization of a system, and to insure operators are not violating system stability.

WAMs can also be used for power system post-event analysis[7]. Since all the data of WAMs are synchronized and time tagged, using WAMs as the data resource for the timeline is an obvious choice. With its help, the system dynamics during an event can be reconstructed and the sequence of events during cascade events can be revealed. Comparing the measured data at different time points with system model-based simulation, one can disclose what happens in the process of the event and what the impact is.

In summary, WAMs, as an emerging new technology for power systems, provide a very useful tool for modern power system operation.

1.3 Introduction of Virginia Tech Synchronized Frequency Monitor Network

As discussed in Chapter 1.1 and Chapter 1.2, PMUs have many merits, such as synchronized high-rate measurements including voltage, current and power flow, which is very attractive for power-system monitoring and stability control. However, the high
Chapter 1

cost of PMU manufacture and installation hamper the rapid deployment of PMUs over the whole power system network. The WAMs based on PMUs are still under construction in North America, and numerous research projects have been done with a focus on how to optimize the selection of PMU installation [15-20], to gain maximum observability with a limited number of PMUs.

<table>
<thead>
<tr>
<th>Table 1.1 Comparing PMU and FDR</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PMU</strong></td>
</tr>
<tr>
<td>![PMU Image]</td>
</tr>
<tr>
<td>Cost</td>
</tr>
<tr>
<td>$10,000 to $25,000 per unit,</td>
</tr>
<tr>
<td>not including installation</td>
</tr>
<tr>
<td>cost</td>
</tr>
<tr>
<td>Installation Site</td>
</tr>
<tr>
<td>Substation</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Phase</td>
</tr>
<tr>
<td>Three Phase</td>
</tr>
<tr>
<td>Communication</td>
</tr>
<tr>
<td>Dedicated communication channel</td>
</tr>
</tbody>
</table>

The Virginia Tech PowerIT research group approaches this cost and deployment problem from another perspective. PMUs are usually installed at substations measuring the voltage, current and power flow from a high-voltage transmission line of 500 kV, 345 kV, 230 kV and 161 kV, etc. A high voltage level means a high isolation standard is

---

1 Phasor Measurement Unit, Macodyne™, Inc Model 1690

5
required. Potential Transfer (PT) and Current Transfer are usually standard equipment for
PMUs for the purpose of isolation. Furthermore, PMUs, which are installed at a
substation, need a dedicated communication line, such as fiber cable. The cost for
isolation and communication are two major components of the cost for PMU applications.
The concept of using a simplified device to measure the power system frequency at the
distribution level was proposed by the Virginia Tech PowerIT group in 2000 by Qiu et al
[21]. With the development of Internet, this concept is further developed to use the
Internet for the communication infrastructure.

As shown in the infrastructure diagram in Fig. 1.2, these simplified PMU devices
measure the system frequency at the distribution level, synchronized with GPS time tag
from the GPS receiver, and send the data back to the Information Management System
(IMS) through the Internet. Then the data will be stored in a database for further
application. The system is called the Frequency Monitoring Network (FNET) [22]. With
the development of FNET, the cost of the WAMs solution in power systems has dramatically dropped, as shown in Table 1.1. FNET is for the first time providing the concept of measuring the synchronized power system frequency at the distribution level.

Figure 1.3 Deployment of FNET in US

Shown in Fig. 1.2, the FNET system consists of two major components: a) Frequency Disturbance Recorders (FDRs), which perform local GPS synchronized frequency measurements and send data to a server through the Internet; b) The Information Management System (IMS), which includes data collection and storage services, data communication services, database operation services, and web services. The FDRs sample the voltages at different locations, calculate the frequency, angle, and magnitude
of the voltage samples, and then send back the data with the GPS synchronized time stamp to the IMS. One of obvious applications of FNET is the real-time power system monitoring as shown in Fig. 1.4.

![Data Streamer Table of 5-second Average Real-time Frequency](image)

Figure 1.4 Data Streamer Table of 5-second Average Real-time Frequency

Although FNET has the merits of a low manufacturing cost and easy deployment, it will face some new challenges. Since the FDRs are installed at the 120V distribution level, the signal-to-noise ratio (SNR) is low; that is, small load changes at the distribution level will cause relatively high noise on the signal measurement. Most FDRs suffer various kinds of interference, such as Electro-Magnetic Interference (EMI), temperature variation, distribution-level load switching on and off, etc. These factors will introduce undesirable components to the measured signals. Besides these signal-to-noise ratio and
various noise component problems, the data discontinuity presents another challenge. If the FDR units lose the synchronization of the GPS signal or if the Internet communications suffer traffic congestion, which rarely happens with dedicated communication, the data transmitted from the FDRs to the IMS will be interrupted. Therefore, unlike PMUs, all the WAMs applications developed based on FNET should consider all these problems.

1.4 Organization of Study

The reminder of this dissertation is organized as follows:

Chapter 2 and Chapter 3 focus on the FNET design. Chapter 2 first addresses the frequency estimation accuracy problem in design. Then the residue problem in the sampling clock of the frequency estimation algorithm, which is a major cause of inaccuracy, is studied. Finally a variable calculation window size design is developed and the test results are demonstrated. Chapter 3 discusses the FDR time-stamp calibration issue with the PMU, and a new design of the automatic GIS information report function is developed.

Chapter 4, Chapter 5 and Chapter 6 focus on situation awareness algorithms developed based on FNET and phasor data. Chapter 4 analyzes the frequency dynamics during disturbance. A Multiple Units-based Frequency Deviation Detection (MUFDD) algorithm is developed.

In Chapter 5, the problems of the disturbance location estimation algorithm based on the Time Difference of Arrival (TDOA) are addressed. After a study of the frequency oscillation magnitude and the distance based on Single Machine Infinite Bus (SMIB) and
the Two-Area model, a frequency-oscillation-based disturbance location estimation algorithm is developed.

Chapter 6 reviews the North American Synchronized Phasor Initial (NASPI) and SuperPDC phasor project. The integration of the FNET with the SuperPDC is addressed. A Virtual Phasor Reference Angle (VPRA) calculation algorithm is developed and demonstrated with the SuperPDC.

Chapter 7 is the conclusion and ideas for future work.
Chapter 2  FNET Frequency Estimation Algorithm

2.1 Overview

An electrical power system is a complex and interconnected system. The most important and unique feature of an electrical power system is that electrical energy cannot easily and conveniently be stored in large quantities. This means that that at any instant the power generated has to meet the power demanded. During a normal situation, the combination of the power demanded is predictable and changes in a relatively slow manner even though the individual consumer load may change rapidly and unpredictably. This predictable system manner would allow the daily generation schedule to be planned and controlled in a predetermined manner [23].

In the system, the generators convert the mechanical energy appearing on the shaft of an engine into electrical energy. The conversion of mechanical energy to electrical energy is achieved by the use of synchronous generator. The synchronous generator feeds the electrical power into the transmission system through the step-up transfer in order to increase the voltage from generator level (10-20 kV) to the transmission level (hundreds of kV). The transmission network connects all the power systems to one system, and transmits and distributes the power to the load centers in an optimal way, creating an interconnected system. The stable operation of a power system depends on the ability to continuously match the electrical output of generators to the electrical demand on the system. Since this is a highly interconnected system, any outages or events that happen on some of the components, like a generator experiencing an unscheduled tripping, a line short circuit or a load center drop; will cause disturbance in the system, affect power quality including voltage and frequency stability, or even cause cascade events and
blackout. Because power system frequency has unique characteristics, especially when measured using synchronized wide-area measurement during disturbances across the whole interconnection, it has drawn particular interest from researchers. FNET was developed and served as a synchronized wide-area frequency measurement network on the distribution level.

2.2 Frequency Estimation in Power System

Since most generators are synchronous machines, and the power transmission system interconnects the different power stations as one system, under normal conditions the whole interconnected power system is running at the nominal frequency, which is 60 Hz in North America and 50 Hz in Europe and China. The power utilities will have a daily generation schedule according to load predication. In real time, the operator will adjust the generation and load in order to match the power generation and demand. The imbalance between the total amount of generation and the total amount of demand will cause the system frequency to deviate from the nominal value and become either greater or lower, in this manner: if the total amount of power generated is greater than the demand, which means the mechanical input into generator is greater than the electrical demand on the generator, the rotor of the generator will increase the spinning speed, causing the frequency to increase; in contrast, if the total amount of power generated is lower than the electrical demand on the generator, the rotor of the generator will decrease the spinning speed as the system frequency decreases. The swing equation, shown below, dominates this physics phenomenon [24]:
\[ J \frac{d\omega_m}{dt} = T_a = T_m - T_e \]  

where

- \( T_a \) = accelerating torque in N.m
- \( T_m \) = mechanical torque in N.m
- \( T_e \) = electromagnetic torque in N.m
- \( J \) = combined moment of inertia of generator and turbine, kg.m\(^2\)
- \( \omega_m \) = mechanical angular velocity of the rotor, rad/s, and
- \( t \) = time, s.

The imbalance of generation and demand may be caused by a load change, generation change or events like generator or line tripping etc. Slow system frequency change and variation are expected and acceptable, and these can be compensated by proper control actions. While the sudden and sharp changes in system frequency are usually caused by system events, which may introduce instability into the system if the operators are not fully aware of the system dynamics in time. System frequency is one of key parameters indicating the system dynamic. Numerous research projects have been done in the past thirty years on how to estimate power system frequency by taking advantage of the modern digital computer. As mentioned in [25], the techniques used for power system frequency estimation can be divided into four major categories:

1. Zero Crossing Technique
2. Least Square Error Technique
3. Frequency Domain Analysis
4. Kalman Filter Technique

As stated in [25], the noise in power system has three main components:
Chapter 2

- White Noise
- Harmonics Noise
- Random Spike Noise

The white noise contains the flat power over the entire frequency spectrum. The model with limited bandwidth for white noise will be used for analysis because the power system uses finite bandwidth. Harmonics noise is induced by harmonics components, like capacitors or inductance, and it will be the fundamental frequency times an integer number. The harmonics noise could be eliminated with a carefully designed low-pass filter. Random spike noise is generated by a random load, generation or system topology change, like capacitor switching on or off, lightning on the transmission line, etc. It is not necessary to deal with random spike noise in frequency estimation techniques design since it only happens occasionally. All the frequency estimation techniques use voltage instead of current for measurement because the voltage remains reasonably unchanging during normal operation.

After carefully comparing the accuracy, computational burden and response time of the aforementioned frequency estimation methods, the Recursive Discrete Fourier Transform proposed by Phadke [26], which is widely used for PMUs, is chosen for use with the FDR. Unlike the PMU, which uses a three-phase transmission line for the positive sequence measurement, the FDRs measure only one phase. The necessary modification has been done in [25] and the details are briefly described as follows, for the purpose of further analysis:

Assuming there are N sampling points \( \{x_k\}, k \in [1,N] \) per cycle of fundamental frequency, the phasor of the fundamental frequency component is:
\[
X_i = \frac{1}{\sqrt{2}} \left( \frac{2}{N} \sum_{k=1}^{N} x_k \cos \left( \frac{2\pi}{N} k \right) - j \frac{2}{N} \sum_{k=1}^{N} x_k \sin \left( \frac{2\pi}{N} k \right) \right)
\]  
(2.2)

Denoting \( \frac{2}{N} \sum_{k=1}^{N} x_k \cos \left( \frac{2\pi}{N} k \right) \) by \( X_c^{(i)} \) and \( -\frac{2}{N} \sum_{k=1}^{N} x_k \sin \left( \frac{2\pi}{N} k \right) \) by \( X_s^{(i)} \) and using the recursive formula derived in [25], each successive phasor can be calculated with

\[
X_c^{(k+1)} = X_c^{(k)} + \frac{2}{N} (x_{k+1} - x_{k+N}) \cos \left( \frac{2\pi}{N} k \right) \\
X_s^{(k+1)} = X_s^{(k)} - \frac{2}{N} (x_{k+1} - x_{k+N}) \sin \left( \frac{2\pi}{N} k \right)
\]  
(2.3)

Hence, a new phasor is available for every incoming sample after initialization. The angle of the \( k \)th phasor is given by:

\[
\phi(k) = \tan^{-1} \left( \frac{-X_s^{(k)}}{X_c^{(k)}} \right)
\]  
(2.4)

Assuming the phasor angle varies as a polynomial function with respect to the sampling number. The detailed analysis in [25] shows that the linear function doesn’t have enough accuracy, while the cubic function improves the accuracy only carrying the heavy computation burden. Therefore the quadratic function is used:

\[
\phi(k) = a_0 + a_1 k + a_2 k^2
\]  
(2.5)

Considering the computational window of \( M \) to estimate the coefficients \( a_0, a_1 \) and \( a_2 \):

\[
\phi(1) = a_0 + a_1 + a_2 \cdot 1^2 \\
\phi(2) = a_0 + a_1 \cdot 2 + a_2 \cdot 2^2 \\
\vdots \\
\phi(M) = a_0 + a_1 M + a_2 M^2
\]  
(2.6)

Rewriting Eq. (2.6) in matrix form:
\[
\begin{bmatrix}
\phi(1) \\
\phi(2) \\
\vdots \\
\phi(M)
\end{bmatrix} =
\begin{bmatrix}
1 & 1 & 1^2 \\
1 & 2 & 2^2 \\
\vdots & \vdots & \vdots \\
1 & M & M^2
\end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
a_2
\end{bmatrix}
\]  \hspace{1cm} (2.7)

or

\[
\Phi = Xa
\]  \hspace{1cm} (2.8)

The unknown coefficient matrix \( a \) can be estimated by the least square error method:

\[
a = [X^T X]^{-1} X^T \Phi
\]  \hspace{1cm} (2.9)

Here, the pseudo-inverse matrix \([X^T X]^{-1} X^T\) is known as the gain matrix, and can be computed off-line. As studied by [25], the phasor angle matrix \( \Phi \) should be monotonic and there should not be any wrap-around angles. To keep the error of the estimated value minimal, \( \phi(1) \) is set to be zero and the rest of the phasor angle is subtracted by \( \phi(1) \).

Once the values of \( a_1 \) and \( a_2 \) are calculated, we can use them to compute the frequency, which is the change rate of the phasor angle:

\[
\frac{d\phi}{dk} = a_1 + 2a_2k
\]  \hspace{1cm} (2.10)

The relationship between the sample number and the time is:

\[
k = Nf_0 t \quad \text{or} \quad \frac{dk}{dt} = Nf_0
\]  \hspace{1cm} (2.11)

where the \( f_0 \) is the fundamental frequency and \( N \) is the sample point being cycled.

Hence, derived by the chain rule:

\[
\frac{d\phi}{dt} = \frac{d\phi}{dk} \cdot \frac{dk}{dt} = (a_1 + 2a_2k)Nf_0 = (a_1 + 2a_2Nf_0 t)Nf_0
\]  \hspace{1cm} (2.12)

Since for recursive phasor angle, \( \frac{d\phi}{dt} \approx \Delta \sigma \),

\[
\Delta \sigma \approx Nf_0 (a_1 + 2a_2Nf_0 t) \Rightarrow \Delta f = \frac{1}{2\pi} Nf_0 (a_1 + 2a_2Nf_0 t)
\]  \hspace{1cm} (2.13)
As mentioned by [25], the frequency estimation obtained at the end of the computation window is the actually the frequency at the half cycle point before the end of the computation window. Since the power system frequency has a relatively slow change rate, it could be that the frequency remains the same within the computation window, and usually a longer computation window is used. To get a more accurate estimation, re-sampling is needed because the frequency estimation will be closer to the true value if the actual frequency approaches the sampling frequency. This also requires a long computation window. After a numerical analysis of the estimated frequency accuracy regarding a computation window size for a 60 Hz fundamental frequency [25], 8 cycles of fundamental frequency is chosen as the computation window.

By re-sampling, one can do the interpolation with the current voltage sampling points to maintain the constant sampling point per cycle, which is 24 point per cycle here, regardless of the actual frequency.

Figure 2.1 Waveform Re-sampling
Chapter 2

Referring to Fig. 2.1, the values of \( z_1 \) and \( z_2 \) are the original voltage sampling point. They can be expressed as:

\[
z_1 = Z_m \sin(\varphi) \tag{2.14}
\]
\[
z_2 = Z_m \sin(\varphi + \alpha) = Z_m \sin \varphi \cos \alpha + Z_m \cos \varphi \sin \alpha \tag{2.15}
\]

where \( Z_m \) is the amplitude

\( \varphi \) is the angle at the sampling instant for the new frequency

\( \alpha \) is the interval angle between the two samples \( z_1 \) and \( z_2 \) based on the new frequency and it is \( 2\pi f_{new}/(Nf_0) \).

Hence,

\[
Z_m \cos \varphi = \frac{(z_2 - z_1 \cos \alpha)}{\sin \alpha}. \tag{2.16}
\]

Let \( x \) be the fractional distance between \( z_1 \) and \( z_2 \), the re-sampling point \( z' \) is then given by:

\[
z' = Z_m \sin(\varphi + x\alpha)
= Z_m \sin \varphi \cos x\alpha + Z_m \cos \varphi \sin x\alpha \tag{2.17}
= z_1 \cos x\alpha + (z_2 - z_1 \cos \alpha) \frac{\sin x\alpha}{\sin \alpha}
\]

After the re-sampling points based on the sampling frequency of \( f_{\text{re-sample}} = f_0 + \Delta f' \), have been found, the angles of the new phasor will be calculated and another correction of frequency \( \Delta f' \) will be estimated. Hence the final frequency will be:

\[
f_{\text{final}} = f_0 + \Delta f' + \Delta f' \tag{2.18}
\]

2.3 Residue problem and Variable Computation Window

Solution

As described in Section 2.2, the technique proposed by [25] provides theoretically
accurate frequency estimation. However, it is based on the assumption that the sampling clock is perfect. For example, there are 24 sampling points per cycle for a 60 Hz assumed fundamental frequency, and there is a total 1440 sampling points per second. Therefore the sampling clock will be 1440 Hz as synchronized by the GPS. However the sampling clock generated by the clock division is not a perfect 1440 Hz clock pulse. Then the residue problem is presented. This residue problem will cause the inaccurate time skew in the measurement and will induce inaccuracy in the frequency estimation. The cause of residue problem and its solution is described in the following sections.

2.3.1 Residue problem

![Diagram of Sampling System of FDR](Image)
As shown in Fig. 2.2 of the FDR hardware implementation, first the step-down transformer converts the 120 VAC grid voltages to 10 VAC input voltage. After a low-pass filter, the input voltage signal is sampled by the sampling clock and converted from analog signal to a 16-bit digital signal by the analog-to-digital circuit board. On the main board, the modular I/O system (MIOS) [27] receives the synchronized pulse per second (PPS) from the GPS receiver, counts the internal 20 MHz system clock and divide the system clock to the sampling clock. The 32-bit central processor unit (CPU) cooperates with the floating processor unit, performing the Recursive Discrete Fourier Transform (RDFT) described in Section 2.2. The calculated voltage, angle and frequency, tagged by the GPS time sample are sent back to the data concentrator through the Internet at 10 points/sec.

The sampling clock generation and synchronization is done by the Modular I/O System (MIOS). Each MIOS has one interrupt sub-module, ten double-action sub-modules (DASM), and eight dedicated PWM sub-modules (PWMSMs), as shown in Fig. 2.3. The MIOS double action sub-module has two 16-bit consecutive input-captures or two 16-bit output-compare functions that occur automatically without software intervention. Here the MIOS double action sub-module channel 15 is configured to be in continuous pulse generation mode, which has two registers, register A and register B, configured as two continuous input captures. As show in Fig. 2.2 and Fig. 2.4, once the GPS receiver has locked onto the stable GPS signal, the stable PPS from the GPS receiver is input as an edge trigger. Whenever the edge detector detects the rising edge of the PPS, the value of register A, which is the captured 16-bit Counter Bus (CB) value at the previous rising edge of the PPS, is passed to register B. In the meantime, register A captures the current 16-bit value from Counter Bus. The Counter Bus is driven by the
system clock, which is 20 MHz here (40 MHz clock from the oscillator has already been divided by two). When the capture is done, the MIOS will assert an interrupt, and the software module could use the two 16-bit captures of register A and register B combined with an overflow counter to calculate the total system clock pulse number between the two consecutive PPSs. Then this value is divided by the total sampling number per second as the input value for MIOS pulse-width-modulation sub-module (PWMSM).

Figure 2.3 MIOS Block Diagram

---

2 Motorola MPC555 manual
PWMSM is the module used to generate the desired sampling frequency and it works as a frequency divider. The value of the total system clock pulse between two consecutive PPSs divided by the sampling number per second is the number of system clock pulses between adjacent sampling pulses:

\[ N_{\text{clipus\_sampling}} = \text{floor}\left( \frac{N_{\text{clipus\_PPS}}}{M} \right) \]  

(2.19)

where \( N_{\text{clipus\_sampling}} \) is number of the system clock pulses between two adjacent sampling pulses.

\( N_{\text{clipus\_PPS}} \) is number of the system clock pulses between two consecutive PPSs

\( M \) is the sampling point per second

---

**Figure 2.4 DASM Block Diagram**

---

3 Motorola MPC 555 Manual
\[ N_{clkpus_{PPS}} = R_A - R_B + clockTicks \] 

(2.20)

\[ clockTicks = overflowCNT \times 65536 \] 

(2.21)

where \( R_A \) and \( R_B \) are the values from capture register A and register B of MIOS.

\( overflowCNT \) is the overflow counter of the 16 bits Counter Bus.

\[ N_{clkpus_{PPS}} = R_A - R_B + clockTicks \]

\[ N_{clkpus_{sampling}} = \text{int}(N_{clkpus_{PPS}} / M) \]

Fig. 2.5 shows the flowchart of the sampling clock generation from the PPS and the system clock divide. As indicated by Eq. (2.19), \( N_{clkpus_{sampling}} \) should be an integer in order to generate a sampling clock. But \( N_{clkpus_{PPS}} \) divided by \( M \) may not be an integer, which means there will be a residue. For example, if the system clock is 20.125 MHz (the nominal system frequency is 20 MHz = 40 MHz /2, but the actual frequency may be a
little bit different since the voltage and temperature fluctuate over the voltage-control oscillator), and the number of sampling points is 1440 per second, then:

\[ N_{\text{clk pu} \_\text{PPS}} / M = 20.125 \times 10^6 / 1440 = 13975.69444 \]  

(2.22)

Since \( N_{\text{clk pu} \_\text{PPS}} \) will be an integer, we use 13975 as the value for \( N_{\text{clk pu} \_\text{PPS}} \), therefore the residue is:

\[ \text{Residue} = 20.125 \times 10^6 - (1440 \times 13975) = 1000 \]  

(2.23)

One thousand counts of the system clock at 20.125 MHz are equal to 0.0497x10^{-3}s. Even though the residual is tiny, it will cause small sampling point shifting every second. If we don’t correct the time shifting, the residue from each second will be cumulated and the sampling data point could be lost:

Figure 2.6 Sampling Time Shift Caused by Residue Problem
If there is no residue and the grid frequency is a perfect 60 Hz, the first sampling point of each second will start at the same position in reference to the beginning of each second. However, as shown in Fig. 2.6, the residue problem will cause the first sampling point of each second to move forward. After a certain period of time, the residue accumulated will drive the sampling point, which should be the first sampling point after PPS, to fall into the previous second, as shown as in the bottom diagram of Fig. 2.6. Since it is restricted by the algorithm, there will be only 1440 sampling points per second used for frequency estimation. Therefore the extra sampling point will be discarded. In terms of calculation, this is equal to moving the whole sampling sequence one point ahead from the discarded point. Hence there will be a relatively huge “jump” (discontinuity) in the sampling voltage waveform used for frequency estimation and the phasor angle used for frequency estimation. According to Eq. (2.12), the frequency that is estimated based on the derivative of the phasor angle will encounter “spikes” too. Here the phasor angle is a relative angle that is the actual phasor angle less the reference phasor angle. The “spikes” in the estimated frequency are caused by losing sampling voltage points instead of a real frequency change in system. Consequently, the “spikes” introduce inaccuracy to the frequency estimation algorithm.

In summary, hardware limitations cause the residue problem in the sampling clock. The residue problem, which has not been carefully considered by the frequency estimation algorithm, will introduce inaccuracy. The rest of this section will discuss the residue impact in terms of quantity.

As shown in Eq. (2.3), the phasor is described as:
where $k$ is the $k$th sampling point.

If one sampling point is discarded by the residue problem, Eq. (2.24) becomes:

\[
X_c^{(k+1)} = X_c^{(k)} + \frac{2}{N} (x_{k+1} - x_{k+1-N}) \cos \left( \frac{2\pi}{N} k \right)
\]

(2.25)

\[
X_s^{(k+1)} = X_s^{(k)} - \frac{2}{N} (x_{k+1} - x_{k+1-N}) \sin \left( \frac{2\pi}{N} k \right)
\]

(2.26)

Compared with Eq. (2.24),

\[
\Delta X_c^{(k+1)} = X_c^{(k+1)} - X_c^{(k)} = \frac{2}{N} (x_{k+2} - x_{k+1}) \cos \left( \frac{2\pi}{N} k \right)
\]

\[
\Delta X_s^{(k+1)} = X_s^{(k+1)} - X_s^{(k)} = \frac{2}{N} (x_{k+2} - x_{k+1}) \sin \left( \frac{2\pi}{N} k \right)
\]

The phasor angle error is:

\[
\Delta \phi (k)_{err} = \tan^{-1} \left( \frac{X_s^k}{X_c^k} \right) - \tan^{-1} \left( \frac{X_s^k}{X_c^k} \right)
\]

(2.27)

Then the frequency error is:

\[
\Delta f_{err} = \frac{\Delta \sigma_{err}}{2\pi} \approx \frac{d\Delta \phi (k)_{err}}{dt} = \frac{d\Delta \phi (k)_{err}}{dk} \frac{dk}{dt} = Nf_0 \frac{d\Delta \phi (k)_{err}}{dk}
\]

\[
d \left[ \tan^{-1} \left( \frac{X_s^k}{X_c^k} \right) - \tan^{-1} \left( \frac{X_s^k}{X_c^k} \right) \right]
\]

(2.28)

\[
= Nf_0 \left[ \frac{d \left( \frac{X_s^k}{X_c^k} \right)}{dk} \frac{X_c^k}{1 + \left( \frac{X_s^k}{X_c^k} \right)^2} - \frac{d \left( \frac{X_s^k}{X_c^k} \right)}{dk} \frac{X_c^k}{1 + \left( \frac{X_s^k}{X_c^k} \right)^2} \right]
\]

Since \( \frac{X_s^k}{X_c^k} \approx \frac{X_s^k}{X_c^k} \) and \( X_c^k \approx X_s^k \), Eq. (2.28) becomes:
$\Delta f_{\text{err}} \approx N f_0 \left[ \frac{1}{X_s^k} \frac{d(-X_s^k + X_s^k)}{d k} + \frac{d}{d k} \left( \frac{2}{N} (x_{k+1} - x_{x}) \sin \left( \frac{2 \pi}{N} k \right) \right) \right]$

$\approx N f_0 \left[ -X_c^k \frac{d}{d k} \left( \frac{2}{N} (x_{k+1} - x_{x}) \cos \left( \frac{2 \pi}{N} k \right) \right) \right]$

$= 4 \pi f_0 \left[ -X_c^k \frac{d}{d k} \left( \frac{2}{N} (x_{k+1} - x_{x}) \cos \left( \frac{2 \pi}{N} k \right) \right) \right]$

(2.29)

Considering the sampling rate is much higher than the nominal frequency, we assume \( \frac{d(x_{k+1} - x_{x})}{d k} \approx 0 \). From Eq. (2.29), we know that the estimated frequency error depends on \( X_c^k, X_s^k, \cos \left( \frac{2 \pi}{N} k \right) \) and \( (x_{k+1} - x_{x}) \). In other words, the frequency estimation error could be positive or negative. The phenomenon of this residue problem with unexpected spikes on the estimated frequency is shown in Fig. 2.7. The input signal for the test is 60 Hz generated by function generator. The estimated frequency is a straight line of 60 Hz with huge “spikes” periodically. One should noticed that the RDFT algorithm described in Section 2.2 uses the least square error to estimate the final frequency, and the re-sampling technique is used to refine the final result. The final frequency estimation error will be slightly different from the result of Eq. (2.30), which is based on the first derivative of the phasor angle from the definition.
Figure 2.7 Testing Result of Losing Sampling Point

Figure 2.8 Angle Discontinuity Caused by Residue Problem
The angle discontinuity can also be seen in Fig. 2.8, which is also the consequence of losing sampling points due to the residue problem. The proposed solution to this residue problem is described in the next section.

2.3.2 Variable Computation Windows

One of the approaches to solving the residue problem is to increase the system’s oscillator frequency. A higher system frequency leads to a smaller time interval between the consecutive system clock pulses. Since the residue is less than 1440 (the denominator of Eq. (2.22)) multiplied by the system clock intervals, a high system frequency will reduce the impact of the residue. However, the hardware change is not easily done for the MPC555 evaluation board. Another approach is to correct the residue each second. Therefore even though the residue still exists, it will be compensated each second and will not be cumulated.

To synchronize the sampling clock with the PPS and correct the residue problem each second, we define the fixed waiting time $t_w$ as the fixed time the sampling clock will wait to start after the edge of the PPS:

$$
t_w = n_w \times t_{sys}
$$

$$
t_{sys} = \frac{1}{f_{sys}}
$$

(2.30)

where $n_w$ is the number of the system clock to count for the waiting, and $t_{sys}$ is the time interval of system clock pulse.

$n_w$ is a fix number for all the FDR units, and we assume $t_{sys}$ is the same across all the units (this depends on individual hardware, but we neglect the slight difference on the
system oscillator since it is very high frequency). The waiting time $t_w$ should be small to make the sampling clock start every second as close as possible to the stable edge of PPS.

As shown in Fig. 2.9, the two units A and B may have a different residue at the end of every second (caused by the slight difference in the system clock; the difference is actually not as significant as shown here). However the residue problem will be corrected at the beginning of each new second by waiting for a fixed time after the PPS rising edge. Therefore the hardware differences are restricted within each second and will not be cumulated. In the meantime, all the units’ sampling clocks are synchronized by the PPS.

Although waiting for a fixed time after the PPS will correct the residue problem every second, it causes another problem: the sampling interval between the last pulse of the previous second and the first pulse of the current second is different from other sampling intervals. The problem is illustrated in Fig. 2.10.
As we can see from Fig. 2.10, the sampling intervals of \( t_0 \) and \( t_1 \) are different. From Eq. (2.7), we know that the algorithm requires the computation window across several fundamental frequency cycles. For the static frequency waveform, the longer the computational window, the better result of the least square error estimation of the frequency. However the computation window size cannot be infinite due the dynamic of the frequency. Considering the power system frequency characteristics, and compromising for the sake of the frequency dynamic requirement, we are using eight fundamental frequency cycles for frequency estimation [25]. This computation window will provide very good accuracy in frequency estimation when it crosses even sampling intervals, but it cause bring glitches when it crosses uneven intervals. Since the FDR outputs ten frequency estimations per second, the first frequency estimation output is 0.1 second, which is 6 cycles, after PPS. Eight cycles as a computation window will definitely cross the uneven sampling intervals of \( t_0 \) and \( t_1 \).

As shown in Fig. 2.11, if the problematic sampling interval is shorter than the normal interval, the estimated frequency is lower than the true frequency, because measuring these sample intervals has the same effect as keeping the sampling interval unchanged.
and lowering the input signal frequency; in contrast, if the problematic sampling interval is longer than the normal interval, the estimated frequency is higher than the true frequency.

\[
\text{If } T_i < T_o \text{ then } f_{\text{out}} < f_{\text{true}} \\
\text{If } T_i > T_o \text{ then } f_{\text{out}} > f_{\text{true}}
\]

Figure 2.11 Impact of Different Sampling Interval

The testing shows the frequency dips every second, which is caused by the uneven
sampling interval crossing the start of each second. One way to solve this problem is to reduce the computation window from eight cycles to six cycles. Then the computation window for the first frequency estimation of each second will not cross the uneven interval. Even though the even sampling intervals still exist, the output frequency (ten points per second for the FDR) will appear to not be affected by the uneven sampling intervals. But a smaller computation window means less accuracy for all the frequency estimation. After carefully considering the algorithm in Section 2.2 and the problem described in Section 2.3.1, a variable computation window method is proposed. In general this method will use two computation windows; one is six cycles for the first frequency estimation of each second, which guarantees the computation window will not cross the uneven sampling intervals, and the other is eight cycles for the rest of the frequency estimations. This method preserves the accuracy offered by the longer computation windows while solving the uneven sampling interval problem by changing the computation window. Several modifications of the original methods need to be done.

First, we will generate two separate gain matrixes similar to Eq. (2.7):

\[
\begin{bmatrix}
\phi(1) \\
\phi(2) \\
\vdots \\
\phi(M)
\end{bmatrix}
= \begin{bmatrix} 1 & 1 & 1^2 \\
1 & 2 & 2^2 \\
\vdots & \vdots & \vdots \\
1 & M & M^2 \end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
a_2
\end{bmatrix}
\text{ or } \Phi = Xa 
\]  
\text{(2.31)}

\[
\begin{bmatrix}
\phi(1) \\
\phi(2) \\
\vdots \\
\phi(M')
\end{bmatrix}
= \begin{bmatrix} 1 & 1 & 1^2 \\
1 & 2 & 2^2 \\
\vdots & \vdots & \vdots \\
1 & M' & M'^2 \end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
a_2
\end{bmatrix}
\text{ or } \Phi = X'a
\]  
\text{(2.32)}

\(M\) is the window size for the normal computation window and \(M'\) is the window size for the reduced computation window. Therefore the two gain matrixes will be:
\[ \mathbf{G} = [\mathbf{X}^T \mathbf{X}]^{-1} \mathbf{X}^T \quad \text{and} \quad \mathbf{G}' = [\mathbf{X}'^T \mathbf{X}']^{-1} \mathbf{X}'^T \]

(2.33)
Figure 2.13 shows the detailed flowchart of the variable computation window implementation: after the sampling clock is generated, the phasor angle will be calculated. If the routine detects it is the first frequency estimation of the second, the reduced computation window with the corresponding gain matrix will be used. Otherwise, the normal computation window will be used. The testing result is shown in the next section.

2.4 Testing and conclusion

To test the variable computation window algorithm, we are using two hardware-identical FDRs. One FDR (Unit 4) is loaded with the firmware based on the old algorithm with a fixed computation window; the other (Unit 10) is loaded with the firmware based on a fixed waiting time and the variable window size algorithm. Both of these units are connected to the same power strip, which guarantees that both of them have the same input.

Figure 2.14 Testing Environment Setup
As shown in Fig. 2.14, both FDR units send data back to the Information Management Server (IMS) with a synchronized GPS time tag. The data collected at IMS are compared below:

(a) Overall comparisons of voltage angle

Figure 2.15 Voltage Angle Comparisons with Fix and Variable Computation Window

(b) Detailed comparison of power angle
Figure 2.15 and Fig. 2.16 show the voltage angle and frequency comparisons between

Figure 2.16 Frequency Comparisons with Fixed and Variable Computation Window

Figure 2.15 and Fig. 2.16 show the voltage angle and frequency comparisons between
the old method and the new method. The voltage angle plot from Unit 4 is showing zigzags in the waveform, which is caused by the residue problem. Meanwhile, the voltage angle from Unit 10, which is the unit with the variable computation window size method, shows a smooth change of the actual voltage angle. By comparing this output with the estimated frequency, one can also easily find many abrupt spikes from the Unit 4 output, which are induced by the residue problem instead of the true value of power grid.

In summary, this chapter discusses the RDFT algorithm used by FDRs to estimate the power grid frequency, and the inaccuracy caused by the imperfect sampling clock due to hardware restrictions, like the residue problem in clock division. To solve this problem, we generate a new sampling clock pulse train every second, thereby correcting the residue problem every second; however this introduces another problem – an uneven sampling interval. To maintain a wider computation window size, which means relatively high accuracy for frequency estimation, and to eliminate the uneven sampling interval problem, the variable computation window size method is proposed. Even though the one phase Recursive Discrete Fourier Transfer (RDFT) algorithm is proven to be very accurate for theoretical power system frequency measurement, due to the insufficient consideration of hardware limitations like the sampling clock flaw, the FDFT didn't get the expected accuracy. After the modification in this chapter, which developed the variable calculation window method, for the first time an accurate synchronized wide-area frequency measurement from a power system distribution network has been obtained.
Chapter 3 Time Stamp Calibration and Automatic GIS

Info Report Design

3.1 Introduction

As introduced in Chapter 1, Wide-area Measurements (WAMs) are using the Global Positioning System (GPS) to synchronize measurements. GPS was developed by the United States Department of Defense, utilizing a constellation of at least 24 Medium Earth Orbit satellites that transmit precise microwave signals. The system enables a GPS receiver to determine its location, speed, direction, and time [28-30]. At least four satellites are needed for a GPS receiver to acquire a very accurate local time and $x$, $y$ and $z$ position coordinate variables. After the position information is determined, only one satellite is needed to maintain accurate local time. According to [4], the synchronization accuracy achieved by GPS can be 1μs or better, which is equal to 0.021° in relation to a 60 Hz signal. To combine the measurements from a PMU with FNET for further phasor applications, measurements from both devices should be aligned. However, the comparison of frequency waveforms during the system dynamic transient from a commercial PMU (transmission level) and an FDR (distribution level) reveals that there is a several seconds’ built-in time difference. The reason for this problem is analyzed in following section and a solution is developed.

Due to the easy deployment of FDRs, it is difficult to track an FDR’s location since it may be moved from one place to another frequently. PMUs are usually tied with substations and will not be moved as frequently. Therefore combining GPS position information in the data stream from FDR to Data Concentrator will be good design
solution to track the deployment location of FDRs.

### 3.2 FDR time stamp calibration

After numerous comparisons between the FDR and the PMU frequency waveforms, it is clear that there is a 10-second difference between these two kinds of device. The waveform from the FDR is ten seconds ahead of the PMU waveform:

![Figure 3.1 10-Second Difference between FDR and PMU](image)

Further study of the FDR reveals that the timing buffer causes the time difference problem. Figure 3.2 shows the flowchart of how the GPS time stamp is tagged with measured frequency data.

From the chart, one can see that the received GPS messages are first stored in the 9-cell GPS message buffer. Then the extracted time stamps from the GPS message buffer are stored in a two-cell time stamp buffer. The message and time stamp buffer are considered as a pipeline: the time stamp at the bottom of the pipeline is the current time,
while the time stamp at the top of the pipeline is from ten seconds ago. After the frequency is estimated in main module, the time stamp will be applied and combined with frequency data sent back to the Data Concentrator. Since the most updated frequency estimation is tagged with a time stamp from ten seconds ago, the FDR output frequency waveform is ten seconds ahead of the PMU data. The buffer also causes uncertainty in the time difference. For example, sometimes the PPS is received by the GPS, while the GPS message followed by that PPS may not be completed due to noise on the communication channel, although this situation rarely happens. Then the buffer may not be filled with messages due to the loss of the GPS message. Therefore the time difference could be changed to nine seconds instead of ten seconds in a normal situation.

Figure 3.2 Flowchart of Tagging Time stamp with Measured Data by FDR

To calibrate the FDR time stamp and eliminate the time difference, it would be straightforward to get the time stamp directly from the GPS receiver instead of the buffer. Figure 3.3 shows the new procedure combining time stamp with the estimated frequency.
As shown in Fig. 3.3, the PPS from the GPS receiver is sent to the main module generating the sampling clock. The generated sampling clock will trigger the A/D module, which converts the analog voltage signal to a 16-bit digital signal, and the frequency estimation module, which implements the algorithm depicted in Chapter 2 to estimate the frequency. The communication module requests and receives the updated time stamp back from the GPS receiver module and tags it with the estimated frequency. To be compatible with the old FDR deployed at the field, which has the ten-second problem, the GPS receiver module can artificially add an offset to the time stamp. The offset is fixed and will not fluctuate like the problem caused by buffer. On the Data Concentrator side, the software can easily compensate the time stamp offset by the software, both the artificial time offset and the ten-second problem caused by the buffer. The function source code of adding an artificial time offset in the GPS receiver module is listed in Appendix A.

Figure. 3.3 Flowchart of Tagging Time Stamp without Buffer by FDR

Besides the ten seconds of time difference, the further detailed comparison of the
FDR and the PMU shows there is another tiny sub-second difference, as shown in Fig. 3.4, observed during lab testing [31]. This is caused by the different frequency dynamic response of various frequency estimation algorithms and hardware implementations. After the modification of the timing system design of FDRs, the frequency measurement data from PMUs and FDRs can now be aligned for further applications.

![Figure 3.4 Sub-second Time Difference between FDR and PMUs [31]](image)

### 3.3 Automatic GIS Information Report Design

Unlike the PMU, which is relatively stationary as it is deployed at a substation, FDRs are easy to deploy at any residential or business area and may be moved from one place to another frequently. Therefore it is necessary to track the current location of FDRs. Since GPS not only provides accurate time information but also provides the accurate latitude and longitude, a new function design, which extracts latitude and longitude information from the GPS receiver module and inserts the Geographic Information
System (GIS) information into the data stream, will enable FDRs to automatically track the current deployment location. This will be useful for many location-related applications, like real-time system monitoring, system disturbance location estimation, etc. However there is no such function described in synchronized phasor measurement standards [5, 32] for commercial PMUs, and it is the first time for a synchronized phasor measurement network of distribution level to have automatic GIS information tracking functionality.

To reduce the data transmitted, the GIS information will only be reported every minute instead of every sub-second. The original GPS message format is shown below:

<table>
<thead>
<tr>
<th>Date</th>
<th>Longitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>mm</td>
<td>month</td>
</tr>
<tr>
<td>dd</td>
<td>day</td>
</tr>
<tr>
<td>yy</td>
<td>year</td>
</tr>
<tr>
<td></td>
<td>ddd</td>
</tr>
<tr>
<td></td>
<td>mm.mmm</td>
</tr>
<tr>
<td></td>
<td>w</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UTC Time</th>
<th>Height</th>
</tr>
</thead>
<tbody>
<tr>
<td>hh</td>
<td>hours</td>
</tr>
<tr>
<td>mm</td>
<td>minutes</td>
</tr>
<tr>
<td>ss</td>
<td>second</td>
</tr>
<tr>
<td></td>
<td>s</td>
</tr>
<tr>
<td></td>
<td>hhhh.h</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Latitude</th>
<th>Velocity</th>
</tr>
</thead>
<tbody>
<tr>
<td>dd</td>
<td>degree</td>
</tr>
<tr>
<td>mm.mmm</td>
<td>minutes</td>
</tr>
<tr>
<td>n</td>
<td>direction</td>
</tr>
<tr>
<td></td>
<td>sss.s</td>
</tr>
<tr>
<td></td>
<td>Hhh.h</td>
</tr>
</tbody>
</table>

As we can see from Table 3.1, both latitude and longitude need three parameters: degree, minutes and direction. Instead of transmitting the three parameters separately, we combine all of them in one floating number, following these steps:

1. Convert the minutes to degrees:

\[ 0.mm = \frac{mm.mmm}{60.0} \]  \hspace{1cm} (3.1)
2. Add the minutes to the degrees:

\[
\text{Latitude : } dd\,nnnn = dd + 0.0000 \\
\text{Longitude : } ddd\,nnnn = ddd + 0.0000
\]  

(3.2)

3. Add signs indicating the direction:

\[
\text{Latitude : } (+/-)dd\,nnnn \quad + : \text{North Hemisphere} \quad - : \text{South Hemisphere} \\
\text{Longitude : } (+/-)ddd\,nnnn \quad + : \text{East Hemisphere} \quad - : \text{West Hemisphere}
\]  

(3.3)

4. The sequence of data transmission:

Latitude, Longitude, Number of satellite tracked

The flowchart depicting how the GIS information is encapsulated to the data stream is shown below:

![Flowchart of Encapsulate the GIS Information into Data Stream](image)

Figure 3.5 Flowchart of Encapsulate the GIS Information into Data Stream
Chapter 3

The firmware code of this new functionality is listed in Appendix A. Figure 3.6 shows an example of the FDR automatically reporting the GIS information of the deployment location. This unit is deployed at Muscle Shoals, TN. The GIS information (Latitude: 34.7740 and Longitude -87.6510) and number of satellites tracked are automatically reported to Data Concentrator every first second of minutes.

![Figure 3.6 Data Stream with GIS Information](image)

3.4 Conclusion

This chapter first discusses the alignment problem between FDRs and PMUs, in which it appears that FDRs’ frequency waveforms are ten seconds ahead of the PMUs.
The problem is caused by the design of both the GPS message buffer and the time stamp buffer in the FDR firmware. To compensate the time stamp problem, the author proposes a buffer-free design for the timing system in FDRs. An artificial time offset is added as an option to make the new FDRs compatible with the old ones. Software time stamp compensation is available at the Data Concentrator. The sub-second time difference problem caused by different frequency dynamic responses is also discussed.

As FDRs can be easily deployed and relocated, a new design, which will enable FDRs to automatically report GIS information, is developed. This design is innovative when compared with the current WAMs functions, and it is the first to provide distribution-level synchronized phasor measurement network ability for automatically tracking measurement locations.
Chapter 4  Power System Dynamic Disturbance Detection Based on FNET

4.1 Power System Dynamics and Operation States

Power systems are complex interconnection systems connecting generation, transmission and distribution in a large area. Because of the interconnection, variable dynamic interactions are possible, some of which will affect a fraction of the whole system, while others may affect the whole system’s operation. Being aware of the system dynamics and studying the system response towards the dynamic disturbance is critical for the power system operator and researcher.

Changing power demand and various disturbances are the two major sources of system dynamics. Since the system dynamic response time is critical, usually the dynamics phenomena are categorized into several groups [23, 33, 34]: wave, electromagnetic, electromechanical and thermodynamic. Figure 4.1 shows the time frames accompanied by different system dynamics [23].

![Figure 4.1 Time Frame of the Power System Dynamic Phenomena [23]](image)

Much research has been done in the past decades studying the system dynamics. In order to assess system security and design an appropriate control strategy, one approach is classifying system operations states into several states, and developing an appropriate control strategy for each state.

As mentioned in [24, 35-38], five states are usually used for power system operation states: normal, alert, emergency, in extremis and restorative. Figure 4.2 describes the
transitions between these states.

![Diagram of Power System Operation States]

FNET is providing observability for up to tenth of a second, which means it is capable of capturing electromagnetic phenomena such as a sudden drop in frequency, or frequency decline propagation etc. It is of great interest that FNET can provide close to real-time system state monitoring and alarm if the system goes into emergency state or alert state. Early detection of system instability will give operators time to take appropriate control action.

### 4.2 Frequency Deviation and Power System Disturbance

A basic fact of power system operation is that there is no major energy storage device in power systems [23]. Therefore the power generated should generally be equal to the power consumed. Many studies show that the composite load pattern is predictable even though individual loads appear to be random. Therefore, the power system operation planners are able to predict the load pattern of the next day considering various factors such as weather and human activities. Following the load pattern prediction, the power system operators will make the daily operation schedule a day ahead and adjust the generation in real time to meet the requirements of the load. However, if there is unscheduled sudden power change in the power generated or power consumed caused by generator trip or load trip, or if the transmission network topology changed due to an
unexpected line trip, the balance can be broken temporarily or for a long time. The system may get into alert state or emergency state and critical system variables, like system frequency or voltage, will deviate from their normal value. As a synchronized wide-area frequency-monitoring network, frequency deviation during power dynamics is of major concern of FNET.

From swing equation [24], the relationship between the power imbalance and generator angle is illustrated as the following:

\[
\frac{2H}{\omega_0} \frac{d^2 \delta}{dt^2} = \bar{T}_m - \bar{T}_e - \frac{K_D}{\omega_0} \frac{d\delta}{dt}
\]  

(4.1)

where

\[
\bar{T}_m \quad \text{is the per unit mechanical torque}
\]
\[
\bar{T}_e \quad \text{is the per unit electrical torque}
\]
\[
\omega_0 \quad \text{is the rated electrical angular velocity}
\]
\[
K_D \quad \text{is the damping factor}
\]
\[
H \quad \text{is the inertia constant}
\]
\[
\delta \quad \text{is the angular position of the rotor in electrical radials with respect to the synchronously rotating reference.}
\]

Since \( \frac{d^2 \delta}{dt^2} = \frac{d\omega_r}{dt} = \omega_0 \frac{d\tilde{\omega_r}}{dt} \), from Eq. (4.1):

\[
2H \frac{d\tilde{\omega_r}}{dt} = 2H \frac{dj}{dt} = \bar{P}_m - \bar{P}_e - K_D \tilde{f}
\]  

(4.2)

where

\[
\omega_r \quad \text{is the rotation angular velocity,}
\]
\[
\bar{P}_m \quad \text{is the per unit mechanical power, and}
\]
\[
\bar{P}_e \quad \text{is the per unit mechanical power.}
\]

Neglecting the damping factor, we find the relationship between the system frequencies changes the power imbalance from Eq. (4.2). Instead of using the instantaneous derivative of the frequency, \( \Delta f \), which is the frequency deviation over a short period of time, to estimate the power imbalance we use [39]:

50
where \( \Delta f = f_{\text{ax2}} - f_{\text{ax1}} \)

\[ \Delta P = P_m - P_e = \beta \cdot \Delta f \]  

\[ \sum_{i=1}^{N} P_{mi} = \sum_{i=1}^{N} P_{ei} \]  

\( \beta \) is the load-frequency sensitivity coefficient. Its value depends on the system inertia and load conditions. For U.S. eastern interconnection, the load-frequency sensitivity ranges from -2,600 MW to -4,600 MW/0.1 Hz [40]. The negative sign indicates the inverse power loss relationship with the frequency deviation. \( \Delta P \) is the total system power imbalance. \( P_m \) and \( P_e \) are the sums of the mechanical power and the electrical power of the system generator, respectively. Hence,

\( P_m = \sum_{i=1}^{N} P_{mi} \)

\( P_e = \sum_{i=1}^{N} P_{ei} \)  

where \( P_{mi} \) and \( P_{ei} \) are the individual generator’s mechanical power input and electrical power output. For each generator, the electrical power output can be calculated using the admittance matrix:

\[ P_{ei} = \text{Re}[V_i^* I_i] = \text{Re}[V_i^* \sum_j Y_{ij} V_j] = -\sum_j |V_i| |V_j| b_{ij} \sin(\delta_i(t) - \delta_j(t)), \quad (i = 1, \ldots, n) \]  

Therefore, when there is a generation trip, load rejection, or line trip in the power system, an imbalance between the generation and the load will cause the system frequency to suddenly change. The rate of system frequency change can be used as an indicator of a disturbance [39]. An estimate of the amount of tripped generation or load rejection in fault analysis is based on the relationship between frequency change and active power imbalance in the system, according to Eq. (4.3). It is known that, in power systems, the real-time frequency and the rate of frequency change \( df/dt \) of the whole system are the most important parameters in calculation or estimation of the imbalance between load and generation.
4.3 Multiple Units-based Frequency Deviation Detection (MUFDD) Algorithm

From [39-41], the \( df/dt \) is the proportion of imbalance between the system mechanical power input and the electrical power output, neglecting the damping factor. Therefore it can be used as a disturbance indicator, because sudden disturbances cause sharp changes in power system frequency. Scheduled generator maintenance or load-shedding usually does not cause such sudden changes in frequency. After the disturbance is found, \( \Delta f \) can be used to estimate the total amount of generation tripping or load rejection according to Eq. (3.3). The \( \beta \) term is usually an empirical coefficient that can be obtained from known cases. Some values of \( \beta \) for EUS and WECC can be found in [40].

Now, with the help of the wide-area measurement network FNET, the frequency and the rate of frequency change can be easily measured and provided for this application. However, the following concerns should be noted during the frequency deviation detection algorithm development: FDRs are installed at the distribution level, which is vulnerable to local noise such as a small location load change; a poor GPS signal may prevent the GPS receiver from receiving the time synchronization signal continuously; and Internet congestion may stop the data streaming from some FDRs. All these factors may induce frequency data discontinuity for an individual FDR unit.

Since FDR units are deployed at the distribution level, local noise, the majority of which is caused by rapid and random local load changes, will generate spikes in the measured frequency data. However, local noise is random and unique to each unit. Only system-level disturbances will be detected by all or most of the units in the same interconnection. Therefore multiple units’ measurement in the same interconnection will be considered for the disturbance detection algorithm, instead of using just a single unit’s data. Another advantage of using multiple units over a single unit is that a single unit will be offline sometimes because of loss of GPS signal or network congestion, etc., while the chance for multiple units going offline is small, and the disturbance detection algorithm will not work during those periods of time if it is completely dependant on any single unit. Currently the FNET project has deployed 32 units in the Eastern Interconnected System (EUS), six units in Western Electricity Coordinating Council (WECC) and two units in Electric Reliability Council of Texas (ERCOT). Therefore the
multiple-unit-based disturbance detection algorithm is possible for each interconnection.

Below are the steps of the Multiple Units based Frequency Deviation Detection (MUFDD) algorithm:

1) Check the available FDR units: if the data loss rate within the calculation windows is higher than 10%, discard the unit; otherwise perform the linear data interpolation before further processing and synchronize all the units’ data by the time tag.

For example, a chunk of the frequency data sequence of the unit \(i\) is:

\[
\{f_{i,j}, f_{i,j+1}, \ldots, f_{i,j+k}, \ldots, f_{i,m}\}
\]

\(m\) is the total frequency points in this chunk. The data between \(f_{i,j}\) and \(f_{i,k}\) are lost. Hence the linear interpolation is as following:

\[
f_{i,j+1} = f_{i,j} + \frac{l(f_{i,j} - f_{i,k})}{k-j}, \quad (l \in [1, k-j-1])
\]  

(4.6)

2) De-noise the measured data from each FDR unit for the each interconnection using a moving, 10-point average.

3) Calculate the rate of system frequency change in continuous manner \(df(t)/dt\):

\[
\frac{df(t)}{dt} = \frac{1}{m(t)} \sum_{i=1}^{n(t)} \frac{df_i(t)}{dt} = \frac{1}{m(t)} \sum_{i=1}^{n(t)} \frac{m(t)}{kT} \sum_{l=1}^{m(t)} f_{i,l} - f_{i,l+kT}
\]  

where \(1 \leq n_i \leq n\), \(n\) is the total number of the FDRs in the interconnection, and \(m(t) \leq n\). \(m(t)\) is the number of available FDRs at the time \(t\). In discrete form, Eq. (4.7) will be converted to:

\[
\frac{\Delta f(t)}{kT} = \frac{1}{m(t)} \sum_{i=1}^{n(t)} \frac{m(t)}{kT} \sum_{l=1}^{m(t)} f_{i,l} - f_{i,l+kT}
\]  

where \(T\) is the time interval between the two adjacent data points, and here this is 0.1 second. \(k\) is the window size, calculating the frequency change rate. As shown in Fig. 4.3, during a frequency drop disturbance, there are usually three characteristic points [40]: Point A, pre-disturbance frequency, typically close to 60 Hz. Point C is the maximum excursion point, while point B is the settling frequency point of the interconnection’s frequency. The transient between point A and point C typically ranges from two seconds to four seconds. In order to get the maximum frequency change rate without being trapped by the spikes of
noise, the calculation window should be close to this time frame. Therefore, \( k \) is selected as 40, which is equal to four seconds in the calculation window.

![Frequency Excursion during Frequency Drop Disturbance](image)

4) Check the rate of system frequency change \( \frac{df(t)}{dt} \) or \( \frac{\Delta f(t)}{kT} \) with a pre-set threshold. If \( |\frac{df(t)}{dt}| \) or \( |\frac{\Delta f(t)}{kT}| \) is smaller than the threshold, the system is in normal operation mode; otherwise report the disturbance time to a log file. This calculation window keeps moving until it comes to the end of the data chunks.

5) Get new chunks of data and repeat these steps from (1).

The thresholds for each interconnection are different since the inertias of different interconnections are various. The same amount of generation trip or load rejection will cause different rates of system frequency change. In other words, the load-frequency sensitivities \( \beta \) of each interconnection are different. After calculating several months’ data for EUS, we found that the rate of system frequency change is normally within \( \pm 0.002 \) Hz/sec and seldom reaches \( \pm 0.004 \) Hz/sec in normal operation conditions. We set 0.005 Hz/sec as the threshold for the EUS. It works well and most of the events detected by the filter are obviously disturbances; only few of them are suspicious. Table 4.1 lists the thresholds for each interconnection:

<table>
<thead>
<tr>
<th>Interconnection</th>
<th>EUS</th>
<th>WECC</th>
<th>ERCOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threshold</td>
<td>0.005 Hz/Sec</td>
<td>0.006 Hz/Sec</td>
<td>0.008 Hz/Sec</td>
</tr>
</tbody>
</table>
Figure 4.4 shows the raw frequency data for Apr 22, 2005 from a single FDR unit deployed in the EUS. There are lots of spikes in the curve caused by local noise. It is therefore hard to tell if an event happened during the day from the raw frequency data alone. The derivative, or $df/dt$, of the frequency from this single unit is shown in Fig. 4.5(a), and the $df/dt$ based on multiple units is shown in Fig. 4.5(b). In Fig. 4.5(b) there appears to be a greater difference in the $df/dt$ value between the system in normal operation and the system during a disturbance than in the $df/dt$ value of Fig. 4.5(a). This difference is good for choosing a threshold for distinguishing between normal system operation and disturbance conditions. Using frequency data from multiple units in different places can eliminate the local random noise and provide a better view of system frequency dynamics. Additionally Fig. 4.5(a) and Fig. 4.5(b) are showing that the rate of system frequency change is usually within $\pm 0.002$ Hz/sec for EUS and seldom reaches $\pm 0.004$ Hz/sec in normal operation conditions.
Fig 4.5(b) shows a huge spike in the multiple unit based $df/dt$ calculation, which indicates that the $df/dt$ value exceeds the threshold of 0.005 Hz/sec. This event signature is typical of a system-level disturbance. The detail of the system frequency dynamics and corresponding multiple-unit based $df/dt$ are shown in Fig. 4.6. The frequency signature is clearly similar to a generation trip. The event time is 7:18:47 (EST) Apr 22, 2005 and has been confirmed as a generation trip of 1400 MW at W. H. Zimmer Nuclear Power Plant.
Similarly, we set 0.006 Hz/sec as the threshold for the WECC. The rate of system frequency change in the WECC under normal operation conditions is within ±0.003 Hz/sec and seldom reaches ±0.005 Hz/sec. Figure 4.7(a) shows the \( \frac{df}{dt} \) result of WECC frequency data for a day. Figure 4.7(b) and Fig. 4.7(c) show a load rejection-like and a generation trip-like disturbance. The disturbance times are 8:36:43 EST, Feb 14\(^{th}\), 2005 and 9:24:56 EST, Feb 14\(^{th}\), 2005 respectively.
4.4 System Frequency Dynamics during Disturbance

The MUFDD was first developed off-line. Testing with historical data shows it is very efficient. It has detected 15 disturbances in the EUS and 51 disturbances in the WECC from June to September 2005. The source code of this algorithm is listed in Appendix B. Table 4.2 indicates the numbers of disturbances detected for each interconnection:

<table>
<thead>
<tr>
<th>Interconnection</th>
<th>Generation Trip-Like</th>
<th>Load Rejection-Like</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>EUS</td>
<td>13</td>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td>WECC</td>
<td>46</td>
<td>5</td>
<td>51</td>
</tr>
</tbody>
</table>

From this table, one can observe that more disturbances have been detected in the WECC than in the EUS. This is partially because the system inertia of the WECC is much smaller than that of the EUS, and even a small generator trip or load rejection will cause a much more significant system frequency change than a similar event in the EUS. Most of the events are detected as generation trips.

From further studies on the disturbances captured by the event filter, some interesting
observations have been made:

(1) Comparing Fig. 4.8(a) and Fig. 4.8(b), disturbances in the WECC have sharper frequency change patterns than those in the EUS. Also the WECC system has a shorter recovery time than the EUS. In Fig. 4.8(a), it took about two minutes for the system frequency to reach the steady state after the disturbance and it only took about 40 seconds for WECC system in Fig. 4.8(b). It is generally true that the WECC has shorter system recovery times and sharper outlines in system frequency disturbance patterns than the EUS for all the disturbances captured by the MUFDD.
(2) In power systems, low-frequency oscillations take place as synchronous generators swing against each other. The frequency range of the oscillations is from 0.1 to 2.5 Hz. Much research has been done to suppress and eliminate these oscillations [42-48]. Figure 4.9 shows the local low-frequency oscillation captured by the unit located at Grand Rapids, Michigan, coming along with a system disturbance. The frequency of the oscillation before the disturbance was 0.49 Hz, and it was 0.42 Hz after the disturbance.

![Figure 4.9 Local Low Frequency Oscillations with Disturbance](image)

(3) In [49-51], the entire power system, including transmission lines, generators, and loads, was considered as a continuum for the study of electromechanical wave propagation in power systems. The frequency pattern in Fig. 4.10 shows some real pictures of the electromechanical wave propagation in the power system. This disturbance has been confirmed as happening close to New York City. Both FDR 18 and FDR 24, which are located in New York City and Toronto, respectively, are close to the disturbance location. The frequency plots of FDR 18 and FDR 24 have obvious swings that other FDRs do not. This disturbance case partially shows that the power system disturbance will affect the frequency differently at different locations depending on the distance to the disturbance source. These measurements can serve as a validation of ongoing studies in the area of electromagnetic wave propagation.
Figure 4.10 Electromechanical-wave Propagation in Disturbance

Figure 4.11 shows the trip of the DC transmission line that links south California and the Bonneville Power Administration (BPA) in Oregon. The FDR units recorded the frequency drop and oscillations. But the oscillations recorded by FDR16 (located in California) and FDR 14 (located in Arizona) compared with oscillations recorded by FDR 21 (located in Washington) have a 180-degree shift in phase from each other. These three FDR units are close to the two terminals of the DC transmission line.

Figure 4.11 DC Transmission Line Trip
(5) The case shown in Fig. 4.12 can help reconstruct the system scenario for the LA blackout on Sep 12, 2005. First it was a local event (a sharp dip captured by the unit located at LA). Then about five minutes later, there was a sharp frequency rise, which was caused by the tripping of overloaded transmission lines, followed by a sharp frequency drop. The sharp frequency drop was the result of the shutting down of several power plants.

![Figure 4.12 System Frequency in LA Blackout on Sep 12, 2005](image)

**4.5 On-line Event Database**

With the help of MUFDD, a MySQL database and a web server, FNET created an On-line Event Database. The FDRs’ data are first collected by the Data Receiver. Then the data will be plotted and displayed by Real-time Frequency Monitoring. In the meantime, the MUFDD module will check the frequency data to determine whether a disturbance occurred. All the raw frequency data will be stored by the Historical Storage Database.

If there is a disturbance detected, the piece of frequency data during system transient will be stored in the Event Data Storage Database and will be plotted through the web server. Also the event information, such as event type, event magnitude etc. will be extracted from the data and the information and be stored by Event Information Database.
All the event information will be tabled by the web server. Figure 4.13 shows the flowchart.

![Flow Chart of On-line Event Database](image)

**Figure 4.13 Flow Chart of On-line Event Database**

With the On-line Event Database, one can view the captured event information instantaneously and near real-time alerting can be easily implemented through a system such as an automatic email system. The timely detection of system disturbances and aggregated handy disturbance information provided by the On-line Event Database gives operators and researchers a convenient way to analyze system events, assess the system operation situation and guide system operation.
4.6 Conclusion

This chapter proposes a system disturbance detection algorithm based on multiple units from FNET measurements. Considering the fact that the frequency data is collected from a distribution network and numerous factors will affect the measurements (noise and data discontinuity problem, etc.), the developed Multiple-Units based Frequency Deviation Detection (MUFDD) algorithm provides a robust mechanism to extract system events from noisy distribution level measurements. Different threshold settings for this algorithm to cope with different interconnections are also studied. The testing result shows this algorithm is efficient, and numerous significant system disturbance dynamics are demonstrated. This is the first time the distribution-level synchronized wide-area measurement network is used to detect power system events.

Based on the MUFDD, the On-line Database is developed. With the help of this database, events can be detected, plotted and reported on-line in timely manner. This database can provide operators with guidance and help them to be aware of system operation states in near real-time. Furthermore, the stored event data can also be used for post-disturbance analysis and scenario reconstruction purposes.
Chapter 5 Event Location Estimation Based on Frequency Oscillation

5.1 Steady Frequency and Frequency Dynamics during Disturbance

Power system frequency remains the same across different parts of an interconnection system during steady state. But this is not the case during system dynamics caused by disturbance, such as a generator trip or load rejection, which induce an active power imbalance and causes the system to operate without a new equilibrium of active power. During this period of time, the system is dominated by the swing equation, which describes the generator rotor speed change and the active power imbalance. During the transient period, the system frequency and the change rate of system frequency are not the same throughout the system. Further study of the power system in continuous mode reveals that the frequency deviation propagates as electromechanical wave [49, 50, 52-56], or so-called “Frequency Wave”. Observations from the synchronized wide-area frequency measurement find that, during the transient period, the change rate of system frequency is related to the distance between the measurement point to the disturbance location, or disturbance resource. The distance here is defined as “electrical distance” which is generally the inverse proportion of the inductance between the “source” and measurement. To make it simple, we roughly assume the “electrical distance” is proportional to the radial geographic distance between the “source” and measurement. Using the frequency wave information to estimate the location of the disturbance source is of great interest in the FNET application.

5.2 Single Machine Infinite Bus (SMIB) and Frequency Deviation Dynamic

In [57], the Single Machine Infinite Bus (SMIB) system is used as the model to study the frequency deviation dynamic regarding distance. The study results are summarized here with several modifications.

As shown in Fig. 5.1, in the SMIB system generator is connecting an infinite bus
system through the tie line:

\[
E_g \angle \delta(t) \quad \text{SMIB} \quad \omega_0
\]

\[
E_\infty \angle 0
\]

Figure 5.1 Single Machine Infinite Bus System

It is important to notice that the infinite bus has a phase angle that is held stiff at zero degrees. Therefore the infinite bus frequency is also stiff at the rated frequency. Neglecting the damping factor, the frequency of the machine’s internal voltage angle will deviate according to the swing equation as follows:

\[
\frac{2H}{\omega_0} \frac{d^2 \delta(t)}{dt^2} = \bar{P}_m - \bar{P}_e
\]  

(5.1)

where \( H \) is the machine’s inertia constant

\( \omega_0 \) is the rated synchronous angular speed in rad/sec

\( \delta \) is the internal voltage angle in radians

\( \bar{P}_m \) is the per-unit mechanical power being injected into the rotor train

\( \bar{P}_e \) is the per-unit electrical power being drawn from the generators electrical terminal

and \( \bar{P}_e \) may be expressed as follows for a lossless system:

\[
\bar{P}_e = \frac{E_g E_\infty}{X} \sin \delta
\]  

(5.2)

where \( E_g \) is the per unit generator’s internal voltage magnitude

\( E_\infty \) is the per-unit infinite bus voltage magnitude, it is always 1.0

\( X \) is the per-unit reactance that connects the internal voltage to the infinite bus

\( \delta \) is the angle difference between the generator’s internal voltage and the infinite bus angle (set at zero)

Since \( \bar{P}_m \) is generally considered constant due to slow action on the generator
governor’s part, then Eq. (5.2) becomes:

$$\frac{2H}{\omega_0} \frac{d^2 \delta(t)}{dt^2} = \frac{P_m}{X} - \frac{E_g}{E_m} \sin \delta(t)$$

(5.3)

This equation, when rearranged and linearized about a quiescent operating point $\delta_0$, becomes:

$$\frac{d^2 \Delta \delta(t)}{dt^2} + \frac{\omega_0 P_{max} \cos \delta_0}{2H} \Delta \delta(t) = 0$$

(5.4)

where $P_{max}$ is the per unit maximum value of $P_e$, and $\Delta \delta(t)$ is the deviation in machine angle from the quiescent point.

For disturbances that result in an initial value of $\Delta \delta_0$, the solution to the equation is:

$$\Delta \delta(t) = \Delta \delta_0 \cos \omega_n t \quad \text{and} \quad \delta(t) = \delta_0 + \Delta \delta(t)$$

(5.5)

Regardless of initial conditions, the natural frequency $\omega_n$, may be expressed as

$$\omega_n = \sqrt{\frac{\omega_0 P_{max} \cos \delta_0}{2H}}$$

(5.6)

![Figure 5.2 Lossless Transmission Voltage Profiles with Different Load Condition](image)

We assume that the line that connects the generator terminal to the infinite bus has a
uniformly distributed inductance and negligible shunt admittance, and assume that the power flow injected into the line is equal to the power flow at all points on the line (lossless).

Here the sending end voltage is the generator terminal bus voltage $E_s$ and the receiving end voltage is the infinite bus voltage $E_\infty$. As the definition of infinite bus, the voltage $E_\infty$ should be constant. Figure 5.2 shows the voltage profile between the sending end and the receiving end under different load conditions [58]. Here we assume the system is running with a Surge Impedance Load (SIL), and then the voltage magnitude along the transmission line is always the same from the infinite bus to the generator terminal. Since it is a lossless transmission line, the active power at all points on the line is the same. Therefore,

$$\frac{E_s E_\infty}{(l-z)/l} \sin \delta_z(z,t) = \frac{E_s E_\infty}{X} \sin \delta(t).$$  \hspace{1cm} (5.7)

Here the $l$ is the total distance from the generator terminal to the infinite bus, and $z$ is the distance from generator terminal to the measurement point. Since the voltage magnitude is the same along the line and for small angle $\delta(t) \approx \sin \delta(t)$, Eq. (5.7) will be:

$$\frac{\delta_z(z,t)}{(l-z)/l} \approx \frac{\delta(t)}{X}.$$  \hspace{1cm} (5.8)

Therefore,

$$\delta_z(z,t) \approx \left(\frac{l-z}{l}\right) \delta(t).$$  \hspace{1cm} (5.9)

This shows the voltage angle along the line decreases in a linear fashion (with respect to the distance from the generator) from the generator’s internal voltage to the infinite bus. Following a disturbance, it can be seen that the angle deviation at any point on the line varies in a linear fashion along the line:

$$\Delta \delta_z(z,t) \approx \frac{(l-z)}{l} \Delta \delta(t) = \frac{(l-z)}{l} \Delta \delta_0 \cos(\omega_n t)$$  \hspace{1cm} (5.10)

Figure 5.3 illustrates relation of the phasor angle with the measurement distance:
Due to the relationship $\frac{d\delta(t)}{dt} = \Delta \omega$, the frequency deviation from the rated frequency will have maximum amplitude at the generator’s internal voltage and will linearly decrease along the line until it reaches zero at the infinite bus:

$$|\Delta \omega_\circ(z,t)| = \left| \frac{\partial \delta_\circ(z,t)}{\partial t} \right| = \left| \frac{l-z}{l} \Delta \delta_\circ \omega_\circ \sin(\omega_\circ t) \right|$$

and the maximum value is

$$|\Delta \omega_\circ(z,t)|_{z=0} = |\Delta \delta_\circ \omega_\circ \sin(\omega_\circ t)|.$$  \hspace{1cm} (5.12)

Notice that the oscillations in both angle and frequency have the same frequency, the system natural frequency; but the oscillation amplitude varies along the line. Expanding this notion from a single line connection to the infinite bus to have a mesh connection to the infinite bus, one could postulate that even though oscillation frequencies would be constant throughout the mesh, the magnitude of the oscillations must necessarily be, in general, different throughout the system – the closer the measurement to the generator that has the disturbance, the higher the amplitude of frequency oscillation; the closer the measurement to the infinite bus, the lower the amplitude of frequency oscillation.

### 5.3 Two-Area Model and Frequency Deviation Dynamic

Section 5.2 studied the frequency deviation with distance regarding the SMIB model. In [59], the analysis is extended with a two-area model, which is closer to real system operation conditions. The study result is summarized here with several modifications.
Chapter 5

As shown in Fig. 5.4, there are two generators connected through “tie line” $x_{12}$. In area one, $G_1$ is the generator, and the load is represented by $D_1 \omega_1$. $D_1$ is the load damping factor. To simplify the discussion, only frequency-related loads are considered here, while the non-frequency loads are ignored. Similarly, in area two, $G_2$ is the generator and the load is represented by $D_2 \omega_2$. A sudden active power change in Area One is represented by $\Delta P$. Area one is considered to be the close part to the active power change while Area Two considered as the remote part. The inter-area power is $P_{12}$ from area one to area two.

From the transfer function of swing equation,

$$\Delta \omega_j = \frac{\bar{T}_m - \bar{T}_e}{2HS} = \frac{\bar{P}_m - \bar{P}_e}{2HS} = \frac{\Delta \bar{P}_m - \Delta \bar{P}_e}{2HS}$$  \hspace{1cm} (5.13)

where $\bar{T}_m$ is the per-unit mechanical torque

$\bar{T}_e$ is the per-unit electrical torque.

$\bar{P}_m$ is the per-unit mechanical power

$\bar{P}_e$ is the per-unit electrical power

$H$ is the per-unit inertia constant (MW-Sec/MVA)

![Figure 5.5 Transfer Function Relating Speed and Power](image-url)

Figure 5.5 Transfer Function Relating Speed and Power
Therefore, in Area One and Area Two, we have:

\[
\Delta \omega_1 = \frac{\Delta \bar{P}_{m1} - \Delta \bar{P}_{e1}}{2H_1 S} \quad (5.14)
\]

\[
\Delta \omega_2 = \frac{\Delta \bar{P}_{m2} - \Delta \bar{P}_{e2}}{2H_2 S} \quad (5.15)
\]

\[
\Delta \bar{P}_{e1} = -\Delta \bar{P} + \Delta \bar{P}_{e11} + \Delta \bar{P}_{e12} = -\Delta \bar{P} + D_1 \Delta \bar{w}_1 + \Delta \bar{P}_{e12} \quad (5.16)
\]

\[
\Delta \bar{P}_{e2} = \Delta \bar{P}_{e21} - \Delta \bar{P}_{e22} = D_2 \Delta \bar{w}_2 - \Delta \bar{P}_{e22} \quad (5.17)
\]

\[
\bar{P}_{12} = \frac{E_1 E_2}{x_{12}} \sin(\delta_1 - \delta_2) = \frac{E_1 E_2}{x_{12}} \sin \delta_{12} \quad (5.18)
\]

\[
\Delta \bar{P}_{12} = \frac{\partial \bar{P}_{12}}{\delta_{12}} \Delta \delta_{12} = \frac{E_1 E_2}{x_{12}} \cos \delta_{12} \Delta \delta_{12} = \frac{E_1 E_2}{x_{12}} \cos \delta_{12} (\Delta \delta_1 - \Delta \delta_2) \quad (5.19)
\]

We assume \( E_1, E_2 \) are at their rating value 1.0, therefore

\[
\Delta \bar{P}_{12} = \frac{1}{x_{12}} \cos \delta_{12} (\Delta \delta_1 - \Delta \delta_2) = \frac{1}{x_{12}} (\Delta \delta_1 - \Delta \delta_2) \quad (5.20)
\]

where
\[
x_{12}' = \frac{x_{12}}{\cos \delta_{12}}
\]

Since \( \delta_1 \) and \( \delta_2 \) are in radians, we have:

\[
\Delta \delta_1 = \frac{\Delta \omega_1}{S} = \frac{\omega_0 \Delta \bar{w}_1}{S} = \frac{2\pi f_0}{S} \Delta \bar{w}_1 = \frac{377}{S} \Delta \bar{w}_1 \quad (5.21)
\]

\[
\Delta \delta_2 = \frac{377}{S} \Delta \bar{w}_2 \quad (5.22)
\]

Substituting Eq. (5.21) and 5.22 into Eq. (5.20), we have:

\[
\Delta \bar{P}_{12} = \frac{1}{x_{12}} \left( \frac{377}{S} \Delta \bar{w}_1 - \frac{377}{S} \Delta \bar{w}_2 \right) \quad (5.23)
\]

In the short term, the mechanical power can be considered as constant. Therefore

\[
\Delta \bar{P}_{m1} = 0 \quad \text{and} \quad \Delta \bar{P}_{m2} = 0. \quad (5.24)
\]

Combining Eq. (5.14) - (5.24), the system transfer function diagram is shown in Fig. 5.6.
Figure 5.6 Equivalent System Transfer Function Diagram of Two-area Model

After some simplification, the system transfer function diagram becomes:

Figure 5.7 Simplified System Transfer Function Diagram

Redrawing the diagram in Fig. 5.7 as a block diagram:
where,
\[ G_1(S) = \frac{1}{2H_1S + D_1} \quad (5.25) \]
\[ G_2(S) = \frac{1}{2H_2S + D_2} \quad (5.26) \]
\[ H(S) = \frac{377}{\bar{x}_{12}S} \quad (5.27) \]

From the block diagram of Fig. 5.8, one has:
\[ \Delta \bar{\omega}_1 = G_1(S)(\Delta \bar{P} - \Delta \bar{P}_{12}) = G_1(S)[\Delta \bar{P} - H(S)(\Delta \bar{\omega}_1 - \Delta \bar{\omega}_2)] \quad (5.28) \]
\[ \Delta \bar{\omega}_2 = G_2(S)\Delta \bar{P}_{12} = G_2(S)[H(S)(\Delta \bar{\omega}_1 - \Delta \bar{\omega}_2)] \quad (5.29) \]

or
\[ [1 + G_1(S)H(S)]\Delta \bar{\omega}_1 - G_1(S)H(S)\Delta \bar{\omega}_2 = G_1(S)\Delta \bar{P} \quad (5.30) \]

and
\[ [G_2(S)H(S)]\Delta \bar{\omega}_1 - [1 + G_2(S)H(S)]\Delta \bar{\omega}_2 = 0 \quad (5.31) \]

Therefore, solved from Eq. (5.30) and (5.31):
\[
\Delta \bar{\omega}_1 = \frac{G_1(S)[1 + G_2(S)H(S)] \Delta P}{1 + G_1(S)H(S) + G_2(S)H(S)}
\]
\[
\Delta \bar{\omega}_2 = \frac{G_1(S)G_2(S)H(S) \Delta P}{1 + G_1(S)H(S) + G_2(S)H(S)}
\]
\[
\frac{\Delta \bar{\omega}_1}{\Delta \bar{\omega}_2} = \frac{[1 + G_1(S)H(S)]}{G_2(S)H(S)} = 1 + \frac{1}{2H_2S + D_2 \frac{377}{\bar{x}_{12}S}} = 2H_2 \frac{\bar{x}_{12}}{377} S^2 + D_2 \frac{\bar{x}_{12}}{377} S + 1
\]

Eq. (5.34) shows the following results:

(1) If \( S \to \infty \), from initial value theorem,

\[
\frac{\Delta \bar{\omega}_1}{\Delta \bar{\omega}_2} \to \infty
\]

This means right after the disturbance,

\[
|\Delta \bar{\omega}_1| >> |\Delta \bar{\omega}_2|
\]

Therefore,

\[
\left| \frac{df_1}{dt} \right| >> \left| \frac{df_2}{dt} \right|
\]

Hence, following a disturbance, such as generator trip, the absolute value of the frequency deviation is much greater for close-by the disturbance than the remote site.

(2) If \( S \to 0 \), from final value theorem,

\[
\frac{\Delta \bar{\omega}_1}{\Delta \bar{\omega}_2} \to 1
\]

This means the two-area frequency will eventually become equal. Therefore the whole system frequency will finally be the same.

(3) \( \frac{\Delta \bar{\omega}_1}{\Delta \bar{\omega}_2} \) only depends on the tie line \( \bar{x}_{12} \), the load damping factor of remote site \( D_2 \) and the remote site generator inertia constant \( H_2 \).

This section has proved that the system frequency will have greater oscillation where it is closer to the disturbance resource from the two-area model perspective. Although the real power system topology will be much more complex than this two-area model, it is
true from historical observations that the closer the measurement is to a system disturbance, the bigger the frequency oscillation magnitude from the measurement data will be during system dynamics. Therefore the magnitude of the oscillations at different parts of system can be used to estimate the location of the disturbance resource.

### 5.4 Electromechanical Wave Propagation Based Disturbance Location Estimation

Reference [60] proposed another disturbance location estimation method based on the electromechanical propagation study of FNET. Generally the frequency perturbation caused by the disturbance propagates throughout the system. Therefore, theoretically, the closer the measurement point to the event location, the earlier it will detect the frequency perturbation. The frequency perturbation arrival time or the frequency propagation sequence is the critical piece of information used to estimate the event location.

Based on the relation between the distance and the time of wave travel, [60] gives the following equations:

\[
(x_i - x_h)^2 + (y_i - y_h)^2 = V^2 (t_i - t_h)^2
\]

For each responding FDR:

\[
(x_1 - x_h)^2 + (y_1 - y_h)^2 = V^2 (t_1 - t_h)^2
\]

\[
(x_2 - x_h)^2 + (y_2 - y_h)^2 = V^2 (t_2 - t_h)^2
\]

\[\vdots\]

\[
(x_n - x_h)^2 + (y_n - y_h)^2 = V^2 (t_n - t_h)^2
\]

where \((x_n, y_n)\) is the \((x, y)\) coordinates of \(n^{th}\) FDR

\((x_h, y_h)\) is the \((x, y)\) coordinates of hypocenter (origin of the disturbance)

\(t_n\) is the frequency perturbation arrival time measured by \(n^{th}\) FDR

\(t_h\) is the time when the disturbance happened

\(V\) is the mean value of the electromechanical wave propagation

By subtracting the consecutive pairs of equations, a set of linear equations is obtained:

\[
(x_{i+1} - x_i)x_h - (y_{i+1} - y_i)y_h - V^2 (t_{i+1} - t_i) = C_i
\]

where \(C_i\) is defined as:
Chapter 5

\[ C_i = \frac{1}{2} \left[ V^2(t_{i+1} - t_i) + x_{i+1}^2 - x_i - y_i \right] \]  

(5.42)

In matrix form, Eq. (5.41) can be written as:

\[ C = Hx \]  

(5.43)

where:

\[ C = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_n \end{bmatrix}, \quad x = \begin{bmatrix} x_h \\ y_h \\ t_h \end{bmatrix} \]

\[ H = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 & V^2(t_2 - t_1) \\ x_3 - x_2 & y_3 - y_2 & V^2(t_3 - t_2) \\ \vdots & \vdots & \vdots \\ x_n - x_{n-1} & y_n - y_{n-1} & V^2(t_n - t_{n-1}) \\ x_1 - x_n & y_1 - y_n & V^2(t_1 - t_n) \end{bmatrix} \]  

(5.44)

which yields the final least square solution as:

\[ x = (H^T H)^{-1} H^T C \]  

(5.45)

Theoretically this method is feasible for estimating the disturbance location based on the aforementioned assumption, such as when the velocity of the wave propagation is known. However there are several practical issues with this method:

1. There is no single fixed electromechanical wave propagation speed [49]. The speed may range from about 200 miles/sec to 660 miles/sec in the Eastern Interconnection (EUS) [61].

2. The electromechanical wave medium is not isotropic [49].

3. The measured system frequency curves may cross over and render the frequency perturbation arrival time or sequence difficult to determine.

Figure 5.9 shows a typical frequency waveform created by a generator trip in the EUS. The measured frequency curves show some crossover. Figure 5.10 compares the measured frequency waveform and the simulation result in PSS/E. The frequency perturbation arrival time or sequences are somewhat different in measurement and simulation[62]. Therefore, this method may not be accurate enough.
Figure 5.9 Typical Frequency Drop of Generator Trip in EUS [63]

Considering the complexity of a power system and all the practical issues above, it is very difficult to gain reasonable accuracy of disturbance location estimation solely based...
on the electromechanical wave (frequency wave) propagation arrival time. For the study in Sections 5.2 and 5.3, it is reasonable to use the frequency oscillation magnitude as one of the criteria for disturbance location estimation.

5.5 Using Frequency Oscillation to Improve Event Location Estimation

Historical data and PSS/E simulation shows the measurements from the sites that are closer to the event location usually have more intense oscillation swings in frequency compared with measurements from other sites. Sections 5.2 and 5.3 also proved that the frequency deviation magnitude is related to the distance between the measurement point and the disturbance location.

![Frequency curves for Bruce case](image)

(a) Frequency curves for Bruce case
Figure 5.11 Frequency Waveforms in Bruce Unit Tripping

As a typical example, Fig. 5.11(a) shows the frequency wave of the generator trip at Bruce Generating Unit, Ontario 12:56:19 (UTC), May 14, 2006. Toronto, Ontario is about 100 miles away from the Bruce Generating Unit. From the plot, the frequency measurement of FDR24 at Toronto shows a severe swing, while the frequency swing of FDR27 (RPI) at Albany, New York is less intense than FDR24, but much more severe than other locations.
Figure 5.12 shows the location of the Bruce plant, Toronto, RPI and VT. From the map, one can see that the frequency perturbation caused by the generator trip of the Bruce Unit travels through Toronto, RPI and VT. Therefore the “wave” will be attenuated along its path of propagation, which shows the different intensities of the frequency swings.

To use the swing intensity as the parameter to estimate the disturbance location, the Oscillation-based Disturbance Location Estimation (ODLE) algorithm is developed, and the steps are listed below:

1) Pre-conditioning the frequency data using a moving medium and moving mean.
2) Fitting the data after pre-conditioning to a high-order polynomial. Here we are only curve-fitting a chunk of data which includes the sudden frequency drop and one or two seconds of data before and after the sudden drop. This will obtain good results for the sudden frequency drop region.

Assuming the polynomial form is:

\[ y(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_2 x^2 + a_1 x + a_0 \]  \hspace{1cm} (5.46)

and there are \( k \) pairs of data \((x_i, y_i)\), where \( i \in (1, k) \), for each measurement unit (FDR) data, \( x_i \) is the time or the sequence and \( y_i \) is the corresponding frequency. \( a_n \) is the coefficient of the \( n^{th} \) power.

The matrix form is:

\[ Y = AX \]  \hspace{1cm} (5.47)

The least square method is used to solve the coefficient matrix \( A \):

\[ \bar{A} = (X^T X)^{-1} X^T Y \]  \hspace{1cm} (5.48)

The fitting curve is:

\[ \bar{Y} = \bar{A}X \]  \hspace{1cm} (5.49)

Since the degree of curve fitting \( n \) is much smaller than the constraint points \( k \), the fitting curve will be stable except at the boundary. The “bad fitting” of the extra data before and after the frequency sudden drop region will not be used for next step.

Figure 5.13 shows the curve fitting result of the data of the aforementioned Bruce case and Fig. 5.14 shows the absolute value of the difference between the fitting curve and the original curve. The “bad fitting” data only appears at the boundary and will be discarded.
Figure 5.13 Curve-fitting of Frequency Waveform

Figure 5.14 Difference between curve fitting and original data

3) Calculate the first order derivation of the curve fitting data:

\[ y'(x) = na_1x^{n-1} + (n-1)a_{n-1}x^{n-2} + \ldots + 2a_2x + a_1 \]  

(5.50)
which will have \( k \) pairs data set \((x_i, y'_i)\), where \( i \in (1, k') \) and \( k' < k \) since the “bad fitting” boundary is discarded. Figure 5.15 shows the first derivation of the fitting curves.

![Figure 5.15 First Order Deviation of Curve Fitting](image)

**Figure 5.15 First Order Deviation of Curve Fitting**

4) Calculate the maximum value of the moving standard deviation of the \( y'(x) \):

First calculate the moving average of \( y' \):

\[
MA_i = \frac{1}{N} \sum_{j=1}^{N} y'_{i+j}
\]

where \( i \in (1, k' - N) \). \( N \) is the window size.

Then calculate the moving standard deviation:

\[
MD_i = \sqrt{\frac{1}{N-1} \sum_{j=1}^{N} (y'_{i+j} - MA)^2}, i \in (1, k' - N)
\]

Find the maximum value of \( MD_i \):

\[
MaxD = \max \{ MD_i, i \in (1, k' - N) \}
\]

5) For \( k^{th} \) FDR, we have the maximum value of moving standard deviation \( MaxD_k \), where \( k \in (1, m) \) and \( m \) is the total number of FDRs. For each point on the map \( P(i, j) \), we calculate the weight \( W_k(i, j) \), which is related to the distance between point \((i, j)\) and the \( k^{th} \) FDR, estimated generator trip amount.
\[ \Delta P \text{ and } \text{Max}D_k. \]

\[ \Delta P = \beta \cdot \Delta f \]  \hspace{1cm} (5.54)

where \( \beta \) is the frequency response sensitivity [40, 64], and \( \Delta f \) is the frequency drop.

Then we add up the weights:

\[ W(i, j) = \sum_{k=1}^{m} W_k(i, j) \]  \hspace{1cm} (5.55)

The estimated event location is \( (i_e, j_e) \), where

\[ W(i_e, j_e) = \max_{i,j} \{W(i, j)\} \]  \hspace{1cm} (5.56)

The developed method uses frequency swing intensity as one of the key parameters for determining the distance between the event location and the FDR location. The weight function combines the consideration of all the FDRs. The point at which the weight is higher will have the higher possibility that the predicted event location coincides with the actual event location.

### 5.6 Weight Functions

There are many ways to construct the weight functions. The general rule is that a larger \( \text{Max}D_k \) leads to larger \( k^{th} \) FDR weight functions \( W_k(i, j) \). Here is one of the examples of the weight functions:

1) First we define a ring region of \( k^{th} \) FDR. The center of the ring is the location of the FDR. The weight functions \( W_k(i, j) \) are defined within the ring region, but always zero outside the ring region. The radius of the internal and external circle of the ring are defined by:

\[ \text{rad}_{\text{min}}(k) = 1.0 \ast C_1 \ast \text{Max}D_k / \Delta P \]  \hspace{1cm} (5.57)

\[ \text{rad}_{\text{max}}(k) = 1.5 \ast C_1 \ast \text{Max}D_k / \Delta P \]  \hspace{1cm} (5.58)

where \( C_1 \) is constant.

2) The weight constant for the \( k^{th} \) FDR:

\[ C_*(k) = C_2 \ast ((10^{8} \ast (\text{Max}D_k \ast C_3)) / \Delta P) \]  \hspace{1cm} (5.59)

where \( C_2, C_3 \) are constants.
3) The weight function of $k^{th}$ FDR:

If the distance $d_{ik}(i,j)$ between point $(i,j)$ and the $k^{th}$ FDR location $(i_k,j_k)$ is within the ring region defined by (5.57) and (5.58), then:

$$W_k(i,j) = C_n(k)*\left[C_4 + C_5*(C_6-r_{d_{max}}(k))/r_{d_{max}}(k)\right]*r_{d_{min}}(k)*\left(r_{d_{max}}(k)-r_{d_{min}}(k)/r_{d_{max}}(k)-r_{d_{min}}(k)\right)$$

otherwise the $W_k(i,j)$ will be zero. $C_4$, $C_5$ and $C_6$ are constants.

The historical cases are used to train the constant $C_j$. Equation (5.59) uses the exponential function of $MaxD_k$ to increase the effect of the frequency oscillation intensity on the weight function.

5.7 Testing and Case Analysis

Testing shows that the developed method improves the accuracy of the event location estimation when oscillations are present, especially for cases in which the measurement units (FDRs) are close to the actual event location. The source code of the Oscillation based Disturbance Location Estimation (ODLE) algorithm is listed in Appendix C.

For the aforementioned Bruce Unit case, Fig. 5.16 shows the distance between the center of the estimated region and the actual event location. The triangle is the actual event location. The rectangle is the estimated region by the new method.

Figure 5.16 Location Estimation for Bruce Case
Figure 5.17 and Fig. 5.18 show the frequency waveform and the improvement of the oscillation method for the generator trip at Vogtle Plant, 10:31:31(UTC), Aug 27, 2006. In this case, the FDR installed at University of Florida, which is in close proximity to the event location, captured the large swing of the frequency perturbation.

![Figure 5.17 Frequency Perturbation of Vogtle Plant Case](image1)

![Figure 5.18 Location Estimation for Vogtle Plant Case](image2)
Table 5.1 Disturbance Location Estimation Comparison

<table>
<thead>
<tr>
<th>Case</th>
<th>Plant</th>
<th>Date</th>
<th>Time (EST)</th>
<th>Amount (MW)</th>
<th>Lat</th>
<th>Long</th>
<th>Distance (TDOA) miles</th>
<th>Distance (ODLE) miles</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>WH Zimmer</td>
<td>2006-2-19</td>
<td>16:08:04</td>
<td>1183</td>
<td>39.178</td>
<td>-84.515</td>
<td>41</td>
<td>72</td>
</tr>
<tr>
<td>2</td>
<td>Hope Creek- Salem</td>
<td>2006-3-8</td>
<td>11:09:38</td>
<td>1019</td>
<td>39.481</td>
<td>-75.506</td>
<td>253</td>
<td>32</td>
</tr>
<tr>
<td>3</td>
<td>Nine Mile-Indep.- Fitz.</td>
<td>2006-3-9</td>
<td>22:14:07</td>
<td>967</td>
<td>43.431</td>
<td>-76.200</td>
<td>142</td>
<td>53</td>
</tr>
<tr>
<td>4</td>
<td>Sequoyah</td>
<td>2006-3-22</td>
<td>16:24:16</td>
<td>1312</td>
<td>35.270</td>
<td>-85.162</td>
<td>80</td>
<td>78</td>
</tr>
<tr>
<td>5</td>
<td>Wansley</td>
<td>2006-3-31</td>
<td>10:37:04</td>
<td>1721</td>
<td>33.435</td>
<td>-85.170</td>
<td>167</td>
<td>81</td>
</tr>
<tr>
<td>6</td>
<td>OCONEL</td>
<td>2006-4-12</td>
<td>12:35:58</td>
<td>978</td>
<td>33.737</td>
<td>-82.946</td>
<td>59</td>
<td>209</td>
</tr>
<tr>
<td>7</td>
<td>Gavin</td>
<td>2006-5-04</td>
<td>13:05:15</td>
<td>875</td>
<td>38.943</td>
<td>-82.136</td>
<td>73</td>
<td>70</td>
</tr>
<tr>
<td>8</td>
<td>Gavin</td>
<td>2006-5-05</td>
<td>12:04:46</td>
<td>1104</td>
<td>38.943</td>
<td>-82.136</td>
<td>73</td>
<td>74</td>
</tr>
<tr>
<td>9</td>
<td>Bruce</td>
<td>2006-5-14</td>
<td>07:56:28</td>
<td>1005</td>
<td>45.731</td>
<td>-81.642</td>
<td>814</td>
<td>81</td>
</tr>
<tr>
<td>10</td>
<td>Catawba</td>
<td>2006-5-20</td>
<td>13:01:33</td>
<td>1769</td>
<td>35.653</td>
<td>-81.234</td>
<td>77</td>
<td>105</td>
</tr>
<tr>
<td>11</td>
<td>Watts Bar</td>
<td>2006-5-30</td>
<td>16:00:08</td>
<td>1120</td>
<td>35.688</td>
<td>-84.828</td>
<td>110</td>
<td>68</td>
</tr>
<tr>
<td>12</td>
<td>Cumberland</td>
<td>2006-7-07</td>
<td>08:26:36</td>
<td>1190</td>
<td>36.374</td>
<td>-87.632</td>
<td>137</td>
<td>102</td>
</tr>
<tr>
<td>13</td>
<td>FERMI</td>
<td>2006-7-29</td>
<td>14:50:24</td>
<td>919</td>
<td>41.988</td>
<td>-83.298</td>
<td>62</td>
<td>43</td>
</tr>
<tr>
<td>14</td>
<td>Cumberland</td>
<td>2006-8-23</td>
<td>01:42:23</td>
<td>1183</td>
<td>36.374</td>
<td>-87.632</td>
<td>60</td>
<td>81</td>
</tr>
<tr>
<td>15</td>
<td>Vogtle</td>
<td>2006-8-27</td>
<td>05:31:41</td>
<td>1104</td>
<td>33.081</td>
<td>-81.994</td>
<td>176</td>
<td>23</td>
</tr>
<tr>
<td>18</td>
<td>Harris</td>
<td>2006-9-19</td>
<td>08:59:59</td>
<td>910</td>
<td>35.633</td>
<td>-78.973</td>
<td>422</td>
<td>53</td>
</tr>
<tr>
<td>19</td>
<td>Lacygne</td>
<td>2006-9-21</td>
<td>00:25:34</td>
<td>917</td>
<td>38.351</td>
<td>-94.742</td>
<td>316</td>
<td>236</td>
</tr>
<tr>
<td>20</td>
<td>Belews Creek</td>
<td>2006-10-01</td>
<td>22:17:09</td>
<td>1129</td>
<td>36.298</td>
<td>-80.144</td>
<td>180</td>
<td>215</td>
</tr>
</tbody>
</table>

Table 5.1 compares the result of the two methods used for disturbance location estimation. The first method is the Time Difference of Arrival (TDOA), which is based on the different arrival time of the frequency wave described in Section 5.4, while the second method is ODEL using the oscillation magnitude to estimate the disturbance location as described in Section 5.5. As shown in the table, the second method has better performance than the first method. The average error of the first method is 153.05 miles, while the average error is 93.6 miles for the second method. However we notice that for some cases, like Case 9, 15 and 18, the second method improved the estimation greatly.
because of the close proximity of one or two FDR units, while in some other cases, like Cases 6, 16 and 19, the second method is worse than the first method, or both are inaccurate. Further analysis of Cases 6, 16 and 19 shows that there are no FDR units close to the disturbance, or even worse, like Case 19 the disturbance is located on the boundary of the interconnection and all of the FDR units are on one side of this disturbance, as shown in Fig. 5.19.

![Figure 5.19 Case 19 – Lacygne and FDR Position](image)

### 5.8 Conclusion

This chapter provides another method of power system event location estimation based on the intensity of the frequency perturbation swing or oscillations. The new method is more robust, especially for the case which has a measurement close to the actual event location. Both observation of the FNET data and frequency oscillation analysis based on SMIB and the two-area model show that the frequency perturbation swing and oscillation are related to the distance between the measurement point and the
actual event location. The testing result in Section 5.7 shows the developed method has more accurate estimation result than the TDOA method. However, it is believed that the dense measurement is necessary for the developed method to get better performance, since only a close-by unit will detect significant frequency oscillation, which is crucial for this method. With low manufacturing cost and almost no installation cost for the FDR, which is a distinct advantage of the FDR over the traditional PMU, it is not difficult for FNET to create a dense measurement network.

The weight function used for this method could be refined with more historical event data and a more sophisticated model. Section 5.6 is just an example based on experiential data. Eventually this method could be implemented on-line and provide fast event location estimation information to power system operators.
Chapter 6  FNET and Phasor Application Based on SuperPDC

6.1 Introduction of SuperPDC

The power system wide-area measurement and control (WAMC) is an emerging new technology in power system operation. Traditional power systems rely on Supervisory Control and Data Acquisition (SCADA) as the major method to acquire the system status. However, without synchronized measurements and limited by the measurement resolution, SCADA can be aware of only a static and partial status of the power system, while the system’s dynamic situations during transients remain unknown. With the development of information technology, especially GPS and the Internet, the WAMC will provide the power system a feasible, synchronized, high-resolution and geographically distributed measurement solution. With the dynamic situation of power system presented by WAMC, more precise system stability analysis and control will be possible. The North American SynchroPhasor Initiative (NASPI) is a nation-wide project to propel the infrastructure construction and application of this new technology. The goal of NASPI will be embodied in two phases [65]: Phase I targets the use of expertise and equipment developed under auspices of the U.S. Department of Energy to deliver immediate value to project participants; Phase II seeks to deliver value within the operations environment using new inter-regional information and measurement systems. Both stages require a reliable and scalable platform to host synchronized measurements data from various devices.

To achieve these purposes, the Super Phasor Data Concentrator (SuperPDC), developed by the Tennessee Valley Authority (TVA) is designed for handling different protocols, storing a huge amount of measurement data, and providing fast queries of historical data and streaming data for real-time application. Currently, there are 41 PMUs in the East and 61 PMUs in the West, while 16 more PMUs are in progress [66], as illustrated in Fig. 6.1. As more PMUs are deployed in the system, the number of measurement points increase rapidly. A rough estimation of over 1 billion measurement data points in the eastern interconnection flow to SuperPDC daily [66] by different
protocols, such as IEEE C37.118, IEEE 1344, BPA PDCstream and Virginia Tech FNET [5, 22, 32]. This number is expected to double or triple in the following several years as more measuring devices are deployed on the field and brought online. Meanwhile, applications for phasor measurements are proposed both for real-time and non-real-time, such as stability analysis, situation awareness, state estimation, state calculation, and event analysis. Therefore it is very critical for the SuperPDC design to satisfy all the aforementioned requirements.

6.2 SuperPDC Structure and Integration with FNET

SuperPDC has several major components, as shown in Fig. 6.2: a real-time data acquisition module, an interface to DatAware assemblies, data pre-processing module and a real-time data broadcast module.
Figure 6.2 SuperPDC Structure\(^5\)

\(^5\) By courtesy of Tennessee Valley Authority (TVA)
The real-time data acquisition module interprets the data packets from various devices. It supports different protocols and is easy to expand for new protocols. To achieve this goal, the “Interface” technology in Visual Basic.NET is used. Interfaces, like classes, define a set of properties, methods, and events; but unlike classes, they do not provide implementations. They are implemented by classes, and defined as separate entities from classes. An interface is like a contract, once it is published, it cannot be changed.

As shown in Fig. 6.3, there is only one interface, “IDataCell,” for different protocols. The protocols IEEE 1344, C37.118, BPA PDCStream and Virginia Tech FNET all share this interface for data acquisition. When a data packet comes, the interface will load the corresponding protocol class to process and shield the protocol details from upper-level applications. The “IDataCell” interface has fields, properties and methods: fields define the internal parameters; properties define the method to retrieve the internal parameter; and methods define the action for processing. For example, the property “Analog Values” defines the format of voltage, current and frequency values from the measurement devices, and the method “GetObjectData” defines the action to retrieve the binary content of the data packets. Every protocol has its own class implementation of the interface “IDataCell”. For example, the class “BPA PDC Stream” implements all the fields, properties and methods defined by interface “IDataCell”. Under the definition of class “BPA PDC Stream”, the interface shows the fields, properties and methods with its own unique implementations (those with default definitions aren’t shown here) and those that are not defined in the interface. For example, the property “DataRate” is only defined by the class “BPA PDC Stream”. On the left of Fig. 6.3, the diagram illustrates the relationship between the interfaces “IDataCell” and “IChannelCell”; the serial of data packets, which is the data stream from the same device, composes the “Channel Cell” (IChannelCell), and several “Channel Cells” may compose the “Channel” if several devices share the same communication channel. The top right of Fig. 6.3 shows the relationship between the user-defined interfaces and the Visual Basic pre-defined interfaces, from which the user-defined interfaces are inherited.
Figure 6.3 Data Structure of Interface and Different Protocol

By courtesy of Tennessee Valley Authority
With this “interface” technique, the upper-level applications, like power system real-time monitoring and stability analysis, don’t need to know the details of each protocol for further procession. Instead, all they need to know is the structure of the interface. This gives the application developer flexibility without being bound by a particular protocol. This also makes SuperPDC flexible enough to support upcoming protocols.

To connect with a database, the DatAware archive assembly of the SuperPDC provides an interface to access to database. DatAware is a proprietary SQL Database developed by TVA, which provides an API for both the client side and the server side with high efficiency of data archive and query ability. To increase the system archive capability, the DatAware server assemblies for SuperPDC has three identical servers: P0, P1 and P2, as shown in Fig. 6.2. The loads are evenly distributed among these servers. As more measurement devices are added to SuperPDC, this structure can be easily expanded by adding more DatAware servers to these assemblies. When one DatAware server fails, the traffic towards this server can be redirected and picked up by the others. It also enhances the reliability of data archiving.

The real-time data broadcast ability is another merit of the SuperPDC. The extracted data-stream from a real-time data acquisition module will be pre-processed by a data pre-process module. This module provides two major functions: synchronization and encapsulation. Synchronization provides a unified data rate for real-time applications. Even though all phasor device measurements are tagged with time stamps, they may work at different rates: some may use 60 points/sec, while others may use 30 points/sec or 10 points/sec, as in the case of FNET. The data pre-processing module will synchronize the different rate data streams to a single data rate according to different real-time applications. For example, for system visualization 10 points/sec may be enough, but 30 points/sec is required for stability analysis. Therefore, for visualization, the high-resolution data stream will lower the data rate by averaging; while for stability analysis, extrapolation may be used to increase the data rate of low data rate resources. Encapsulation means combining synchronized measurement data from multiple measurement devices into a single binary data stream to reduce the size of the total data transmitted. With a fixed bandwidth of the communication channel, less traffic size means less transmission time. As shown in [67],

94
the communication delay in WAMs can be considered to be the combination of serial delay, between-packet delay, propagation delay and routing delay. With encapsulation, even though the size of individual packet size will increase, the total packet size transmitting the same amount of data will decrease because of the reduction of the total packet number and packet headers. Therefore the serial delay, which is equal to the packet size divided by the data rate, as well as the between-packet delay, will be reduced. This will provide good performance for real-time application as [12-14, 67-69] indicate that communication delay is a critical issue for real-time WAM applications.

### 6.3 Virtual Phasor Reference Angle Application Base on SuperPDC

Recommended by the Performance Requirement Task Team (PRTT) of the Eastern Interconnection Phasor Project (EIPP, the ancestor of NASPI), one of the applications with for the SuperPDC is the Virtual Phasor Reference Angle (VPRA). Several adjacent PMUs are used to determine the phase angle reference, which is transmitted to all other locations that calculate relative phase angles locally [70]. This is helpful for determining the angle relationship between buses for visualization. To address this problem, three adjacent PMUs are used here (the actual application may use different ones), Volunteer, Cordova and Lowndes, as illustrated in Fig. 6.4. The algorithm seems to be straightforward, as Eq. (6.1):

\[ \theta_{\text{ref}} = \frac{1}{3}(\theta_{\text{Cordova}} + \theta_{\text{Volunteer}} + \theta_{\text{Lowndes}}) \]  

(6.1)
However, the following issues should be considered: the PMUs’ measurements provide wrapped phase angles; PMUs may lose data points or a PMU may be off-line; PMU data may be corrupted due to problems with communication links; and so on. Therefore, special handling is required to calculate the average of several measured bus angles. The original concept is proposed by [71] and the algorithm to calculate the Virtual Phasor Reference Angle is developed here.

Illustrated in Fig. 6.5, we assume there are two PMUs in an unwrapped angle, shown as the black and blue curves. The red curve is the correct average angle, which is the virtual reference angle.

![Figure 6.5 Phasor Angle Unwrapping with Starting Point](image)

Illustrated in Fig. 6.6, if the unwrapping starts from point B, the process is similar.

![Figure 6.6 Unwrapping Start from Point B](image)
The original concept proposed by [71] has a problem with the starting point. If the calculation of reference angle starts from Point A in Fig. 6.5, it will give the correct average phase angle, which is the red curve in Fig. 6.5. But if the calculation of the reference angle starts from Point B in Fig. 6.5, the result will be the red curve in Fig. 6.6, which is different from the red curve in Fig. 6.5. The problem is that the algorithm in [71] is using the difference between the current angle value and the previous angle value to determine whether the current angle value needs to be unwrapped. For the starting point, there is no “previous” angle value to determine if it needs to be unwrapped. Therefore there are two different results shown in Fig. 6.5 and Fig. 6.6 considering the starting Points A and B, respectively. To solve this problem, some judgments should be made about the initial values. The developed algorithms with detailed procedures are listed below:

1) Initialization: Assume there are total \( m \) PMUs, and \( A[x] \ (x \in \{1, m\}) \) is the current phase angle of PMU \( x \). We use \( A[1] \) of PMU 1 as the reference and calculate the unwrapping offset for each PMU at the start point:

   'Initialize the Unwrapoffset
   \[
   \text{angleRef} = A[1]
   \]

   for \( x = 1 \) to \( m \)
   
   \[
   \text{angleDelta0} = \text{abs} (A[x]-\text{angleRef}) \quad // \text{abs is the absolute function}
   \]
   
   \[
   \text{angleDelta1} = \text{abs} (A[x]+360-\text{angleRef}) \quad // 360 \text{ is the degree}
   \]
   
   \[
   \text{angleDelta2} = \text{abs} (A[x]-360-\text{angleRef})
   \]

   If (\text{angleDelta0} < \text{angleDelta1}) And (\text{angleDelta0}<\text{angleDelta2}) Then
   
   \[
   \text{UnwrapOffset}[x] = 0
   \]

   Elseif (\text{angleDelta1} < \text{angleDelta2})
   
   \[
   \text{UnwrapOffset}[x] = 360
   \]

   Else
   
   \[
   \text{UnwrapOffset}[x] = -360
   \]

   EndIf

   Next For

As shown above, the pseudo code compares \( \text{abs}(A[x]-\text{angleRef}) \), \( \text{abs} \)
\((A[x]+360-angleRef)\) and \(\text{abs}(A[x]-360-angleRef)\): if \(\text{abs}(A[x]-angleRef)\) is the minimum, no unwrapping is needed, therefore Unwrapoffset\([x]\) is set to 0; if \(\text{abs}(A[x]+360-angleRef)\) is the minimum, positive unwrapping is needed therefore Unwrapoffset\([x]\) is set to 360; if \(\text{abs}(A[x]-360-angleRef)\) is the minimum, negative unwrapping is needed and Unwrapoffset\([x]\) is set to -360;

2) Phase unwrapping: compare the difference between the current point and the previous point with a large angle, said, 300 degrees. Depending on whether the wrapping is positive wrapping or negative wrapping, add 360 or -360 to the Unwrapoffset\([x]\] instead of the raw phase angle measurements. The original algorithm adds -360 or 360 to the raw phase angle measurement. Here is the description:

\[\Delta A = A_1 - A_0,\] where \(A_0\) and \(A_1\) are the previous value and the current value, respectively.

- if \(\Delta A > 300 \text{ degrees}\), then \(A_1' = A_1 - 360\) (negative wrap).
- if \(\Delta A < -300 \text{ degrees}\), then \(A_1' = A_1 + 360\) (positive wrap).

It is easy to get confused: should we replace the current value \(A_1\) with \(A_1'\)? Whether we replace the current \(A_1\) with \(A_1'\) or not will determine whether we induce error, as illustrated in Fig. 6.7.

![Figure 6.7 Unwrapping Problem in Previous Algorithm](image-url)
In Fig. 6.7, we could see: If we don’t replace $A1$ with $A’1$, which is an unwrapped angle, then the unwrapped angle of $A2$ will still be $A2$ instead of $A’2$ since the absolute angle difference between $A1$ and $A2$ is less than 300; if we replace $A1$ with $A’1$, then $A2$ will be unwrapped to $A’2$ correctly, but $A4$ will be unwrapped to $A’4$ instead of $A’’4$, which is supposed to be the unwrapped angle of $A4$. We could call this the partial unwrap problem that the $A4$ is not fully unwrapped. To avoid the partial unwrap problem, we use the $Unwrapoffset[x]$ to accumulate the unwrapping offset:

\[ \Delta A = A[x] - LastA[x], \text{ where } A[x] \text{ and } LastA[x] \text{ are the current value and the previous value of PMU } x, \text{ respectively.} \]

- if $\Delta A > 300$ degrees, then $Unwrapoffset[x] <= Unwrapoffset[x] - 360$ (positive wrapping)
- if $\Delta A < -300$ degrees, then $Unwrapoffset[x] <= Unwrapoffset[x] + 360$ (negative wrapping)

3) Phase reset: the $Unwrapoffset[x]$ need to be reset in order to avoid numerical overflow.

- if $Unwrapoffset[x] > 360 * N$, then $Unwrapoffset[x] = Unwrapoffset[x] - 360 * N$
- if $Unwrapoffset[x] < -360 * N$, then $Unwrapoffset[x] = Unwrapoffset[x] + 360 * N$

where $N$ is the number of PMUs used for calculating phase angles.

4) Phase averaging: the average of phases can then be calculated using the original phasor angle and unwrap offset:

\[ A_{avg} = \frac{1}{N} \sum (A[x] + Unwrapoffset[x]) \]

5) Phase modulus operation: convert the averaged phase angle to a value in range [-180, 180] degrees.

\[ A_{avg\_mod} = \text{mod}(A_{avg}, 360) \]

If $A_{avg\_mod} > 180$, then

\[ A_{avg\_mod} = A_{avg\_mod} - 360 \]
If there is no new PMU coming online and no PMU dropping offline, the whole procedure will go from Step (2) to Step (5) iteratively; if there a new PMU comes online or one of the current PMUs drop offline, we need to clear \(\text{UnwrapOffset}[x]\) and do Step (1) to Step (5), which includes the initialization.

The real-time calculation snapshot of Virtual Phasor Reference Angle with SuperPDC is shown below. The black line is the calculated reference angle while the other three are actual PMU angles. The source code of this algorithm is listed in Appendix D.

![Figure 6.8 Virtual Phasor Reference Angle with SuperPDC](image)

### 6.4 Proposed Future Applications based on the SuperPDC

The SuperPDC provides an ideal platform for various applications of synchronized phasor data. Several applications have been proposed for the next step of developments with the SuperPDC:

- a) Event Analysis: Post-event analysis of time-synchronized high-sample-rate data.
This application make it possible to reconstruct a high–resolution system dynamic scenario during disturbance

b) Stability Analysis: Oscillatory condition alarming through real-time processing of high-sample-rate data. With early detection, the operator can take further steps to damp out oscillation.

c) Situation Awareness: Display of time-synchronized, sub-second data to operators. This application will help the operator to quickly acquire the system status.

d) State Estimation/State Calculation: Use time-synchronized, sub-second data to improve solution quality and the speed of traditional state estimators; this can also enable phasors to be used directly to calculate the state of the bulk transmission grid.

Overall, these phasor applications provide a set of powerful tools for system monitoring and analysis that can be used to enhance system reliability.

6.5 Conclusion

This chapter first introduces the infrastructure of the SuperPDC, which is developed by TVA, providing a powerful platform to archive, manage and apply the phasor data for diverse applications. Expansibility, reliability and versatility are the major characteristics of the SuperPDC. Based on these features, FNET has been integrated with this platform without difficulty. The combination of FNET data and PMU data provides more observability both on the transmission level and the distribution level.

A new Virtual Phasor Reference Angle (VPRA) algorithm is developed. The real-time VPRA calculation developed is based on the SuperPDC platform. This is the first time there is a correct VPRA estimation from real phasor measurements, which provides critical angle reference for future phasor applications. Additional advanced future phasor applications are also discussed.
Chapter 7 Conclusion and Future Work

7.1 Conclusion

As the Wide Area Measurement (WAM) emerges in power system monitoring and operation, FNET provides a low-cost and easy-to-deploy solution for this technology. Unlike the traditional Phasor Measurement Unit (PMU) measuring the transmission line, FNET measures the frequency from distribution network and transmits back data through the Internet, which dramatically reduces the deployment cost that is a hurdle for the wide usage of WAM technology. In the FNET, the FDR is the sensor, which samples the voltage from a 120 V wall outlet and transfers the sampling data to voltage frequency synchronized with a GPS time tag. This research work focuses on two perspectives: FNET Design and The Situation Awareness Algorithm Develop based on this platform.

The first FNET design breakthrough of this research work is to improve the frequency estimation algorithm accuracy, which is very critical to this measurement network. As we discussed, the residue problem in the sampling clock, which is not considered by the algorithm, induces error to the frequency estimation. The analysis shows that the accumulation of the residue will create another problem: losing a sampling point. The consequence of losing sampling point is positive or negative spikes in the frequency estimation output. The author proposes a variable computation window method and double-gain matrixes implementation. This method preserves the advantage of using a longer computation window, but avoids the residue problem, which is a result of a lack of consideration in the original algorithm. The testing with a signal generator and real frequency measurement shows that the result of this method is reliable and accurate. This method makes it possible for the RDFT frequency estimation algorithm to achieve as high accuracy as is theoretically proven.

Another challenge in FNET design is to align FNET data with commercial PMUs. Time stamp calibration and sub-second time difference issues are addressed. A solution for these problems is proposed that makes it possible to combine the FNET and PMUs data for future phasor applications. Another innovative design developed by this research work is the Automatic GIS Information Report function. A detailed design flowchart is presented and data format is addressed. This is the first time the GIS information is
integrated into the WAMs data and provides an automatic tracking function for the WAMs devices.

The second part of this research work focuses on the Situation Awareness Algorithm and related applications development. Three major algorithms and applications are developed based on FNET and phasor measurements: the Multiple Units-based Frequency Deviation Detection (MUFDD), the Oscillation-based Disturbance Location Estimation (ODLE) and the Virtual Phasor Reference Angle (VPRA) calculation algorithm.

MUFDD provides an efficient and accurate way to detect system disturbances from distribution-level measurements of FNET. As the relationship between system frequency change and power imbalance is revealed by a swing equation, MUFDD utilizes multiple units to estimate the system frequency deviation. Then the frequency deviation information from each interconnection will be compared using different criteria based upon interconnections. Excursions, which indicate drastic system frequency change, will be detected as an indication of system disturbance. This algorithm is fast, efficient and accurate, and has overcome several disadvantages of distribution-level measurements, including noise and the relatively high rate of data discontinuity of individual measurement unit. Abundant system dynamic patterns during the disturbance are discovered by this algorithm. An On-line Event Database is also established to detect and manage system events timely and conveniently.

ODLE is the developed algorithm for estimating the system disturbance location based on oscillation information. Based on the electromechanical wave (frequency wave) propagation theory, the Time Difference of Arrival (TDOA) algorithm has been proposed by others’ research work to estimate the disturbance location. However, due to the fact that the frequency wave propagation speed is not isotropic and not constant, and the problem that it is hard to determine the arrival sequence, the TDOA may not provide disturbance estimation with good accuracy. Studied with the Single Machine Infinite Bus (SMIB) and the two-area model, it is clear that the frequency oscillation magnitude is related to the distance between the measurement and the disturbance location. ODLE is using this concept to build a relationship between the oscillation magnitude and distance through several numerical procedures. Testing results show that it markedly improves the
Chapter 7

estimation results, especially for cases with measurements close by the disturbance, which will capture big swings in the frequency waveform. With the low cost of FNET deployment, it is not difficult to build up a dense measurement network across the whole interconnection that will tremendously help the accuracy of event location estimation.

The VPRA calculation algorithm developed in this research work is a method of utilizing several adjacent PMUs’ measurements from the transmission line to establish the reference phasor angle, which is very useful for future phasor applications, like phasor angle visualization, dynamic power flow calculation, and system stability analysis. Under the guidance of North American Synchronize Phasor Initiative (NASPI), the real-time VPRA calculation application, based on this algorithm, has been developed with SuperPDC, which is a phasor data platform developed by TVA.

7.2 Contribution

This dissertation work provides several unique findings:

- Developed a variable computation window size design for an RDFT frequency estimation algorithm. RDFT is proven to be a very accurate frequency estimation algorithm. However, due to the insufficient consideration of hardware limitations, it didn’t get the expected accuracy in the field. After the developed design in this work, for the first time FNET provides an accurate synchronized wide-area frequency measurement from power system distribution network.

- Developed an Automatic GIS Information Report function. This is the first time WAMs devices have the ability to track device location automatically.

- Developed the Multiple-Units based Frequency Deviation Detection (MUFDD) algorithm. This algorithm provides an efficient and accurate method to detect power system disturbances from synchronized, distribution-level wide-area measurements.

- Developed the Oscillation based Disturbance Location Estimation (ODLE) algorithm. This algorithm is the first time to use the concept that the oscillation is related to the distance between the measurements and the disturbance source to estimate the event location from distribution-level synchronized frequency
measurements.

- Developed the Virtual Phasor Reference Angle (VPRA) calculation algorithm and demonstrated the application of the real-time VPRA calculation on the SuperPDC. This application, based on the developed algorithm, is the first time to provide the correct reference phasor angle, which is very critical for future phasor applications, from actual phasor measurements.

Based on these contributions, some manuscripts have been published or submitted as shown below:

-Paper:


5. Hengxu Zhang, Chunyi Wang, Ning Zhang, Jian Zuo, Dawei Fan, New Fuzzy Excitation Controller Design Based on Coordination of Voltage Regulation and Dynamic Stability, 2007 PES General Meeting

6. Tao Xia, Hengxu Zhang, Robert Gardner, Jason Bank, Jian Zuo, Yilu Liu Wide-area Frequency Based Event Location Estimation, 2007 PES General Meeting


8. Tao Xia, Hengxu Zhang, Robert Gardner, Jason Bank, Jian Zuo, Yilu Liu, Lisa Beard, Peter Hirsch, Guorui Zhang, Rick Dong, Wide-area Frequency Based Event
7.3 Future Work

Here we provide a list of suggested future work:

- Develop the disturbance classification based on the Multiple Units-based Frequency Deviation Detection.
- Design adaptive frequency deviation detection. Different users or customers may have various interests regarding system events and the system operation conditions change with time. Adaptive frequency deviation detection could provide more accurate event detection and more specific event alerting for different applications.
- Improve the on-line database design. Add more features to the database, like disturbance type and statistic data.
- Improve the weight function for oscillation-based disturbance location estimation.
- Combine FNET and PMU data for the application of disturbance monitoring and location estimation.
References


[27] Motorola Inc. MPC555 user manual.


[40] "North American Electric Reliability Council: Understand and calculate frequency response."
References


References


Appendix A. Time stamp function module of GPS receiver module

FNET_MGPS.cpp

(`/***********************************************************************
  * NAME: ParseGPSMessage
  * PURPOSE: Parse Eq GPS message to get the proper time
  * IN: buffer - character buffer
  *     times - time queue
  * OUT: None
  * *************************************************************************/
  void UTC_TIME::ParseGPSMessage(TBuffer<96>& buffer, TQueue< int, 2 >& times)
  {
    int  i=0,j=0;
    char ch, charbuf[96];
    char lat1[2],lat2[7];
    char lon1[3],lon2[7];

    // copy message (96 bytes total) to a string
    ch=' ';                       //Changed by Jzuo Feb 1,2005
    while (!buffer.isEmpty() && ch != '@') {
      buffer.dequeue(ch);
      charbuf[i++] = ch;
    }
    if(EqMessage(charbuf)) {
      tdata.tm_year=ParseMessage(11,charbuf);
      tdata.tm_mon=ParseMessage(5,charbuf)-1;
      tdata.tm_sec=ParseMessage(20,charbuf);
      tdata.tm_min=ParseMessage(17,charbuf);
      tdata.tm_hour=ParseMessage(14,charbuf);
      tdata.tm_mday=ParseMessage(8,charbuf);

      if (tdata.tm_sec>10)
        tdata.tm_sec = tdata.tm_sec-10;
      else {
        tdata.tm_sec = tdata.tm_sec+50;         // tdata.tm_sec+60-10;
        if (tdata.tm_min>=1)
          tdata.tm_min = tdata.tm_min-1;
        else {
          tdata.tm_min = tdata.tm_min+59;        // tdata.tm_min+60-1;
          if (tdata.tm_hour>=1)
            tdata.tm_hour = tdata.tm_hour-1;
          else
            tdata.tm_hour = tdata.tm_hour+23;      // tdata.tm_hour+24-1;
        }
      }
    }
sat= ParseMessage(81, charbuf);
times.enqueue(tdata.tm_hour*10000 + tdata.tm_min*100 + tdata.tm_sec);
est = false;
i = 0;
for (j=23;j<25;j++)
{
    lat1[i]=charbuf[j];
i++;
}
latitude = atof(lat1);
i = 0;
for (j=26;j<33;j++)
{
    lat2[i]=charbuf[j];
i++;
}
latitude = latitude + atof(lat2)/60;
if ((charbuf[34]=='s')|| (charbuf[34]=='S')) //South
    latitude = latitude * (-1.0);
i = 0;
for (j=36;j<39;j++)
{
    lon1[i]=charbuf[j];
i++;
}
longitude = atof(lon1);
i = 0;
for (j=40;j<47;j++)
{
    lon2[i]=charbuf[j];
i++;
}
longitude = longitude + atof(lon2)/60;
if ((charbuf[48]=='w') || (charbuf[48]=='W')) //West
    longitude = longitude * (-1.0);
Appendix B. Multiple-Unit based Frequency Deviation Detection (MUFDD)

Trigger.m

function varargout = TrigProg2(varargin)
    gui_Singleton = 1;
    gui_State = struct('gui_Name', mfilename, ...'
        'gui_Singleton', gui_Singleton, ...'
        'gui_OpeningFcn', @TrigProg2_OpeningFcn, ...'
        'gui_OutputFcn', @TrigProg2_OutputFcn, ...'
        'gui_LayoutFcn', [], ...'
        'gui_Callback', []);
    if nargin && ischar(varargin{1})
        gui_State.gui_Callback = str2func(varargin{1});
    end
    if nargout
        [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
    else
        gui_mainfcn(gui_State, varargin{:});
    end
% End initialization code - DO NOT EDIT

% --- Executes just before TrigProg2 is made visible.
function TrigProg2_OpeningFcn(hObject, eventdata, handles, varargin)
    % This function has no output args, see OutputFcn.
    % hObject    handle to figure
    % eventdata  reserved - to be defined in a future version of MATLAB
    % handles    structure with handles and user data (see GUIDATA)
    % varargin   unrecognized PropertyName/PropertyValue pairs from the
        % command line (see VARARGIN)

    % Choose default command line output for TrigProg2
    handles.output = hObject;
    % Update handles structure
    guidata(hObject, handles);
    % UIWAIT makes TrigProg2 wait for user response (see UIRESUME)
    % uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = TrigProg2_OutputFcn(hObject, eventdata, handles)
    % varargout    cell array for returning output args (see VARARGOUT);
    % hObject      handle to figure
    % eventdata    reserved - to be defined in a future version of MATLAB
    % handles      structure with handles and user data (see GUIDATA)

    % Get default command line output from handles structure
    varargout{1} = handles.output;

114
function year_Callback(hObject, eventdata, handles)
% hObject    handle to year (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of year as text
%        str2double(get(hObject,'String')) returns contents of year as a
double
year = str2double(get(hObject, 'String'));
if isnan(year)
    set(hObject, 'String', 0);
    errordlg('Input must be a number','Error');
end

% Save the new volume value
handles.metricdata.year = year;
guidata(hObject,handles)

% --- Executes during object creation, after setting all properties.
function year_CreateFcn(hObject, eventdata, handles)
% hObject    handle to year (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'))
end

function Month_Callback(hObject, eventdata, handles)
% hObject    handle to Month (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Month as text
%        str2double(get(hObject,'String')) returns contents of Month as a
double
month = str2double(get(hObject, 'String'));
if isnan(month)
    set(hObject, 'String', 0);
    errordlg('Input must be a number','Error');
end

% Save the new volume value
handles.metricdata.month = month;
guidata(hObject,handles)
function Day_Callback(hObject, eventdata, handles)
% hObject    handle to Day (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Day as text
%        str2double(get(hObject,'String')) returns contents of Day as a double
day = str2double(get(hObject, 'String'));
if isnan(day)
    set(hObject, 'String', 0);
    errordlg('Input must be a number','Error');
end

% Save the new volume value
handles.metricdata.day = day;
guidata(hObject,handles)

% --- Executes on button press in Calculate.
function Calculate_Callback(hObject, eventdata, handles)
% hObject    handle to Calculate (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
axes(handles.axes1);
cla;

%event_analyzer(handles.metricdata.year, handles.metricdata.month, handles.metricdata.day);
year = handles.metricdata.year;
month = handles.metricdata.month;
day = handles.metricdata.day;

year_c = num2str(year); month_c = num2str(month); day_c = num2str(day);
if length(month_c)==1,
    month_c = ['0',month_c];
end
if length(day_c)==1
    day_c = ['0',day_c];
end
datafile = ['D:\Ryan\fdr-',year_c,'-',month_c,'-',day_c,'-EST.mat'];
units = [2 3 4]; % 6 7 9 11 17 20];
unitsname = ['UMR   ';'ARI   ';'VT    ';'ABB   ';'MISS  ';'UFL   ';'CALV    ';'TULANE';'TVA   '];
start_time = [year month day 00 00 00];
end_point = 900000;
dfdt = zeros(1,end_point);
factor = zeros(1,end_point);
tmp_freq = zeros(1,6000);
max_index = 0;

for k = 1 : length(units)
Appendix B

warning off
eval(['load ',datapath,datafile,' FDR',num2str(units(k)),';']);
warning on
if exist(['FDR',num2str(units(k))],'var')
tmp_start_time = start_time;
tmp_end_time = tmp_start_time+[00 00 00 10 00];
eval(['FDRdata = FDR',num2str(units(k)),';']);
if length(FDRdata.freq)>0
    freq_outrangeindx = union(find(FDRdata.freq>60.5),...
        find(FDRdata.freq<59.5));
disp(['Frequency error: ',num2str(length(freq_outrangeindx))]);
for j=1:length(freq_outrangeindx)
    FDRdata.freq(freq_outrangeindx(j))= ...
FDRdata.freq(max(freq_outrangeindx(j)-1,1));
end
iteration_time = end_point/6000;
for j = 1 : iteration_time     %time
    index_beg =
        min(find(datenum(FDRdata.date)>datenum(tmp_start_time)));
    index_end =
        max(find(datenum(FDRdata.date)<=datenum(tmp_end_time)));
    index_dif = index_end - index_beg+1;
    if (index_dif > 5910)  % The data is good
        if (index_end>max_index)
            max_index = index_end;
        end
        tmp_freq(1:index_dif) = FDRdata.freq(index_beg:index_end);
        for l = 1 : (6000-index_dif)  % Interpolate
            tmp_freq = [tmp_freq(1:l*60),tmp_freq(l*60), ...
                tmp_freq(l*60+1:length(tmp_freq)-1)];
        end
        dfdt((j-1)*6000+1:(j-1)*6000+6000) =
        dfdt((j-1)*6000+1:(j-1)*6000+6000)+...
            tmp_freq(1:6000);
        factor((j-1)*6000+1:j*6000) =
        factor((j-1)*6000+1:j*6000)+1;
    end
    tmp_start_time = tmp_end_time;
    tmp_end_time = tmp_end_time+[00 00 00 10 00];
end
end
end
for k = 1 : end_point
    if (factor(k) == 0)
        dfdt(k) = dfdt(k)/(factor(k)+1);
    else
        dfdt(k) = dfdt(k)/factor(k);
    end
end
%average process,filter
windowSize = 40;
dfdt = filter(ones(1,windowSize)/windowSize,1,dfdt);  % running average

dfdt =
(dfdt(windowSize:end-windowSize)-dfdt(2*windowSize:end))/windowSize*100

dfdt_outrange =
union(find(dfdt(1:min(end_point,max_index)-2*windowSize)>10),
    find(dfdt(1:min(end_point,max_index)-2*windowSize)<-10))

for i = 1:length(dfdt_outrange)
    dfdt(dfdt_outrange(i)) = 0;
end

dfdt_event =
union(find(dfdt(1:min(end_point,max_index)-2*windowSize)>=0.5),
    find(dfdt(1:min(end_point,max_index)-2*windowSize)<=-0.5))

eventfilename = ['event_',year_c,'_',month_c,'_',day_c,'.txt'];

if length(dfdt_event)>0
    event.date(1,:) = FDRdata.date(dfdt_event(1,:));
    event.dfdt(1) = dfdt(dfdt_event(1));
    org_eventnumber = dfdt_event(1);
    j=1;
    for i=2:length(dfdt_event)
        if (dfdt_event(i)-org_eventnumber)>50
            j=j+1;
            event.date(j,:) = FDRdata.date(dfdt_event(i,:));
            event.dfdt(j) = dfdt(dfdt_event(i));
            org_eventnumber = dfdt_event(i);
        elseif abs(dfdt(dfdt_event(i)))>abs(event.dfdt(j))
            event.date(j,:) = FDRdata.date(dfdt_event(i,:));
            event.dfdt(j) = dfdt(dfdt_event(i));
            org_eventnumber = dfdt_event(i);
        end
    end
    for i = 1 : j
        fprintf(fid,'date:%d   %2d   %2d   %2d:%2d:%2.2f 
... 
dfdt:%6.6f
',event.date(i,1),event.date(i,2),
    event.date(i,3),event.date(i,4),event.date(i,5),event.date(i,6),event.dfdt(i));
    end
else
    fprintf(fid,'No event is detected today!');
end
fclose(fid);

plotflag =1;
if (plotflag == 1)
    figure;
    subplot(2,1,1);
    plot(dfdt(1:min(end_point,max_index)-2*windowSize));
xlabel('Time (0.1s)');
ylabel('delta Freq (mHz)')
figuretitle = ['Event Detector ', year_c,'/',month_c,'/',day_c,',' ',unitsname(3,:)];
title(figuretitle);
subplot(2,1,2);
plot(FDRdata.freq);
xlabel('Time (0.1s)');
ylabel('Raw Frequency (Hz)');
end

% --- Executes on button press in reset.
function reset_Callback(hObject, eventdata, handles)
 hObject    handle to reset (see GCBO)
 eventdata  reserved - to be defined in a future version of MATLAB
 handles    structure with handles and user data (see GUIDATA)
 initialize_gui(gcf, handles, true);

function initialize_gui(fig_handle, handles, isreset)
% If the metricdata field is present and the reset flag is false, it means
% we are we are just re-initializing a GUI by calling it from the cmd line
% while it is up. So, bail out as we dont want to reset the data.
if isfield(handles, 'metricdata') & ~isreset
 return;
end

handles.metricdata.year = 2005;
handles.metricdata.month = 2;
handles.metricdata.day = 1;
handles.metricdata.unitk = 3;
set(handles.year, 'String', handles.metricdata.year);
set(handles.month, 'String', handles.metricdata.month);
set(handles.day, 'String', handles.metricdata.day);
set(handles.unitk, 'String', handles.metricdata.unitk);
set(handles.unitgroup, 'SelectedObject', handles.english);

% Update handles structure
guidata(handles.figure1, handles);
Appendix C. Oscillation based Disturbance Location
Estimation (ODLE)

Oscillation.m

% Prompt the dialog for inputting data file and configuration file
prompt = {'Data file path', 'data file (e.g. 20060514-125617)', 'Configuration file', 'Map file', 'Trip amount (MW)'};
defans = {'D:\Ryan\Research\Dissertation\Slope\', 'USAMap.bmp', '1000'};
fields = {'f_path', 'd_file', 'c_file', 'g_file', 'trip_amount'};  % File path, data file, config file, graphic file
info = inputdlg(prompt, 'Input the raw data file', 1, defans);
if ~isempty(info)
    info = cell2struct(info, fields);
f_path = info.f_path; % File path
d_file = info.d_file; % Data file
c_file = info.c_file; % Configuration file
g_file = info.g_file; % Graphic file
trip_amount = str2num(info.trip_amount); % Trip amount
u_info = readCfg(f_path, c_file);
if isempty(u_info)
    return;
end
u_info.eventtime = d_file;
i = 1;
while (i < u_info.num)
    d_file_tmp = strcat(d_file, '-', u_info.name(i), '.txt');
    u_data = readData(f_path, d_file_tmp);
    if isempty(u_data)
        if (i < u_info.num)
            for j = i:u_info.num - 1
                u_info.name(j) = u_info.name(j + 1);
                u_info.id(j) = u_info.id(j + 1);
                u_info.latitude(j) = u_info.latitude(j + 1);
                u_info.longitude(j) = u_info.longitude(j + 1);
                u_info.x(j) = u_info.x(j + 1);
                u_info.y(j) = u_info.y(j + 1);
                u_info.region(j) = u_info.region(j + 1);
            end
            i = i + 1;
        else
            u_info.data(i) = u_data;
        end
    end
end
PlotFreq(u_info, 'Raw Data');
median_windowsize = 5;
u_info_median = data_median(u_info, median_windowsize);
PlotFreq(u_info_median, [num2str(median_windowsize), ' Points Moving Median']);
avg_windowsize = 5;
u_info_avg = data_avg(u_info, avg_windowsize);

120
Appendix C

PlotFreq(u_info_avg,[num2str(avg_windowsize),' Points Average']);
start_point=1;
end_point=350-median_windowsize-avg_windowsize;
u_info_fit=gen_fit_curve(u_info_avg,start_point,end_point);
start_point=150;
end_point =250;
PlotFreq(u_info_fit,'Curve Fitting Freq');
PlotSlope(u_info_fit,'Curve Fitting Slope');
u_info_fit_part=gen_fit_curve(u_info_avg,start_point,end_point);
PlotFreq(u_info_fit_part,'Curve Fitting');
for i=1 : u_info.num
    u_info.data(i).freq=u_info_fit_part.data(i).diff;
    u_info.data(i).time=u_info_fit_part.data(i).time;
end
PlotFreq(u_info,'Absolute Difference between Curve Fitting and Original Data');
    u_info_fit_part=PlotSlope(u_info_fit_part,'Curve Fitting First Order Deviation');
PlotFFT(u_info_fit_part,'FFT Plot');
    Triangulation(u_info_fit_part,f_path,g_file,d_file,trip_amount);
End
Appendix C

=======================================================================

Triangulation.m

%Estimate the disturbance location based on weight functions
function Triangulation(u_info,f_path,g_file,d_file,trip_amount)
    format long
    img_file_rd=strcat(f_path,g_file);
    g_matrix=imread(img_file_rd,'bmp');
    [high,width,color]=size(g_matrix);
    %Calculate the radius and weight for each unit
    k1=5;                     %coefficient for radius
    k2=1000;                     %coefficient for weight
    std_tmp=u_info.std(1:u_info.num);
    max_std=max(std_tmp);
    if max_std>=0.05
        k3=70;
    elseif max_std>=0.04
        k3=90;
    elseif max_std>=0.03
        k3=115;
    else k3=150;
    end
    k3=200;
    for i=1:u_info.num
        radius_min(i)=1.0*k1/(std_tmp(i)*2500/trip_amount); %miles
        radius_max(i)=1.5*k1/(std_tmp(i)*2500/trip_amount); %miles
        weight(i)=k2*((10^(std_tmp(i)*k3))/trip_amount);
    end
    event_pos=zeros(high,width);
    ratio_pixel_miles= 2.7;  %One pixel interval equal to 2.7miles
    weight
    radius_min
    radius_max
    k3
    max_std
    for i=1:u_info.num
        for j=1:high
            for k=1:width
                x=u_info.x(i);
                y=u_info.y(i);
                dis=sqrt((k-x)^2+(j-y)^2)*ratio_pixel_miles;
                if (dis<=radius_max(i)) && (dis>=radius_min(i))
                    event_pos(j,k)=event_pos(j,k)+weight(i)*(10+10*(150-radius_min)/radius_min*(radius_max-dis)/(radius_max-radius_min));
                end
            end
        end
        max_weight=0;
        for j=1:high
            for k=1:width
                if event_pos(j,k)>max_weight
                    max_weight=event_pos(j,k);
                end
            end
        end
    end
end
avg_x=0;
avg_y=0;
max_weight_num=0;
for j=1:high
  for k=1:width
    if (event_pos(j,k)==max_weight)
      g_matrix(j,k,:) = [1 1 1];
      avg_x=avg_x+k;
      avg_y=avg_y+j;
      max_weight_num=max_weight_num+1;
    end
  end
end
avg_x=avg_x/floor(avg_x/max_weight_num)
avg_y=avg_y/floor(avg_y/max_weight_num)

dis=sqrt((avg_x-775)^2+(avg_y-87)^2)*ratio_pixel_miles;
if (dis<150)
  avg_x=avg_x-(775-692);
  avg_y=avg_y+(163-87);
end

contour(event_pos);
mesh(event_pos);
img_file_wr=strcat(f_path,d_file,g_file);
imwrite(g_matrix,img_file_wr,'bmp');
end
Appendix D. Virtual Phasor Reference Angle (VPRA) Calculation

ReferenceAngleCalculator.vb

Imports System.Text
Imports System.Math
Imports TVA.Common
Imports TVA.Math.Common
Imports TVA.Measurements
Imports TVA.Collections.Common
Imports TVA.IO
Imports TVA.IO.FilePath
Imports InterfaceAdapters

Public Class ReferenceAngleCalculator
    Inherits CalculatedMeasurementAdapterBase

    Private Const BackupQueueSize As Integer = 10
    Private m_phaseResetAngle As Double
    Private m_lastAngles As Dictionary(Of MeasurementKey, Double)
    Private m_unwrapOffsets As Dictionary(Of MeasurementKey, Double)
    Private m_latestCalculatedAngles As List(Of Double)
    Private m_measurements As IMeasurement()

    #If DEBUG Then
    Private m_frameLog As LogFile
    #End If

    Public Sub New()
        m_lastAngles = New Dictionary(Of MeasurementKey, Double)
        m_unwrapOffsets = New Dictionary(Of MeasurementKey, Double)
        m_latestCalculatedAngles = New List(Of Double)
        #If DEBUG Then
        m_frameLog = New LogFile(GetApplicationPath() & "ReferenceAngleLog.txt")
        #End If
    End Sub

    Public Overrides Sub Initialize(ByVal calculationName As String, ByVal configurationSection As String, ByVal outputMeasurements As IMeasurement(), ByVal inputMeasurementKeys As MeasurementKey(), ByVal minimumMeasurementsToUse As Integer, ByVal expectedMeasurementsPerSecond As Integer, ByVal lagTime As Double, ByVal leadTime As Double)
        MyBase.Initialize(calculationName, configurationSection, outputMeasurements, inputMeasurementKeys, minimumMeasurementsToUse, expectedMeasurementsPerSecond, lagTime, leadTime)
        MyClass.MinimumMeasurementsToUse = minimumMeasurementsToUse
    End Sub

    Public Overrides Property MinimumMeasurementsToUse() As Integer
        Get
            Return MyBase.MinimumMeasurementsToUse
        End Get
        Set(ByVal value As Integer)
        MyBase.MinimumMeasurementsToUse = value
    End Set
End Class
Appendix D

MyBase.MinimumMeasurementsToUse = value
m_phaseResetAngle = value * 360
End Set
End Property

Public Overrides ReadOnly Property Status() As String
Get
Const ValuesToShow As Integer = 4
With New StringBuilder
    .Append(MyBase.Status)
    .Append("  Last " & ValuesToShow & " calculated angles: ")
    SyncLock m_latestCalculatedAngles
        If m_latestCalculatedAngles.Count > ValuesToShow Then
            .Append(ListToString(m_latestCalculatedAngles.GetRange(m_latestCalculatedAngles.Count - ValuesToShow, ValuesToShow), ",","c))
        Else
            .Append("Not enough values calculated yet...")
        End If
    End SyncLock
    .Append(Environment.NewLine)
    Return .ToString()
End With
End Get
End Property

''' <summary>
''' Calculates the "virtual" Eastern Interconnect reference angle
''' </summary>
''' <param name="frame">Single frame of measurement data within a one second sample</param>
''' <param name="index">Index of frame within the one second sample</param>
''' <remarks>
''' The frame.Measurements property references a dictionary, keyed on each measurement's MeasurementKey, containing all available measurements as defined by the InputMeasurementKeys property that arrived within the specified LagTime. Note that this function will be called with a frequency specified by the ExpectedMeasurementsPerSecond property, so make sure all work to be done is executed as efficiently as possible.
''' </remarks>
Protected Overrides Sub PublishFrame(ByVal frame As IFrame, ByVal index As Integer)
Dim calculatedMeasurement As Measurement = Measurement.Clone(OutputMeasurements(0), frame.Ticks)
Dim angle, deltaAngle, angleTotal, angleAverage, lastAngle, unwrapOffset As Double
Dim key As MeasurementKey
Dim dataSetChanged As Boolean
Dim x As Integer
#If DEBUG Then
    LogFrameDetail(frame, index)
#End If

' Attempt to get minimum needed reporting set of composite angles used to calculate reference angle
If TryGetMinimumNeededMeasurements(frame, m_measurements) Then
    ' See if data set has changed since last run
    If m_lastAngles.Count > 0 AndAlso m_lastAngles.Count = m_measurements.Length Then

For x = 0 To MinimumMeasurementsToUse - 1
    If Not m_lastAngles.ContainsKey(m_measurements(x).Key) Then
        dataSetChanged = True
        Exit For
    End If
Next
Else
    dataSetChanged = True
End If

' Reinitialize all angle calculation data if data set has changed
If dataSetChanged Then
    Dim angleRef, angleDelta0, angleDelta1, angleDelta2 As Double

    ' Clear last angles and unwrap offsets
    m_lastAngles.Clear()
    m_unwrapOffsets.Clear()

    ' Calculate new unwrap offsets
    angleRef = m_measurements(0).AdjustedValue
    For x = 0 To MinimumMeasurementsToUse - 1
        angleDelta0 = Abs(m_measurements(x).AdjustedValue - angleRef)
        angleDelta1 = Abs(m_measurements(x).AdjustedValue + 360.0R - angleRef)
        angleDelta2 = Abs(m_measurements(x).AdjustedValue - 360.0R - angleRef)
        If angleDelta0 < angleDelta1 AndAlso angleDelta0 < angleDelta2 Then
            unwrapOffset = 0.0R
        ElseIf angleDelta1 < angleDelta2 Then
            unwrapOffset = 360.0R
        Else
            unwrapOffset = -360.0R
        End If
        m_unwrapOffsets(m_measurements(x).Key) = unwrapOffset
    Next
End If

' Total all phase angles, unwrapping angles if needed
For x = 0 To MinimumMeasurementsToUse - 1
    ' Get current angle value and key
    With m_measurements(x)
        angle = .AdjustedValue
        key = .Key
    End With

    ' Get the unwrap offset for this angle
    unwrapOffset = m_unwrapOffsets(key)

    ' Get angle value from last run (if there was a last run)
    If m_lastAngles.TryGetValue(key, lastAngle) Then
        ' Calculate angle difference from last run
        deltaAngle = angle - lastAngle

        ' Adjust angle unwrap offset, if needed
        If deltaAngle > 300 Then
            unwrapOffset -= 360
        ElseIf deltaAngle < -300 Then
            unwrapOffset += 360
        End If
    End If
Appendix D

' Reset angle unwrap offset, if needed
If unwrapOffset > m_phaseResetAngle Then
    unwrapOffset -= m_phaseResetAngle
ElseIf unwrapOffset < -m_phaseResetAngle Then
    unwrapOffset += m_phaseResetAngle
End If

' Track last angle unwrap offset
m_unwrapOffsets(key) = unwrapOffset
End If

' Total composite angles so average can be calculated
angleTotal += (angle + unwrapOffset)
Next

' We use modulus function to make sure angle is in range of 0 to 359
angleAverage = (angleTotal / MinimumMeasurementsToUse) Mod 360

' Track last angles for next run
m_lastAngles.Clear()
For x = 0 To MinimumMeasurementsToUse - 1
    m_lastAngles(m_measurements(x).Key) = m_measurements(x).AdjustedValue
Next
Else
' Use stack average when minimum set is below specified angle count
angleAverage = Average(m_latestCalculatedAngles) Mod 360

' We mark quality bad on measurement when we fall back to stack average
calculatedMeasurement.ValueQualityIsGood = False

#If DEBUG Then
    'RaiseCalculationException(
        LogFrameWarning("WARNING: Minimum set of PMU's not available for reference
        angle calculation - using rolling average")
    #End If
End If

' Slide angle value in range of -179 to +180
If angleAverage > 180 Then angleAverage -= 360
If angleAverage < -179 Then angleAverage += 360
calculatedMeasurement.Value = angleAverage

#If DEBUG Then
    LogFrameWarning("Calculated reference angle: " & angleAverage)
#End If

' Provide calculated measurement for external consumption
PublishNewCalculatedMeasurement(calculatedMeasurement)

' Add calculated reference angle to latest angle queue used as backup in case
' needed minimum number of PMU's go offline or are slow reporting
SyncLock m_latestCalculatedAngles
    With m_latestCalculatedAngles
        .Add(angleAverage)
        While .Count > BackupQueueSize
            .RemoveAt(0)
        End While
    End With
End SyncLock
End Sub
#If DEBUG Then

Private Sub LogFrameDetail(ByVal frame As IFrame, ByVal frameIndex As Integer)
    With m_frameLog
        .AppendLine("***************************************************************")
        .AppendLine(" Frame Index: " & frameIndex)
        .Append(" Keys: ")
        For Each measurement As IMeasurement In frame.Measurements.Values
            .Append(measurement.Key.ToString().PadLeft(10) & " ")
        Next
        .AppendLine(""
        .Append(" Values: ")
        For Each measurement As IMeasurement In frame.Measurements.Values
            .Append(measurement.Value.ToString("0.000").PadLeft(10) & " ")
        Next
        .AppendLine(""
    End With
End Sub

Private Sub LogFrameWarning(ByVal warning As String)
    m_frameLog.AppendLine([" & DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss.fff") & "]: " & warning)
End Sub
#End If

End Class