OSCILLATION ANALYSIS AND MODEL SECURITY IN POWER SYSTEM OPTIMIZATION PROBLEMS

by

Alexander R. Borden

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy
(Electrical Engineering)

at the

UNIVERSITY OF WISCONSIN-MADISON

2014

Date of final oral examination: January 11, 2014

The dissertation is approved by the following members of the Final Oral Committee:

Bernard C. Lesieutre, Professor, Electrical and Computer Engineering
Christopher L. DeMarco, Professor, Electrical and Computer Engineering
Parameswaran Ramanathan, Professor, Electrical and Computer Engineering
Michael C. Ferris, Professor, Computer Science
Ian Dobson, Professor, Electrical and Computer Engineering

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to the numerous people who have helped me throughout my academic career and personal life during my time as an undergraduate and graduate student at the University of Wisconsin - Madison. Starting with my PhD committee, I feel thankful and fortunate to have interacted with and learned from each one of the professors.

I would like to thank Dr. Christopher DeMarco for positively impacting my academic career right from the start. My study of electrical engineering was significantly enhanced through his courses on electronic circuits and advanced power system analysis. I was given a broader view by seeing his seminars on nonlinear stability and control theory in power systems and by his hosting of the 2013 PSERC IAB meeting. I also had the pleasure of assisting Dr. DeMarco as the teaching assistant in a semester of ECE 342, which was a constructive experience for me.

I would like to thank Dr. Michael Ferris for his imparting of knowledge on optimization theory. As a graduate student, I was able to enroll in Dr. Ferris' courses on nonlinear optimization and computational methods for large sparse systems. I give credit to Dr. Ferris for my understanding of optimization, as his super enlightened understanding of all things optimization related is unrivaled. With Dr. Ferris' direction, I was able to develop some of the major theories presented in this dissertation.

I would like to thank Dr. Parameswaran Ramanathan for sharing his intriguing research ideas and for extending his research to the field of power systems. His work on confidential integer linear programming served as inspiration for the latter half of research work in this dissertation. The research on confidential computing has drawn interest from the National Science Foundation and is being continued on by Dr. Ramanathan and his graduate student. I wish them all the best and will always be available for assistance.

I would like to thank Dr. Ian Dobson for his instruction on system theory. His course on nonlinear systems taught me advanced power systems concepts including stability analysis, energy functions and bifurcation theory. I appreciate his enthusiasm for power systems research, and I am grateful to have studied under one of the field's top experts.

I must thank my advisor, Dr. Bernard Lesieutre. I cannot express my gratitude enough for him facilitating my success as a graduate student. I am indebted to his guidance. Dr. Lesieutre enabled me to attend conferences and business meetings all around the country, placing me in connection with important industry members and academics in power engineering. With Dr. Lesieutre's passion for power system research, I was fortunate to be involved on projects including: market power analysis, model validation, semidefinite programming, modal analysis, model security, and a confidential optimal power flow. Beyond his superb professional qualities, he is a humorous presenter, a humble individual and a genuinely great person.

There are a number of other professors at the University of Wisconsin – Madison who have had positive influences on me throughout my education. Some of those professors include Dr. Stephen Wright, Dr. Thomas Jahns, Dr. Giri Venkataramanan, Dr. Daniel Cobb, Dr. John Booske and Dr. John Gubner. I am also thankful for the friendship and advice from many current and former graduate students. These include Daniel Molzahn, Vikas Dawar, JongMin Lim, Honghao Zheng, Fatou Thiam, Chaitanya Baone, Taedong Kim, Jesse Holzer, Sowmya Acharya, Hillary Brown, Nishant Mehta, Rafael Viloria, Kanishka Kumar, Heng (Kevin) Chen, Mike Schlindwein, Daniel Schwarting, Aditya Shastri and Rahul Srinivasan.

I must express my deepest gratitude to my family and loved ones: to my sweet and loving mother Catherine, to my steady role-model father Garry, to my adventurous and unique twin brother Michael, and to my supportive and bright minded girlfriend Megan. I extend my gratitude and love to the rest of my family and friends as well.

I gratefully acknowledge the support of the Consortium for Electric Reliability Technology Solutions, the National Science Foundation, Bonneville Power Administration, the Grainger Power Engineering Award, research assistantships funded by grants from the Department of Energy, and a teaching assistantship from the University of Wisconsin – Madison.

I will be taking the knowledge and skillsets I developed at the university with me to the Midcontinent Independent System Operator following this time.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

The work in this dissertation is motivated by the need to increase power system model accuracy, stability and security. Practical issues associated with these topics are addressed, along with improvements in advanced methods and tools from optimization. Results include a robust method for estimating dynamic oscillatory features from data and also a method supporting secure exchange of power system models. These advancements will help create more reliable and secure power system operations. Improvements in model accuracy, stability and security will benefit electricity operators, customers and researchers alike.

First a modal analysis problem involving power system stability is presented. Sudden disturbances in the electric grid tend to cause oscillatory ringing of power, voltage and other units of measure. When the differential equations describing the model's dynamics are available, the natural modes of the system can be calculated. These modes characterize the stability properties of the system. However, often times the differential equations describing the model's dynamics are unavailable. Instead, modal information must be extracted from oscillatory data collected during a disturbance event. The industry standard modal analysis technique Prony analysis is discussed along with its shortcomings. A nonlinear least-squares optimization problem that is superior to Prony analysis and other traditional modal analysis methods, is presented. The approach is extended to include nonlinear behavior motivated by the theory of normal form analysis. The methods and theories are fully brought to fruition with the creation of a user friendly tool and graphic user interface, which has been distributed with success to power industry engineers.

The second half of work involves a model security problem that is motivated by the highly confidential nature of power system models and the desire to share actual models in power system communities. As a starting point, the frequently encountered optimal power flow optimization problem is initially addressed. The optimal power flow problem is ever increasing in size

and complexity, and with that comes an increased need for computational resources. The possibility of solving the optimal power flow problem with powerful remote computers is thus intriguing; however, the highly confidential nature of power system models requires that any optimal power flow problem sent to the cloud must first undergo some form of masking. A method for preserving confidentiality in an optimal power flow problem is presented in the model security chapters of this report. The method was primarily developed for use in cloud computing, but it is also of value for any scenario requiring data-secure optimal power flow computation. The confidentiality preserving optimal power flow method is finally examined for a separate model security problem, where an existing power system model is transformed into an entirely new and legitimate power system model. The transformation allows for the secure exchange of power system models, that are based off confidential counterparts, which can by freely shared within power system communities and amongst researchers.

This dissertation sheds insight and solutions into two fundamental power system engineering problems: oscillation analysis for system stability and model confidentiality for system security. Both unique problems share in common current inadequate approaches to system reliability and security in need of improvement, while having an underlying theme on optimization.

Chapter 1
Introduction

The work in this dissertation is motivated by the need to increase power system model accuracy, stability and security. Advancements resulting from this work will better ensure the reliable and secure operation of the electric power system. The connecting theme throughout this report is the development of tools that address key power system problems in need of improvement; with a core emphasis on optimization methods. The topics are further related by the need for accurate power system models for reliable grid operation, and also by the desire to share realistic models within power system communities. Introductions to power system problems of modal analysis and model security are presented here in Chapter 1. Research work associated with modal analysis is presented in Chapters 2–5, while the research work associated confidential optimal power flow problems is presented in Chapters 6–7. Conclusions and proposed continuations of this research work are presented in Chapter 8.

1.1 Motivation

The North American power system may be the world's single largest technological mechanism, spanning all of the United States, Canada and a portion of Baja California in Mexico. The entire system is divided into four major synchronous Interconnections, Eastern, Western, ERCOT and Quebec, where the Interconnections are linked by DC transmission facilities [1]. Each Interconnection acts a single large machine, with every generator in the island pulling in tandem to supply the electrical demand of its customers. This wide spread connectivity and interoperability creates an abundance of unique issues concerning stability and security in the system. To today's society, electric power has become as essential as water and heating services. In order to support reliable electric power services, robust analysis tools must be developed to address the issues of grid stability and security. This need for analysis tools, concerning stability and security, motivates the work in this dissertation report.

A byproduct of inadequate analysis tools is unplanned electricity outages. Outages have damaging wide-spreading effects, directly affecting electricity consumers. A major electric outage can leave millions of commercial, residential and industrial customers without power. The consequence of electric outages varies in type, though some studies estimate the economic cost of interruptions to U.S. electricity consumers is about $79-80 billion annually [2,3]. It is clear that electric reliability is important, and has serious social and economic consequences when failures occur. The work in this report seeks to better ensure reliability by improving methods of stability assessment and by examining the security of power system models for confidential operations.

1.2 Power System Oscillations and Modal Analysis

The dominant oscillating modes of a power system characterize its stability properties. Severe faults, such as the loss of major transmission lines or generators, will cause oscillations in the system. Once started, the oscillations either damp-out, sustain or continuously grow towards system collapse. In these cases, real-time modal analysis techniques allow for situational awareness of the power system's stability [4]. The oscillations can be categorized differently depending on generator locations and the frequency of oscillations. Intra-area oscillations occur in localized regions and have typical oscillation frequencies of 1 Hz or higher. In contrast, inter-area oscillations are observed over large parts of the network. They are associated with coherent groups of generators swinging against each other at 1 Hz frequency or less. Inter-area oscillations are of primary interest here due to their wide-spreading effects. The damping characteristics of the inter-area oscillations are dictated by tie-line strength, the nature of loads and the dynamics of generators with their associated controls [5].

An example of unstable power oscillations causing widespread outages was the August 10, 1996 event in the Western Interconnection. A 500-kV transmission-line carrying 1300 MW sagged close to a tree and flashed over causing the line to trip. Power was rerouted through parallel lines loaded up to 115% of their thermal ratings, and more lines tripped due to relay failures

and sagging into trees. Depressed voltage levels caused sequential tripping of generator units, which started the power and voltage oscillations [6,7]. A 0.25 Hz growing mode was sustained on the California–Oregon Intertie, resulting in the intertie tripping and the system separating. The long, intricate cascading event shed 30 GW of load and ultimately disconnected 7.5 million customers, or 24 million people, for up to 9 hours [8]. Since then, preventative measures have been taken to stabilize poorly damped modes.

Power System Stabilizers (PSS) were developed in the mid 1960's in response to lightly damped inter-area oscillations having low frequencies of less than 1 Hz. The PSS control system works in conjunction with the excitation system of a synchronous machine to increase system damping and extend power transfer limits [9]. PSS are installed to damp out local intra-area oscillations, and multiple PSS in coordination may be installed at remote power stations to jointly damp out widespread inter-area oscillations [10,11,12]. Flexible AC Transmission System stabilizers (FACTS) can alternatively be installed at locations close to tie-lines and long-distance transmission corridors for damping inter-area modes [13]. Stabilizing oscillations with PSS and FACTS involves two main issues of design: properly tuning parameters for feedback and selecting key installation locations. Accurate information of mode observability and controllability are thus required. PSS and FACTS feedback parameters can be tuned to stabilize poorly damped modes identified by modal analysis methods, and the modal analysis participation factors reveal key installation locations for the devices.

When fully represented dynamic models of a power system are available, the modes of the system can be calculated using small-signal analysis or linearization. The differential equations of the state-space model can be linearized about an operating point, and eigenanalysis of the linearized state matrix gives the eigenvalues and eigenvectors of the system. The eigenvalues correspond to the system's natural damping factors and frequencies, and the eigenvectors construct to the participation factors of each mode [14]. Linearization is a simple yet powerful tool for determining system modes and stability properties; however power systems are inherently nonlinear. At the most fundamental level, the power flow equations are trigonometric. Installation of

FACTS devices, high-voltage direct current (HVDC) transmission and power electronics only increase the system's nonlinearity [15-17]. Furthermore, to increase economic benefit, the modern power system is operated close to its stability limit. Highly nonlinear dynamic behavior is exhibited under these stressed conditions. A linearized solution acquired by small-signal analysis may poorly capture the dynamics of a highly nonlinear power system. More accurately, normal form analysis can be used to represent a system's nonlinear dynamics.

Normal form analysis captures nonlinear characteristics with high order interaction between the system's natural modes. Similar to linearization, a stable equilibrium point is expanded about; but comparatively, the normal form solution has increased accuracy at deviations further away from the equilibrium. In the normal form method, near identity coordinate transformations eliminate higher order terms, thus yielding a closed form approximate solution [15-21]. Normal form analysis is commonly used for identifying system bifurcation types [22]; which is done by expanding the differential equations about a bifurcation point. At the bifurcation point, some higher order terms will fail to eliminate, and the left over irreducible equation reveals the bifurcation type, i.e. saddle-node, pitchfork, Hopf, etc.

Alternative to model-based modal analysis methods, such as linearization, are model-free (or measurement-based) modal analysis methods which can be used to analyze data signals from recorded events. These methods decompose an oscillating data signal into a finite sum of oscillating components. Standard methods such as Prony analysis [23,24] and the Matrix Pencil method [25,26], determine modal content by estimating a linear system to match the data. The techniques, however, have shortcomings and can struggle to fit highly nonlinear or noisy disturbance data, leaving room for improvement. A nonlinear least-squares optimization approach yields better results in comparison [27-30]. The optimization can be built upon by using the theory of normal form analysis [31]. This work is presented in the next three chapters of the report.

1.3 Model Validation

As suggested, measurement-based modal analysis methods are valuable for many reasons. One additional purpose for these methods is for performing model validation. For example, an

actual power system model can be digitally constructed using simulation software. The simulation models are then used for making decisions on how to operate the actual power system. It is expected that the simulations will accurately reflect the behavior of the system in reality. A disparity between simulated and actual events, will thus infer an inaccuracy in the simulated model. In the case of ring-down disturbance events, the modes from actual data can be compared against modes from the simulation.

The power system simulation tools often appear as black-boxes to the user. In these cases, small-signal analysis cannot be performed on the state-space model to calculate system modes. Measurement-based modal analysis methods can be used instead for estimating the dominant oscillatory modes from both the actual and simulated data. Important features of interest, such as dominant mode damping, frequency, amplitude and phase can be compared between the actual and simulated data, thus validating the simulation model. Improvements in modal analysis techniques will in turn improve model accuracy, further motivating the modal analysis work in this report.

Model validation studies are used to ensure simulation models are accurate and up-to-date. Power system engineers use the models to make a wide variety of decisions, including: transmission planning, stability assessment and generation dispatch. An overly pessimistic model can result in overly conservative grid operation, under-utilization of network capacity, unnecessary capital investment and increased cost of electric power [32]. On the other hand, overly optimistic models result in grid under-investment, unsafe operating conditions and potentially widespread power outages. Take for example the August 10, 1996 blackout in the Western Interconnection. After the event, power engineers attempted to reproduce simulations of the widespread power outage. The dissimilarity in simulation results and actual disturbance data was alarming. Beforehand, the power transfer limits for critical paths had been determined by inaccurate simulation models. Afterward, the operating transfer capabilities of critical paths were significantly de-rated based on results of model validation studies [32].

Figure 1.1 depicts the California–Oregon Intertie (COI) power flow on August 10, 1996 before the intertie tripped [6]. The actual data shows the 0.25 Hz growing oscillation, where the

simulated response entirely omits it. This is an example of an overly optimistic simulation model.

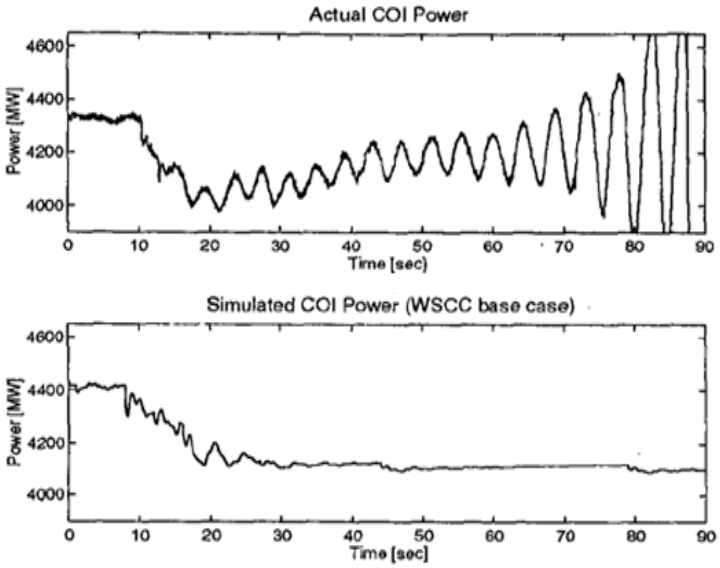

Figure 1.1: Actual vs. Simulated COI power flow, Aug. 10, 1996

In the modal analysis work of this report, an optimization procedure is presented for computing the modal solutions. As mentioned, the solutions can be used to assess the accuracy of power system models. The remaining second half of work in this report addresses another optimization and model based problem, that being the optimal power flow problem. The optimal power flow problem is one of the most prevalent and extensively studied optimization problem encountered in power system engineering. This report seeks to study it from a new angle which involves privacy, security and especially the sharing of confidential models.

1.4 Optimal Power Flow Levels of Complexity

The electric power industry has experienced many changes and new challenges over recent decades. The change to deregulated electricity markets has introduced competition and reduced costs, but it has also created uncertainty in generation forecasting. Non-dispatchable renewable energy resources have only further exaggerated this uncertainty. The increase in consumer demand has outpaced the development of electrical infrastructure, which has placed stress on aging

equipment. All the while, system stability must be assured for a multitude of contingency scenarios. For these reasons, it is evident that the optimal power flow problem has been represented in a variety of forms, with its complexity evolving over time [33,34]. It is one of the most widely studied optimization problems of the past half-century.

In general the optimal power flow problem seeks to minimize a given cost, subject to the engineering limits and physical constraints of the system. The cost to be minimized is typically the monetary cost of generation dispatch, but could also include objectives for planning, reliability and more. The full AC optimal power flow problem is a nonlinear, non-convex, large-scale optimization problem. It may contain both continuous and integer optimization variables. It may also be independent of time or be a dynamic program considering current operation points and generation ramping constraints [33].

Advancements in optimization techniques have offered real measurable monetary savings. For example, mixed-integer programming for unit commitment problems saved United States electric system operators over $500 million annually in 2011, with forecasted savings of $1 billion annually by 2015 [35]. Just ten years ago the unit commitment problem stretched the capabilities of computer hardware and optimization algorithms. Before then, unit commitment applications relied on human judgment and dynamic programming heuristics [36].

1.5 Model Security and Confidentiality Preserving OPF

It is clear optimization problems in the electric power industry are becoming more complicated for many reasons. Therefore any advancements in optimization techniques can help counteract growing complexity, helping save time, money and resources. Computational needs continue to increase for solving the optimal power flow (OPF) problem; therefore, powerful computing infrastructure and advanced optimization tools have become a necessity for the electric power industry. This computational need intrigues the possibility of using powerful remote computing resources available through cloud computing.

The concept of cloud computing was introduced in the 1960's by John McCarthy, whose opinion was that "computation may someday be organized as a public utility". Cloud computing

can be defined as a computing environment where computing needs by one party can be outsourced to another. With cloud computing, powerful computing resources and database storage are made available via internet access. For cloud computing customers, computing processes and data handling is moved away from their desktops and into large data centers. It is appealing that they no longer have to pay for personal computing infrastructure, installation, manpower and maintenance. Access to cloud resources are available to the customer, regardless of their device used, their location or time of use [37].

A similar idea was demonstrated with the Network Enabled Optimization System (NEOS) server and the Condor pool at the University of Wisconsin. NEOS provides access to solvers for optimization problems via the internet; coming complete with automatic differentiation tools and sparsity pattern identification for nonlinear problems. A user provides minimum problem specifications, and all other necessary information is determined by the NEOS solver. Condor, a distributed resource management system, provides the computational resource of large workstation clusters that would otherwise be running idle [38]. A sample of applications the NEOS server has been used for include: chemistry, physics, circuit design and power system engineering problems [39].

For the electric power industry, cloud computing resembles an intriguing possibility for outsourcing complex problems. As noted in [41], different applications in power system planning and operations have varied computational requirements. Some applications may be computed only on occasion, where others may be computed nearly continuously. For example, renewable energy integration planning may be performed up to ten years in advance, requiring computations on a low frequency basis. These studies are extensive however, and are simultaneously contingent on various load, generation and weather conditions. Studying the feasibility of renewable energy integration into constrained transmission networks requires sophisticated simulations of various load, generation and weather conditions. These analysis are highly parallelizable and are thus prime candidates to benefit from the elasticity provided by cloud computing. In contrast, locational marginal pricing (LMP) is computed in real-time spot markets. The LMP is contingent on the OPF problem, which itself may be a simple quadratic optimization or

could be as complex as a security constrained, stochastic dynamic program over a receding horizon. Cloud services may also be of interest for reliable data storage for power system applications. Traditional SCADA historians may require utilities to generate and store 100TB of data annually. Now with the advent of Phasor Measurement Units, which sample the system 30 times a second compared to every few minutes with SCADA systems, the requirement for data storage will increase significantly [41].

A major obstacle to the adoption of cloud computing in the electric power industry, is the confidentiality of their data and models. Cloud computing security may already be good, as the provider devotes resources to solving security issues that many customers cannot afford individually [37,40,42]. Nonetheless, data security is of primary importance to the electric power industry. Without a secure computing environment, the possibility of utilizing the cloud for their computational needs may not be considered. With appropriate risk mitigation approaches, the risk versus reward may warrant the use of cloud services for certain applications.

The model security work in this report addresses several security issues, inspired by the potential of cloud computing or for uses in any application where secure computation is required. Initially, linear programming masking techniques detailed in [43] and [44] are built upon but tailored specifically to the linear DC optimal power flow problem [45]. The work is extended to the full nonlinear AC optimal power flow, with a discussion on the limitations encountered in the nonlinear case. Other applications of the masking procedure are discussed, including the possibility of multi-party optimization.

Lastly a related problem of transforming the characteristics of an existing power system model to a legitimate and entirely different power system is presented. The transformation may open doors for sharing models provably relatable to other confidential models, which may fundamentally transform the process of power system research developments and data sharing. Currently, models of confidential power systems must be obtained through nondisclosure agreements. In this way, results cannot be independently checked, which largely prevents against the standard scientific protocol of verifying and replicating results.

1.6 Outline and Contributions

In Chapter 2, an introduction to modal analysis is provided with discussion of both model-based and measurement-based methods. Traditional measurement-based modal analysis methods of Prony Analysis, the Matrix Pencil method and the Eigensystem Realization Algorithm are presented. Contribution are made by extending the Matrix Pencil method and Eigensystem Realization Algorithm for use in multi-signal modal analysis problems, where numerous oscillatory data-signals are simultaneously analyzed for shared modal content.

In Chapter 3, a nonlinear least-squares optimization method called the variable projection method is discussed for use in measurement-based modal analysis problems. Data fitting using nonlinear least-squares approaches is not a new concept, but it has gone overlooked for modal identification power system problems. The method outperforms Prony analysis and other traditional methods in estimating modal content of data. The traditional modal analysis methods from Chapter 2 are used as initial conditions for the optimization for improved solution speed and likelihood of locating global minimizers. Contributions to the method include extensions for multi-signal analysis and computational speed improvements. Extensions for polynomial data-detrending and constrained optimization are detailed as well.

In Chapter 4, the theory of normal form analysis is presented for analyzing nonlinear state-space models. By example, it is shown that normal form analysis provides a closed form solution to the model which is superior to the solution obtained by linearization and small-signal analysis. The observation is made that high order combinations of natural modes are observed in the normal form solution structure for nonlinear systems. This observation motivates an extension to the variable projection method for modal analysis problems, which seeks to better estimate the natural modes or eigenvalues of nonlinear ring-down disturbance data. A similar extension is also presented for systems with repeated eigenvalues.

In Chapter 5, the culmination of the modal analysis techniques from the prior chapters are consolidated into a user friendly tool and graphic interface developed for industry use. The tool has been met with positive reception by industry members, and it has served to advance the goals

of this dissertation of improving stability and reliability in real electric power systems. Discussion is provided on the types of optimization algorithms best suited for this problem. An example is provided for analyzing real power system disturbance data. The example provides a platform for demonstrating the effective approaches to modal analysis problems, such as the issue of determining appropriate model order selection (number of modes). The example demonstrates the need for flexible tools, which allow user guidance and interactivity for optimizing performance.

In Chapter 6, model security work is addressed for confidential optimal power flow problems. Recently developed masking approaches for securely solving linear programs in cloud computing are specifically applied to the linear DC OPF problem. Contributions are made by detailing procedures for extracting dual variables and locational marginal prices from fully masked OPF problems, procedures for masking quadratic cost functions, and approaches for preserving problem sparsity. The nonlinear AC OPF problem is discussed for similar masking procedures, along with its security limitations imposed by its nonlinear nature.

In Chapter 7, a unique power system transformation problem is presented for transforming a legitimate power system OPF model to a totally new and different OPF model. The transformation preserves a mapping between optimal cost and variables between the two problems. The transformation is detailed for linear programs, quadratic programs, and piece-wise linear cost functions. The work is motivated by current nondisclosure policies for handling confidential power system models, with an end goal of facilitating free exchange of credible models within power system communities and research groups.

Chapter 8 concludes the dissertation with reflections on the contributions and ideas presented in this report. Future work directions for areas of research involving modal analysis, model security, and optimization are proposed.

Chapter 2

Modal Analysis

Electromechanical disturbances in the electric power system tend to cause power oscillations that either damp out, sustain or grow. The disturbances are sources of undesired oscillatory ring-down behavior, and the dominant oscillating modes characterize the system's stability properties and reveal important information about the system. Phasor Measurement Unit (PMU) data of these ring-down events can be analyzed to extract important modal information of the power system. It is the purpose of this chapter to investigate the importance of accurately estimating modes from data. Accurate mode estimation will assist system stability analysis, improve model validation, and guide control mechanisms for stabilizing poorly damped modes. This chapter provides an introduction to modal analysis problems, along with an examination of traditional analysis methods.

2.1 Small-Signal Analysis

Power systems are modeled using coupled differential and algebraic equations which characterize the system's interconnections and dynamic variables. Algebraic power balance equations describe the power flow, the load demand and the system's topological interconnections. Differential swing equations are used to model generator dynamics describing the turbine inertia and damping constants and control mechanisms. Combining the power balance and swing equations yields a set of nonlinear ordinary differential equations (ODE's) which describes the total system representation [46]. Small-signal analysis and linearization about an operating point can be performed on the nonlinear ODE's to obtain a linearized state-space model in (2.1), where x is the state vector, y is the output vector, u is the input vector, A is the state matrix, B is the input matrix, C is the output matrix and D is the feedthrough matrix.

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{2.1}$$
$$y(t) = Cx(t) + Du(t)$$

Generally, the input vector u can be dropped if the system in unforced, which simplifies (2.1). Eigenanalysis can be performed to determine the eigenvalues of (2.1). Consider a system with p states. Denote V as the matrix with the eigenvectors of matrix A as its columns, and denote the eigenvalues of A as $\lambda_1, \dots, \lambda_p$. The solution for $x(t)$ in (2.1) is shown in (2.2), where $x(0)$ is the state vector at time $t = 0$, and $diag(V^{-1}x(0))$ creates a diagonal matrix from the vector $V^{-1}x(0)$.

$$x(t) = V * diag\left(V^{-1}x(0)\right) * \begin{bmatrix} e^{\lambda_1 t} \\ e^{\lambda_2 t} \\ \vdots \\ e^{\lambda_p t} \end{bmatrix} \tag{2.2}$$

The solution for $x(t)$ in (2.2), and correspondingly the output vector $y(t)$, is therefore a summation of damped or undamped sinusoids and exponentials. In power system problems, measurements of state variables $x(t)$ are less accessible than measurements of power flow, voltage, etc., which are considered output measurements in $y(t)$.

For large scale power systems, it may be less practical to perform eigenanalysis due to the inaccuracies of modeling the entire state-space system. Therefore measurement-based modal analysis methods can be used to estimate the system's eigenvalues and mode shapes. The system model can be validated using disturbance data analyzed by the methods. Traditional modal analysis methods are detailed in the next section.

2.2 Traditional Modal Analysis Methods

As detailed in the prior section, power systems are modeled by sets of differential and algebraic equations. The system dynamics are characterized by modes having amplitude, frequency, damping and phase. Therefore one may advocate the explicit use of these distinct features for validating system models, with data represented as a sum of damped or undamped sinusoids. Efficient mathematical techniques have been developed to perform the task of estimating modal content of data. This section examines three traditional linear modal analysis techniques: Prony analysis, the Matrix Pencil method and the Eigensystem Realization Algorithm (ERA).

Consider an oscillatory data waveform denoted as y having m discrete data points, evenly spaced over the time interval t and starting at time 0 seconds.

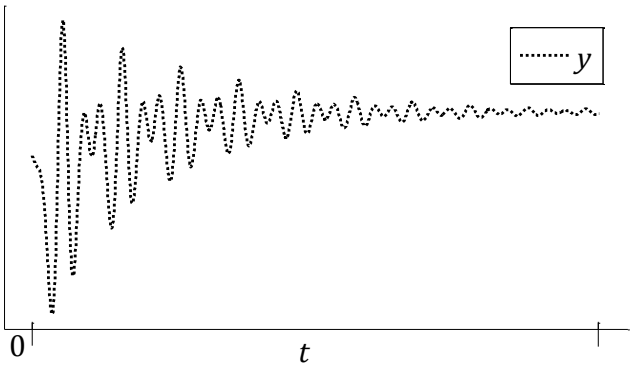

Figure 2.2: Example data-set y with m data points

The modal analysis methods seek to fit a function $\hat{y}(t)$, shown in (2.3), consisting of damped or undamped exponentials, to the oscillatory data-signal y.

$$\begin{aligned}\hat{y}(t) &= \sum_{i=1}^{q} A_i\, e^{\sigma_i t}\, \cos(\omega_i t + \phi_i) \\ &= \sum_{i=1}^{q} A_i e^{\sigma_i t}[\cos(\phi_i)\cos(\omega_i t) - \sin(\phi_i)\sin(\omega_i t)]\end{aligned} \tag{2.3}$$

Computationally, the data is analyzed as if it were generated by an impulse response of a linear system whose coefficients are calculated using temporal correlations in the data [23,24]. The natural modes of the linear system correspond to $\sigma_i + j\omega_i$ in (2.3). The frequency ω_i has units rad/s, and the damping σ_i, sometimes referred to as the *neper frequency*, has units $1/s$ [47]. The mode shape amplitude and phase correspond to the complex coefficients $A_i \angle \phi_i$. In practice, the data comprises equally-spaced discrete-time sample points, where Δt is the time-sample period. The discrete-time representation of (2.3) may be represented as shown in (2.4).

$$\hat{y}[n] = \hat{y}(n\Delta t) = \sum_{i=1}^{p} b_i z_i^n \tag{2.4}$$

The signal $\hat{y}[n]$ is a finite summation of p mode pairs (b_i, z_i), where $b_i \in \mathbb{C}$ is the output residue coefficient for the discrete-time pole $z_i \in \mathbb{C}$ [23]. The discrete-time poles z_i are related

to continuous-time eigenvalues λ_i by $z_i = e^{\lambda_i \Delta t} = e^{(\sigma_i + j\omega_i)\Delta t}$. Note the series summation to q in (2.3), as opposed to the series summation to p in (2.4). It follows that $q \leq p$ due to the cancellation from complex conjugate mode pairs (b_i, z_i) and (b_i^*, z_i^*) as shown below.

$$b_i\, z_i^{t/\Delta t} + b_i^*\, z_i^{*t/\Delta t} = A_i\, e^{\sigma_i t}\, \cos(\omega_i t + \phi_i)$$

If there are $n_{\mathbb{C}}$ number of pairs of complex conjugate modes and $n_{\mathbb{R}}$ number of real modes, then $q = n_{\mathbb{C}} + n_{\mathbb{R}}$ and $p = 2n_{\mathbb{C}} + n_{\mathbb{R}}$. Therefore if p is an odd number, then at minimum there must be one real-valued mode.

The conversion between continuous-time parameters in (2.3) and discrete-time parameters in (2.4) depends on the time-sampling period Δt. For complex-conjugate mode pairs, the residue coefficient $b_i = \frac{A_i}{2} e^{j\phi_i}$ and discrete-time pole $z_i = e^{(\sigma_i + j\omega_i)\Delta t}$. Real-valued mode pairs simply relate by $b_i = A_i$, $z_i = e^{\sigma_i \Delta t}$, $\omega_i = 0$ and $\phi_i = 0$.

2.3 Multi-Signal Analysis

Prony analysis, matrix pencil and ERA can all be extended for simultaneously analyzing multiple data-signals of a particular disturbance event. As detailed in [24], multi-signal Prony analysis tends to improve modal estimate accuracy, and similar observations can be made for matrix pencil and ERA.

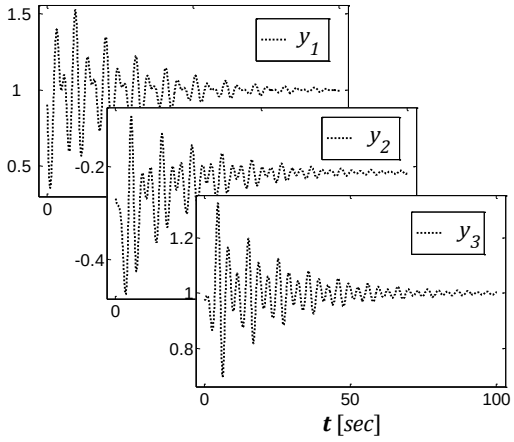

Figure 2.3: Multi-signal example data-set

Consider a data-set where multiple signals are collected, such that $y = [y_1, \dots, y_{n_{sig}}]$, where y is an $m \times n_{sig}$ matrix. The multiple signals belong to the same system, so it is assumed each signal exhibits the same p natural modes/discrete-time poles z_i. If the data-set had just one signal, then the residue coefficient b_i would be a complex scalar in the mode pair (b, z_i). However, when the data-set has multiple signals, the residue coefficient b_i becomes a vector of length n_{sig}.

In this chapter, the traditional methods of Prony analysis, matrix-pencil and ERA are detailed for multi-signal modal analysis. Each method is performed as a two-step procedure: first, modal damping and frequency is estimated, then second, modal amplitude and phase is computed. The next section details the first step of these two-step procedures.

2.4 Solving for a Linear Model and Discrete-time Poles z_i

Before using Prony analysis, matrix pencil or ERA for estimating discrete-time poles z_i, the data-signal y may require some preconditioning. The data-signal can be preconditioned so that its dc-offset is subtracted; which by doing so, the methods are alleviated from approximating a zero eigenvalue mode, i.e. a dc-offset mode. A reasonable choice for dc-offset subtraction is the signal's final value as time goes to infinity, as steady-state is presumed to be approached; however, this conjecture may worsen for unstable or lightly damped data. Additional preconditioning is required if the data-signal is not uniformly spaced by Δt. For unequally spaced data, interpolation must be performed to enforce uniformly spaced time-samples; which is required because analysis will be done in the discrete-time domain. Finally, if necessary, a noise reduction filter may be applied for overly noisy data.

Prony analysis, matrix pencil and ERA are all performed as two-step procedures. In the first-step, the discrete-time poles z_i of the system are estimated, thereby solving for a linear model to match the data. Each of the three methods performs this first-step differently, as shown in the following subsections.

A. Prony Analysis

Prony analysis has proven to be a valuable tool in estimating the modal content of power oscillations from measured ring-downs. The method was developed by Gaspard Riche de Prony in 1795 but did not see its best use until the arrival of modern computing. Early adoption of Prony's method in power system problems can be seen in [48], where it is applied to several Western Interconnection ring-down events. Since that time, Prony's method has been incorporated into many commercial power system analysis software's [49,50].

The user of Prony's method must predetermine the number of modes p. The p discrete-time poles z_i are calculated using the linear prediction property of linear systems [23,24]. The linear prediction property of linear systems is presented here. Consider a linear system with exactly p modes, i.e. p discrete-time poles $z_1, \dots, z_p$. Construct the characteristic polynomial with the p discrete-time poles $z_1, \dots, z_p$ as its roots as in (2.5). Denote the coefficients of the characteristic polynomial as $a_0, \dots, a_p$.

$$\sum_{j=0}^{p} a_j z^{-j} = \prod_{j=1}^{p} \left(1 - z_j z^{-1}\right) \tag{2.5}$$

If the discrete-time poles $z_1, \dots, z_p$ are known, then the characteristic polynomial coefficients $a_0, \dots, a_p$ can be computed by matching like terms in (2.5). Consider a linear system with exactly p modes and p data points evenly spaced in time by Δt, $y(i\Delta t) = y[i]$ for $i = 0, \dots, p-1$. The next data point in time, $y[p]$, can then be exactly computed by using the linear prediction property of linear systems, as shown in (2.6).

$$y[p] = -\begin{bmatrix} a_1 & a_2 & \dots & a_{p-1} & a_p \end{bmatrix} \begin{bmatrix} y[p-1] \\ y[p-2] \\ \vdots \\ y[1] \\ y[0] \end{bmatrix} \tag{2.6}$$

This linear prediction property is exploited in Prony's method to help estimate the discrete-time poles z_i. Generally, linear prediction uses knowledge of the discrete-time poles $z_1, \dots, z_p$ to calculate the unknown characteristic polynomial coefficients $a_0, \dots, a_p$, but in Prony analysis the opposite is done. In Prony analysis, an approximation of the characteristic polynomial coefficients $a_0, \dots, a_p$ is used to calculate the discrete-time poles $z_1, \dots, z_p$, as shown next.

Assume n_{sig} number of oscillatory data-signals $y_1, \dots, y_{n_{sig}}$ are collected, each having m data points evenly spaced in time, $y_i = [\, y_i[0] \;\; \dots \;\; y_i[m-1] \,]^T$. The user of Prony's method must specify the number of modes p assumed to be present. An $(m-p) \times p$ Toeplitz matrix T_i is formed, and an $(m-p) \times 1$ column vector $y_{i,t}$ is also formed from the tail-end of data points in y_i for $i = 1, \dots, n_{sig}$.

$$T_i = \begin{bmatrix} y_i[p-1] & y_i[p-2] & \cdots & y_i[1] & y_i[0] \\ y_i[p] & y_i[p-1] & \ddots & y_i[2] & y_i[1] \\ y_i[p+1] & y_i[p] & \ddots & y_i[3] & y_i[2] \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ y_i[n-2] & y_i[n-3] & \cdots & y_i[n-p] & y_i[n-p-1] \end{bmatrix} \quad for\ i = 1, \dots, n_{sig} \qquad (2.7)$$

$$y_{i,t} = \begin{bmatrix} y_i[p] \\ y_i[p+1] \\ y_i[p+2] \\ \vdots \\ y_i[n-1] \end{bmatrix} \quad for\ i = 1, \dots, n_{sig}$$

The n_{sig} Toeplitz matrices T_i and vectors $y_{i,t}$ are stacked into one large matrix T and vector y_t.

$$T = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{n_{sig}} \end{bmatrix} \qquad y_t = \begin{bmatrix} y_{1,t} \\ y_{2,t} \\ \vdots \\ y_{n_{sig},t} \end{bmatrix} \qquad (2.8)$$

A pseudoinverse operation is performed to estimate the characteristic polynomial coefficients $a_1, \dots, a_p$.

$$a = -T^{\dagger} y_t \tag{2.9}$$

With the coefficients $a_1, \dots, a_p$ approximated from (2.9), the discrete-time poles $z_1, \dots, z_p$ can be solved by matching like terms in (2.5). An observation is made here that Prony analysis fits its solution to the coefficients $a_1, \dots, a_p$ and not the data itself. This approach can result in inadequate solutions when compared to other methods.

B. Matrix Pencil Method

Like Prony analysis, the Matrix Pencil method matches data to coefficients of a linear system, and the method has recently found use in power system analysis [25,26]. The Matrix Pencil algorithm for multi-signal modal analysis is performed as follows.

As before, n_{sig} number of oscillatory data-signals $y = [\, y_1 \,, \dots, y_{n_{sig}}]$ are collected, each having m data points evenly spaced in time, $y_i = [\, y_i[0] \;\; \dots \;\; y_i[m-1] \,]^T$. To begin, the pencil parameter L is introduced. The pencil parameter is chosen such that $L = floor(m/2)$, which is chosen because the performance of the method becomes close to the optimal Cramer-Rao bound at this value [26]. Next, a Hankel matrix H_i is formed for each of the data signals y_i.

$$H_i = \begin{bmatrix} y_i[0] & y_i[1] & \cdots & y_i[L] \\ y_i[1] & y_i[2] & \cdots & y_i[L+1] \\ \vdots & \vdots & \ddots & \vdots \\ y_i[m-L-1] & y_i[m-L] & \cdots & y_i[m-1] \end{bmatrix} \quad for\ i = 1 \,, \dots, n_{sig} \tag{2.10}$$

The Hankel matrices are stacked into one large matrix H.

$$H = \begin{bmatrix} H_1 \\ \vdots \\ H_{n_{sig}} \end{bmatrix} \tag{2.11}$$

The SVD of H is computed, where an economy size SVD suffices.

$$H = U\,\Sigma\,V^T \tag{2.12}$$

The ratio of each singular value σ to the largest singular value σ_{max} is compared to a threshold value θ. Any singular value having ratio $\sigma / \sigma_{max} \geq \theta$ will be kept, while all singular values having ratios less than θ are discarded. The authors in [26] suggest a threshold value $\theta \approx 10^{-3}$, though a threshold value of $\theta = 0.16$ appeared more adequate when tested on power system data. The number of singular values kept thus determines the model order p. If desired the user could specify p, which would eliminate the need for the threshold determination.

The right singular vector matrix has $m-L$ columns, $V = [\, v_{\bullet,1}\, , \ldots , v_{\bullet,m-L}\,]$. Define matrix V' as the first p right singular vectors, $V' = [\, v_{\bullet,1}\, , \ldots , v_{\bullet,p}\,]$. Matrices V_1 and V_2 are formed from offset rows of V'.

$$V_1 = \begin{bmatrix} v'_{1,\bullet} \\ \vdots \\ v'_{p-1,\bullet} \end{bmatrix} \qquad V_2 = \begin{bmatrix} v'_{2,\bullet} \\ \vdots \\ v'_{p,\bullet} \end{bmatrix} \tag{2.13}$$

The discrete-time poles z_i may be found as the generalized eigenvalues of the matrix pair $\{\, V_2^T V_1\, , V_1^T V_1\,\}$, i.e. the eigenvalues of matrix $(V_1^T V_1)^{-1} V_2^T V_1$.

C. Eigensystem Realization Algorithm

Literature on the Eigensystem Realization Algorithm (ERA) can be reviewed in [51] and [52]. The procedure for computing discrete-time poles z_i using ERA for multi-signal modal analysis is presented here. For each signal in $y = [\, y_1\, , \ldots , y_{n_{sig}}\,]$, a Hankel matrix $H_{i,0}$ is created from data-points $y_i[0], \ldots , y_i[m-2]$, and also a shifted Hankel matrix $H_{i,1}$ is created from $y_i[1], \ldots , y_i[m-1]$. Define the parameter N, which is analogous to the pencil parameter in the Matrix Pencil method, as $N = floor(m/2)$.

$$H_{i,0} = \begin{bmatrix} y_i[0] & y_i[1] & \cdots & y_i[N-1] \\ y_i[1] & y_i[2] & \cdots & y_i[N] \\ \vdots & \vdots & \ddots & \vdots \\ y_i[m-N-1] & y_i[m-N] & \cdots & y_i[m-2] \end{bmatrix} \quad for\ i = 1\ , \dots , n_{sig} \tag{2.14}$$

$$H_{i,1} = \begin{bmatrix} y_i[1] & y_i[2] & \cdots & y_i[N] \\ y_i[2] & y_i[3] & \cdots & y_i[N+1] \\ \vdots & \vdots & \ddots & \vdots \\ y_i[m-N] & y_i[m-N+1] & \cdots & y_i[m-1] \end{bmatrix} \quad for\ i = 1\ , \dots , n_{sig}$$

The Hankel matrices are stacked as follows.

$$H_0 = \begin{bmatrix} H_{1,0} \\ \vdots \\ H_{n_{sig},0} \end{bmatrix} \qquad H_1 = \begin{bmatrix} H_{1,1} \\ \vdots \\ H_{n_{sig},1} \end{bmatrix} \tag{2.15}$$

The SVD of H_0 is computed, where an economy size SVD will suffice.

$$H_0 = U\Sigma V^T = \begin{bmatrix} U_p\ U_s \end{bmatrix} \begin{bmatrix} \Sigma_p & 0 \\ 0 & \Sigma_s \end{bmatrix} \begin{bmatrix} V_p^T \\ V_s^T \end{bmatrix} \tag{2.16}$$

The p largest singular values in H_0 are kept. Either the user can predetermine the model order p, or it can be determined by a threshold value as done in the Matrix Pencil method. The state matrix is computed by $A = \Sigma_p^{-1/2}\ U_p^T\ H_1\ V_p\ \Sigma_p^{-1/2}$. The eigenvalues of A are the discrete-time poles z_i.

2.5 Solving for Continuous-time Function $\hat{\boldsymbol{y}}(\boldsymbol{t})$

As mentioned, the traditional modal analysis methods are two-step processes. First the discrete-time poles are estimated, as outlined in the prior section. Once the discrete-time poles are estimated, the residue coefficients b_i are computed. Typically, the residue coefficients are computed in the discrete-time domain by forming a Vandermonde matrix from the discrete-time

poles. However, the computation for residue coefficients can be performed in the continuous-time domain as well, where evenly spaced time-samples are no longer necessary. With the discrete-time poles z_i estimated from one of the traditional modal analysis methods, the continuous-time eigenvalues λ_i can be found by (2.17), for $i = 1 , \dots , p$.

$$\lambda_i = \sigma_i + j\omega_i = \frac{1}{\Delta t}\ln(z_i) \tag{2.17}$$

The remaining computations will be performed in the continuous-time domain. A set of p basis functions consisting of (un)damped cosines, sines and exponentials will be used. Construct the matrix $\Phi(\sigma, \omega)$ which contains the p basis functions, evaluated at all m points in time. Preferably, the matrix $\Phi(\sigma, \omega)$ also has an additional vector of ones to be used as a dc-offset basis function. Recall $n_{\mathbb{C}}$ denotes the number of pairs of complex conjugate modes, $n_{\mathbb{R}}$ denotes the number of real modes, and that $q = n_{\mathbb{C}} + n_{\mathbb{R}}$ and $p = 2n_{\mathbb{C}} + n_{\mathbb{R}}$.

$$\begin{aligned}\Phi(\sigma, \omega) = \big[\, & e^{\sigma_1 t}\cos(\omega_1 t) \, , \, e^{\sigma_1 t}\sin(\omega_1 t) \, , \, \dots \, , \\ & e^{\sigma_{n_{\mathbb{C}}} t}\cos(\omega_{n_{\mathbb{C}}} t) \, , \, e^{\sigma_{n_{\mathbb{C}}} t}\sin(\omega_{n_{\mathbb{C}}} t) \, , \, e^{\sigma_{(n_{\mathbb{C}}+1)} t} \, , \, \dots \, , \, e^{\sigma_q t} \, , \, 1_{m\times 1} \,\big]\end{aligned} \tag{2.18}$$

The $(p+1)$ x 1 vector of residue coefficients $b = [\, b_1 \, , \, \dots \, , \, b_{p+1} \,]^T$ can now be solved for. Consider the continuous-time approximating signal $\hat{y} = \Phi(\sigma, \omega)\, b$. To minimize the least-squares error in $\|y - \hat{y}\|_2^2$, a pseudo-inverse operation is performed in (2.19).

$$b = \Phi(\sigma, \omega)^{\dagger}\, y \tag{2.19}$$

Mode shape amplitudes A_i and phases ϕ_i can be extracted from b, specifically by using the relationship in (2.3), and the approximating function $\hat{y}$ is completed.

2.6 Method Comparison and Example

In this section, an example is used for comparison of the traditional modal analysis methods discussed in this chapter. Power System Analysis Toolbox is used for simulating a fault in the IEEE 14-bus system [53]. A detailed Simulink model of the 14-bus that is used for the dynamic simulation is shown in Figure 2.4. The model has 56 state-variables with six state variables for each of the five synchronous machines, four state variables for each of the five exciters, and three state variables for each of the two turbine governors.

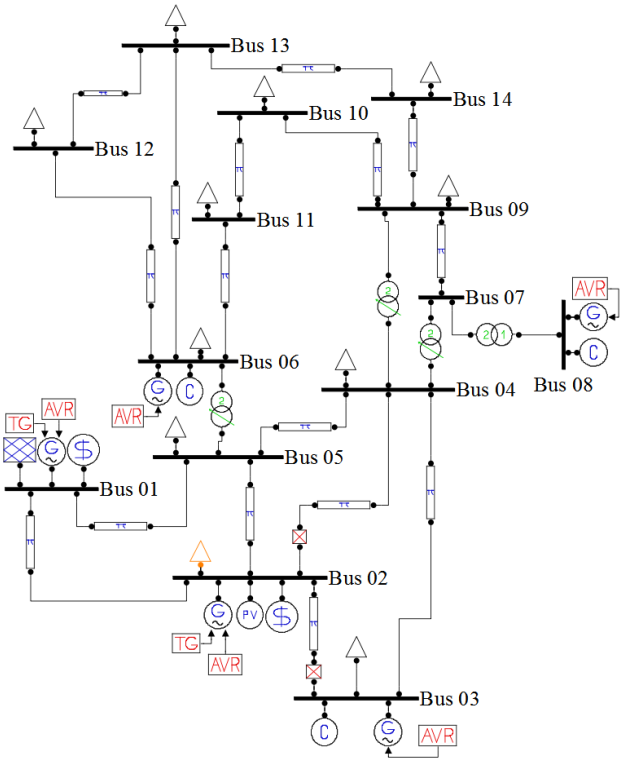

Figure 2.4: IEEE 14-bus Simulink Model

A fault is applied at bus 7 at 1 second and cleared at 1.083 seconds. The dynamic simulation for all 56 state variables is recorded as data to be analyzed using multi-signal Prony analysis and ERA with 9 modes specified along with a dc-offset. The time window for the analysis is from 1.8 to 9 seconds with uniformly spaced data by step-size $\Delta t = 0.01$ seconds. Each signal is weighted by one over its standard deviation to unbias signals of different units and varied magni-

tudes, which is discussed in more detail in Section 3.2-B. Additionally, the weighting helps to visualize signals of comparable magnitudes on the same axes for plotting purposes. In the following figures, the state variables are plotted with the dc-offset of each signal subtracted out, which allows for better viewing of the oscillatory components in the data. The summed error from Prony analysis and ERA to the actual data is also plotted; which is calculated as the summation of errors for similar state variables against time. As seen in the figures, multi-signal ERA provided an obvious closer fit to the actual data compared to Prony analysis. The Matrix Pencil method solutions are not plotted because their solution is nearly exact to that from ERA.

The modes estimated by each method are shown in Table 2.1. The system has 56 actual eigenvalues, and some of the closer matching eigenvalues to the estimated modes are shown in the table. Judging from the following figures and by Table 2.1, it is clear that Prony analysis resulted in poorly estimated modes. Generally, Prony analysis requires a large number of modes to be specified in order to achieve a good fit to the data. Results from Prony analysis are also highly sensitive to the time step-size Δt.

ERA	Matrix Pencil	Actual		Prony
$-0.690 \pm j0.840$	$-0.687 \pm j0.840$	$-0.614 \pm j0.690$		$-51.7 \pm j69.19$
$-1.036 \pm j1.346$	$-1.035 \pm j1.347$	$-1.074 \pm j1.388$		$-36.9 \pm j141.93$
$-1.865 \pm j7.796$	$-1.865 \pm j7.797$	$-2.221 \pm j8.293$		$-0.888 \pm j8.631$
$-0.844 \pm j9.201$	$-0.844 \pm j9.201$	$-0.921 \pm j9.244$		$-28.2 \pm j222.79$
-0.698	-0.693	-0.857		
		-1.000		-0.973

Table 2.1: Comparison of estimated modes and actual system eigenvalues

The computation time for the three methods is shown in Table 2.2, with the evaluations performed in MATLAB R2011a on a computer with 64-bit Intel i5-560M Dual Core CPU at 2.67 GHz with 4 GB of RAM. Prony analysis is the fastest computation of the methods discussed in this chapter. The computation speed comes as a tradeoff against solution accuracy compared to matrix pencil and ERA.

	ERA	Matrix Pencil	Prony
Time [sec]	6.324	4.646	0.0782

Table 2.2: Method evaluation times for example

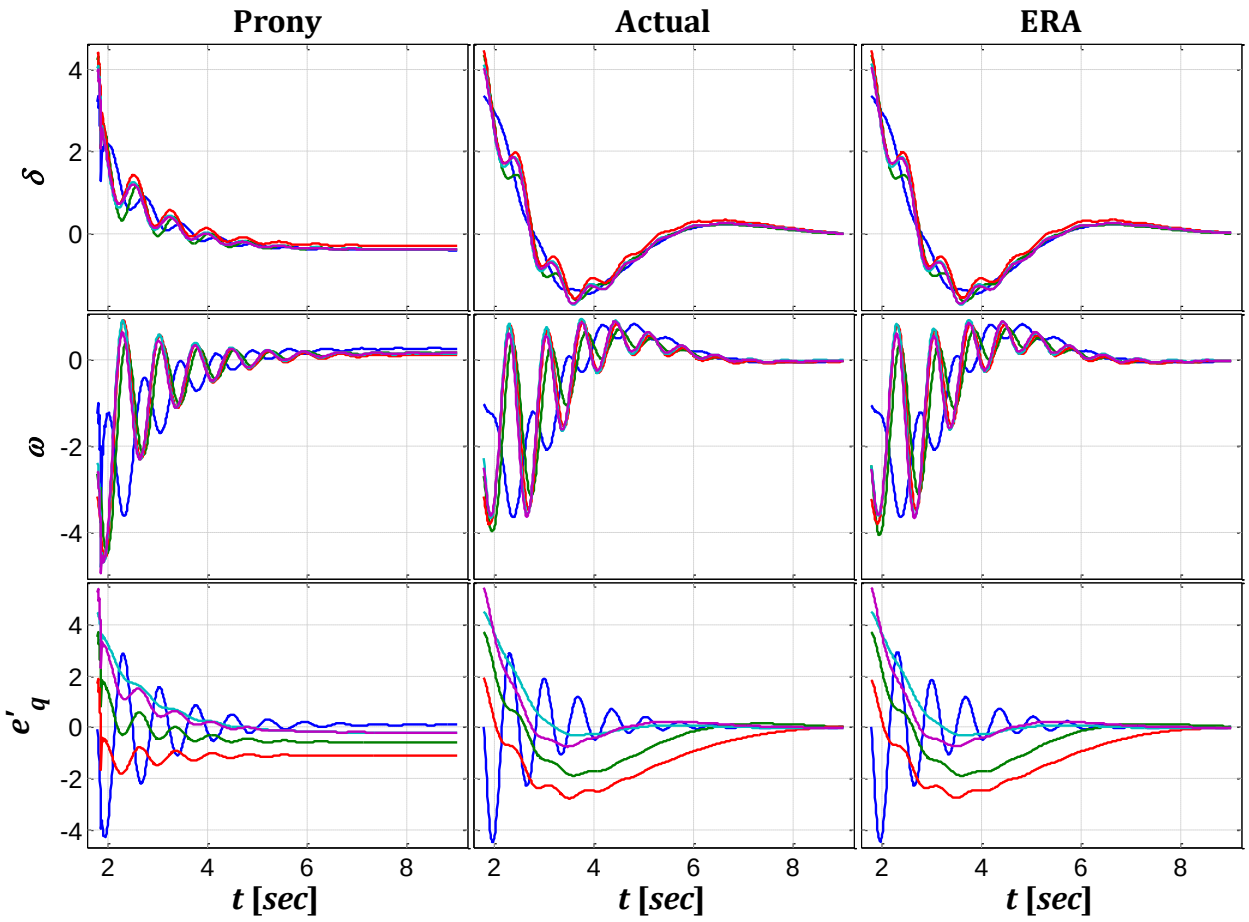

Figure 2.5: Simulated and estimated state variables δ, ω and e'_q

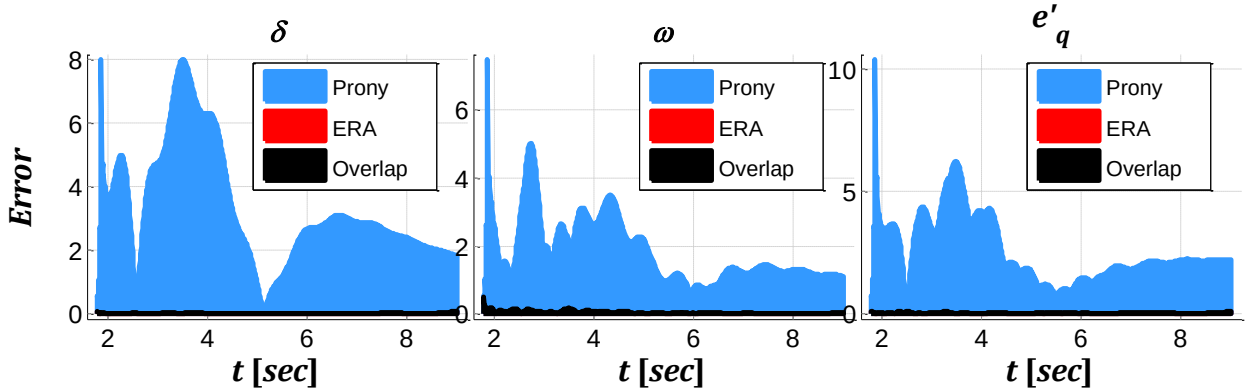

Figure 2.6: Estimation error for state variables δ, ω and e'_q

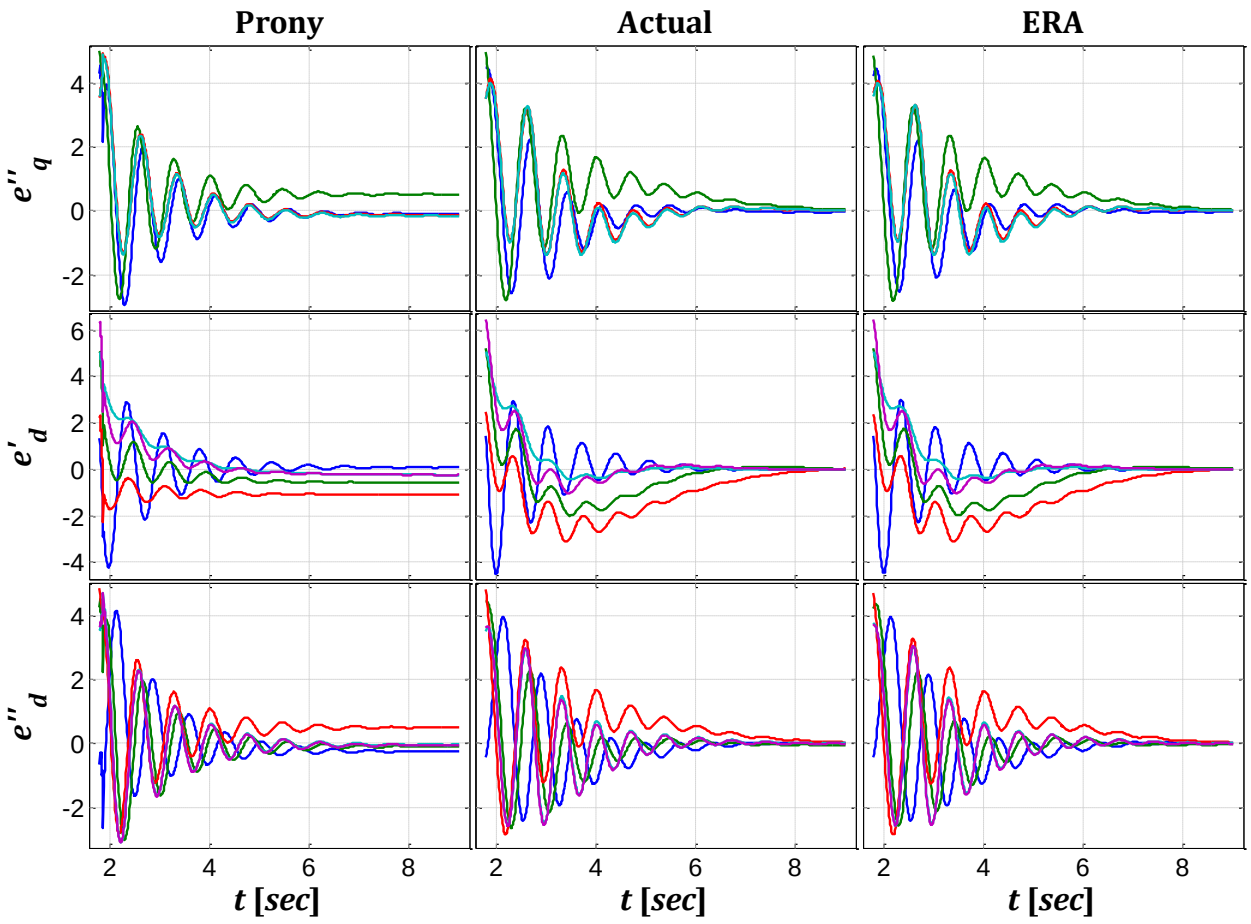

Figure 2.7: Simulated and estimated state variables e''_q, e'_d and e''_d

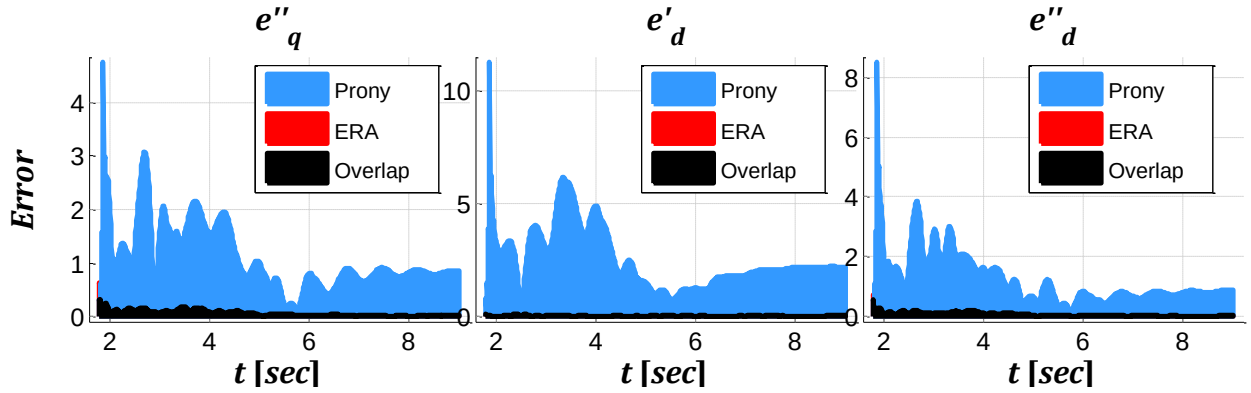

Figure 2.8: Estimation error for state variables e''_q, e'_d and e''_d

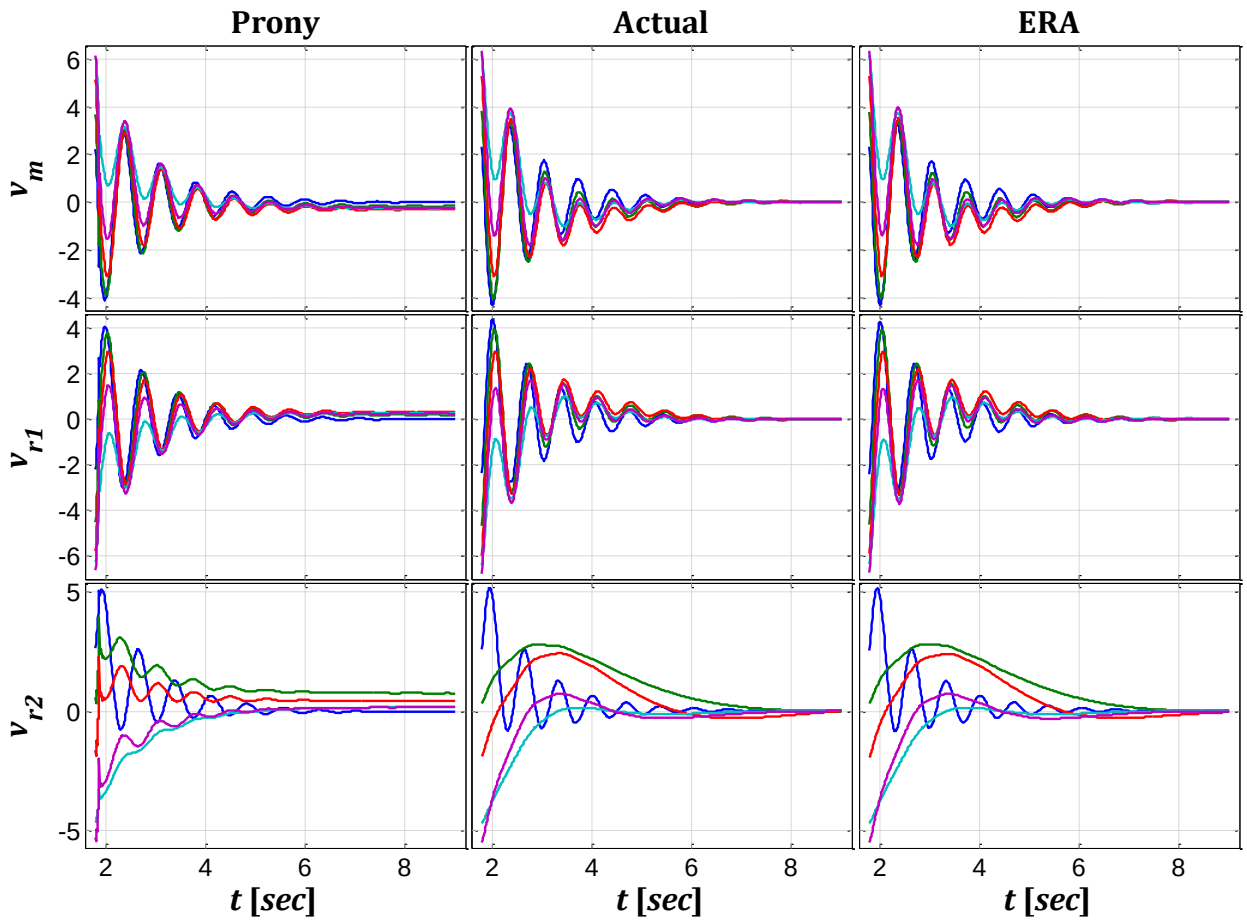

Figure 2.9: Simulated and estimated state variables v_m, v_{r1} and v_{r2}

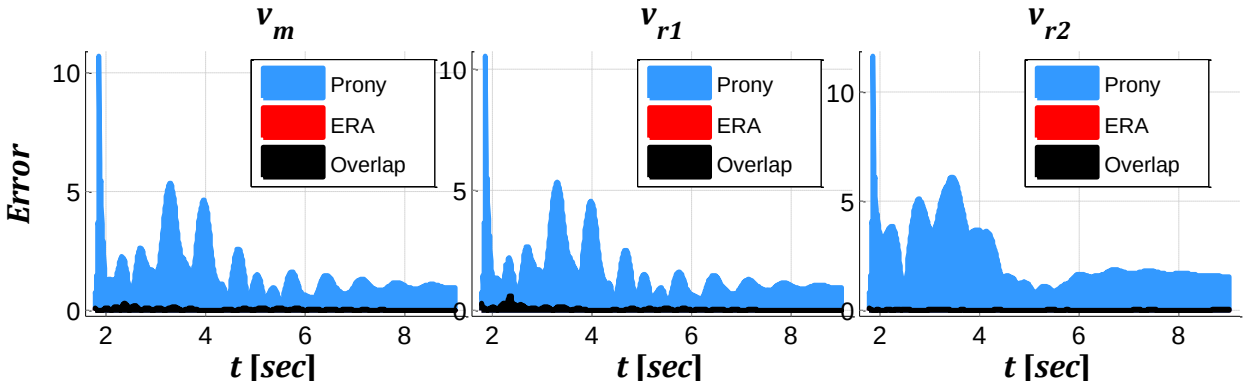

Figure 2.10: Estimation error for state variables v_m, v_{r1} and v_{r2}

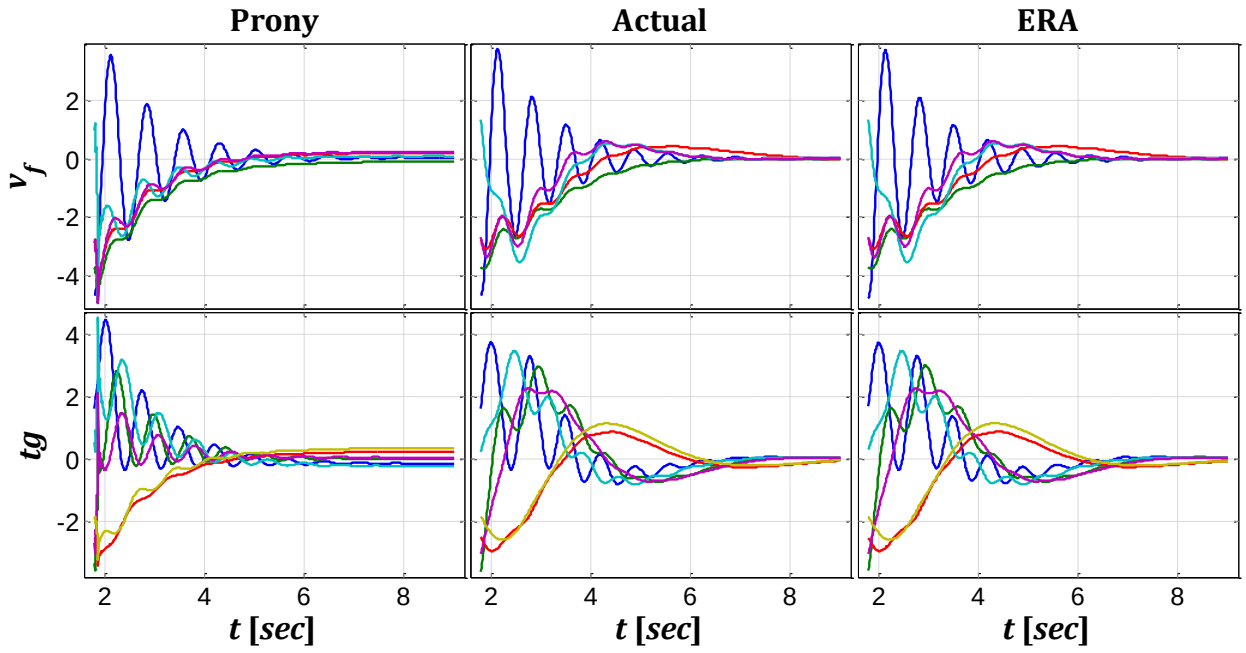

Figure 2.11: Simulated and estimated state variable v_f and turbine governor state variables

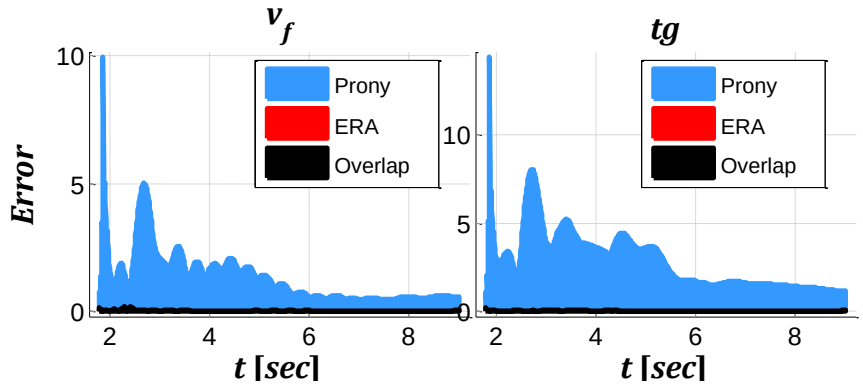

Figure 2.12: Estimation error for state variable v_f and turbine governor state variables

2.7 Mode Energy

It is often useful to determine which modes appear most dominantly in the data. One suggested approach for assessing the significance of each mode is by calculating the mode energy. Consider a set of p modes, with $n_{\mathbb{C}}$ number of pairs of complex conjugate modes and $n_{\mathbb{R}}$ number of real modes, such that $q = n_{\mathbb{C}} + n_{\mathbb{R}}$ and $p = 2n_{\mathbb{C}} + n_{\mathbb{R}}$. Consider the j^{th} signal $\hat{y}_j(t)$ in (2.20) for multi-signal analysis with n_{sig} total number of signals.

$$\hat{y}_j(t) = \sum_{i=1}^{q} A_{i,j}\, e^{\sigma_i t}\, \cos\left(\omega_i t + \phi_{i,j}\right) \quad for\ j = 1, \dots, n_{sig} \tag{2.20}$$

The mode energy is the integral of each modes contribution to the j^{th} signal squared, $\hat{y}_j(t)^2$, from the start time $t_{[0]} = 0$ to stop time $t_{[m-1]}$ in the data, summed across all n_{sig} signals. Therefore the mode energy of the i^{th} complex conjugate mode is that shown in (2.21a) and the mode energy of the i^{th} real valued mode is that shown in (2.21b).

$$i^{th}\ complex\ mode\ energy\ = \sum_{j=1}^{n_{sig}} \int_0^{t[m-1]} \left[A_{i,j}\, e^{\sigma_i t}\, \cos\left(\omega_i t + \phi_{i,j}\right)\right]^2 dt \tag{2.21a}$$

$$= \sum_{j=1}^{n_{sig}} \frac{1}{4}\frac{A_{i,j}^2}{\sigma_i^3 + \sigma_i \omega_i^2} \left[e^{2\sigma_i t[m-1]} \begin{bmatrix} \omega_i^2 + \sigma_i^2 \cos^2\left(\omega_i t[m-1] + \phi_{i,j}\right) + \\ 2\sigma_i\omega_i \cos\left(\omega_i t[m-1] + \phi_{i,j}\right) \sin\left(\omega_i t[m-1] + \phi_{i,j}\right) \end{bmatrix} \right.$$
$$\left. -2\sigma_i^2 \cos^2 \phi_{i,j} - 2\sigma_i\omega_i \cos\phi_{i,j} \sin\phi_{i,j} - \omega_i^2 \right]$$

$$i^{th}\ real\ mode\ energy\ = \sum_{j=1}^{n_{sig}} \int_0^{t[m-1]} \left[A_{i,j}\, e^{\sigma_i t}\right]^2 dt\ = \sum_{j=1}^{n_{sig}} \frac{1}{2}\frac{A_{i,j}}{\sigma_i}\left(e^{2\sigma_i t[m-1]} - 1\right) \tag{2.21b}$$

The metric of mode energy can be used for ranking the dominance of estimated modes, with high energy implying dominance. It is suggested the mode energies are calculated for the signals weighted by one over their standard deviations. This is logical because many of the signals are of different units. Calculating mode energies for unweighted signals would tend to favor the modes most prevalent in signals with large numerical units. The energies of the modes estimated by ERA, from the example in Section 2.6, can be calculated using (2.21a) and (2.21b). They are ranked in Table 2.3.

Frequency $[Hz]$	Damping $[1/s]$ & Frequency $[rad/s]$	Mode Energy
0.134	$-0.690 \pm j0.840$	338.33
0.214	$-1.036 \pm j1.346$	278.60
–	-0.698	244.41
1.464	$-0.844 \pm j9.201$	133.54
1.241	$-1.865 \pm j7.796$	60.89

Table 2.3: Energy of modes estimated by ERA

From Table 2.3, it can be seen that the two low frequency modes and the single real valued mode have the highest energies. These modes create the slow shaping of the waveforms in Figures 2.5-2.12. These slower modes provide more of a trending fit to the data, and often don't complete a single cycle before damping out. In general, the fast oscillating modes are of more interest to power system grid operators. From that perspective, the high frequency mode at 1.464 Hz is seen oscillating most predominantly in the figures; therefore this mode would be of most concern.

2.8 Conclusion

In testing the three linear methods, it was observed that ERA and matrix pencil performed most reliably, with ERA typically achieving the minimum least-square error most of the time. Prony analysis especially was shown to have shortcomings when analyzing highly nonlinear data. Comparatively, the Matrix Pencil method and Eigensystem Realization Algorithm provide a much better fit to the data, but at the expense of longer computational speeds. Prony analysis requires a large number of modes to be specified to obtain a proper fit to the data. To its benefit, Prony analysis is a relatively instantaneous computation and may therefore be better suited for real-time applications where computation speed is important.

It is observed that the measurement-based modal analysis methods discussed in this chapter do not directly minimize the least-square error to the data. It is reasonable to argue that least-square error is a valid metric for assessing the accuracy of the modal analysis methods. Assuming that least-squares error to data positively correlates to accurate mode estimations, then another approach exists that is superior for estimating modal content. A nonlinear least-squares approach, called the variable projection method, can be used for obtaining an optimal fit to the data. The nonlinear least-squares approach is presented next in Chapter 3.

Chapter 3

Nonlinear Least-Squares

This chapter details a nonlinear least-squares approach to modal analysis problems. It can be shown that the two-step modal analysis procedures in Chapter 2 are not optimal in the least-squares sense, and the methods have shortcomings when approximating highly-nonlinear dynamic disturbance data. In contrast, the nonlinear least-squares approach directly matches a model to data without directly imposing an underlying linear system assumption and optimally captures the dynamics present in the data.

The nonlinear least-squares variable projection method was originated in 1973 by Golub and Pereyra [27,28]. They observed that in many nonlinear least-squares problems there are optimization variables that appear nonlinearly and optimization variables that appear linearly. It was observed that the linear variables have a dependent relationship corresponding to the optimal solution for the nonlinear variables. Golub and Pereyra identified the separability between the two variable types and called these problems *separable least-squares problems*. They obtained the *variable projection functional* by eliminating the linear variables from the optimization problem. In addition, they outlined the gradient equation of the variable projection functional; allowing for the use of gradient based line-search and trust region optimization methods.

In 1977, Bolstad released a FORTRAN implementation of the variable projection method with the varpro.f code. Since then, the code and method have been improved, with a thorough examination of contributions presented in [54] and [56]. A third-party MATLAB implementation is available in the varpro.m code [55], which has found its use among a wide area of disciplines.

This chapter motivates the use of the variable projection method for use in power system problems that have traditionally relied on Prony analysis or other methods for estimating modal content of data. An example disturbance event is analyzed with results suggesting the nonlinear least-squares approach is superior.

3.1 Variable Projection Method

There is existing literature detailing the variable projection method and least-squares methods [27–30]. An overview of the method is given in this section. Consider a data waveform denoted as y having m data points over the time interval t, with starting time at 0 seconds. Referring to the methods in Chapter 2, observe that the solution for residue coefficients, $b = \Phi(\sigma, \omega)^{\dagger} y$, is entirely dependent on the discrete-time poles z_i and damping σ and frequency ω. Therefore the residual error $\|y - \hat{y}\|_2^2$ is not optimally minimized in the two-step process, as it is biased by the previous solution for damping and frequencies. The optimal solution can be obtained by simultaneously solving for σ, ω and b with a nonlinear optimization method. In relation to the variable projection method, the damping and frequencies, σ and ω, are the independent nonlinear parameters and the coefficients b are the dependent linear parameters. These variables types are separable, which will be shown next.

To begin the method, define the set of parameters $\alpha = [\, \alpha_1 \, , \ldots , \alpha_p \,]$. Define the set of n basis function vectors $\phi_1(\alpha) \, , \ldots , \phi_n(\alpha)$, which are nonlinear functions of α evaluated at all m points in time. An approximating function $\hat{y}(\alpha)$ can be constructed that is composed of a summation of the n basis functions $\phi_i(\alpha)$.

$$\hat{y}(\alpha) = \sum_{i=1}^{n} b_i \, \phi_i(\alpha) \tag{3.1}$$

In (3.1), $\alpha_1 \, , \ldots , \alpha_p$ denote the nonlinear optimization variables and b_i denotes the linear optimization variables which are to be eliminated from the problem. For modal analysis problems, complex-valued modes are enforced with basis functions as pairs of damped cosines and sines, $\phi_i(\alpha) = e^{\alpha_i t} \cos(\alpha_{i+1} \, t)$ and $\phi_{i+1}(\alpha) = e^{\alpha_i t} \sin(\alpha_{i+1} \, t)$. Real-valued modes are enforced with exponentials $\phi_i(\alpha) = e^{\alpha_i t}$. With the basis functions $\phi_1(\alpha), \ldots , \phi_n(\alpha)$ defined, construct the $m \times n$ matrix $\Phi(\alpha) = [\, \phi_1(\alpha) \, , \ldots , \phi_n(\alpha) \,]$, such that $\hat{y}(\alpha) = \Phi(\alpha) \, b$.

The residual vector $r(\alpha, b) = y - \hat{y}(\alpha)$, for a given α, can be expressed as a linear least-squares optimization problem solving for b, as in (3.2).

$$\min_{b|\alpha} \frac{1}{2} \left\| r(\alpha, b) \right\|_2^2 = \min_{b|\alpha} \frac{1}{2} \left\| y - \Phi(\alpha)\, b \right\|_2^2 \tag{3.2}$$

For a given α, the least-squares error in the residual vector is minimized by $b = \Phi(\alpha)^{\dagger} y$, where † denotes the pseudoinverse. Substituting $b = \Phi(\alpha)^{\dagger} y$ recasts the linear least-squares problem in (3.2) as a nonlinear least-squares optimization problem, solving for parameters α, as in (3.3).

$$\min_{\alpha} \frac{1}{2} \left\| r(\alpha) \right\|_2^2 = \min_{\alpha} \frac{1}{2} \left\| (I - \Phi(\alpha)\Phi(\alpha)^{\dagger})\, y \right\|_2^2 \tag{3.3}$$

In (3.3), $\frac{1}{2} \left\| r(\alpha) \right\|_2^2$ was termed the *variable projection functional* by Golub and Pereyra [27,28]. For modal analysis problems, the solution to (3.3) gives the optimal fit to the data-set y for a predetermined number of modes p. The method is not limited to uniformly spaced time-samples, as analysis is done completely in the continuous-time domain. The variable projection functional's gradient equation can also be determined as shown in (3.4), where the i^{th} element equals $\frac{\partial}{\partial \alpha_i}\left(\frac{1}{2}\left\| r(\alpha) \right\|_2^2\right)$ for $i = 1, \ldots, p$.

$$\nabla \frac{1}{2} \left\| r(\alpha) \right\|_2^2 = J^T r(\alpha) \tag{3.4}$$

The gradient can thus be determined once the $m \times p$ Jacobian matrix J is calculated, where $J = \left[\frac{\partial r(\alpha)}{\partial \alpha_1} \ \ldots \ \frac{\partial r(\alpha)}{\partial \alpha_p} \right]$ and its j^{th} column $J_{\bullet,j} = \frac{\partial r(\alpha)}{\partial \alpha_j}$.

$$\frac{\partial r(\alpha)}{\partial \alpha_j} = -\left[\frac{\partial \Phi(\alpha)}{\partial \alpha_j} \Phi(\alpha)^{\dagger} + \Phi(\alpha) \frac{\partial \Phi(\alpha)^{\dagger}}{\partial \alpha_j} \right] y \tag{3.5}$$

Golub and Pereyra showed $\frac{\partial r(\alpha)}{\partial \alpha_j}$ can be rewritten as in (3.6) [27,28], where $P^{\perp} = I - \Phi(\alpha)\, \Phi(\alpha)^{\dagger}$.

$$\frac{\partial r(\alpha)}{\partial \alpha_j} = -\left[\left(P^{\perp}\frac{\partial \Phi(\alpha)}{\partial \alpha_j}\Phi(\alpha)^{\dagger}\right) + \left(P^{\perp}\frac{\partial \Phi(\alpha)}{\partial \alpha_j}\Phi(\alpha)^{\dagger}\right)^{T}\right] y \tag{3.6}$$

For ease of notation the parameter α will be dropped from $\Phi(\alpha)$. The full Moore-Penrose pseudoinverse $\Phi^{\dagger}$ is not necessary for the calculation; instead the symmetric generalized inverse Φ^{-} satisfying $\Phi\Phi^{-}\Phi = \Phi$ and $P = (\Phi\Phi^{-})^{T} = \Phi\Phi^{-}$ suffices. Denote the projector on the orthogonal complement of the column space as $P^{\perp} = I - P$. The conversion between (3.5) and (3.6) is proven using the following three observations [27,28].

1) $$P\Phi = \Phi \quad \text{by} \quad \Phi\Phi^{-}\Phi = \Phi$$

$$\frac{\partial P\Phi}{\partial \alpha_j} = \frac{\partial P}{\partial \alpha_j}\Phi + P\frac{\partial \Phi}{\partial \alpha_j} = \frac{\partial \Phi}{\partial \alpha_j}$$

$$\Longrightarrow \frac{\partial P}{\partial \alpha_j}\Phi = \frac{\partial \Phi}{\partial \alpha_j} - P\frac{\partial \Phi}{\partial \alpha_j} = (I-P)\frac{\partial \Phi}{\partial \alpha_j} = P^{\perp}\frac{\partial \Phi}{\partial \alpha_j}$$

$$\therefore \frac{\partial P}{\partial \alpha_j}P = \frac{\partial P}{\partial \alpha_j}\Phi\Phi^{-} = P^{\perp}\frac{\partial \Phi}{\partial \alpha_j}\Phi^{-}$$

2) $$\left(\frac{\partial P}{\partial \alpha_j}P\right)^{T} = P^{T}\frac{\partial P^{T}}{\partial \alpha_j} = P\frac{\partial P}{\partial \alpha_j}$$

$$\text{by} \quad P = (\Phi\Phi^{-})^{T} = \Phi\Phi^{-}$$

3) $$P = P^{2}$$

$$\Phi\Phi^{-} = (\Phi\Phi^{-})(\Phi\Phi^{-})$$

$$\text{by} \quad \Phi\Phi^{-}\Phi = \Phi$$

Therefore,

$$\frac{\partial P}{\partial \alpha_j} = \frac{\partial P^{2}}{\partial \alpha_j} = \frac{\partial P}{\partial \alpha_j}P + P\frac{\partial P}{\partial \alpha_j}$$

$$= \frac{\partial P}{\partial \alpha_j}P + \left(\frac{\partial P}{\partial \alpha_j}P\right)^{T}$$

$$= \left(P^{\perp}\frac{\partial \Phi}{\partial \alpha_j}\Phi^{-}\right) + \left(P^{\perp}\frac{\partial \Phi}{\partial \alpha_j}\Phi^{-}\right)^{T}$$

$$J_{\bullet,j} = \frac{\partial r(\alpha)}{\partial \alpha_j} = \frac{\partial}{\partial \alpha_j}(I - P)y = -\frac{\partial P}{\partial \alpha_j}y$$
$$= -\left[\left(P^{\perp}\frac{\partial \Phi}{\partial \alpha_j}\Phi^{-}\right) + \left(P^{\perp}\frac{\partial \Phi}{\partial \alpha_j}\Phi^{-}\right)^{T}\right]y$$

With the Jacobian matrix J calculated the gradient of the variable projection functional can be determined by (3.4) [27,28]. By having a closed-form expression for the gradient $\nabla \frac{1}{2} \| r(\alpha) \|_2^2$, a line-search or trust region method can be employed to determine the optimal α values in the solution to (3.3). A singular value decomposition (SVD) is required to compute the pseudoinverse $\Phi^{\dagger}$. True computational savings are achieved because $\Phi^{\dagger}$ shows up in both the cost function (3.3) and the gradient equation (3.5). Therefore, after the SVD computation, $\Phi^{\dagger}$ can be stored and reused for calculating both the cost and gradient.

A well-known singular value decomposition trick can be used to ensure faster solve-time during the optimization. An economy size SVD can be used to determine the pseudoinverse $\Phi(\alpha)^{\dagger}$, where $\Phi(\alpha)$ is an $m \times n$ matrix. Typically $m > n$, with m being the number of time samples and n being the number of basis functions. The left singular vectors comprise the $m \times n$ matrix U. With an economy size SVD only the first n columns of the orthogonal right singular vector matrix V are calculated, where Σ is an $n \times n$ matrix.

$$\Phi(\alpha) = U\,\Sigma\,V^{T}$$
$$\Phi(\alpha)^{\dagger} = V\,\Sigma^{-1}\,U^{T}$$
$$\Phi(\alpha)\,\Phi(\alpha)^{\dagger} = U\,U^{T}$$

Therefore $P^{\perp} = I - \Phi(\alpha)\,\Phi(\alpha)^{\dagger} = I - U\,U^{T}$ can be used for faster calculation speed in (3.6).

3.2 Additional Features

The variable projection method is robust, though the quality of solution depends on several conditions supplied by the user. The method also give the user valuable degrees of flexibility, as

will be described in the following subsections. Specific details on solver and computation details are provided in Chapter 5.

A. Initial Conditions and Model Order Selection

Many nonlinear optimization solvers require initial conditions to be specified in order to begin a line-search or trust region method [57]. Properly selected initial conditions are critical for fast solve-times and for improved likelihood of finding a global minimizer rather than a local minimizer. Potential initial conditions can be obtained by multi-signal Prony analysis, matrix pencil or ERA as outlined in Chapter 2.

The variable projection method allows for some flexibility that is not offered by traditional modal analysis methods. The user not only determines the number of modes to be estimated, but he or she may also determine the exact number of complex-conjugate and real-valued modes estimated. The set of basis functions in $\Phi(\alpha)$ is altered depending on the number of complex and real valued modes, which can be seen in (2.18) from Section 2.5. With traditional modal analysis methods, the user is stuck with however many number of complex and real-valued modes that the method predicts. In many cases, a better approximation to the data can be obtained by adjusting the number of complex and real-valued modes.

B. Multi-Signal Analysis

Often times it is necessary to analyze multiple signals simultaneously [24]. For multi-signal analysis the objective function in (3.3) is modified. The summed least-squares error over the number of signals becomes the objective function as shown in (3.7). To avoid biasing signals of varying magnitude, each signal is multiplied by a scaling multiplier β_i. One possible multiplier is $\beta_i = 1/\sigma_i$, where σ_i is the standard deviation of each signal. By weighting each signal by one over its standard deviation, each of the weighted signals will have standard deviation equal to 1. Alternatively, the signals could be grouped in classes according to their units, where each signal in a class is scaled equivalently. This approach would ensure signals of different units are

comparable in numerical magnitude, while preserving numerical deviations of signals in the same classes comparatively. Selecting appropriate scaling multipliers β_i is an important issue, as changes to the cost function in (3.7) have significant impacts on optimization results.

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{n_{sig}} \left\| \beta_i \left(I - \Phi(\alpha)\Phi(\alpha)^{\dagger} \right) y_i \right\|_2^2 \tag{3.7}$$

If desired, the scaling multiplier could be a function of time, $\beta_i(t)$, if there are temporal regions of the signal of more concern then others. The multiple signals belong to the same system, so it is assumed they share the same natural modes in α. Consequently, no additional optimization variables are introduced for multi-signal analysis.

C. Data-Detrending

In power system problems there is often a need to detrend the data [58]. For example, there is an obvious need for first-order detrending when considering instantaneous bus angle data like that seen in Figure 3.1, as opposed to comparative angles to a reference bus. Figure 3.1 is actual bus angle data from a real disturbance event.

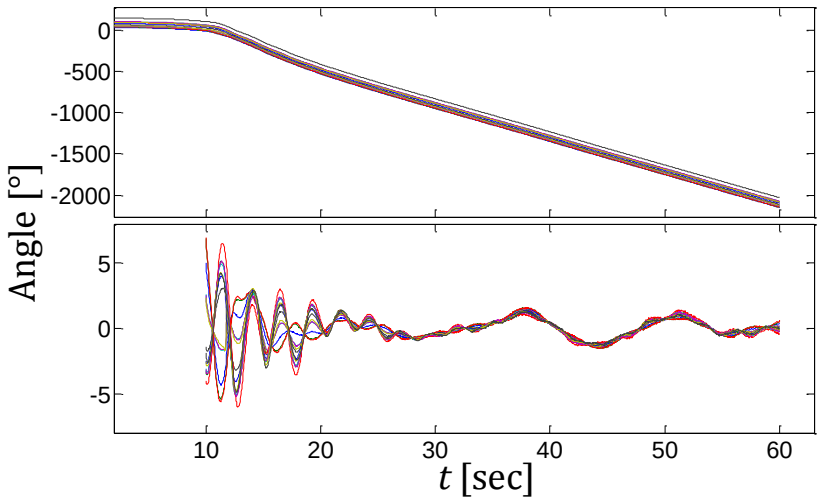

Figure 3.1: Bus angle disturbance data
(a) without trending removed (b) with trending removed

In Figure 3.1, a disturbance occurs around 10 seconds. Before the disturbance, the grid frequency remains steady, and the bus angles remain constant relative to 60 Hz synchronous speed. After the disturbance, the grid frequency decreases, and the bus angles monotonically decrease relative to 60 Hz synchronous speed. At first glance, Figure 3.1 (a) appears to show no oscillatory components; however, the oscillations are far more evident after subtracting out a linear curve and trending from real-valued modes, shown in Figure 3.1 (b).

Fortunately, the variable projection method allows for the inclusion of any possible basis function, including polynomial basis functions for data-detrending. Example polynomial trending basis functions could be linear $\phi_i(\alpha) = t$, quadratic $\phi_i(\alpha) = t^2$, or a constant dc-offset $\phi_i(\alpha) = 1$. The linear coefficients b_i that multiply the trending basis functions are dynamically optimized as the nonlinear program iterates toward a local or global minimizer.

D. Constrained Optimization

The variable projection method as presented thus far, is an unconstrained nonlinear optimization problem. The optimization will never be unbounded due to the least-squares objective function, and it will always be feasible due to the absence of constraints. If desired however, constraints can be imposed on the optimization problem. The most basic constraints would be lower and upper bounds imposed on the damping σ and frequency ω variables, which are in units of $1/s$ and rad/s respectively. However, power system communities interested in oscillation analysis studies, prefer to express frequency (f) in Hertz and damping as the unit-less percent damping ratio.

$$f = {}^{\omega}/_{2\pi}\ [Hz] \tag{3.8a}$$

$$\%\ damping = -100\frac{\sigma}{\sqrt{\sigma^2 + \omega^2}} \tag{3.8b}$$

Therefore, if lower and upper bounds are placed on frequency and percent damping, then an additional step is needed to convert these constraints to $1/s$ and rad/s. Bounds on frequency f $[Hz]$ are easily converted to bounds on ω $[rad/s]$ via (3.8a). However, converting bounds on

percent damping to constraints on σ and ω is more complicated. Consider placing lower and upper bounds, $\%d_{lb}$ and $\%d_{ub}$ on percent damping as shown below.

$$\%d_{lb} \leq -100\frac{\sigma}{\sqrt{\sigma^2+\omega^2}} \leq \%d_{ub}$$

If the frequency is greater than or equals to zero, then the bounds on percent damping can equivalently be expressed as the following linear inequality constraints.

$$\sigma + \frac{\%d_{lb}}{\sqrt{100^2-(\%d_{lb})^2}}\omega \leq 0$$

$$-\sigma - \frac{\%d_{ub}}{\sqrt{100^2-(\%d_{ub})^2}}\omega \leq 0$$

The practical implementation of the constrained optimization is discussed in more detail in Section 5.1.

3.3 Example

In this section, the nonlinear least-squares variable projection method is compared against results from ERA in a modal analysis example. A 16-machine 68-bus test system is used which is shown in Figure 3.2 [59], which is a simplified New England/ New York interconnected system [60]. The system has 235 state variables. The 16 synchronous machines are represented by sixth-order dynamical models, and a first-order exciter model for each machine. There are 29 induction motor loads represented by third-order dynamical models. Lastly, there are 12 power system stabilizers represented by third-order dynamical models. The dynamic response of all 235 state variables is simulated using Power System Toolbox [61].

At time t=0.1 seconds, a three-phase fault is applied at bus-1, with bus-30 at the far end of the line. The fault is cleared at bus-2 at t=0.15 seconds, and is cleared at bus-38 at t=0.16 seconds. The simulated response of all 235 state variables is collected and is shown in Figure 3.3. In Figure 3.3, the top subplot is the unaltered state variable data from the disturbance simulation. The middle subplot is the data with dc-offsets subtracted for each signal. The time-frame used in

the analysis is from t=0.8 to t=8 seconds shown in the bottom subplot, with a time-step Δt=0.01 seconds. The bottom subplot shows the data with dc-offsets subtracted and also weighted by the inverse of their standard deviations β_i. The preconditioned data in the bottom subplot is input to ERA for analysis.

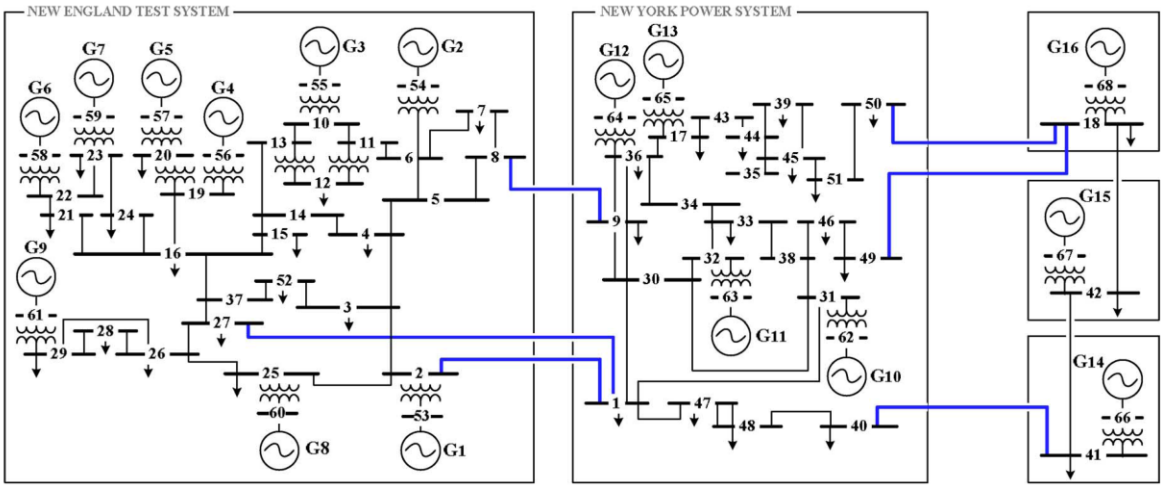

Figure 3.2: New England/ New York 16-machine 68-bus test system

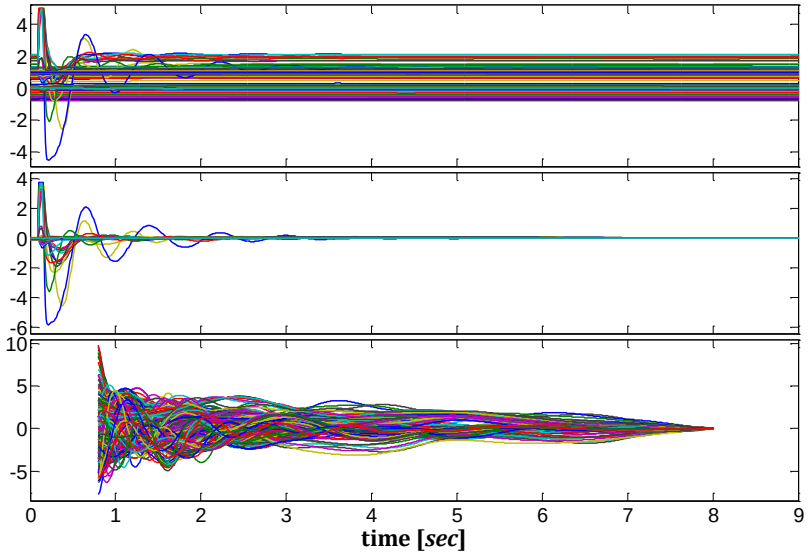

Figure 3.3: 16-machine 68-bus test system disturbance
(a) Unaltered disturbance data
(b) Data with dc-offsets subtracted
(c) Data weighted by inverse standard deviation and with dc-offsets subtracted

The weighted signals are input to multi-signal ERA with the number of modes specified as $p=11$, where one of the modes will approximate a zero eigenvalue used as a dc-offset. The solution from ERA is then used as initial conditions for multi-signal variable projection. The total least-squares error obtained by ERA was 6246.02, and the error obtained by variable projection was 1886.11. Table 3.4 breaks down the contributing errors by state variable type, where each signal is weighted by its inverse standard deviation β_i, and where $r_i(\alpha)$ is the residual error for the i^{th} signal.

	Signals	Var. Proj.	ERA
Synchronous Generator (16)	δ	60.49	563.97
	ω	285.37	407.07
	E_d'	172.17	359.11
	E_q'	184.63	338.89
	ψ_k	192.03	369.76
	ψ_{kq}	207.51	293.60
Exciter (16)	E_{fd}	393.53	419.78
Induction Motor (29)	V_d'	105.67	740.68
	V_q'	20.69	1257.91
	$slip$	123.36	804.06
Power System Stabilizer (12)	$pss1$	8.55	183.24
	$pss2$	67.86	257.98
	$pss3$	64.26	249.97
$\sum_{i=1}^{235} \frac{1}{2} \left\| \beta_i r_i(\alpha) \right\|_2^2$		1886.11	6246.02

Table 3.4: Weighted error per signal type

In Table 3.4, the error per signal type was calculated by summing each signals' contributions to total error $\sum_{i=1}^{235} \frac{1}{2} \left\| \beta_i r_i(\alpha) \right\|_2^2$, for signals of the same state variable type. For example, variable projection gave lesser least-squares error compared to ERA for the synchronous machine angles δ. There are 16 of the angles δ. Each angle is weighted by β_i and the summation of

the 16 signal contributions to $\sum_{i=1}^{235} \frac{1}{2} \left\| \beta_i \, r_i(\alpha) \right\|_2^2$ equals 60.49 for variable projection and 563.97 for ERA. With this analysis, variable projection provided a lower least-squares error compared to ERA for all state variable types.

The following figures display the weighted estimated solutions from variable projection and ERA compared to the actual data. For ease of display, each signal has its dc-offset subtracted to view oscillatory content better. The differences between the estimated solutions and actual data are subtle, therefore plots of the (absolute value) error between actual and estimated signals versus time are also shown to emphasize the differences.

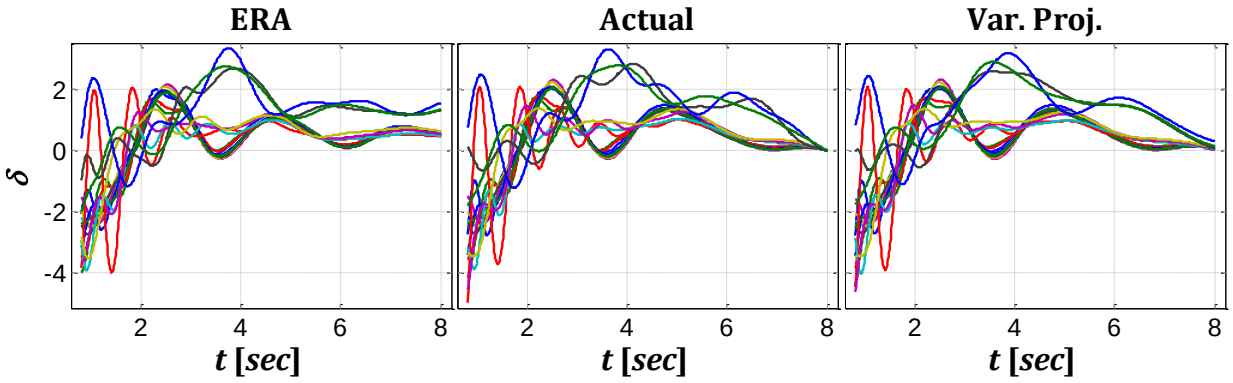

Figure 3.4: Simulated and estimated state variables δ

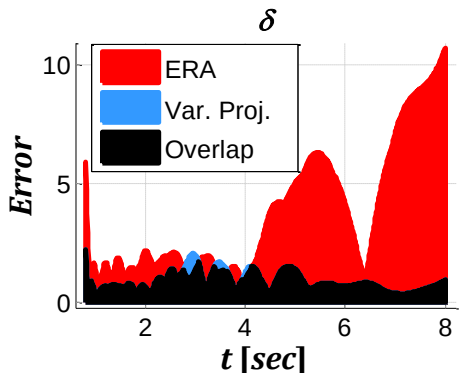

Figure 3.5: Estimation error for state variables δ

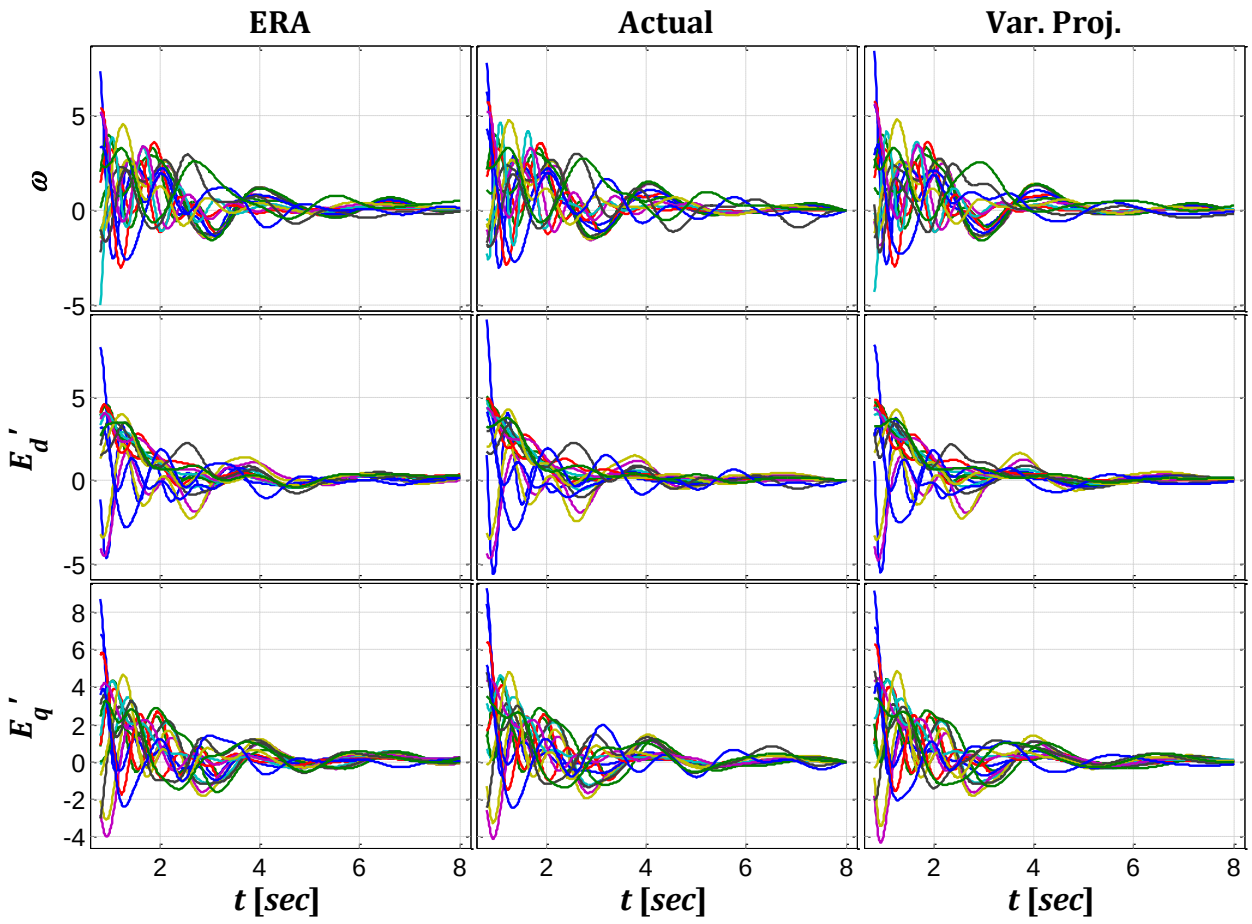

Figure 3.6: Simulated and estimated state variables ω, E'_d and E'_q

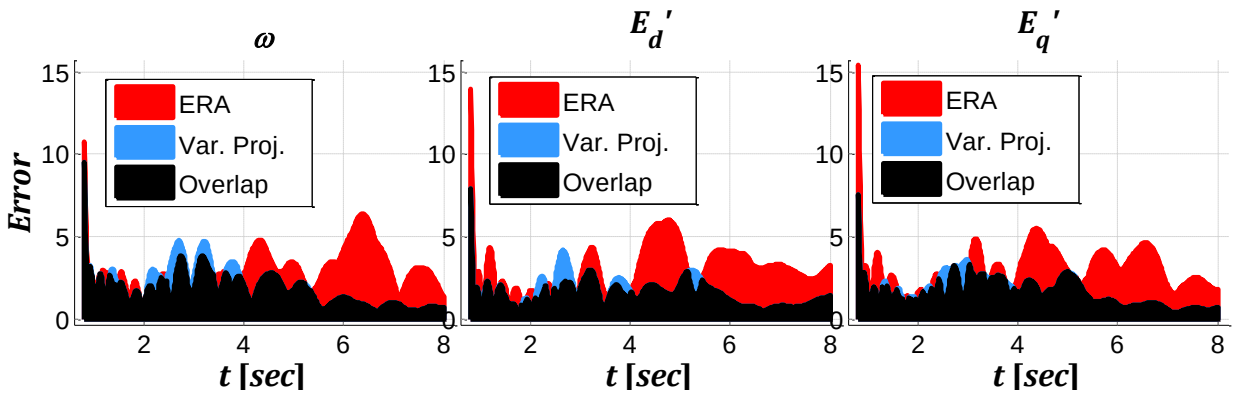

Figure 3.7: Estimation error for state variables ω, E'_d and E'_q

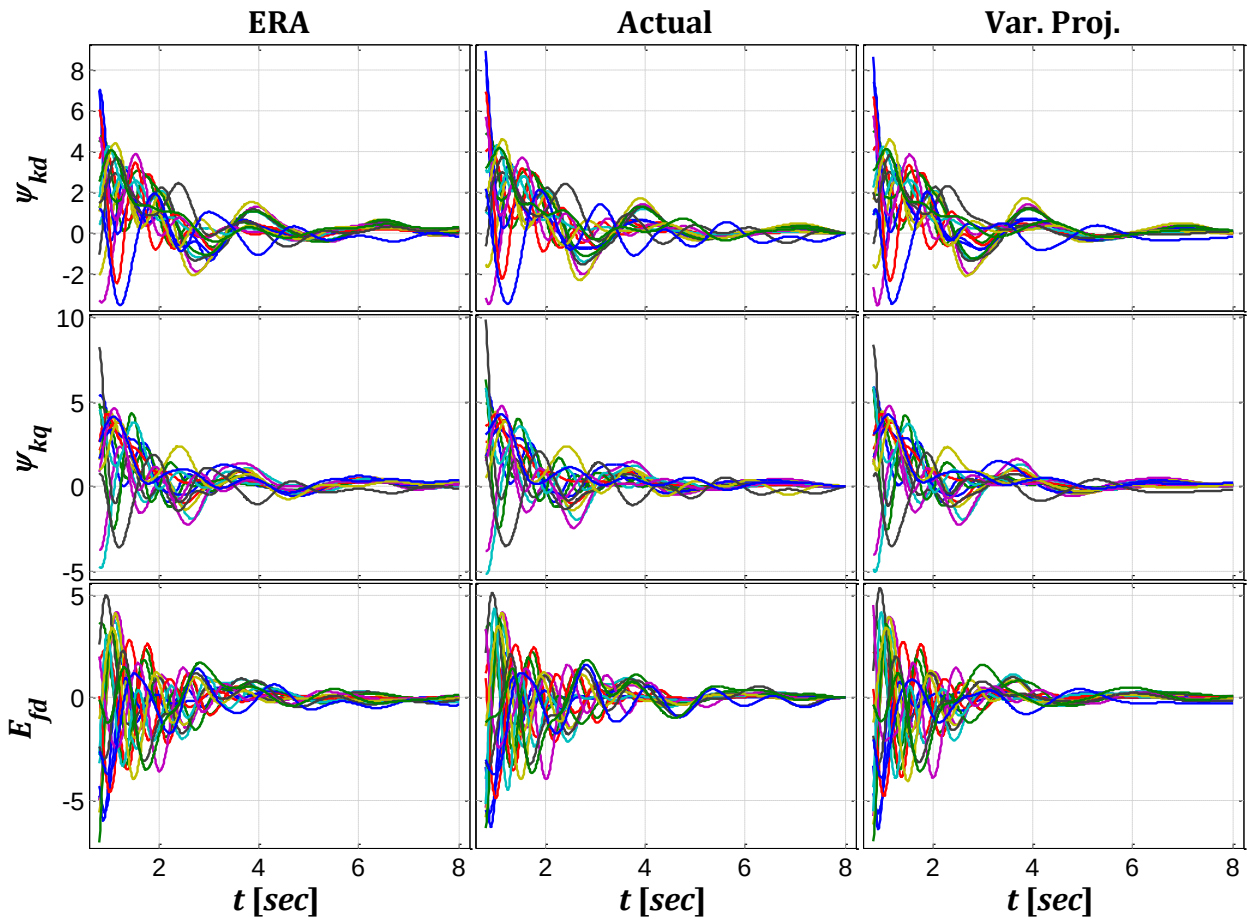

Figure 3.8: Simulated and estimated state variables Ψ_{kd}, Ψ_{kq} and E_{fd}

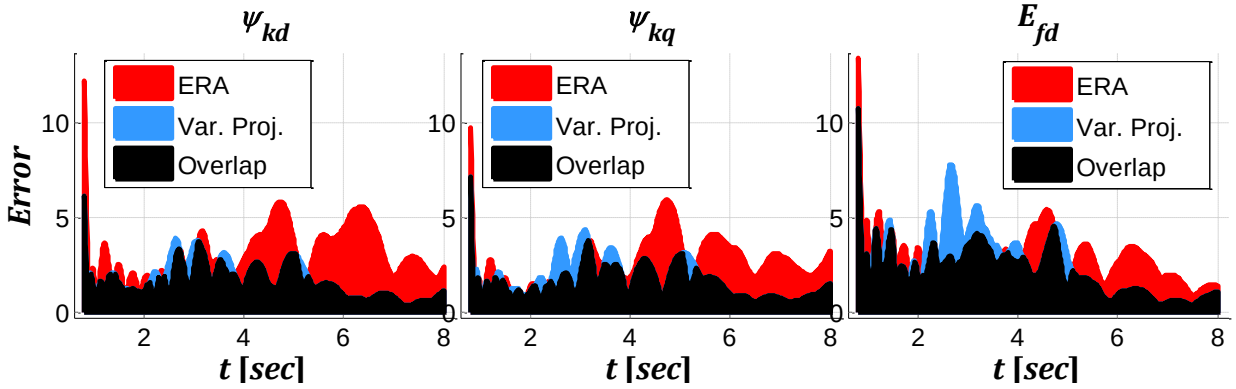

Figure 3.9: Estimation error for state variables Ψ_{kd}, Ψ_{kq} and E_{fd}

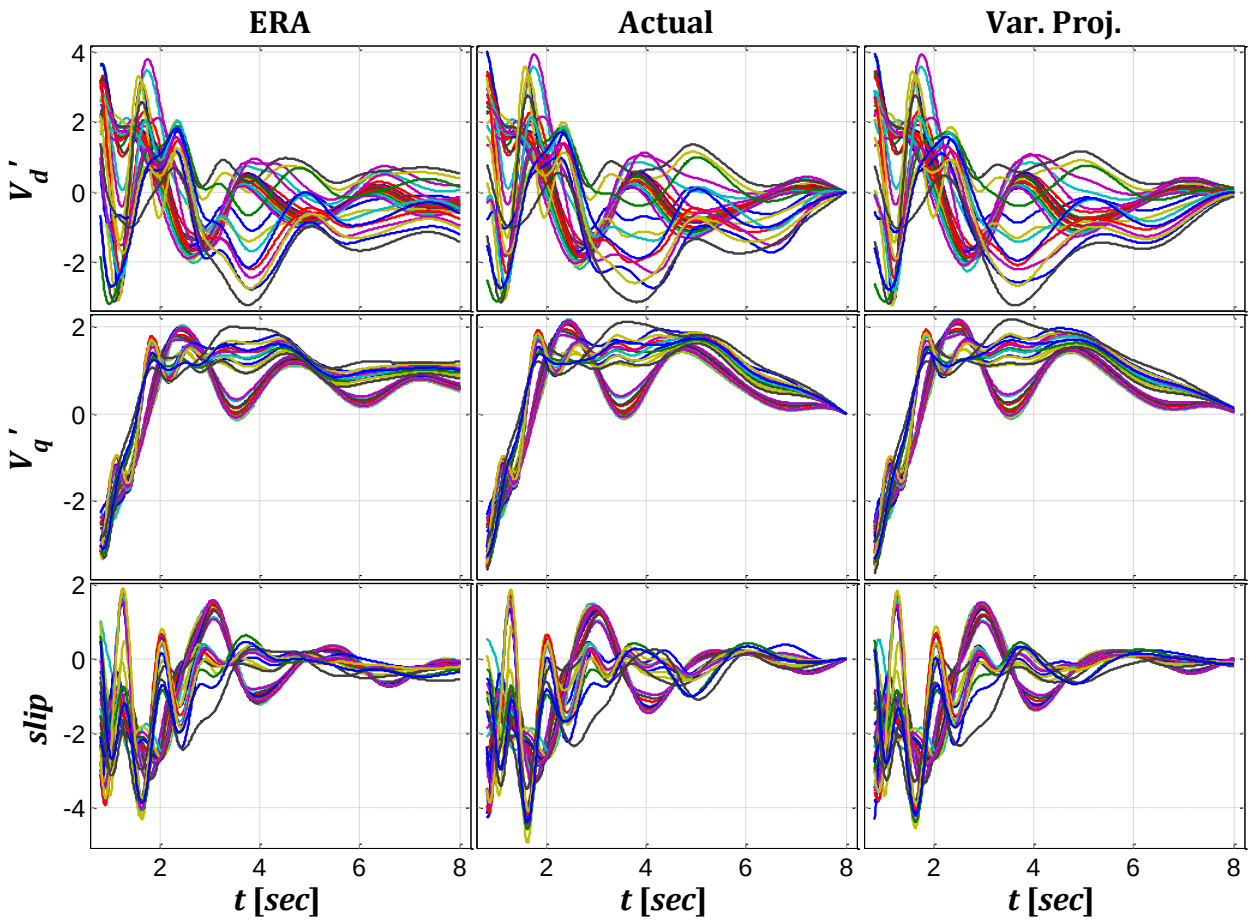

Figure 3.10: Simulated and estimated state variables V'_d, V'_q and *slip*

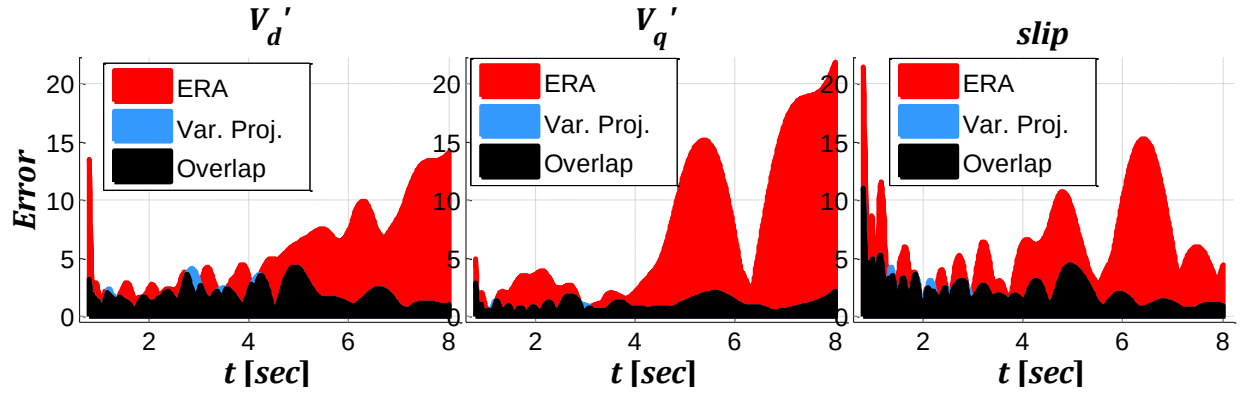

Figure 3.11: Estimation error for state variables V'_d, V'_q and *slip*

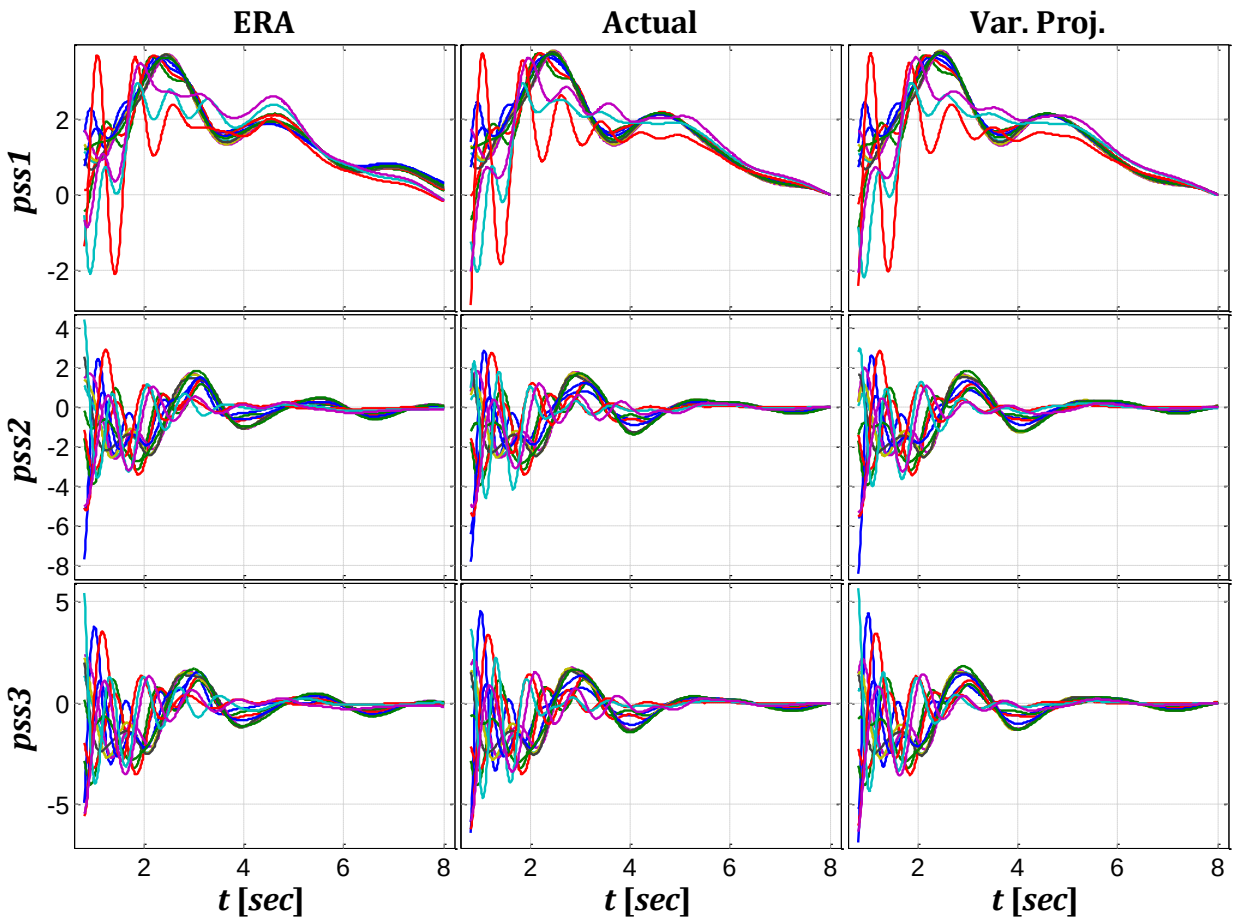

Figure 3.12: Simulated and estimated state variables *pss*1, *pss*2 and *pss*3

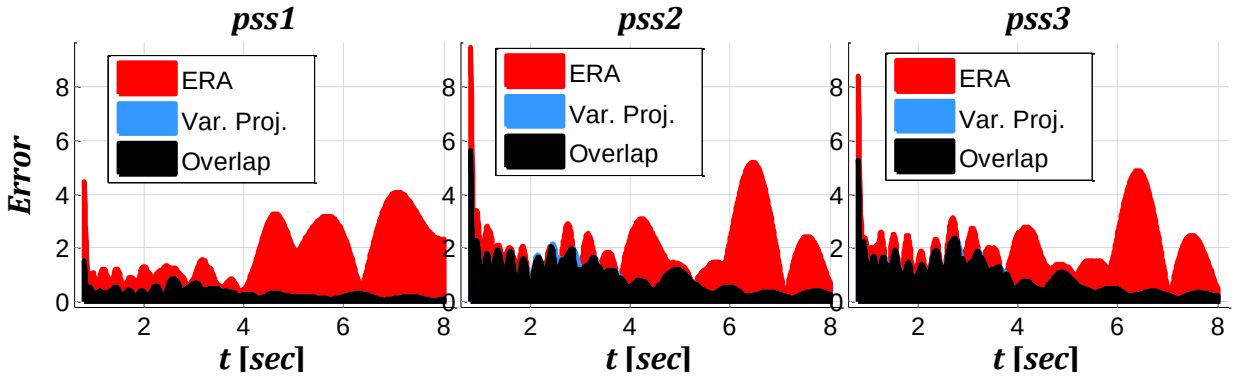

Figure 3.13: Estimation error for state variables *pss*1, *pss*2 and *pss*3

From the figures, it can be seen that ERA provided a good fit to the data for 11 modes. However, it can clearly be seen that variable projection improved on the solution from ERA by providing an even lower least-squares error to the data.

The modes estimated by variable projection and ERA are shown in Table 3.5. The state space model of the system is known, and therefore the actual eigenvalues of the system can be compared against the modes predicted by the modal analysis methods. Some of the 235 eigenvalues that roughly match the modes predicted by variable projection and ERA are additionally shown in Table 3.5. As a cautionary point, all of the estimated modal damping and frequencies should not be expected to exactly match all of the system's actual eigenvalues. In the 16-machine 68-bus system, not only does the system have 235 states, but the dynamic response of the system is nonlinear. The modal analysis methods approximate this high-order nonlinear response using a subset of just 11 modes (with one mode being a zero eigenvalue).

Var. Proj.	Actual	ERA
24.71% @ 1.564	28.78% @ 1.608	
	24.12% @ 1.394	16.57% @ 1.440
16.78% @ 1.165	14.77% @ 1.190	
	14.50% @ 1.106	14.42% @ 1.106
	8.599% @ 0.793	11.43% @ 0.699
16.85% @ 0.569	11.95% @ 0.570	
16.68% @ 0.355	14.47% @ 0.354	14.15% @ 0.407
		43.63% @ 0.174
99.99% @ 1.25×10^{-10}	99.99% @ 5.83×10^{-7}	

Table 3.5: Estimated % damping and frequency [Hz] vs. Actual Eigenvalues

As detailed in in Section 2.7, the estimated modes can be ranked by their dominance by computing the mode energies from the weighted signals, shown in Table 3.6. The mode at

99.99% @ 1.25×10^{-10} estimated by variable projection is a slow trending mode, and it essentially provides a near dc-offset to the data (despite there being a zero eigenvalue mode as well). Naturally this mode has very large energy by the calculation from Section 2.7. In Table 3.6, this mode is shown to have the largest energy, however it is not the primary interest for this analysis. Therefore the modes are ordered by their mode energy, excluding the slow trending mode which is listed last.

Var. Proj.	Mode Energy
16.68% @ 0.355	619.83
16.78% @ 1.165	514.69
16.85% @ 0.569	255.74
24.71% @ 1.564	241.09
99.99% @ 1.25×10^{-10}	5614.88

ERA	Mode Energy
14.15% @ 0.407	607.34
43.63% @ 0.174	383.52
14.42% @ 1.106	348.58
16.57% @ 1.440	237.96
11.43% @ 0.699	123.00

Table 3.6: Energy of modes estimated by variable projection and ERA

From Tables 3.5 and 3.6, it can be seen that variable projection located different modes than ERA. Most of the modes estimated by ERA and variable projection resemble one of the system's actual 235 eigenvalues, with perhaps the exception of the 43.63% @ 0.174 mode predicted by ERA and the 99.99% @ 1.25×10^{-10} predicted by variable projection. According to the mode energies in Table 3.6, there is a slow dominant mode somewhere around 0.35–0.4 Hz. The next dominant mode is perhaps the fast oscillatory mode at 1.1 Hz.

Some specifications can be given on the details of the optimization, with the estimations from ERA used as initial conditions for variable projection. Table 3.7 displays some of the evaluation times from the analysis, evaluated in MATLAB R2011a on a computer with 64-bit Intel i5-560M Dual Core CPU at 2.67 GHz with 4 GB of RAM. In the table, it can be seen that the ERA computation was the most time consuming at 61.42 seconds. The evaluation time for the Matrix Pencil Method was also determined, having a comparatively lower time of 40.53 seconds. Matrix Pencil provided a slightly worse error to the data at 6281.45 compared to ERA which

gave an error of 6246.02; however, the variable projection method converged to the same solution when using either of the estimations from matrix pencil or ERA as initial conditions.

Operation	Time [sec]
ERA	61.42
Matrix Pencil	40.53

Variable Projection	12.36
SVD	0.02
Cost function evaluations	1.12
Jacobian evaluations	1.45

Table 3.7: Evaluation times from example

From the initial conditions, the optimization iterated 13 times before stopping due to convergence. As will be discussed in Section 5.1, a trust region method is used in the optimization. Therefore the cost function, the gradient and the SVD (needed for calculating the cost and gradient), are all computed just 14 times each in the optimization. Table 3.7 shows that the variable projection method took 12.36 seconds to converge. The table shows the cumulative computation time taken for the 14 evaluations of the SVD, cost function and Jacobian matrix. These computations are not significantly time consuming. The SVD in particular is a very fast computation because it is evaluating the $\Phi(\alpha)$ matrix with dimensions as the number of time points by the number of modes (721×11 in this example). The remaining 9.77 seconds in the variable projection evaluation come from the nonlinear least squares solver used on the problem. Specifically the *lsqnonlin* MATLAB function employed here, solves a 2D subspace trust region method at each iteration of the optimization [62]. A solve-time hierarchy analysis reveals preconditioning of the trust region method accounts for most of the 9.77 seconds. In the preconditioning, a QR factorization and a column approximate minimum degree permutation are the bottleneck computations. The matrix the preconditioning is applied to, has a very large number of rows because of the multi-signal analysis on 235 signals. The matrix has row dimension as the number of time points multiplied by the number of signals and column dimension as the number of modes

([721×235]×11 in this example). If computation time is important for implementing this method, then the optimization solver and its options could be changed for the likeliest improvements.

This example has demonstrated how multi-signal variable projection reduces least-squares error when compared to ERA for an equivalent number of modes. This result should always be expected as long as variable projection is initialized with good initial conditions. If the modal damping and frequency results estimated by ERA (or any linear method) are input to variable projection as initial conditions, then variable projection will never do worse in terms of least-squares error to the data. If a linear modal analysis method predicts the perfect solution for some number of modes, then variable projection will show the gradient at that solution is zero, i.e. a local minimum was found. If the gradient at that solution is nonzero, then the variable projection method will iterate towards a better solution. The optimization is nonconvex; therefore its ability to find a global rather than local minimizer is dependent on initial conditions provided to the method.

3.4 Conclusion

This chapter has outlined a nonlinear least-squares optimization method, called the variable projection method, for estimating modal content of oscillatory data. This chapter seeks to encourage the use of the variable projection method in power system problems where linear modal analysis methods have traditionally been used for modal identification.

An example data-set was presented with a simulated disturbance in a 16-machine 68-bus test system. Estimated damping and frequencies from ERA and the variable projection method were compared to the actual eigenvalues of the system, and the modes were ranked by their mode energies. It was shown the variable projection method located modes that minimized error better than the modes estimated by ERA. This result should be expected as long as the variable projection method is provided with quality initial conditions, such as a solution obtained from ERA or the Matrix Pencil method.

The variable projection method offers several advantages over other methods. The method does not require data for analysis to be evenly spaced in time; and it appears to be relatively insensitive to time-sampling rate. The method allows for the inclusion of any possible basis function, such as polynomial basis function for data detrending. Further investigation could examine including nonlinear basis functions having modulated amplitudes and frequencies. The user of the method also has tight control on the type of solution obtained. For example, the user must not only specify the number of modes, but furthermore must specify what number of the modes are complex and real. Additional control can be achieved by constrained optimization if desired. This type of control allows for specific targeting of modal solutions by experienced users.

An extension to the variable projection method will be discussed in the next chapter, which enforces high order interaction between the system's natural modes. As discussed in Section 2.1, linearization is the standard analysis method for approximating solutions to systems of nonlinear differential equations. However the theory of normal form analysis can be used to better approximate solutions. Chapter 4 examines enforcing a normal form solution structure in modal analysis optimization problems. The theory of normal form analysis lends itself well to working with the variable projection method, and this approach could potentially estimate modes of nonlinear disturbance data more accurately.

In Chapter 5, a modal analysis tool developed for industry use is presented, which uses the variable projection method outlined in this chapter. Specific details are provided concerning the optimization algorithms used for solving the nonlinear least-squares problem and effective strategies for performing modal analysis.

Chapter 4

Normal Form Theory

Power system models are inherently nonlinear. The theory of normal form analysis can be used to help better understand nonlinear dynamic behavior evident in the models [15-21]. Normal form analysis can be used to map a model of nonlinear differential equations to a linear model through use of near-identity nonlinear coordinate transformations. The solution structure to the nonlinear differential equations is a summation of fundamental damped or undamped sinusoids and their nonlinear combinations. This chapter uses the theory of normal form analysis to motivate an extension to the variable projection method described in Chapter 3.

In Section 4.1, the complete mathematical procedure of normal form analysis is reviewed. In Section 4.2, normal form analysis is demonstrated on a three-bus power system example. In Section 4.3, the nonlinear variable projection method is extended using the theory of normal form analysis to further improve solution accuracy. An example demonstrating the normal form extension is presented in Section 4.4, and it is shown that the nonlinear program performs well as the least-squares error is minimized and the system's eigenvalues are converged on. Lastly in Section 4.5, a basic example is presented for analyzing system's with repeated eigenvalues.

4.1 Normal Form Analysis

Normal form analysis was introduced by Henri Poincaré in the late 1800's, as a tool to integrate nonlinear systems [19]. Poincaré showed with a systematic sequence of near-identity coordinate changes, higher order terms could be eliminated. In this section the mathematical procedure of normal form analysis is presented.

For the normal form method, a nonlinear set of p differential equations is given describing the dynamics of the system, as shown in (4.1).

$$\dot{x} = f(x) \tag{4.1}$$

Equation (4.1) is preconditioned so that $x = 0$ at the equilibrium point, i.e. $\dot{x}(t = \infty) = 0$ and $x(t = \infty) = 0$. If the final equilibrium point is nonzero, $x(\infty) = x_f \neq 0$, then the coordinate system is shifted, $\tilde{x}(t) = x(t) - x_f$, so that $\tilde{x}(\infty) = x(\infty) - x_f = 0$. Normal form analysis exploits combinatorics of a system's natural modes. Notation is introduced here for taking a vector to an exponential power, for example $y^{(2)}$ where y has 3 entries.

$$y^{(2)} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}^{(2)} = \begin{bmatrix} y_1^2 \\ y_1 y_2 \\ y_1 y_3 \\ y_2^2 \\ y_2 y_3 \\ y_3^2 \end{bmatrix}$$

In the above example, taking y to the 2nd power, results in a vector having every second order combination of the elements in y. The vector exponential notation is frequently reoccurring in the normal form analysis, outlined next.

A. Taylor Expansion and Jordan Form Coordinates

To begin the normal form analysis, the set of differential equations in (4.1) is represented by a Taylor expansion to a desired order N.

$$\begin{aligned} \dot{x}(t) - \dot{x}(\infty) \approx & \left[\frac{\partial f}{dx}\right]_{x=x(\infty)} \big(x(t) - x(\infty)\big) \\ & + \frac{1}{2!}\left[\frac{\partial^2 f}{dx^2}\right]_{x=x(\infty)} \big(x(t) - x(\infty)\big)^{(2)} \\ & + \ldots \\ & + \frac{1}{N!}\left[\frac{\partial^N f}{dx^N}\right]_{x=x(\infty)} \big(x(t) - x(\infty)\big)^{(N)} \end{aligned}$$

With $\dot{x}(\infty) = 0$ and $x(\infty) = 0$, the Taylor expansion reduces to the following.

$$\dot{x}(t) \approx \left[\frac{\partial f}{dx}\right]_{x=x(\infty)} x(t) + \frac{1}{2!}\left[\frac{\partial^2 f}{dx^2}\right]_{x=x(\infty)} x(t)^{(2)} + \ldots + \frac{1}{N!}\left[\frac{\partial^N f}{dx^N}\right]_{x=x(\infty)} x(t)^{(N)}$$

The notation of the Taylor expansion will be simplified to that in (4.2).

$$\dot{x} \approx Ax + X_2 x^{(2)} + \ldots + X_N x^{(N)} \tag{4.2}$$

The Taylor expansion is put into Jordan form by computing the eigenvalues $\lambda_1, \ldots, \lambda_p$ and right eigenvectors V of the state matrix A. By substituting $x = Vy$ and multiplying (4.2) by $U = V^{-1}$, the coordinates $x(t)$ are converted to the Jordan form coordinate system $y(t)$.

$$\dot{y} = UAVy + UX_2(Vy)^{(2)} + \ldots + UX_N(Vy)^{(N)}$$

The above notation of the Jordan form expansion will be simplified to (4.3).

$$\dot{y} = Jy + Y_2 y^{(2)} + \ldots + Y_N y^{(N)} \tag{4.3}$$

In (4.3), the eigenvalues of the state matrix A are the diagonal entries of matrix J.

$$J = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_p \end{bmatrix}$$

B. Normal Form Coordinate Change

The Jordan form coordinate system $y(t)$ is transformed to the normal form coordinate system $z(t)$ via the relationship in (4.4).

$$y = H_1 z + H_2 z^{(2)} + \ldots + H_N z^{(N)} \tag{4.4}$$

Relationship (4.4) will be used to obtain the end goal of a decoupled linear system $\dot{z} = Jz$. Take the time-derivative of (4.4) to obtain (4.5).

$$\dot{y} = \left(H_1 + H_2 \frac{\partial z^{(2)}}{\partial z} + \ \ldots + H_N \frac{\partial z^{(N)}}{\partial z} \right) \dot{z} \tag{4.5}$$

Substitute $\dot{y}$ from (4.5) into (4.3) to obtain (4.6).

$$Jy + Y_2 y^{(2)} + \ \ldots + Y_N y^{(N)} = \left(H_1 + H_2 \frac{\partial z^{(2)}}{\partial z} + \ \ldots + H_N \frac{\partial z^{(N)}}{\partial z} \right) \dot{z} \tag{4.6}$$

Substitute y from (4.4) into (4.6) to obtain (4.7).

$$\begin{aligned} & J\left(H_1 z + H_2 z^{(2)} + \ldots + H_N z^{(N)} \right) \\ & + \ Y_2 \left(H_1 z + H_2 z^{(2)} + \ldots + H_N z^{(N)} \right)^{(2)} = \left(H_1 + H_2 \frac{\partial z^{(2)}}{\partial z} + \ldots + H_N \frac{\partial z^{(N)}}{\partial z} \right) \dot{z} \\ & + \ldots \\ & + \ Y_N \left(H_1 z + H_2 z^{(2)} + \ldots + H_N z^{(N)} \right)^{(N)} \end{aligned} \tag{4.7}$$

The relationship in (4.7), is fully in terms of normal form coordinates $z(t)$. The H matrices can be determined to permit cancellation of terms or order $z^{(2)}$ through $z^{(N)}$.[1] The right hand side matrix multiplying $\dot{z}$ in (4.7) could be inverted and multiplied through in order to solve for $\dot{z}$, as shown below.

$$\dot{z} = \left(H_1 + H_2 \frac{\partial z^{(2)}}{\partial z} + \ \ldots + H_N \frac{\partial z^{(N)}}{\partial z} \right)^{-1} \begin{pmatrix} J\left(H_1 z + H_2 z^{(2)} + \ \ldots + H_N z^{(N)} \right) \\ + \ Y_2 \left(H_1 z + H_2 z^{(2)} + \ \ldots + H_N z^{(N)} \right)^{(2)} \\ + \ldots \\ + \ Y_N \left(H_1 z + H_2 z^{(2)} + \ \ldots + H_N z^{(N)} \right)^{(N)} \end{pmatrix}$$

From there the H matrices could be solved to permit proper cancellation and achieve the end goal of $\dot{z} = Jz$. However, the inversion of the right hand side matrix is computationally expensive. A more common approach is to substitute $\dot{z} = Jz$ into (4.7) as shown in (4.8).

[1] For some nonlinear systems at certain operating points, the nonlinear terms cannot be eliminated. In such cases normal forms are used to construct the simplest nonlinear system model for analysis.

$$\begin{aligned} & J\left(H_1 z + H_2 z^{(2)} + \ldots + H_N z^{(N)}\right) \\ & + Y_2\left(H_1 + H_2 z^{(2)} + \ldots + H_N z^{(N)}\right)^{(2)} \\ & + \ldots \\ & + Y_N\left(H_1 z + H_2 z^{(2)} + \ldots + H_N z^{(N)}\right)^{(N)} \end{aligned} = \left(H_1 + H_2 \frac{\partial z^{(2)}}{\partial z} + \ldots + H_N \frac{\partial z^{(N)}}{\partial z}\right) Jz \tag{4.8}$$

Equating like terms of order $z^{(1)}$ through $z^{(N)}$ on the left and right hand side in (4.8), reveals the H matrix solutions that allow proper cancellation. For example, equating first order terms $z^{(1)} = z$ in (4.8), reveals the solution for matrix H_1.

$$\begin{aligned} JH_1 z &= H_1 Jz \\ H_1 &= I \text{ (identity matrix)} \end{aligned} \tag{4.9}$$

Equating second order terms $z^{(2)}$ and substituting $H_1 = I$ in (4.8), reveals the element-by-element solution for matrix H_2 in (4.10).

$$\begin{aligned} (JH_2 + Y_2)z^{(2)} &= H_2 \frac{\partial z^{(2)}}{\partial z} Jz \\ \Longrightarrow Y_2 z^{(2)} &= H_2 \frac{\partial z^{(2)}}{\partial z} Jz - JH_2 z^{(2)} \\ H_2(i\,,[j_1, j_2]) &= \frac{Y_2(i\,,[j_1, j_2])}{\lambda_{j_1} + \lambda_{j_2} - \lambda_i} \end{aligned} \tag{4.10}$$

The above notation of $H_2(i\,,[j_1, j_2])$ refers to the element in the i^{th} row of matrix H_2 and the column associated with the second order combination of j_1 and j_2. An indexing scheme is presented in Section 4.1-D which would determine the column index associated with the second order combination of j_1 and j_2. Resonance problems can occur if the denominator in (4.10) equals zeros. Issues of resonance will be discussed in more detail in Section 4.3-B.

Matching like terms of order $z^{(3)}$ through $z^{(N)}$ is more complicated. The solutions for matrices H_3 through H_N cannot be written as formally as H_1 and H_2, but they can be solved with a loop algorithm. With the H matrices determined, (4.7) reduces to (4.11).

$$\dot{z} = Jz + 0z^{(2)} + \ldots + 0z^{(N)} + \mathcal{O}(N+1) \tag{4.11}$$
$$\dot{z} \approx Jz$$

In (4.11), $\dot{z}$ maintains a linear term Jz, while eliminating higher order terms up to order N. The higher order terms of $\mathcal{O}(N+1)$ and above are assumed negligible and truncated. Equation (4.11) is a decoupled linear differential equation, therefore $z(t)$ can be solved.

$$z_i(t) = z_{0,i}\, e^{\lambda_i t} \;,\; for\; i = 1, \ldots, p \tag{4.12}$$

In (4.12), $z_{0,i}$ is the i^{th} initial condition for $z(t)$ at time 0 seconds. To obtain the normal form initial condition z_0, start with initial condition x_0. Next solve $y_0 = Ux_0$. Relationship (4.4) is used to convert between y_0 and z_0. A nonlinear program must be implemented to solve (4.4) for z_0. Finally, the normal form coordinates $z(t)$ are solved in (4.12). With $z(t)$ solved, the Jordan form coordinates $y(t)$ are obtained using (4.4), and finally the original coordinates are solved using $x(t) = Vy(t)$.

The solution obtained for $x(t)$ from normal form analysis contains terms for the p natural modes, $\lambda_1, \ldots, \lambda_p$, of the system, but it also contains higher order terms up to order N, exhibiting interaction between the natural modes. The typical solution structure is shown in (4.13).

$$\begin{aligned} x(t) \approx & \sum_{i_1=1}^{p} b_{i_1}\, e^{\lambda_{i_1} t} \\ & + \sum_{i_1=1}^{p} \sum_{i_2=i_1}^{p} b_{i_1,i_2}\, e^{(\lambda_{i_1}+\lambda_{i_2})t} \\ & + \ldots \\ & + \sum_{i_1=1}^{p} \sum_{i_2=i_1}^{p} \ldots \sum_{i_N=i_{N-1}}^{p} b_{i_1,i_2,\ldots,i_N} e^{\left(\lambda_{i_1}+\lambda_{i_2}+\ldots+\lambda_{i_N}\right)t} \end{aligned} \tag{4.13}$$

In (4.13), the b's are complex coefficient vectors solved for in the normal form procedure. The subscripts $i_1, \dots, i_k$ in $b_{i_1,\dots,i_k}$ distinguish the natural mode interactions the coefficient vector is associated with. The solution $x(t)$ is a summation of damped or undamped sinusoids, and one can see exactly how the p natural modes, $\lambda_1, \dots, \lambda_p$, combine and interact to give an approximate solution to (4.1). Note that a first order normal form solution, $N = 1$, is equivalent to the solution obtained from eigenanalysis and linearization, discussed in Section 2.1.

C. Normal Form Combinatorics

It can be seen in (4.13) that there is higher order interaction between each of the p natural modes, $\lambda_1, \dots, \lambda_p$, for order $k = 2, \dots, N$. To clarify, for example if $p = 3$ and $k = 2$, there would be six unique second order combinations of λ_1, λ_2 and λ_3.

$$\{2\lambda_1\,, \quad \lambda_1 + \lambda_2\,, \quad \lambda_1 + \lambda_3\,, \quad 2\lambda_2\,, \quad \lambda_2 + \lambda_3\,, \quad 2\lambda_3\}$$

In general, the number of unique order k combinations for p number of modes can be calculated. This is a unique combinations problem with repetition allowed, such as the number of unique ways to fill k slots with p choices with repetition of the p choices allowed. The number of unique combinations is $\frac{(p+k-1)!}{k!(p-1)!}$ or equivalently $\binom{p+k-1}{k} = \binom{p+k-1}{p-1}$, where $\binom{n}{k}$ is the binomial coefficient n choose k.

D. Normal Form Indexing Scheme

With the combinatorial nature of normal form analysis, indexing can become complicated. Consider needing to know the exact index of a particular combination in vector exponential $y^{(k)}$.

$$y^{(k)} = \begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix}^{(k)}$$

For example, consider the combination $y_2 y_3^2 y_4$ in vector $y^{(4)}$ where y has five entries, $p = 5$.

$$y^{(4)} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix}^{(4)}$$

So long as the combination entries are ordered numerically ascending, i.e. in the example 2,3,3,4, then the index can be conveniently calculated. The index of $y_2 y_3^2 y_4$ in $y^{(4)}$ is 47.

For a general vector exponential $y^{(k)}$ where y has p entries, denote the index to be defined as $i = [i_1 \quad i_2 \quad i_3 \quad \ldots \quad i_k]$. In the example, $i = [2 \quad 3 \quad 3 \quad 4]$.

$$index = 1 + \frac{(p+k-1)!}{(k)!\,(p-1)!} - \frac{(p+k-i_1)!}{(k)!\,(p-i_1)!} + \sum_{j=2}^{k} \frac{(p+k-i_{j-1}-j+1)!}{(k-j+1)!\,(p-i_{j-1})!} - \frac{(p+k-i_j-j+1)!}{(k-j+1)!\,(p-i_j)!}$$

4.2 Normal Form Example

In this section, a three-bus power system disturbance is analyzed to demonstrate normal form analysis. The electromechanical dynamics at each generator are modeled by second-order nonlinear differential swing equations. There are thus six state variables/natural modes: three bus angles ($\theta\,[rad]$) and three turbine speeds $\left(\omega\,\left[\frac{rad}{sec}\right]\right)$. The second-order swing equation are shown in (4.14).

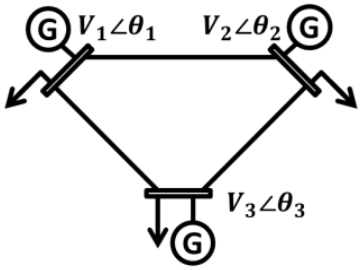

Figure 4.1: Three-bus power system

$$\dot{\theta}_i = \omega_i - \omega_0 \tag{4.14}$$

$$\dot{\omega}_i = P_i - \delta_i(\omega_i - \omega_0) - \sum_{\substack{i=1 \\ j \neq i}}^{\#bus} \frac{V_i V_j}{X_{ij}} \sin(\theta_i - \theta_j)$$

$$for\ i = 1\,,\,2\,,\,3$$

In (4.14), ω_0 is synchronous speed equal to 1, $P_i = P_{i,mech} - P_{i,load}$ is constant where $P_{i,mech}$ is the mechanical input power and $P_{i,load}$ is the active power load, δ_i is the turbine damping constant and X_{ij} is the line reactance. The state-space model is constructed from the system's differential and algebraic equations. To begin, the system is in steady-state. A fault is applied at bus 1 by suppressing its voltage to zero. After 0.1 seconds, bus 1 voltage its restored to its nominal value while simultaneously disconnecting the line connecting buses 1 and 2. An ordinary differential equation (ODE) solver is used to simulate the disturbance. The dynamic response of all six state variables is simulated with the ODE solver, with Figure 4.2 displaying the dynamic response of all six state variables, after the line is disconnected, as the dashed black line.

An approximate solution to the state variables can be obtained using small-signal analysis and linearization of the state-space model in (4.14), as discussed in Section 2.1. In Figures 4.2 and 4.3, the dashed black line is the simulated response obtained from the ODE solver. The solid blue line in Figure 4.2 is the approximate solution obtained by using small-signal analysis and linearization. It can be seen the linearized solution is a relatively poor approximation to the actual nonlinear response from the ODE solver. Alternatively, a closed form approximate solution can be obtained using the normal form analysis outlined in the prior section. Normal form analysis with order $N = 8$ is performed on the state-space model in (4.14), and the closed form solution is plotted as the solid blue line in Figure 4.3 against the nonlinear ODE simulation.

As can be seen, the 8^{th} order normal form solution in Figure 4.3 approximates the nonlinear ODE simulation much closer than the linearization solution in Figure 4.2. The combinatorial interaction of natural modes obtained via normal form analysis effectively captures the nonlinear behavior of the system. This modal interaction should therefore be expected when analyzing disturbance data from a nonlinear system. The theory of normal form analysis will be used for motivating an extension to the nonlinear least-squares variable projection method for estimating modes of ring-down data, which is presented in the next section.

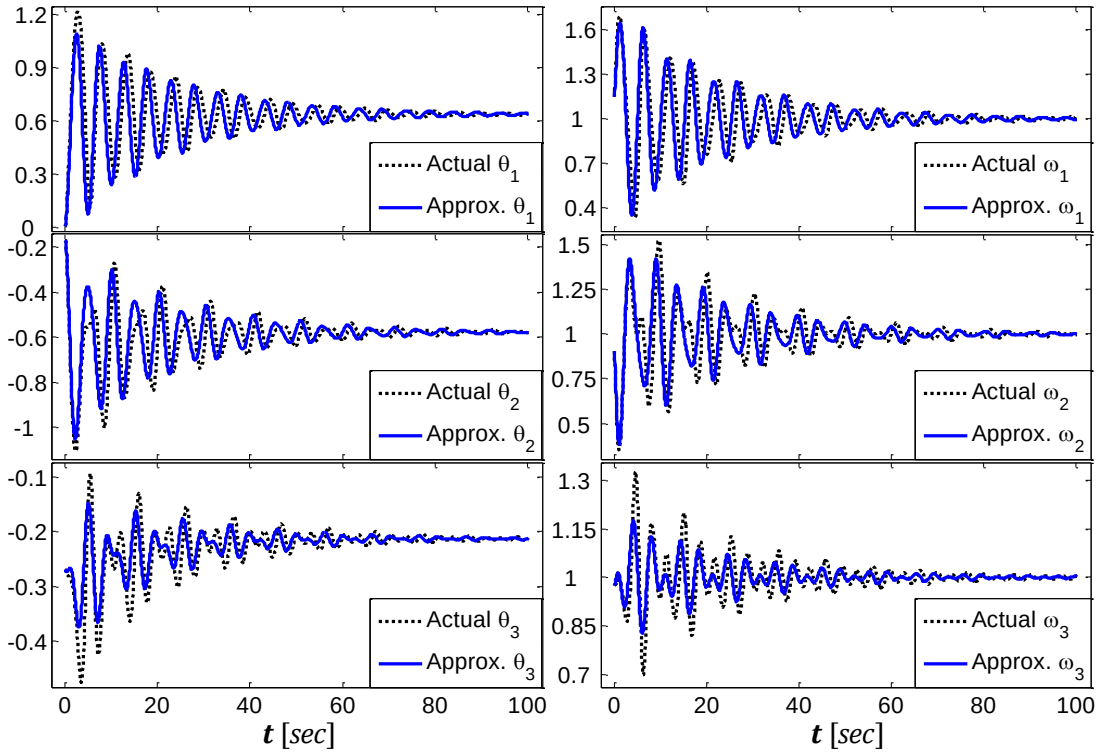

Figure 4.2: Linearization solution (solid blue) vs. Actual nonlinear response (dashed black)

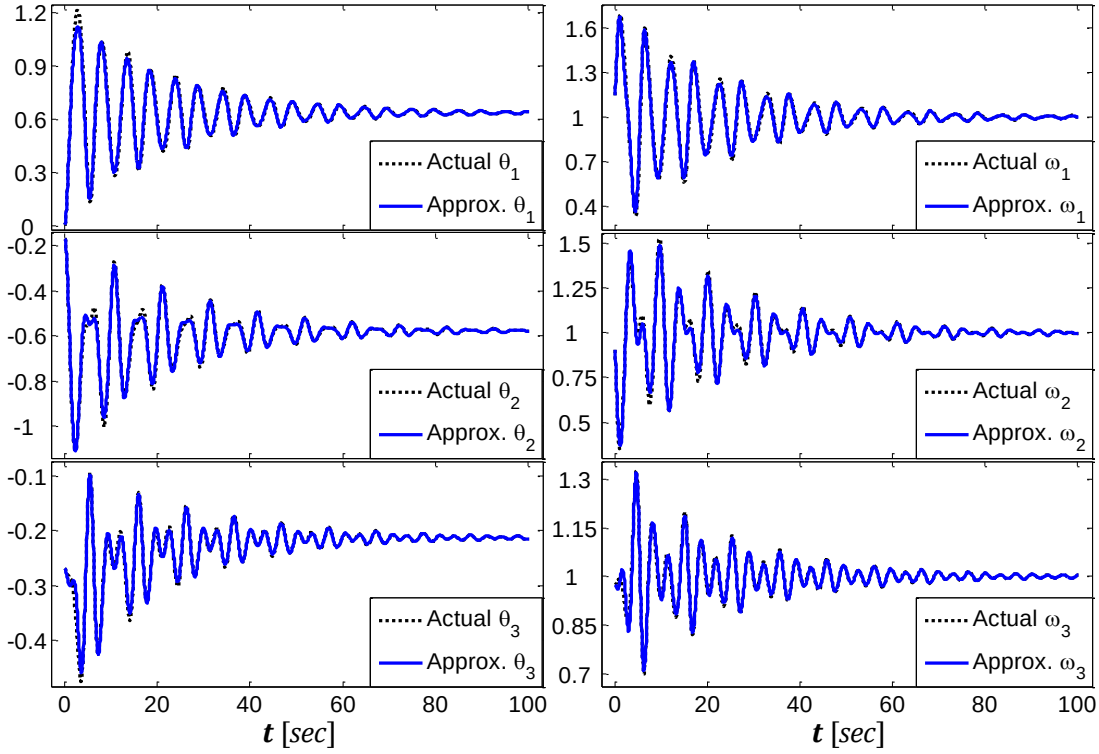

Figure 4.3: 8th order normal form solution (solid blue) vs. Actual nonlinear response (dashed black)

4.3 Extension to the Variable Projection Method

Normal form analysis improves upon a linearized solution to a set of nonlinear differential equations. Section 4.1 detailed the normal form solution structure which motivates the work in this section.

A. Normal Form Nonlinear Program

As was shown in (4.13), the normal form solution $x(t)$ contains all possible modal interactions between the system's p natural modes/eigenvalues up to a desired order N. In this section, the normal form solution structure is enforced on the variable projection nonlinear optimization method of Chapter 3. The normal form extension does not increase the number of optimization variables in the variable projection method. For modal analysis problems, there will remain p number of optimization variables in the vector α (where α was introduced in Section 3.1). The normal form solution structure is imposed by augmenting the matrix of basis functions $\Phi(\alpha)$ with additional basis functions that enforce the interaction between the p natural modes/eigenvalues. For example, consider there being $p = 3$ number of natural modes/eigenvalues λ_1, λ_2 and λ_3. With a 2nd order normal form solution structure, there would be six additional unique 2nd order combinations of λ_1, λ_2 and λ_3.

$$\{2\lambda_1\,,\quad \lambda_1+\lambda_2\,,\quad \lambda_1+\lambda_3\,,\quad 2\lambda_2\,,\quad \lambda_2+\lambda_3\,,\quad 2\lambda_3\}$$

The example has three natural modes, $p = 3$, therefore there is one complex-conjugate mode-pair and one real-valued mode, such that $\lambda_1 = \sigma_1 + j\omega_1$, $\lambda_2 = \sigma_1 - j\omega_1$ and $\lambda_3 = \sigma_2$. The vector α, is typically constructed using real-valued damping σ_i and frequency ω_i components, where in the example $\alpha = [\sigma_1 \quad \omega_1 \quad \sigma_2]$. For simplicity, now consider α to be constructed from the complex-valued eigenvalues, such that $\alpha = [\lambda_1 \quad \lambda_2 \quad \lambda_3]$. In the example, six basis functions of 2nd order mode combinations are appended to the $\Phi(\alpha)$ matrix to emulate a normal form solution structure.

$$\Phi(\lambda) = [e^{\lambda_1 t} \quad e^{\lambda_2 t} \quad e^{\lambda_3 t} \mid e^{2\lambda_1 t} \quad e^{(\lambda_1+\lambda_2)t} \quad e^{(\lambda_1+\lambda_3)t} \quad e^{2\lambda_2 t} \quad e^{(\lambda_2+\lambda_3)t} \quad e^{2\lambda_3 t}]$$

The normal form solution structure in (4.13) is thus enforced. With this formulation, $\Phi(\alpha)$ is constrained so that the higher order modes are combinations of the p natural modes $\lambda_1, \dots, \lambda_p$. The user of the normal form optimization method must specify the desired number of natural modes p and also the normal form order N. In general, the total number of basis functions equals the number of natural modes plus higher order modes, which equals

$$p + \sum_{k=2}^{N} \frac{(p+k-1)!}{k!\,(p-1)!}.$$

The computational speed of the normal form optimization method is relatively unaffected because the number of optimization variables is unchanged, with there being only a small subset of p optimization variables.

B. Resonance

Due to the issue of resonance, it is not always possible to perform normal form analysis with some state-space models. For system's having this resonance issue, higher order nonlinear terms cannot be eliminated by following the method outlined in Section 4.1. Specifically, the analysis breaks down when attempting to solve for the H matrices. Consider the element-by-element solution for H_2 from (4.10).

$$H_2(i\,,[\,j_1, j_2]) = \frac{Y_2(i\,,[\,j_1, j_2])}{\lambda_{j_1} + \lambda_{j_2} - \lambda_i}$$

It is possible this expression could equal infinity if the denominator $\lambda_{j_1} + \lambda_{j_2} - \lambda_i = 0$. This is called resonance. It is entirely dependent on the operating point and eigenvalues of the system, and it can similarly occur with the higher order matrices H_3 and above. Resonance is less common, and generally it can be used for identifying types of bifurcations when the equations become irreducible.

A related resonance issue can arise in measurement-based modal analysis methods. Consider the basis functions in (4.15) resulting from the existence of two nearly identical eigenvalues, $\lambda_1 \approx \lambda_2$, with similar damping and frequency.

$$\begin{aligned} & A_1 e^{\sigma_1 t}\cos(\omega_1 t + \phi_1) + A_2 e^{\sigma_2 t}\cos(\omega_2 t + \phi_2) \\ &= e^{\sigma_1 t}[A_1 \cos\phi_1 \cos(\omega_1 t) - A_1 \sin\phi_1 \sin(\omega_1 t)] \\ &\quad + e^{\sigma_2 t}[A_2 \cos\phi_2 \cos(\omega_2 t) - A_2 \sin\phi_2 \sin(\omega_2 t)] \end{aligned} \tag{4.15}$$

Define and substitute the coefficients $a_1 = A_1 \cos\phi_1$, $b_1 = A_1 \sin\phi_1$, $a_2 = A_2 \cos\phi_2$ and $b_2 = A_2 \sin\phi_2$ to obtain (4.16).

$$e^{\sigma_1 t}[a_1 \cos(\omega_1 t) - b_1 \sin(\omega_1 t)] + e^{\sigma_2 t}[a_2 \cos(\omega_2 t) - b_2 \sin(\omega_2 t)] \tag{4.16}$$

The optimization problem is typically implemented in the above fashion, where a_1, b_1, a_2 and b_2 are the linear coefficients solved by the variable projection method. With a solution for these coefficients, the mode shape amplitude and phase can be solved by $A_1 = \sqrt{a_1^2 + b_1^2}$ and $\phi_1 = \tan^{-1}\left(\frac{b_1}{a_1}\right)$, and A_2 and ϕ_2 are solved the same way. However, with $\sigma_1 \approx \sigma_2$ and $\omega_1 \approx \omega_2$, (4.16) can be rewritten as in (4.17).

$$e^{\sigma_1 t}[(a_1 + a_2)\cos(\omega_1 t) - (b_1 + b_2)\sin(\omega_1 t)] \tag{4.17}$$

Observe that a_1 plus a_2 can equal a constant value $C = a_1 + a_2$. Define $a_2 = C - a_1$ and observe that the scalar coefficients a_1 and a_2 can run off to plus and minus infinity without changing (4.17). This can similarly occur with b_1 and b_2. This is a resonance issue resulting from two or more modes having similar damping and frequencies, which causes mode shape values to explode to unreasonably large magnitudes. Enforcing a normal form solution structure on the problem may only increase this possibility, as the higher order modes could become resonant with one another.

For measurement-based modal analysis methods, fixes to this resonance issue are possible. Having two or more resonant modes, with similar damping and frequencies values, could be an artifact of specifying too many modes in the analysis, i.e. p is too large. In fact, resonance and unreasonable mode shape values could occur using Prony analysis or any traditional modal analysis method. A quick fix would be to decrease the number of specified modes p, in hopes that the method will consolidate these modes of similar damping and frequency into a single dominant mode. The variable projection method offers a different fix in the form of constrained optimization, that is not available with traditional modal analysis methods. As discussed in Section 3.2-D, constraints can be enforced on the variables of the optimization problem. The constraints can be used to prevent against resonance by guiding modes away from one another using appropriate bounds on the damping and frequency variables in the optimization.

4.4 Example

In this section, the three-bus power system disturbance from Section 4.2 is analyzed to demonstrate the variable projection method and the normal form extension to it. The state-space model of this system is known, (4.14); therefore, the natural modes of this model can actually be calculated using linearization and eigenanalysis. The six state variables give rise to six natural modes in the system. One of the six modes is a zero eigenvalue which contributes a dc-offset to each state. In total there is a zero eigenvalue mode, two complex conjugate pairs of modes and one real mode.

An ODE solver was used to simulate the disturbance as described in Section 4.2, and the simulated results are recorded as data. The data is used to demonstrate the variable projection method and its normal form extension. Each data signal is scaled by the inverse of its standard deviation, β_i, which is in accordance with conditioning data for multi-signal analysis. A dc-offset is included in the optimization to account for the zero eigenvalue. The number of natural modes input to the variable projection method is specified as $p = 5$; with two complex conjugate pairs of modes and one real mode. The results from the variable projection optimization are

shown in Figure 4.4 for all six state variables. The ODE simulated data is shown as the dashed black line, and the solution from the variable projection method is shown as the solid blue line.

As seen in Figure 4.4, the solution obtained by the variable projection method is good; however, it is of interest to examine the data while including 2nd and 3rd order combinations of the five natural modes. Therefore, a normal form solution structure with $N = 2$, is imposed on the optimization problem by augmenting the $\Phi(\alpha)$ matrix with additional basis functions.

$$\Phi(\alpha) = [e^{\lambda_1 t} \quad \dots \quad e^{\lambda_5 t} \mid e^{2\lambda_1 t} \quad e^{(\lambda_1+\lambda_2)t} \quad \dots \quad e^{2\lambda_5 t} \mid 1_{m\times 1}]$$

The $\Phi(\alpha)$ matrix has five 1st order basis functions, fifteen 2nd order basis functions and a column of ones used for a dc-offset. The residual error for the i^{th} signal y_i is thus $r_i(\alpha) = (I - \Phi(\alpha)\Phi(\alpha)^{\dagger})\, y_i$. The results from the optimization are shown in Figure 4.5.

The solution obtained by including the 2nd order basis functions provides an even better match to the disturbance data. This is no surprise however, as increasing the number of basis functions will always further minimize the least-squares error. This is even more so the case with the minimal cost obtained by including the 3rd order combinations. The least-squares error objective value for all three solutions is shown in Table 4.1.

	Var. Proj. $p = 5, N = 1$	Var. Proj. $p = 5, N = 2$	Var. Proj. $p = 5, N = 3$
$\sum_{i=1}^{6} \frac{1}{2} \left\| \beta_i r_i(\alpha) \right\|_2^2$	306.61	16.34	0.027

Table 4.1: Objective function of least-squares error to data

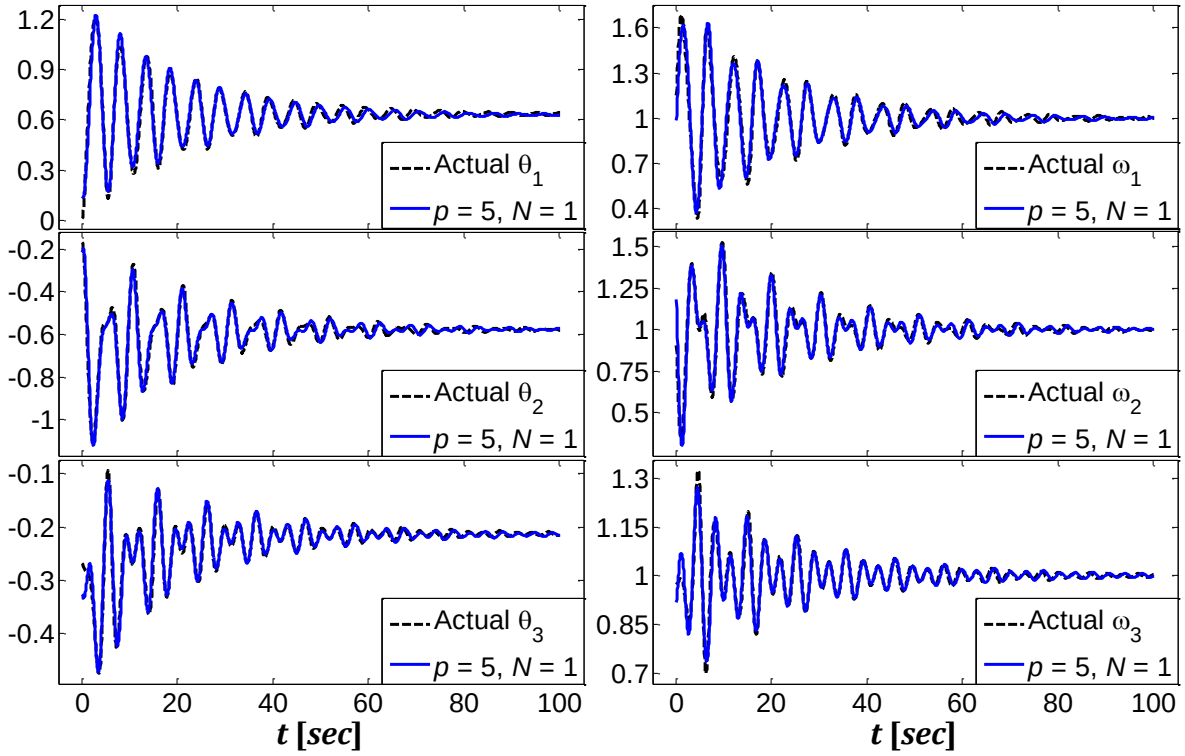

Figure 4.4: Optimal solution $p = 5, N = 1$ (solid blue) vs. Actual nonlinear response (dashed black)

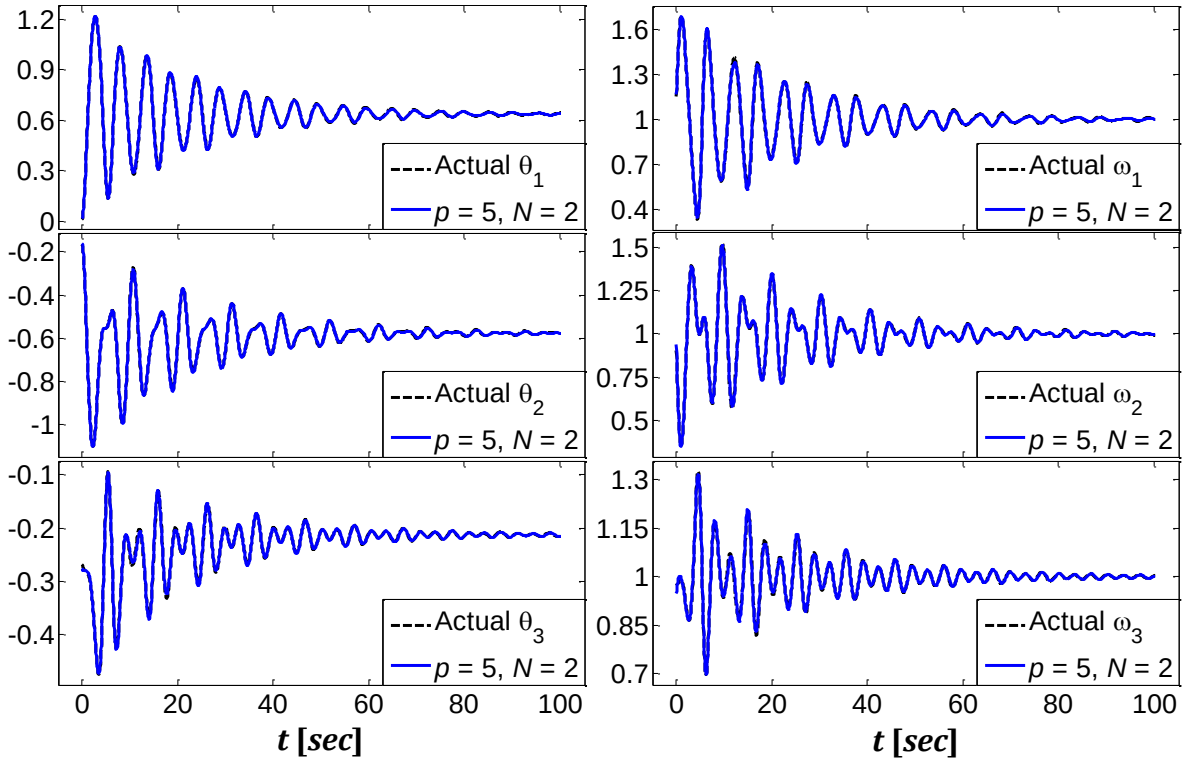

Figure 4.5: Optimal solution $p = 5, N = 2$ (solid blue) vs. Actual nonlinear response (dashed black)

The improvement can be seen in the estimated natural modes that converge closely to the actual eigenvalues of the system. If the state-space model were linear, the estimated modes from the 1st order solution would exactly match the actual eigenvalues of the system. This is not the case, so the normal form solution structure is used to fit the nonlinearity in the disturbance data. The estimated modes from the three solutions of the variable projection method (without the normal form solution structure, $N = 1$, and with 2nd and 3rd order combinations) are shown in Table 4.2.

	Actual Eigenvalues	Var. Proj. $p = 5, N = 1$	Var. Proj. $p = 5, N = 2$	Var. Proj. $p = 5, N = 3$
λ_1, λ_1^*	$-0.0396 \pm j1.845$	$-0.0373 \pm j1.824$	$-0.0396 \pm j1.837$	$-0.0408 \pm j1.845$
λ_2, λ_2^*	$-0.0448 \pm j1.235$	$-0.0485 \pm j1.204$	$-0.0497 \pm j1.229$	$-0.0452 \pm j1.234$
λ_3	-0.0688	-0.0900	-0.1164	-0.108

Table 4.2: Actual eigenvalues vs. estimated continuous-time poles

Excluding the real eigenvalue λ_3, the variable projection method with 3rd order modes had estimations closest to the actual eigenvalues. The method had more difficulty accurately estimating damping, $real\{\lambda_i\}$, compared to estimating frequencies, $imag\{\lambda_i\}$. The damping variables are more sensitive because they are numerically lesser (absolute) valued, than frequency variables. Therefore an equivalent numerical change in both damping and frequency, results in a greater percent change in damping.

4.5 Repeated Eigenvalues

Generally when a system has distinct eigenvalues, the solution structure obtained is a summation of damped cosines and exponentials. This type of solution structure is enforced by the traditional modal analysis methods discussed in Chapter 2. However, it is worthwhile examining the slightly modified solution structure obtained when a system possess repeated eigenvalues. Consider a basic state-space system $\dot{x}(t) = Ax(t)$, with two complex conjugate pairs of

eigenvalues. When the complex conjugate pairs are repeated, the solution structure would be that shown in (4.18).

$$x(t) = A_1 e^{\sigma_1 t}\cos(\omega_1 t + \phi_1) + tA_2 e^{\sigma_2 t}\cos(\omega_2 t + \phi_2) \tag{4.18}$$

Observe the second term in (4.18) is multiplied by time t. This type of solution structure is obtained from systems possessing repeated eigenvalues. A basic example is now provided which demonstrates a system with two complex conjugate pairs of repeated eigenvalues (four modes in total). A signal is constructed having the same formulation shown in (4.18). The values of the damping, frequency, amplitude and phase parameters are shown in Table 4.3, and the signal is recorded as data as shown in Figure 4.6. Initially, the data is analyzed using the variable projection method and with the standard solution structure of damped cosines. First, the optimization is performed using just one complex conjugate pair of eigenvalues (two modes), and a large error of 473.54 is obtained as seen in Table 4.3.

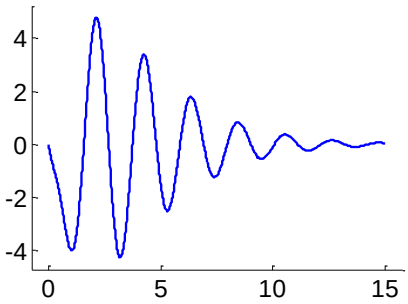

Figure 4.6: Repeated eigenvalue signal

	Actual	Prior Optimization (2 modes)	Prior Optimization (4 modes)	Optimization w/ repeated eigenvalue
σ_1	-0.5	-0.17	-0.50	-0.50
ω_1	3	2.83	3.00	3.00
A_1	4	5.08	2102	4.00
ϕ_1	90°	31.0°	153.2°	90°
σ_2	-0.5		-0.50	-0.50
ω_2	3		3.00	3.00
A_2	7		2100	6.98
ϕ_2	$-22.5°$		$-26.7°$	$-22.7°$
Error		473.54	8.6×10^{-8}	7.6×10^{-8}

Table 4.3: Actual repeated eigenvalue parameters vs. estimated parameters

Next, the optimization is performed again with a standard solution structure and two complex conjugate pairs of eigenvalues (four modes). The frequency variables ω_1 and ω_2 are not restricted to equal one another, but the optimization correctly identifies both frequencies to equal

3.00 rad/s. The same is true with the damping variables σ_1 and σ_2 which are not restricted to equal one another, but both are correctly identified as $-0.5\ 1/s$. The error in this formulation is 8.6×10^{-8} which would seem to indicate an accurate solution to the data had been obtained. However, the estimated mode shape amplitudes A_1 and A_2 in this solution each have magnitudes around 2100, where the actual amplitudes should equal 4 and 7 respectively. These unusually large amplitudes indicate something is wrong with this solution. Therefore lastly, the data is analyzed using a repeated eigenvalue solution structure, where one of the basis functions is multiplied by time as in (4.18). With this solution structure the optimization converges on the correct parameter values as is seen in Table 4.3.

4.6 Conclusion

This chapter first outlined the theory and method behind normal form analysis, which is first presented on a 3-bus power system. This theory motivated an extension to the nonlinear least-squares variable projection method. The extension adds no additional optimization variables to the nonlinear program, which allows for quick and accurate solutions. The 3-bus example was revisited to demonstrate the application of a normal form solution structure for estimating modes of data, and it was observed the estimated modes converged closely onto the actual eigenvalues of the system. The flexibility allowed by the variable projection function allows for the inclusion of basis functions not permitted by traditional modal analysis methods. This chapter exploited this flexibility to enforce the normal form solution structure, and similarly a basic example was demonstrated that enforced a repeated eigenvalue solution structure.

The normal form extension was partly inspired by analyzing results from Prony's method. For nonlinear data, Prony analysis requires a large number of modes p to be specified to get a decent fit. In analyzing the many modes, it was observed that some of the higher order modes appeared to closely resemble combinations of the lower order modes, such as high frequency modes being double the value of lower frequency modes. This observation coupled with the theory of normal form analysis gave rise to the ideas presented in this chapter.

Chapter 5

Modal Analysis Tool for Industry Use

A call for standalone ring-down analysis tools was recently made by the Western Electricity Coordinating Council (WECC), arising from the needs of system planners who perform modal analysis to estimate critical inter-area modes [64]. Current Prony analysis-based applications were reported to suffer from a variety of issues such as: dealing with nonlinear system responses, the sensitivity of results to selected time intervals, and the varying quality of estimations [64]. An application was requested to meet certain requirements, including: user ability to select time intervals by visual inspection of signals, data conditioning such as first-order detrending, and accurate estimations of oscillation frequencies, damping and mode shape. This request provided an opportunity to develop efficient modal analysis techniques into a working application for industry use.

A least-squares nonlinear optimization technique that has been overlooked by the electric power industry is the variable projection method [27,28]. This robust technique for estimating modal content of data was discussed in Chapter 3. It is necessarily true that a properly initialized nonlinear least-squares optimization program, will provide as good or better modal estimation to a data-set when compared to quick linear analysis techniques such as Prony analysis or other traditional methods. The variable projection method is the backbone of the modal analysis tool developed and discussed in this chapter.

The request by WECC for a standalone ring-down analysis tool was an opportunity to effectively deploy the modal analysis methods discussed in Chapters 2–4. The tool has been developed as a graphic user interface (GUI) in a MATLAB environment. Section 5.1 discusses the optimization algorithms used in implementation of the tool. Section 5.2 discusses various application requirements for the modal analysis tool as requested by power industry users. Section 5.3 demonstrates the modal analysis tool on real power system disturbance data.

5.1 Optimization Algorithms

This section seeks to provide insight into the methods and algorithms used in the modal analysis tool which were developed in the MATLAB environment. A publicly accessible variable projection code for nonlinear least-squares problems is available in [54,55]. Modifications to the code were made for multi-signal analysis, constrained optimization and for optimal computational performance.

The solver used in the optimization is dependent on the type of constraints imposed by the user. Section 3.2-D discussed the type of constraints used in the modal analysis problem. If the optimization is unconstrained or if bounds are placed on the damping and frequency optimization variables, σ and ω, then the MATLAB function *lsqnonlin* is called. The nonlinear least-squares function accepts problems having the formulation in (5.1).

$$\begin{aligned} &\min_x \ f(x) = \tfrac{1}{2} \left\| F(x) \right\|_2^2 \\ &s.t. \quad lb \le x \le ub \end{aligned} \tag{5.1}$$

The *lsqnonlin* function uses a trust region method and is based on the interior-reflective Newton method described in [66,67]. Consider the unconstrained trust region method at some point x with a trial step s in a trust region neighborhood N.

$$\begin{aligned} &\min_s \ f(x+s) \\ &s.t. \quad s \in N \end{aligned} \tag{5.2}$$

The current point is updated to be $x+s$; unless if $f(x+s) < f(x)$, which then the region of trust is shrunk and the trial step computation is recomputed. The nonlinear objective function in (5.2) is approximated by the first two terms of the Taylor expansion of f at the current point x, as shown in (5.3).

$$\begin{aligned} &\min_s \ \tfrac{1}{2} s^T H s + s^T g \\ &s.t. \quad \left\| Ds \right\| \le \Delta \end{aligned} \tag{5.3}$$

In (5.3), H is the symmetric Hessian matrix and g is the gradient vector of $f(x)$, D is a diagonal scaling matrix and Δ is a positive scalar. The trust region method in *lsqnonlin* requires the gradient to be supplied by the user (see Section 3.1 for the variable projection gradient). If the user does not supply the Hessian, then the algorithm computes a finite-difference approximation. Good algorithms exist for solving (5.3) [68,69]; where *lsqnonlin* uses a preconditioned conjugate gradient method to solve s. The process iterates with superlinear convergence towards a local minima. The trust region method in *lsqnonlin* can allow bounds or box constraints on the variables, which is handled efficiently using projected searches [70].

The function *lsqnonlin* is effective, but its restrictions on linear inequality or nonlinear constraints make it not always applicable. As detailed in Section 3.2-D, placing bounds on percent damping in the modal analysis problem introduces linear inequality constraints to the optimization. Restated here, consider placing lower and upper bounds, $\%d_{lb}$ and $\%d_{ub}$ on percent damping as shown below.

$$\%d_{lb} \leq -100\frac{\sigma}{\sqrt{\sigma^2+\omega^2}} \leq \%d_{ub}$$

If the frequency is greater than or equals to zero, then the bounds on percent damping can equivalently be expressed as the following linear inequality constraints.

$$\sigma + \frac{\%d_{lb}}{\sqrt{100^2-(\%d_{lb})^2}}\omega \leq 0$$

$$-\sigma - \frac{\%d_{ub}}{\sqrt{100^2-(\%d_{ub})^2}}\omega \leq 0$$

With these linear inequality constraints, the optimization will have the formulation in (5.4).

$$\min_x \ f(x) = \tfrac{1}{2}\left\| F(x) \right\|_2^2 \qquad (5.4)$$

$$s.t. \quad Ax \leq 0$$

For problems with the formulation in (5.4), the tool calls the MATLAB function *fmincon*. The constrained nonlinear optimization is best solved using an interior-point method. The interi-

or-point method is solved by means of adding a logarithmic term to the cost, called a barrier function, with the method described in [71] and [72]. The interior-point approximation of (5.4) is shown in (5.5).

$$\min_{x,\,s} \; f_\mu(x) = f(x) - \mu \sum_i \ln(s_i) \tag{5.5}$$

$$s.t. \quad Ax + s = 0$$

In (5.5), μ is a small positive scalar, sometimes called the barrier parameter. Slack variables s are introduced, and they are restricted to be positive. The approximate problem (5.5) is a sequence of equality constrained problems that are easier to solve than the original inequality constrained problem in (5.4). A step is determined at each iteration of the algorithm using either a Newton step or a conjugate gradient step using a trust region. If the user does not supply the Hessian matrix, then by default it is updated using a BFGS approximation [73]. At each iteration, the algorithm decreases a merit function (5.6).

$$f_\mu(x) + v \left\| Ax + s \right\| \tag{5.6}$$

The parameter v may increase with each iteration in order to force the solution towards feasibility. If an attempted step does not decrease the merit function, the algorithm rejects the attempted step and attempts a new step. The method iterates until a local minima is found. It is observed that convergence is slower for problems of the type in (5.4) compared to (5.1).

5.2 Application Requirements

This section provides discussion concerning application requirements and user requests for developing the modal analysis tool for industry use.

A common data format was recently adopted by the WECC Joint Synchronized Information Subcommittee [65]. Standalone ring-down analysis tools are required to support this data format in both CSV and XML file formats. The data format follows a new standard for signal/channel headings and units, start date and time, etc. The ring-down tool should accommo-

date actual synchrophasor data and simulated data. It should accommodate transient simulations with reported data entries not equally spaced in time. Additionally, simulated data may contain two data entries at the same time during simulated switching events. Actual synchrophasor data may contain bad data points, recorded as NaN or Inf. In summary, the ring-down analysis tool must accept the newly adopted data format in CSV or XML file format, must handle time entry issues in simulated data and must intelligently patch data drop-outs from actual synchrophasor data.

The modal analysis tool was requested to accommodate users of varying levels of experience. The tool should be simple enough to run without much initialization, but should also allow advanced control and guidance by experienced users. For example, the number of modes, or model order, in the analysis should be determined automatically, but it should also be adjustable for more advanced users. Modal analysis methods like Prony analysis, require the number of modes to be predetermined by the user; however, many users would prefer for the model order to be determined automatically. In response to this, the developed tool uses the Matrix Pencil method discussed in Section 2.4-B. The Matrix Pencil method uses a fixed threshold value to determine an appropriate model order for the data. The estimates from the Matrix Pencil method are then used as initial conditions for the variable projection method. Prony analysis and the Eigensystem Realization Algorithm can similarly be used for determining initial conditions if desired.

No initial condition generator is guaranteed to find the ideal initial conditions and model order 100% of the time, as data sets of vastly different sizes, types, and levels of noise will be encountered. After computing initial conditions the variable projection optimization will start and will terminate when a local minimizer is found. Inevitably, the user may wish to redo the analysis by: adding or subtracting modes, adjusting the number of complex and real-valued modes, adding or adjusting constraints, adding or removing polynomial trending, or adjusting the time interval for which the analysis takes place. To hedge against this inevitability, some flexibility has been incorporated into the tool for the user.

If the user is unsatisfied with results obtained from the first trial of the analysis, they can rerun the optimization under different circumstances. It makes sense to rerun the application with the previously solved solution available as initial conditions. This flexibility allows the user to guide the initial conditions and optimization using his or her intuition of the problem. A Fast Fourier transform is incorporated into the tool, which allows users to visually inspect significant frequencies missing in the solution. An example demonstrating these features is presented in Section 5.3.

5.3 Example

In this section, an example is presented which highlights many of the features of the developed modal analysis tool. Synchrophasor data was collected during a large Western Interconnection disturbance. Four different units of measure were observed, voltages [kV], angles [°], and frequencies [Hz] were collected at 14 buses, and the power flow [MW] along 16 transmission lines was collected. The frequencies and angles are shown as deviations from the reference bus angle and frequency. All of these signals/channels are simultaneously shown in Figure 5.1.

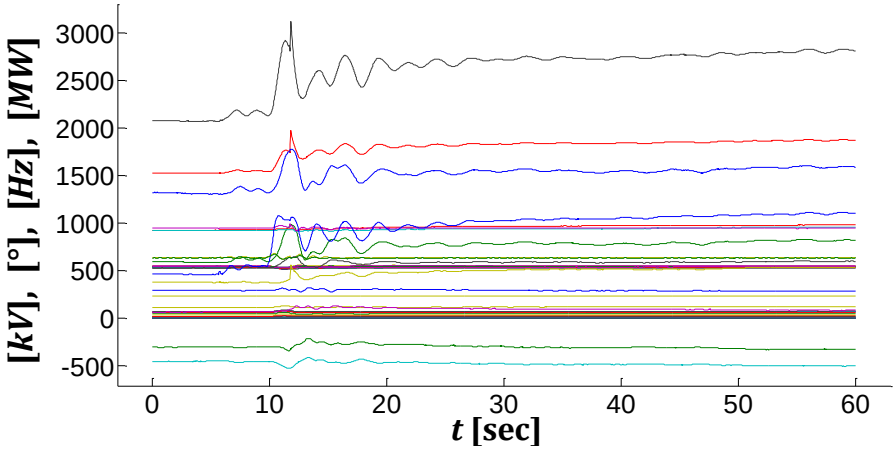

Figure 5.1: Western Interconnect disturbance

The disturbance happens around 12 seconds and settles out around 30 seconds. This disturbance data will be used for demonstrating the GUI's developed for the modal analysis tool. Figure 5.2 displays the GUI developed for initializing the analysis.

Several features discussed throughout the preceding chapters are made available in the tool. First note there are 58 channels of data, and 1801 time sample points. Polynomial data detrending is made available with the 'Polynomial trend' vector. By default 'Polynomial trend' is '[0]', implying a dc-offset will be included in the analysis. First order and higher polynomial trending terms can also be included in the 'Polynomial trend' vector. For example, setting 'Polynomial trend' to '[0 1 3]' includes a dc-offset, a linear term and a cubic polynomial trending term. A similar approach is used with the 'Normal form order' vector (discussed in Chapter 4). By default this vector is '[1]', implying only first order modes are considered and there is no normal form solution structure imposed.

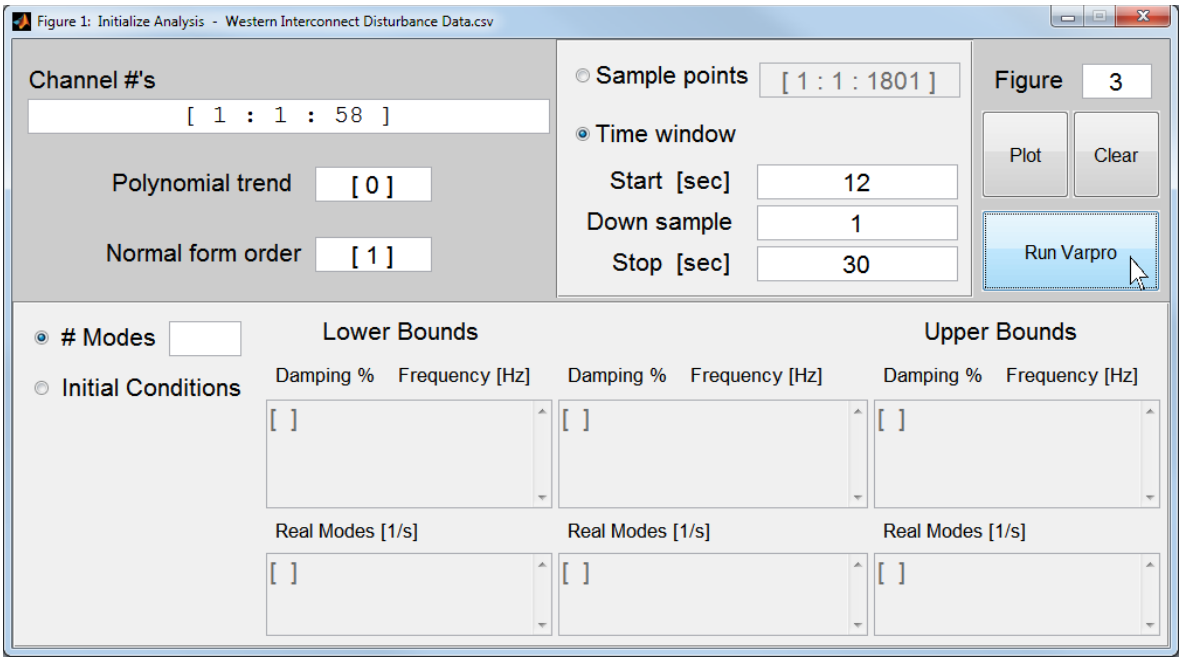

Figure 5.2: GUI for initializing modal analysis

The time window for analysis is changed to 12 to 30 seconds (541 time-sample points). The 'Down sample' integer is by default '1', implying there will be no down-sampling of the data. If 'Down sample' were set to '2', for example, then the analysis would use every other data-point between 12 and 30 seconds. The '# Modes' is left blank as the Matrix Pencil algorithm will be left to decide the model order. All 58 signals are weighted by the inverse of their stand-

ard deviations and multi-signal analysis is performed. The estimations from the Matrix Pencil method are used as initial conditions for the variable projection method, which uses the trust region method to solve the optimization. The results are reported in the GUI shown in Figure 5.3.

The primary role of the GUI in Figure 5.3 is for plotting and viewing results. From the analysis, the Matrix Pencil method determined there to be three significant pairs of complex conjugate modes, i.e. six modes in total. With six modes, the variable projection method located an optimal objective value of 1415. In this example, the Matrix Pencil method provided good initial conditions as the objective value only decreased from 1445 to 1415 in the optimization. The solved damping percentage and frequency $[Hz]$ are reported in Figure 5.3, with frequency $= \frac{\omega}{2\pi}$ and damping $\% = -100\frac{\sigma}{\sqrt{\sigma^2+\omega^2}}$.

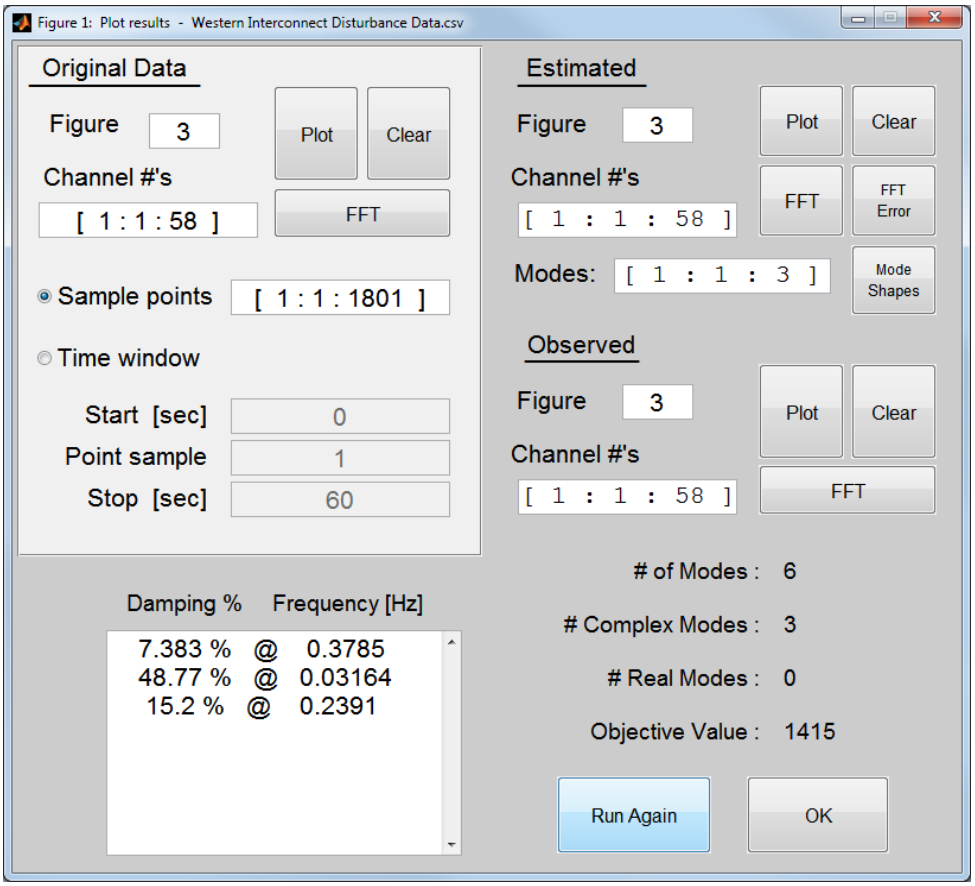

Figure 5.3: GUI displaying solution results

The bus voltages, angles and frequencies and transmission line-flows are plotted in Figure 5.4 showing the comparison of the estimated signals versus the actual data. The measurements are plotted with their dc-offsets subtracted, so as to better visualize the fast oscillations after the disturbance. The complex mode with high percent damping of 48.77% and low frequency of 0.03164 Hz (31.6 seconds per cycle) is contributing more of a trending fit to the data than capturing fast oscillatory behavior. This mode is responsible for the slow shaping that is most evident in the voltage plots in Figure 5.4, where the mode makes a half-cycle of 15.8 seconds before damping out.

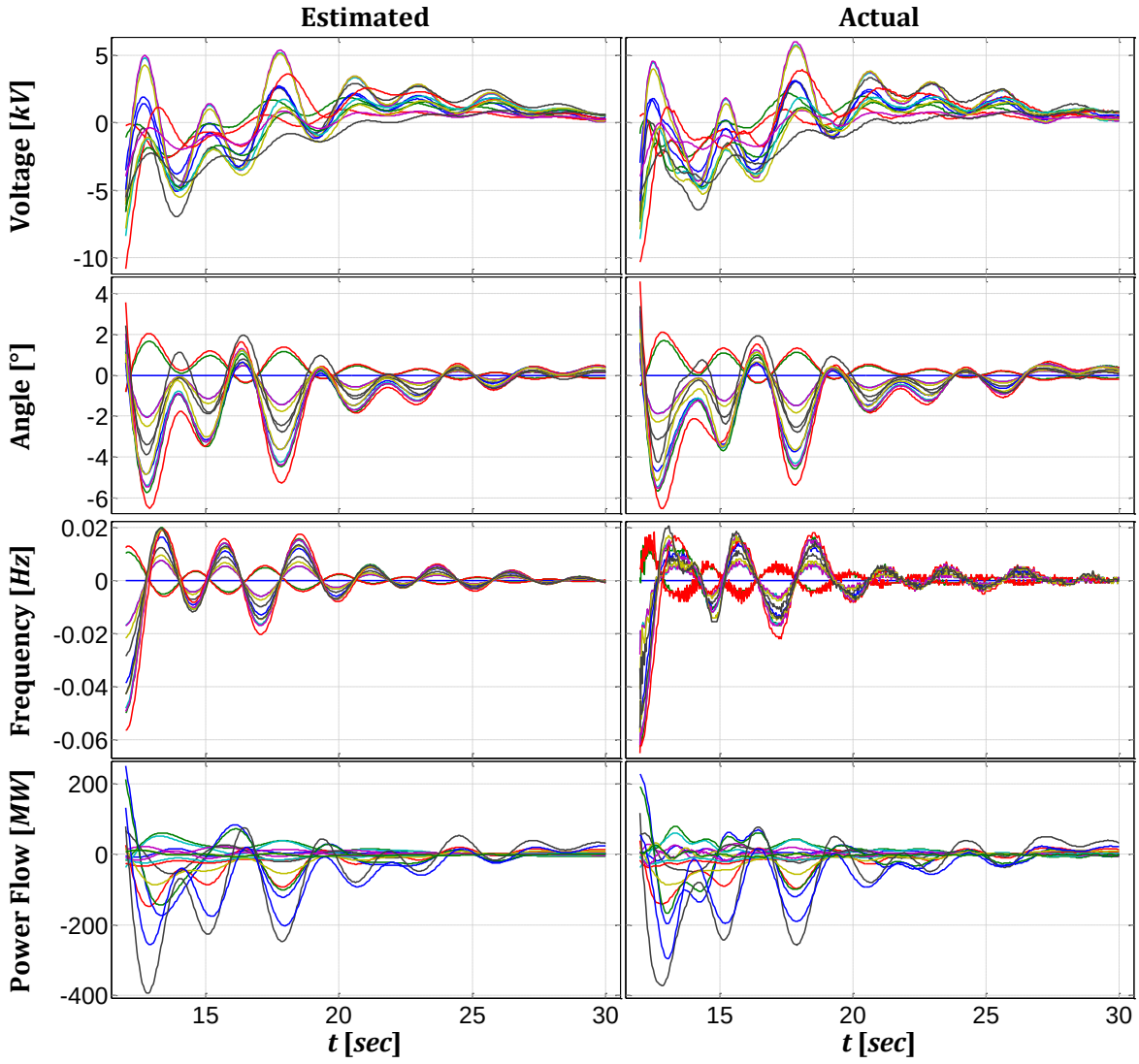

Figure 5.4: Estimated solution with 6 modes vs. Actual data

Some specifications can be given on the details of the optimization. Table 5.1 displays some of the computation times from the analysis, evaluated in MATLAB R2011a on a computer with 64-bit Intel i5-560M Dual Core CPU at 2.67 GHz with 4 GB of RAM.

Operation	Time [sec]
Matrix Pencil	2.734

Variable Projection	0.503
SVD	0.00467
Cost function evaluations	0.0128
Gradient evaluations	0.0924

Table 5.1: Evaluation times from example (trust region)

From the initial conditions, the optimization iterated 5 times before stopping due to convergence. Therefore the cost function, the gradient and the SVD (needed for calculating the cost and gradient), are all computed just 6 times each in the optimization. The variable projection method took just 0.503 seconds to converge. The table shows the cumulative time taken for the 6 evaluations of the SVD, cost function, gradient. These computations are not significantly time consuming. This leads to believe most of the evaluation time occurs elsewhere, such as when approximating the Hessian matrix or when solving the approximate trust region problem shown in (5.3).

A Fast Fourier Transform (FFT) can be taken of the weighted residual error between the actual data and estimated signal. In Figure 5.3, the 'FFT Error' button computes and plots this.

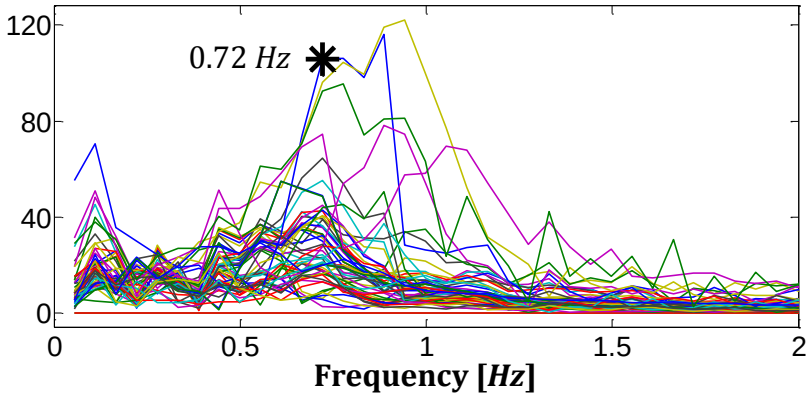

Figure 5.5: FFT of weighted residual error with 6 modes

The FFT plot in Figure 5.3 shows some considerable missing mode activity around frequency 0.72 Hz. The optimization can be computed once more using the 'Run Again' button in Figure 5.3 to reinitialize the analysis with the GUI shown in Figure 5.6.

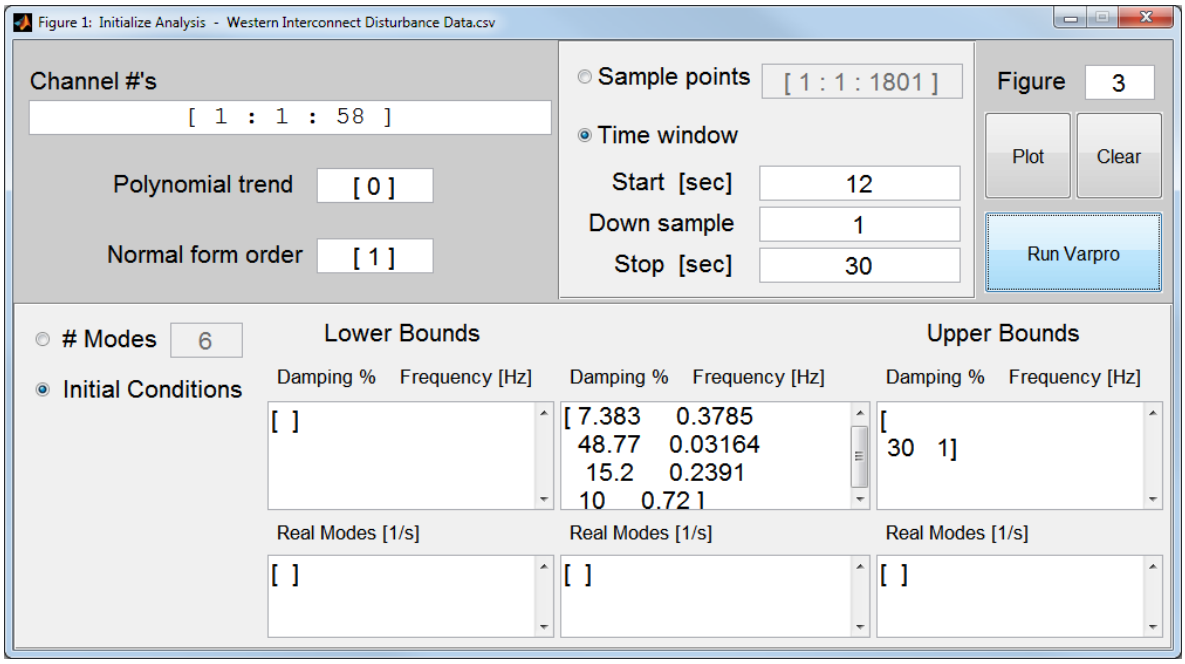

Figure 5.6: GUI for initializing modal analysis with

Now the mode solutions from the first trial are available as initial conditions. The prior mode solutions are kept; however, an additional complex mode at 0.72 Hz with an arbitrary damping around 10% is manually added to the initial conditions. Some power engineers may not like having an estimated mode with percent damping as high as 48.77%. In this new trial of the analysis, a constraint is placed on the 2nd mode to enforce percent damping less than or equal to 30%, and an arbitrarily upper bound of 1 Hz is placed on the frequency. This creates linear inequality constraints in the optimization, thus an interior-point method must be used. The initial conditions for the 2nd mode are not in the problem's feasible space, thus before calling the interior-point method, the damping of the 2nd mode is lowered so that the percent damping is 30% and the initial conditions are feasible. This step is completely optional, but it may provide for faster convergence with a feasible starting point.

The variable projection optimization is ran once more, and the results are reported in Table 5.2. With the additional complex mode at 0.7206 Hz, and with the imposed constraint on the 2nd mode, the objective value decreased from 1415 before to 668.3. As a side point, it appears the mode at 0.72 Hz could resemble a harmonic double frequency of the 0.38 Hz mode, which motivates the normal forms approaches in Chapter 4.

$$\text{Objective Value} = 668.3$$

	Damping % and Frequency [Hz]	Mode Energy
Mode 1	6.997% @ 0.3835	203.11×10^3
Mode 2	30.00% @ 0.0316	169.70×10^3
Mode 3	12.51% @ 0.2440	185.41×10^3
Mode 4	12.71% @ 0.7206	16.57×10^3

Table 5.2: Mode estimations and Mode energy

Some specifications can be given on the details of the optimization. Table 5.3 displays some of the computation times from the analysis.

Operation	Time [sec]
Variable Projection	5.049
SVD	0.054
Cost function evaluations	0.294
Gradient evaluations	1.839

Table 5.3: Evaluation times from example (interior-point)

From the initial conditions, the optimization iterated 24 times before stopping due to convergence. The cost function, gradient and SVD were evaluated 81 times each. The remaining evaluation time occurs elsewhere in the interior-point overhead.

The FFT of the weighted residual error using 8 modes to fit the data is shown in Figure 5.7. Comparing the FFT of Figure 5.7 to the FFT of Figure 5.5 reveals a good portion of modal activity around 0.72 Hz has been suppressed. There remains some modal content around 1 Hz but it is not drastically larger in magnitude then other frequencies. Therefore, as a judgment call, 8 modes is likely a sufficient number for estimating the most dominant modes in the data.

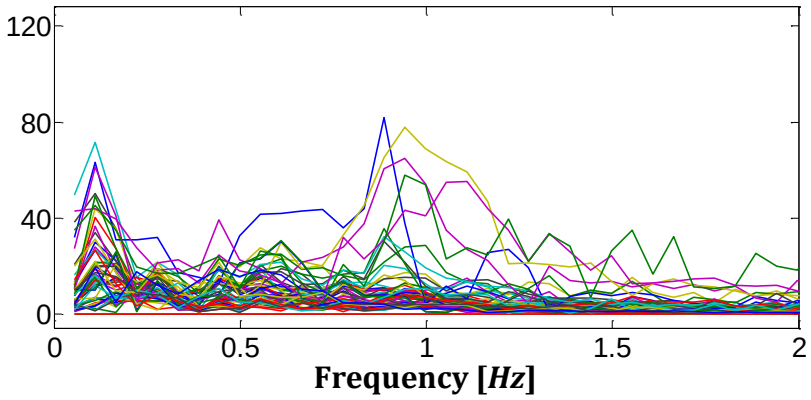

Figure 5.7: FFT of weighted residual error for 8 modes

The new estimated signals are shown in Figure 5.9; and the weighted error between the solution with 6 modes versus 8 modes is shown in Figure 5.8. The weighted errors show that the solution with 8 modes has a lower least-squares error to the data. The 8 mode solution would have had even less error if the 30% upper limit on the 2nd mode percent damping had not been enforced.

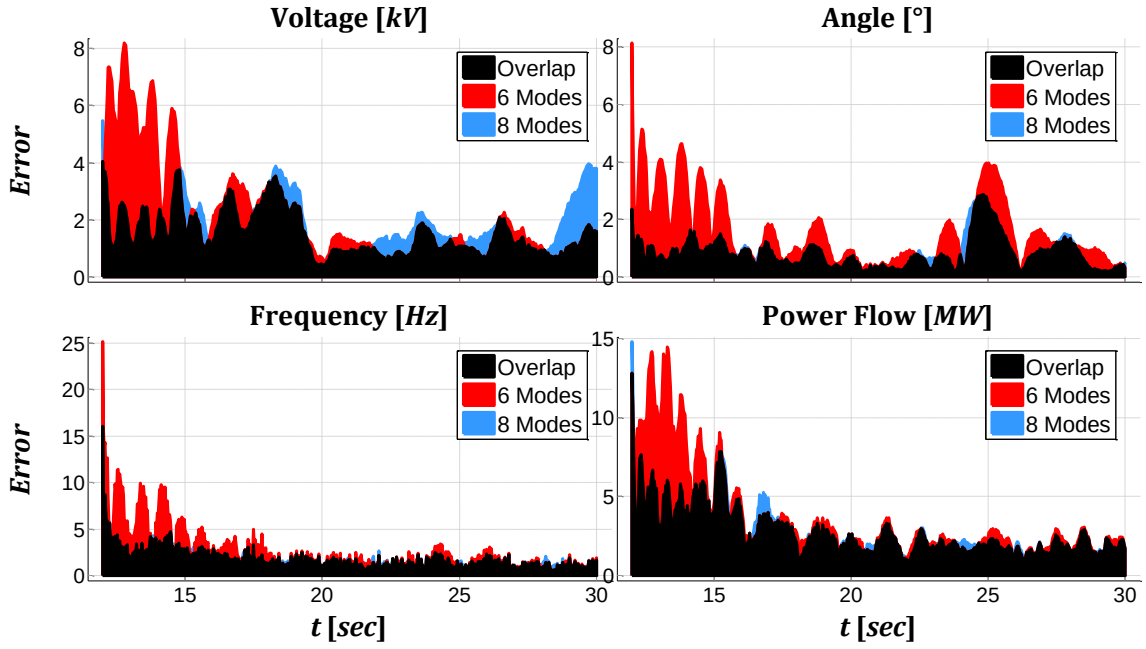

Figure 5.8: Weighted residual error comparison for 6 modes vs. 8 modes

In Figure 5.9, the measurements are again displayed with their dc-offsets subtracted. The 8 mode solution appears to be very accurate for just 4 mode pairs simultaneously estimating 58 signals.

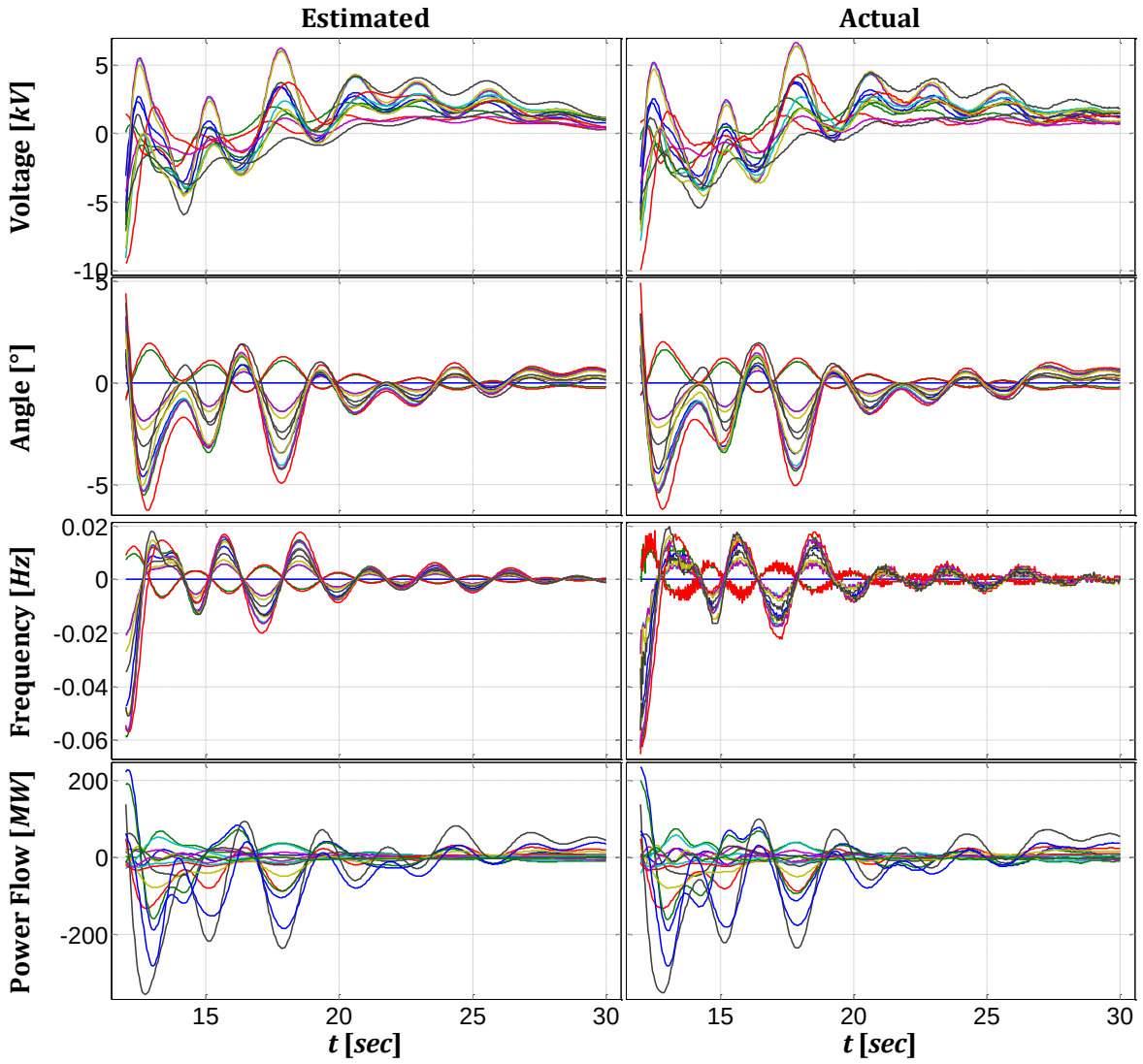

Figure 5.9: Estimated solution with 8 modes vs. Actual data

One last feature of the modal analysis tool, is its mode shape plotting function. The mode shapes are the coefficients or participation factors in each of the 58 data signals. Figure 5.10 displays polar plots of the mode shapes separated by the four signal types. Referring back to (2.3), the magnitude of the mode shapes in Figure 5.10 correspond to amplitudes A_i and the an-

gles correspond to the phase ϕ_i. The modes are numbered as shown in Table 5.2. Judging from the mode shape plots, it appears that the mode 1, with frequency 0.3835 Hz, appears most dominantly in the data. This is confirmed by the mode energy calculations in Table 5.2. The signal measurements have strong correlation with one another, which can be seen most obviously by how the mode shapes for modes 1 and 3 tend to align in phase.

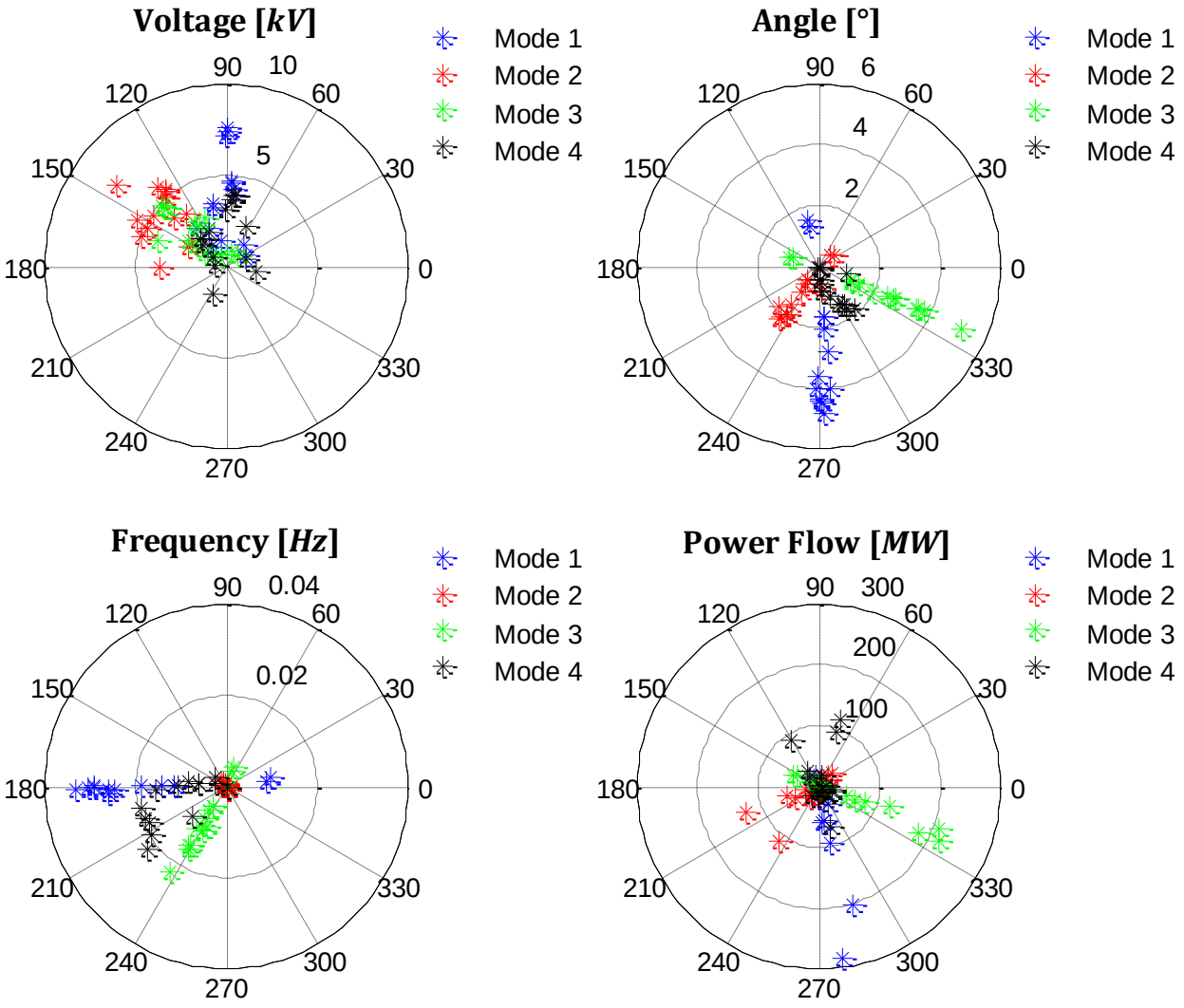

Figure 5.10: Mode shapes of 8 mode solution

This example offers a glimpse at effective modal analysis strategies and how the user-friendly modal analysis tool proves valuable for use by power industry members.

5.4 Conclusion

This chapter has detailed effective strategies for performing modal analysis as the culmination of methods presented in Chapters 2-4. Firstly, the Matrix Pencil method is used to determine the model order in the data and is used to provide good initial conditions for the optimization. The variable projection method is then used to improve on estimations provided by the Matrix Pencil method. The type of optimization algorithm used is dependent on the constraints imposed by the user. If the optimization is unconstrained or has direct bounds on the variables, then a trust region method is used. If the problem has linear inequality constraints due to bounds on percent damping, then an interior-point method is used.

This chapter has showcased a GUI developed for modal analysis, which has been distributed to power industry members with positive reception. The tool includes features for data-detrending, down-sampling, normal form solution structure, constrained optimization, and plotting features for the signal waveforms, Fast Fourier Transforms, and mode shapes. Importantly, the tool allows flexibility for user guidance and interaction, such as the ability to reuse solutions for initial conditions in a new trial. This guided approach greatly aids users of the tool, as this feature was lacking in prior methods used in industry. This chapter has demonstrated how a combination of optimization techniques and streamlined user guidance can create a superior approach for solving modal analysis problems.

This concludes the research work addressing power system modal analysis problems in this dissertation. The remaining chapters will examine security related issues in regards to solving the optimal power flow problem.

Chapter 6

Model Security

The work in this chapter is motivated by current nondisclosure policies for handling confidential power system models. Data for actual power system models are not easily accessible and typically require nondisclosure agreements. The purpose of the work in this chapter is to investigate methods that ensure security of confidential model details. A starting point for this work, borrows from recent publications on obscuring details of linear programs presumed to be solved in unsecure cloud computing environments.

Initial investigations of the model security problem examine a method for masking optimization problems. The masking method obscures model details and in this chapter is applied specifically to the optimal power flow (OPF) problem. The OPF problem is used to determine an optimal operating point in electric power systems. The OPF computes the optimal dispatch of resources needed to supply energy while accounting for a range of physical, engineering, and security constraints. It is solved nearly continuously in some form by grid operators. It takes a number of different forms depending on the particular objective and the scale of interest (planning vs. operations, economics, reliability, etc.). The mathematical representation varies from a linear program (DC OPF [76]) to a nonlinear, nonconvex mixed-integer program (security constrained AC OPF). Generally, all variants include an objective function (commonly quadratic or piece-wise linear), physical network constraints (the power flow equations) and imposed engineering limits (voltage magnitude, active and reactive power generation, transmission line-flow, etc.). The problem can be large with thousands of decision variables and tens of thousands of constraints. In this context, advances in the field of computing are of considerable interest.

An emerging paradigm in computer science and engineering is cloud computing [37]. Cloud computing provides subscribers shared access to powerful remote computing platforms; therefore, the potential to solve OPF problems remotely with cloud computing is an appealing possibility. Realistic OPF problems are nonlinear, nonconvex and very large, potentially having tens of thousands of buses and optimization variables; therefore, the OPF problem seems a prom-

ising candidate for remotely solving in the cloud. Security in cloud computing is a significant concern however [39-42]. With a shared computing platform comes the possible risk of attackers obtaining data sent to the cloud. In the case of power systems, this data is often confidential. Leaks of confidential data can be financially damaging and potentially threatening to national security. For this reason, cloud computing is currently not well suited for power system applications without further security advances.

This confidentiality motivates the need to improve OPF problem security. A masking process can be used to obscure the problem data such that an attacker with access to the masked problem cannot obtain confidential information. The masking process preserves the ability to obtain the original optimal solution; however, knowledge of the masking process details are required in order to extract the original solution from the masked solution. Existing research has investigated techniques for masking optimization problems [43,44]. In [43], the authors outline a systematic approach for masking a general linear program. The approach in [43] seems well suited for the linear DC OPF problem; however, some additions are needed. The approach in [43] only specifies a linear objective function whereas quadratic cost functions are necessary for many practical OPF problems. Furthermore, existing literature does not discuss dual solutions to the original unmasked problem. The dual variables in the OPF problem are important to power system operations, with a subset of them being the locational marginal prices in market contexts. The method in [43] does not obscure the number and type of facilities present in the problem. An additional computational concern is that the approach in [43] destroys problem sparsity, making solutions of large OPF problems computationally intractable. A masking approach that preserves sparsity in integer programs is described in [44]. This approach serves as inspiration to a similar approach for preserving sparsity in the masked OPF problems in this chapter.

This chapter presents a confidentiality preserving optimal power flow method. Several issues are addressed pertaining to the OPF, including dual variable calculations in Section 6.2, controlling the sparsity of a DC OPF in Section 6.3-B, imposing quadratic cost functions in Section 6.3-C, obscuring the number of system facilities in Section 6.4, and masking nonlinear constraints in Section 6.5. Initial focus is on the linear DC OPF and its relation to the existing litera-

ture on cloud computing security, and afterward the nonlinear AC OPF is addressed. An example of both the DC and full AC OPF is shown in Section 6.6.

6.1 DC Optimal Power Flow Problem Overview

The DC OPF uses a power flow model that is a linear approximation of the nonlinear power flow equations. There are four main approximations made in the DC OPF: the bus voltage magnitudes are all equal to one, the voltage angle differences are small so that $\cos(\delta_k - \delta_m) \approx 1$ and $\sin(\delta_k - \delta_m) \approx \delta_k - \delta_m$, the resistance for each branch is negligible and set to zero and all shunt elements are neglected. Reactive power at the loads and generators are not explicitly considered. The DC OPF can be written with linear constraints and quadratic cost function having the following form in (6.1).

$$\min_{P_g,\,\delta} \ \frac{1}{2} P_g^T \boldsymbol{D} P_g + d^T P_g \tag{6.1}$$

$$s.t. \qquad -P_g + \boldsymbol{B}\delta = -P_L$$

$$\delta_{ref} = 0$$

$$P_{g,min} \le P_g \le P_{g,max}$$

$$-P_{flow,max} \le diag(b_{br})\boldsymbol{A_{inc}}\delta \le P_{flow,max}$$

In the formulation for the DC OPF in (6.1), the optimization variables are P_g as the vector of generator power injections and δ as the vector of bus voltage angles. It has a quadratic cost function, where $\boldsymbol{D}$ is a diagonal matrix of generator quadratic cost coefficients and d is a vector of generator linear cost coefficients.

The first equality constraint enforces power balance at each bus. Here the bus susceptance matrix $\boldsymbol{B}$ is the imaginary part of the bus admittance matrix with shunt elements neglected. Reflecting common power system topology, the matrix $\boldsymbol{B}$ is typically sparse. The vector P_L contains the bus active power loads. It is important to note that in the formulation of (6.1), the power generated is in the delivering reference frame (i.e., P_g is nonnegative), and the bus loads are in

the receiving reference frame (i.e., P_L is also nonnegative). The second equality constraint enforces the bus voltage angle at the reference bus to be zero.

The first inequality constraint in (6.1), limits the power generation of each generator to be within a lower and upper bound. In this report, all powers generated are assumed to be greater than or equal to zero ($P_g \geq 0$), i.e. the generators cannot act as loads. The last constraint limits the power flow in both directions on each branch to be less than a maximum flow $P_{flow,max}$. Here the vector b_{br} contains the branch susceptances and $diag(b_{br})$ is the diagonal matrix with the vector b_{br} on the diagonal. The matrix $\boldsymbol{A_{inc}}$ is the bus-to-branch incidence matrix; this matrix has number of rows equal to the number of branches and number of columns equal to the number of buses. Each row has +1 in the column corresponding to the branch's "from" bus and -1 in the column corresponding to the branch's "to" bus.

6.2 Masking Primal and Dual Linear Programs

Recent research details a method for masking a linear program [43]. In this section, the masking method is briefly summarized and a method for recovering the unmasked dual variables is further developed. In Section 6.3, the masking method is applied to the DC OPF problem. The primal notation in this section is adopted from [43]. Start from the standard linear program primal (6.2a) and dual (6.2b) formulations [75].

$$\min_{x} \quad c^T x \qquad (6.2a)$$
$$s.t. \quad \boldsymbol{M_1} x = b_1$$
$$\boldsymbol{M_2} x \leq b_2$$
$$x \geq 0$$

$$\max_{u,v} \quad b_1^T u + b_2^T v \qquad (6.2b)$$
$$s.t. \quad \boldsymbol{M_1^T} u + \boldsymbol{M_2^T} v \leq c$$
$$v \leq 0$$

A random positive monomial matrix $\boldsymbol{Q}$ (i.e., a matrix containing exactly one non-zero entry per row and column) and a random positive vector r are used to hide the cost vector c and the optimization variable vector x, as shown in (6.3).

$$\min_{x} \quad c^T \boldsymbol{Q}(\boldsymbol{Q}^{-1}x + r) \tag{6.3}$$

$$s.t. \quad \boldsymbol{M_1 Q}(\boldsymbol{Q}^{-1}x + r) = b_1 + \boldsymbol{M_1 Q}\, r$$

$$\boldsymbol{M_2 Q}(\boldsymbol{Q}^{-1}x + r) \leq b_2 + \boldsymbol{M_2 Q}\, r$$

$$\boldsymbol{Q}^{-1}x + r \geq r$$

Substituting the masked variable $z = \boldsymbol{Q}^{-1}x + r$ and introducing the random positive diagonal matrix $\boldsymbol{S}$ yields the primal and dual problems (6.4a) and (6.4b).

$$\min_{z} \quad c^T \boldsymbol{Q} z \tag{6.4a}$$

$$s.t. \quad \boldsymbol{M_1 Q} z = b_1 + \boldsymbol{M_1 Q}\, r$$

$$\boldsymbol{M_2 Q} z \leq b_2 + \boldsymbol{M_2 Q}\, r$$

$$\boldsymbol{S} z \geq \boldsymbol{S} r$$

$$\max_{u,v} \quad (b_1 + \boldsymbol{M_1 Q} r)^T u + (b_2 + \boldsymbol{M_2 Q} r)^T v \tag{6.4b}$$

$$s.t. \quad (\boldsymbol{M_1 Q})^T u + (\boldsymbol{M_2 Q})^T v \leq (c^T Q)^T$$

$$v \leq 0$$

The inequality constraints in (6.4a) are converted to equality constraints through the introduction of slack variables z_{sl}. Denote the vector z' as the vector z augmented with the slack variables, $z'^T = [z^T\ z_{sl}^T]$. The cost function vector is augmented with zero entries corresponding to the slack variables, $c'^T = [c^T \boldsymbol{Q}\ \ 0 \ldots 0]$. The dual variable vectors u and v are consolidated into a single vector $u'^T = [u^T\ v^T]$.

The constraint notation is simplified by defining $\boldsymbol{M'}$ and b'.

$$\boldsymbol{M'} = \begin{pmatrix} \boldsymbol{M_1 Q} & \boldsymbol{0} \\ \boldsymbol{M_2 Q} & \\ \boldsymbol{-S} & \boldsymbol{A} \end{pmatrix} \qquad b' = \begin{pmatrix} b_1 + \boldsymbol{M_1 Q} r \\ b_2 + \boldsymbol{M_2 Q} r \\ \boldsymbol{-S} r \end{pmatrix}$$

Here matrix $\boldsymbol{A}$ is a random positive monomial matrix. The formulations in (6.4a) and (6.4b) can be rewritten as seen in (6.5a) and (6.5b).

$$\min_{z'} \quad c'^T z' \tag{6.5a}$$

$$s.t. \quad \boldsymbol{M'} z' = b'$$

$$z' \geq 0$$

$$\max_{u'} \quad b'^T u' \tag{6.5b}$$

$$s.t. \quad \boldsymbol{M'^T} u' \leq c'$$

Lastly the matrix $\boldsymbol{M}'$ and vector b' are hidden using any nonsingular matrix $\boldsymbol{P}$ and a random positive monomial matrix $\boldsymbol{T}$ with $\boldsymbol{M}'' = \boldsymbol{P}\boldsymbol{M}'\boldsymbol{T}$, $b'' = \boldsymbol{P}b'$ and $c''^T = c'^T\boldsymbol{T}$. The $\boldsymbol{P}$ matrix takes linear combinations of the rows in the constraint equations. The $\boldsymbol{T}$ matrix scales and permutes the columns of the constraint matrix $\boldsymbol{M}'$ and cost function vector c'^T. The new primal optimization variable vector is $z'' = \boldsymbol{T}^{-1}z'$ and new dual optimization variable vector is $u'' = (\boldsymbol{P}^T)^{-1}u'$. The linear program is in its final masked primal (6.6a) and dual (6.6b) forms.

$$\min_{z''} \; c''^T z'' \qquad (6.6a)$$
$$s.t. \quad \boldsymbol{M}''z'' = b''$$
$$z'' \geq 0$$

$$\max_{u''} \; b''^T u'' \qquad (6.6b)$$
$$s.t. \quad \boldsymbol{M}''^{\boldsymbol{T}} u'' \leq c''$$

The original optimal primal variable vector x^* can be recovered after solving masked problems (6.6a) and (6.6b) with $\boldsymbol{T}z''^* = z'^* = \left[z^{*T}\ z_{sl}^{*T}\right]^T$ and $x^* = \boldsymbol{Q}(z^* - r)$. The original optimal dual variable vectors u^* and v^* can be solved by $\boldsymbol{P}^T u''^* = u'^* = \left[u^{*T}\ v^{*T}\right]^T$.

6.3 Masking A DC OPF Problem

In this section, the masking techniques developed in Section 6.2 are specifically applied to the DC OPF problem outlined in Section 6.1. First the composition of the matrices in (6.2a) and (6.2b) is specified. In Section 6.3-B a method for constructing the $\boldsymbol{P}$ matrix used in (6.6a) and (6.6b) is detailed. The linear program masking is then extended to include a quadratic cost function which is required for typical OPF problems.

A. Problem Setup

The DC OPF problem in (6.1) is formulated in terms of a standard linear program as in (6.2a). Note that the bus angles δ in (6.1) are free variables; however, the standard linear program formulation in (6.2a) requires nonnegative variables. Therefore the bus angles δ are represented as the difference of two nonnegative variables, $\delta = \delta^+ - \delta^-$ where $\delta^+,\ \delta^- \geq 0$. In

(6.2a), the optimization variable vector $x^T = \begin{bmatrix} P_g^T & \delta^{+T} & \delta^{-T} \end{bmatrix}$. There are $n_g + 2n_b$ elements in x, where n_g is the number of generators and n_b is the number of buses. The quadratic cost function terms in (6.1) are temporarily neglected, but they will be revisited in Section 6.3-C. In (6.2a), the linear cost coefficient vector $c^T = [d^T \quad 0 \ldots 0]$, where d is the linear generator cost coefficients in (6.1), and there are $2n_b$ number of zeros. Consider the equality constraints of (6.2a), $\boldsymbol{M_1} x = b_1$. For DC OPF problems, the matrix $\boldsymbol{M_1}$ is constructed as follows.

$$\boldsymbol{M_1} = [-\boldsymbol{E_g} \quad \boldsymbol{B} \quad -\boldsymbol{B}]$$

Matrix $\boldsymbol{M_1}$ has n_b rows. The $n_b \times n_g$ matrix $\boldsymbol{E_g}$ has a single +1 entry in each column for the rows corresponding to buses with generators and has zeros elsewhere. The $n_b \times n_b$ matrix $\boldsymbol{B}$ is the bus susceptance matrix with shunt elements neglected. In the equality constraints of (6.2a), the column vector $b_1 = -P_L^T$, where P_L is the vector of bus loads. Therefore, the rows of $\boldsymbol{M_1}$ and b_1 enforce power balance at each bus. Consider the inequality constraints of (6.2a), $\boldsymbol{M_2} x \leq b_2$. The matrix $\boldsymbol{M_2}$ is constructed as follows.

$$\boldsymbol{M_2} = \begin{bmatrix} \boldsymbol{I_g} & \boldsymbol{0} & \boldsymbol{0} \\ -\boldsymbol{I_g} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & diag(b_{br})\boldsymbol{A_{inc}} & -diag(b_{br})\boldsymbol{A_{inc}} \\ \boldsymbol{0} & -diag(b_{br})\boldsymbol{A_{inc}} & diag(b_{br})\boldsymbol{A_{inc}} \end{bmatrix}$$

Matrix $\boldsymbol{M_2}$ has $2n_g + 2n_{br}$ rows where n_{br} is the number of branches. Here I_g is the $n_g \times n_g$ identity matrix. As described in Section 6.1, the vector b_{br} contains the branch susceptances, and $diag(b_{br})$ is the diagonal matrix with the vector b_{br} on its diagonal. The matrix $\boldsymbol{A_{inc}}$ is the bus-to-branch incidence matrix, and $diag(b_{br})\boldsymbol{A_{inc}}$ is an $n_{br} \times n_b$ matrix. In the inequality constraints of (6.2a), column vector $b_2 = \begin{bmatrix} P_{g,max}^T & -P_{g,min}^T & P_{flow,max}^T & P_{flow,max}^T \end{bmatrix}^T$. Vectors $P_{g,max}$ and $P_{g,min}$ are the upper and lower power generation limits, and $P_{flow,max}$ is the branch power flow limits.

In (6.1), there is one equality constraint enforcing the reference/slack bus angle to equal zero, $\delta_{ref} = \delta_{ref}^{+} - \delta_{ref}^{-} = 0$. This constraint is best handled by simply removing the two columns in $\boldsymbol{M_1}$ and $\boldsymbol{M_2}$ corresponding to variables δ_{ref}^{+} and δ_{ref}^{-}.

As an example, consider defining $\boldsymbol{Q}$, $\boldsymbol{S}$ and $\boldsymbol{A}$ as identity matrices and r as the zero-valued vector in steps (6.2)–(6.5). For this DC OPF example, matrix $\boldsymbol{M'}$ and vector b' from (6.5a) is constructed for the IEEE 30-bus test system [77], and is shown in Figure 6.1.

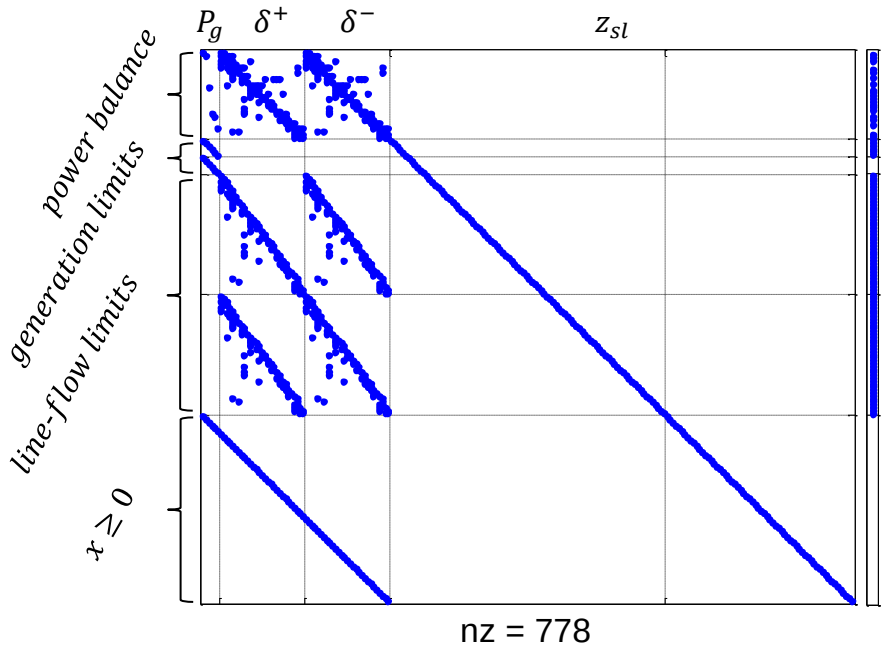

Figure 6.1: M' matrix and b' vector for IEEE 30-bus system

Figure 6.1 indicates the typical problem structure of a DC OPF problem. The topology of the system including bus connections and number of facilities is clearly evident, and these features will be made private by the masking process. In OPF problems, the dual optimization variables are important. The dual optimization variable vector u in (6.2b) corresponds to the equality constraints of the primal problem, and v corresponds to the inequality constraints. The locational marginal prices (LMPs) are the Lagrange multipliers of the power balance equality constraints; therefore the Lagrange multipliers u are the LMPs. In Section 6.2, the method for recovering the unmasked dual variables was shown.

B. Constructing P and T Matrices for a DC OPF

The masking steps of equations (6.2a) through (6.5a) are straightforward as the construction of matrices $\boldsymbol{Q}$, $\boldsymbol{A}$ and $\boldsymbol{S}$ and vector r only require generation of random positive numbers and permutations. However, creation of the nonsingular $\boldsymbol{P}$ matrix used in (6.6a) is not as straightforward in the case of the DC OPF problem. The $\boldsymbol{P}$ matrix is left multiplied by the constraint matrix $\boldsymbol{M}'$ in (6.5a) to permute, scale and take linear combinations of the rows. A carelessly constructed $\boldsymbol{P}$ matrix can largely increase the solve-time of the fully masked DC OPF (6.6a). A positive monomial $\boldsymbol{T}$ matrix is also right multiplied by $\boldsymbol{M}'$ to permute and scale the columns. This section details some effective approaches for carefully constructing the $\boldsymbol{P}$ and $\boldsymbol{T}$ matrices for (6.6a), specifically $\boldsymbol{M}'' = \boldsymbol{P}\boldsymbol{M}'\boldsymbol{T}$.

A good initial approach in constructing the $\boldsymbol{P}$ and $\boldsymbol{T}$ matrices involves calculating the symmetric reverse Cuthill-McKee ordering [78,79] of the Laplacian bus susceptance matrix $\boldsymbol{B}$ as seen in (6.1). Doing so tends to place the nonzero elements in $\boldsymbol{B}$ closer to the diagonal thus decreasing its bandwidth. This is demonstrated on the IEEE 30-bus test system [77] shown in Figures 6.2 and 6.3.

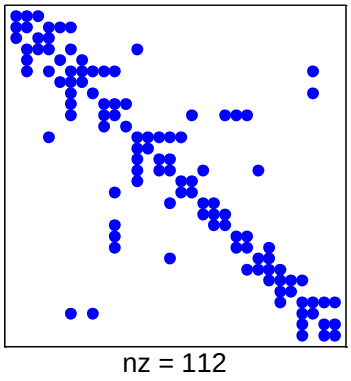

Figure 6.2: IEEE 30-bus system, Bus susceptance matrix *B*

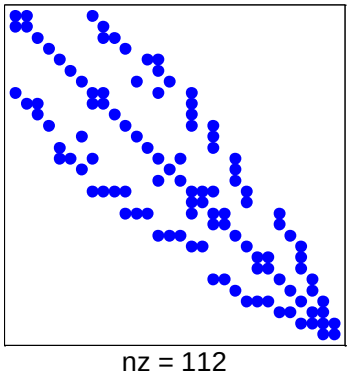

Figure 6.3: IEEE 30-bus system, Symmetric Reverse Cuthill-McKee Ordering

Consider the matrix $\boldsymbol{M}'$ in (6.5a) for DC OPF problems. To clearly illustrate the typical sparse structure of $\boldsymbol{M}'$ in a DC OPF problem, let matrices $\boldsymbol{Q}$, $\boldsymbol{S}$ and $\boldsymbol{A}$ be identity matrices and r be the zero-valued vector, instead of their respective random counterparts. A color-coded spy-

plot displaying the nonzero elements and values in matrix $\boldsymbol{M}'$ and vector $\boldsymbol{b}'$ from (6.5a) is shown in Figure 6.4, for the IEEE 30-bus test system [77].

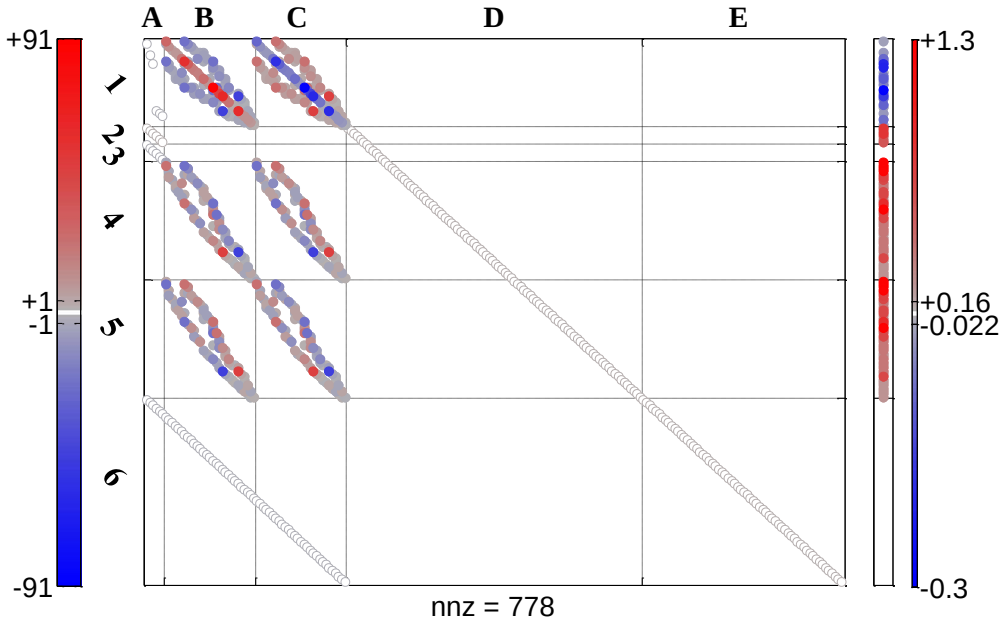

Figure 6.4: Nonzero elements of M' and b' in 30-bus system

The typical structure of matrix $\boldsymbol{M}'$ for the DC OPF can be seen in Figure 6.4 with obvious distinguishable sections; grid lines and labels were inserted to emphasize these sections. Generally, the numerical values of matrix $\boldsymbol{M}'$ in Figure 6.4 will be secure by properly constructing the random matrices $\boldsymbol{Q}$, $\boldsymbol{A}$ and $\boldsymbol{S}$ and vector r discussed in 6.2. An attacker could, however, easily identify the topology of the system by viewing matrix $\boldsymbol{M}'$ in Figure 6.4.

The final step of the masking process requires left multiplying a nonsingular matrix $\boldsymbol{P}$ and right multiplying a random positive monomial matrix $\boldsymbol{T}$, specifically $\boldsymbol{M}'' = \boldsymbol{P}\boldsymbol{M}'\boldsymbol{T}$ in (6.6a). One of the purposes of the $\boldsymbol{P}$ matrix is to increase the number of nonzero elements in order to obscure the number of facility types and the topology of the power system. Using Figure 6.4 as example, it can be seen that the columns in sections A-C are much more dense than the columns in D and E. Naively taking random row combinations of $\boldsymbol{M}'$ would increase the density of the columns in A-C much more than columns in D and E. By counting the number of denser columns, an attacker could determine the number of generators and buses, as there would be n_g+2n_b-2 denser

columns. Having the number of generators and buses, an attacker knowing the DC OPF formulation could then calculate the number of system branches n_{br}. The nonsingular matrix $\boldsymbol{P}$ must carefully be constructed to prevent against this. The structure of $\boldsymbol{M'}$ in Figure 6.4 can be exploited to perform linear row operation tricks for the DC OPF that more evenly distribute nonzero column densities. The row operation tricks will be shown illustratively.

First a matrix $\boldsymbol{P_1}$ is multiplied by $\boldsymbol{M'}$ to get matrix $\boldsymbol{P_1M'}$ in Figure 6.5. The nonzero column density of $\boldsymbol{P_1M'}$ in Figure 6.5 has been more evenly distributed compared to $\boldsymbol{M'}$ in Figure 6.4. The number of nonzero elements has also been reduced from 778 to 523. With some manipulation, matrix $\boldsymbol{P_1M'}$ in Figure 6.5 can be permuted into a type of nonsquare banded matrix as in Figure 6.6. From Figure 6.6, a Dulmage-Mendelsohn decomposition [80,81] can be performed as in Figure 6.7. These steps will be consolidated into matrices $\boldsymbol{P_2}$ and $\boldsymbol{T_1}$ to get $\boldsymbol{P_2P_1M'T_1}$ in Figure 6.7. The matrices in Figures 6.6 and 6.7 have nice fill-reducing structures for solving the partially masked OPF problem.

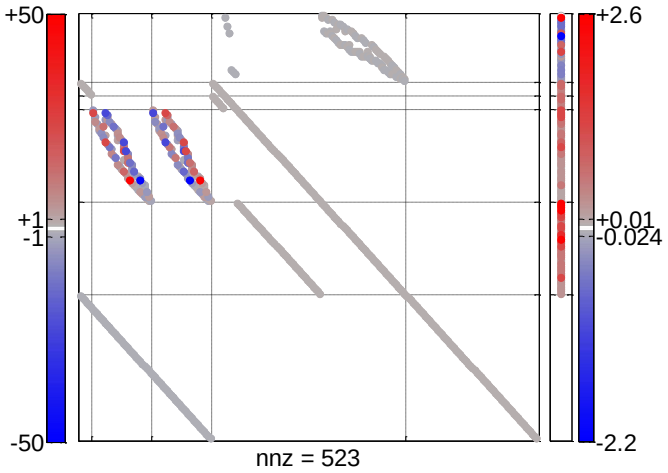

Figure 6.5: Nonzero elements of $\boldsymbol{P_1M'}$ in 30-bus system

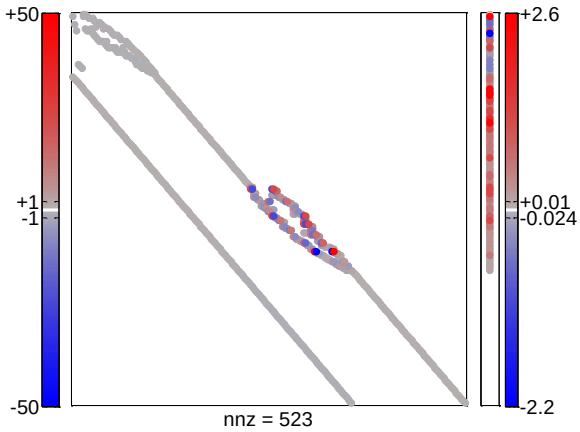

Figure 6.6: Nonsquare banded matrix

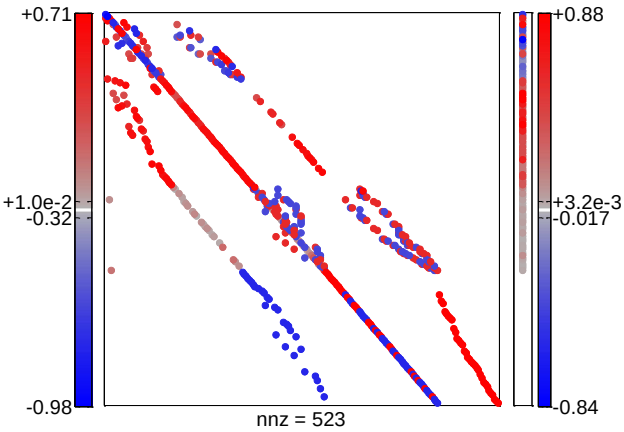

Figure 6.7: Dulmage-Mendelsohn decomposition

Each row in Figure 6.7 has also been divided by its 2-norm to have more uniform values. In the 30-bus system, dividing by each row's norm lowered the condition number of the constraint matrix $\boldsymbol{M'}$ from 550.4 to 79.5. The masking process can be continued to obscure sensitive information like the number of generators, buses and branches. The matrix in Figure 6.7 is well suited for performing linear row combinations that will increase the number of nonzeros.

A nonmonomial matrix $\boldsymbol{P_3}$ is constructed and it is left multiplied by $\boldsymbol{P_2P_1M'T_1}$ from Figure 6.7. In this example, matrix $\boldsymbol{P_3}$ was carefully constructed to take linear combinations of the rows so that each row in Figure 6.8 has a number of nonzeros greater than or equal to 7; chosen because 7 was the maximum row nonzero density in Figure 6.7. Alternatively a random sparse $\boldsymbol{P}$ matrix can quickly and easily be created, though this matrix would likely not evenly distribute nonzero row and column densities. The matrix $\boldsymbol{P_3}$ should be created in such a way to maintain balance between sparsity and security in the problem. The linear row combinations increase the number of nonzeros from 523 in Figure 6.7 to 1530 in Figure 6.8. The matrix $\boldsymbol{P_3}$ could also be constructed to enforce the percentage of nonzeros in the columns.

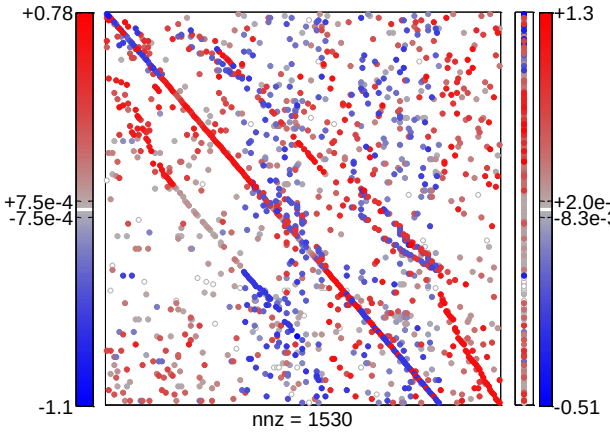

Figure 6.8: Nonzero elements of $P_3P_2P_1M'T_1$ in 30-bus system

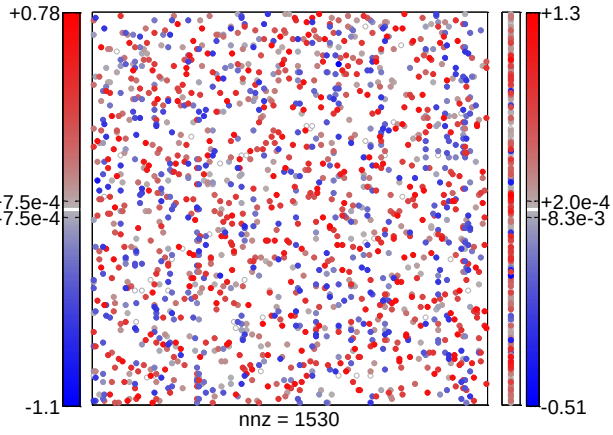

Figure 6.9: Nonzero elements of $M''=PM'T$ in 30-bus system

The final step of permuting the rows and columns of the matrix in Figure 6.8 is performed to completely obscure the matrix structure. Any monomial matrix $\boldsymbol{P_4}$ can be created for permut-

ing rows, and a positive monomial matrix $\boldsymbol{T_2}$ is needed for permuting columns. Altogether, the final obscured matrix is $\boldsymbol{M'' = P_4 P_3 P_2 P_1 M' T_1 T_2 = PM'T}$, as in Figure 6.9.

The matrix $\boldsymbol{M'' = PM'T}$ in Figure 6.9 is the constraint matrix in (6.6a) and (6.6b). The multi-stage process of creating the matrix $\boldsymbol{P}$ outlined above hides the original structure and values of $\boldsymbol{M'}$ while maintaining sparsity. No columns or rows have too great or too low of a non-zero density, which prevents against attacks of that nature. According to [43] the numerical values of the masked problem are sufficiently secure. The system structure in $\boldsymbol{M'}$ should be mostly hidden in $\boldsymbol{M''}$, though to what extent is uncertain.

C. Quadratic Cost Function

In this section, the method for masking a quadratic cost function is detailed, which is typically required in OPF problems. In (6.1), a quadratic cost function was shown for OPF problems, $\frac{1}{2} P_g^T \boldsymbol{D} P_g + d^T P_g$. Rewriting this cost function in terms of $x^T = \begin{bmatrix} P_g^T & \delta^{+T} & \delta^{-T} \end{bmatrix}$, as was done in Section 6.3-A, changes the primal and dual problems to those in (6.7a) and (6.7b).

$$\min_{x} \quad \frac{1}{2} x^T \boldsymbol{C} x + c^T x \tag{6.7a}$$

$$s.t. \quad \boldsymbol{M_1} x = b_1$$

$$\boldsymbol{M_2} x \leq b_2$$

$$x \geq 0$$

$$\max_{u,v,w} \quad -\frac{1}{2} \begin{bmatrix} u \\ v \\ w \end{bmatrix}^T \begin{bmatrix} M_1 C^{-1} M_1^T & M_1 C^{-1} M_2^T & M_1 C^{-1} \\ M_2 C^{-1} M_1^T & M_2 C^{-1} M_2^T & M_2 C^{-1} \\ C^{-1} M_1^T & C^{-1} M_2^T & C^{-1} \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} + \begin{bmatrix} b_1 + \boldsymbol{M_1 C^{-1}} c \\ b_2 + \boldsymbol{M_2 C^{-1}} c \\ \boldsymbol{C^{-1}} c \end{bmatrix}^T \begin{bmatrix} u \\ v \\ w \end{bmatrix} - \frac{1}{2} c^T \boldsymbol{C^{-1}} c$$

$$s.t. \quad v \leq 0 \tag{6.7b}$$

$$w \geq 0$$

Like before, the linear cost vector $c^T = [d^T \;\; 0 \ldots 0]$, but there is now also the quadratic cost matrix $\boldsymbol{C}$ with dimensions $(n_g + 2n_b - 2) \times (n_g + 2n_b - 2)$.

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{D} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}$$

Following the masking procedure outlined in Section 6.2, (6.3) through (6.6a) and (6.6b) gives the final masked primal (6.8a) and dual (6.8b) problem for a quadratic cost function.

$$\min_{z''} \quad \frac{1}{2} z''^T \boldsymbol{C}'' z'' + c''^T z \tag{6.8a}$$

$$s.t. \quad \boldsymbol{M}'' z'' = b''$$

$$z'' \geq 0$$

$$\max_{u'',v''} -\frac{1}{2}\begin{bmatrix} u'' \\ v'' \end{bmatrix}^T \begin{bmatrix} \boldsymbol{M}''\boldsymbol{C}''^{-1}\boldsymbol{M}''^{\boldsymbol{T}} & \boldsymbol{M}''\boldsymbol{C}''^{-1} \\ \boldsymbol{C}''^{-1}\boldsymbol{M}''^{\boldsymbol{T}} & \boldsymbol{C}''^{-1} \end{bmatrix} \begin{bmatrix} u'' \\ v'' \end{bmatrix} + \begin{bmatrix} b'' + \boldsymbol{M}''\boldsymbol{C}''^{-1}c'' \\ \boldsymbol{C}''^{-1}c'' \end{bmatrix}^T \begin{bmatrix} u'' \\ v'' \end{bmatrix} - \frac{1}{2} c''^T \boldsymbol{C}''^{-1} c''$$

$$s.t. \quad v'' \geq 0 \tag{6.8b}$$

Most of the variables above were already defined in Section 6.2 with a few important distinctions. In the linear cost function of Section 6.2, $c'^T = [c^T \boldsymbol{Q} \quad 0 \ldots 0]$ with appended zero entries corresponding to the slack variables; however, now due to the quadratic cost function this changes to $c'^T = [(c^T \boldsymbol{Q} - r^T \boldsymbol{Q}^{\boldsymbol{T}} \boldsymbol{C}^{\boldsymbol{T}} \boldsymbol{Q}) \quad 0 \ldots 0]$ and $c''^T = c'^T \boldsymbol{T}$. There is also the new quadratic cost matrix $\boldsymbol{C}'$.

$$\boldsymbol{C}' = \begin{bmatrix} \boldsymbol{Q}^{\boldsymbol{T}} \boldsymbol{C} \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}$$

Matrix $\boldsymbol{C}'$ has appended zero entries corresponding to the slack variables, and $\boldsymbol{C}'' = \boldsymbol{T}^{\boldsymbol{T}} \boldsymbol{C}' \boldsymbol{T}$. The original optimal primal solution to (6.7a) can be obtained by $\boldsymbol{T} z''^* = z'^* = \begin{bmatrix} z^{*T} & z_{sl}^{*T} \end{bmatrix}^T$ and $x^* = \boldsymbol{Q}(z^* - r)$. The original optimal dual solution to (6.7b) can be obtained by $\boldsymbol{P}^{\boldsymbol{T}} u''^* = u'^* = \begin{bmatrix} u^{*T} & v^{*T} & w^{*T} \end{bmatrix}^T$.

6.4 Further Obscuring

In this section, additional methods for masking an OPF are discussed. Further obscuring methods may be necessary in order to hide the number and type of facilities in the system.

A. Adding Constraints to the Cost Function

Even in the fully obscured problem, there remains sensitive information that could be extracted. By counting the number of nonzero entries in the masked linear cost coefficient c'' (or the masked quadratic cost coefficient $\boldsymbol{C}''$), an attacker could determine the number of generators present in the system. If the zero entries in c'' and $\boldsymbol{C}''$ were filled in with nonzero entries then that particular attack could be prevented. This can be accomplished by adding the linear constraints to the cost function, or by adding the linear constraints squared in the case of a quadratic cost function. Consider a problem with quadratic cost function and linear constraints as in (6.9).

$$\begin{aligned} \min_{x} \quad & \tfrac{1}{2} x^T \boldsymbol{C} x + c^T x \\ s.t. \quad & \boldsymbol{M} x = b \\ & x \geq 0 \end{aligned} \tag{6.9}$$

The equality constraints in (6.9) can be rearranged as $\boldsymbol{M}x - b = 0$, and quadratic combinations of the constraints can be created using a symmetric matrix $\boldsymbol{W}$.

$$(\boldsymbol{M}x - b)^T \boldsymbol{W} (\boldsymbol{M}x - b) = 0$$

This can be added to the objective function in (6.9) without affecting the optimization.

$$\begin{aligned} \max_{x} \quad & \tfrac{1}{2} x^T (\boldsymbol{C} + 2\boldsymbol{M}^T \boldsymbol{W} \boldsymbol{M}) x + (c^T - 2b^T \boldsymbol{W} \boldsymbol{M}) x + b^T \boldsymbol{W} b \\ s.t. \quad & \boldsymbol{M} x = b \\ & x \geq 0 \end{aligned} \tag{6.10}$$

The optimal solutions to problems (6.9) and (6.10) are equivalent. However, the optimization problem of (6.10) has quadratic combinations of the equality constraints added to the objective function, which produces more nonzero entries in the objective function. This approach can be used to prevent against attacks that count the number of nonzero entries in the objective function. In the case of the OPF problem, this obscures the number of generators present.

Figure 6.10 shows the very sparse quadratic, positive semidefinite cost matrix associated with the IEEE 30-bus test system from Figure 6.1. Figure 6.11 shows the same quadratic cost matrix but obscured by using the technique in this section; in the complete masking process this matrix would also be permuted. From Figure 6.10, it can be inferred the system has 6 generators, with the first 6 optimization variables corresponding to the power generation variables. From Figure 6.11, it is not obvious how many generators the system has, or which of the optimization variables are the power generation variables.

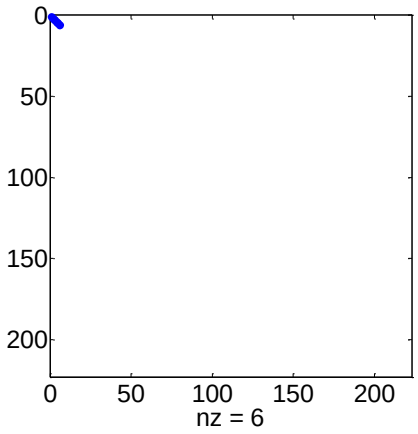

Figure 6.10: IEEE 30-bus quadratic cost matrix

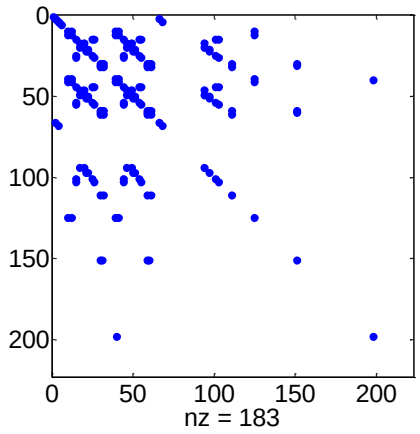

Figure 6.11: Obscured quadratic cost matrix

B. Fictitious Buses, Generators and Loads

The approach in Section 6.4-A masks the number of generators in the cost function, but the total number of rows and columns in $\boldsymbol{M''}$ contains information about the number of system facilities. To be exact the number of rows in $\boldsymbol{M''}$ equals $3n_b+3n_g+2n_{br}-2$ and the number of col-

umns equals $4n_b+4n_g+2n_{br}-4$. An attacker with knowledge of just one of the variables n_b, n_g or n_{br}, could then calculate the two other unknown variables.

To obscure the number of rows and columns, fictitious buses, generators and loads can be created. A fictitious bus can be created by splitting an existing line. This way of adding fictitious buses does not alter the solution of the OPF, but naively adding fictitious generators can alter the solution. However, a fictitious generator with very large cost would not be dispatched; therefore, the addition of an expensive generator should not affect the solution of the OPF. Alternatively, an offsetting fictitious load and generator pair at the same bus with equivalent upper and lower generation limits will also not affect the solution of the OPF.

Figure 6.12 displays the one-line diagram of the IEEE 14-bus system [77], and Figure 6.13 displays the same system with a fictitious bus, generator and line added.

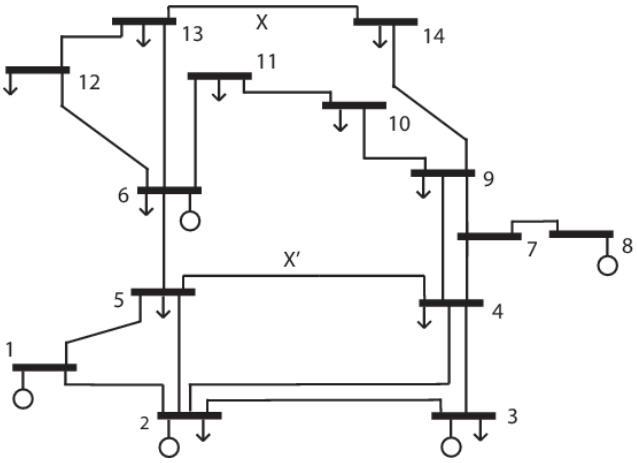

Figure 6.12: IEEE 14-bus system

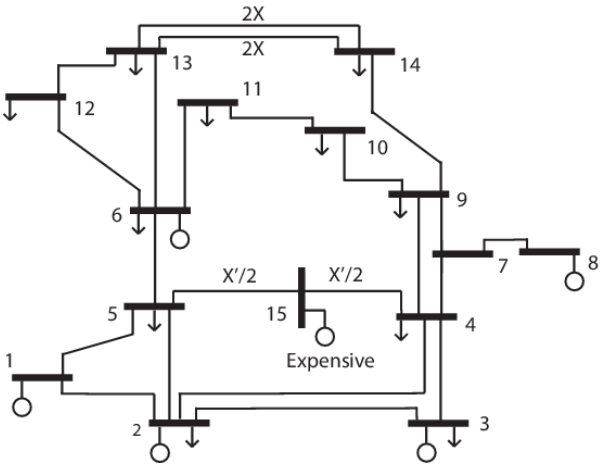

Figure 6.13: IEEE 14-bus system with fictitious bus, generator and lines

In this basic example, a fictitious bus was added between buses 4 and 5, and the new transmission line reactances are set equal to ½ the reactance of the original line connecting buses 4 and 5. At the fictitious bus-15, an expensive generator is placed. In the DC OPF problem, this generator is never dispatched. Lastly, the line connecting buses 13 and 14 is split into two parallel lines, each having double the reactance of the original line. For the DC OPF problem, these modifications to the system do not change the optimal operating point.

6.5 Masking Nonlinear Constraints

The previous methods used for masking linear programs [43], can be extended to masking nonlinear constraints such as those in the AC OPF. Start with a nonlinear optimization problem having quadratic cost function, shown in (6.11)

$$\min_{x} \quad \frac{1}{2}x^T \boldsymbol{C} x + c^T x \tag{6.11}$$
$$s.t. \quad f_{eq}(x) = 0$$
$$f_{ineq}(x) \leq 0$$
$$x \geq 0$$

Substitute the masked variable $z = \boldsymbol{Q}^{-1}x + r$, and neglect the constant cost term created.

$$\min_{z} \quad \frac{1}{2}z^T \boldsymbol{Q^T C Q} z + (c^T \boldsymbol{Q} - r^T \boldsymbol{Q^T C^T Q})z \tag{6.12}$$
$$s.t. \quad f_{eq}\big(\boldsymbol{Q}(z - r)\big) = 0$$
$$f_{ineq}(\boldsymbol{Q}(z - r)) \leq 0$$
$$z \geq r$$

The nonlinear inequality constraints in (6.12) are converted to equality constraints through the introduction of slack variables z_{sl}. Denote the optimization variable vector z' as the prior vector z augmented with the slack variables, $z'^T = [z^T \; z_{sl}^T]$. Define the linear cost coefficient vector $c'^T = [(c^T \boldsymbol{Q} - r^T \boldsymbol{Q^T C^T Q}) \quad 0 \ldots 0]$ with appended zero entries corresponding to the slack variables. Define the matrix

$$\boldsymbol{C'} = \begin{bmatrix} \boldsymbol{Q^T C Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}$$

with appended zero entries corresponding to the slack variables. The nonlinear equality constraints are constructed as $f'_{eq}(z')$.

$$f'_{eq}(z') = \begin{bmatrix} f_{eq}\big(\boldsymbol{Q}(z-r)\big) \\ \Big[f_{ineq}\big(\boldsymbol{Q}(z-r)\big)\Big] + \boldsymbol{A}z_{sl} \\ -\boldsymbol{S}z - \boldsymbol{S}r \end{bmatrix} = 0$$

Here $\boldsymbol{A}$ is a random positive monomial matrix and $\boldsymbol{S}$ is a random positive diagonal matrix. The formulation in (6.12) can then be rewritten as (6.13).

$$\min_{z'} \frac{1}{2} z'^T \boldsymbol{C}' z' + c'^T z' \tag{6.13}$$
$$s.t. \quad f'_{eq}(z') = 0$$
$$z' \geq 0$$

A random positive monomial matrix $\boldsymbol{T}$ scales and permutes the optimization variables with $z'' = \boldsymbol{T}^{-1} z'$. The objective function is modified as $c''^T = c'^T \boldsymbol{T}$ and $\boldsymbol{C}'' = \boldsymbol{T}^{\boldsymbol{T}} \boldsymbol{C}' \boldsymbol{T}$.

There are some issues to consider when constructing the $\boldsymbol{P}$ matrix for nonlinear constraints. Recall, in the linear case, the $\boldsymbol{P}$ matrix could be any nonsingular matrix. For the nonlinear case, a nonmonomial $\boldsymbol{P}$ matrix would create linear row combinations of the nonlinear constraints in $f'_{eq}(z')$. It is true that the global solution of the optimization is unchanged when multiplying a nonsingular $\boldsymbol{P}$ matrix by the nonlinear constraints, however there can be issues of convergence to global versus local minimum. In testing several trials, it was observed that when a nonmonomial $\boldsymbol{P}$ matrix was used the optimization often converged to a local minimum, even when the unmasked nonlinear problem converged to a global minimum using the same initial conditions. Conversely, when using a monomial $\boldsymbol{P}$ matrix the minimum that was found in the unmasked problem was equivalently found in the masked problem. Using a nonsingular $\boldsymbol{P}$ matrix and a positive monomial $\boldsymbol{T}$ matrix, the nonlinear constraints $f'_{eq}(z')$ are transformed as follows.

$$f'_{eq}(\boldsymbol{T}\boldsymbol{T}^{-1} z') = f'_{eq}(\boldsymbol{T} z'')$$
$$f''_{eq}(z'') = \boldsymbol{P} \cdot f'_{eq}(\boldsymbol{T} z'')$$

The final masked problem with nonlinear constraints is (6.14).

$$\min_{z''} \frac{1}{2} z''^T \boldsymbol{C}'' z'' + c''^T z'' \tag{6.14}$$
$$s.t. \quad f''_{eq}(z'') = 0$$
$$z'' \geq 0$$

As before the original optimal primal solution to (6.11) can be obtained by $\boldsymbol{T} z''^* = z'^* = \left[z^{*T}\ z_{sl}^{*T}\right]^T$ and $x^* = \boldsymbol{Q}(z^* - r)$. The original optimal dual solution can be obtained by $\boldsymbol{P^T} u''^* = u'^* = \left[u^{*T}\ \ v^{*T}\ \ w^{*T}\right]^T$.

6.6 Numeric Example

The IEEE 30-bus network [77] is used for the example in the following section. It is presented as both the DC OPF and the full nonlinear AC OPF to show the successful recovery of the original optimal solution from both masked problems. The IEEE 30-bus network has partially been illustrated as an example in Section 6.3-B, Figures 6.3-6.9. It will be examined more closely here. The 30-bus network has 6 generators and 41 branches. Some of the branch-flow limits were tightened to enforce binding constraints in the OPF examples.

A quadratic cost function is assumed in the DC OPF having the formulation shown in (6.1). The masking process requires the generation of a random positive monomial matrix $\boldsymbol{A}$, a random positive diagonal matrix $\boldsymbol{S}$ and a random positive vector r. The $\boldsymbol{Q}$ matrix will be determined using the symmetric reverse Cuthill-McKee ordering. The original primal variable x is substituted with $x = \boldsymbol{Q}(z - r)$. All inequality constraints are converted to equality constraints via introduction of slack variables creating constraint matrix $\boldsymbol{M'}$ as in (6.5a) and in Figure 6.14. The $\boldsymbol{M'}$ matrix in Figure 6.14 was constructed with random numbers and permutations where applicable. A nonsingular matrix $\boldsymbol{P}$ is carefully constructed using the steps outlined in Section 6.3-B, and a random positive monomial matrix $\boldsymbol{T}$ is generated. The constraint matrix $\boldsymbol{M''} = \boldsymbol{PM'T}$ is in the form of (6.8a) and shown in Figure 6.15.

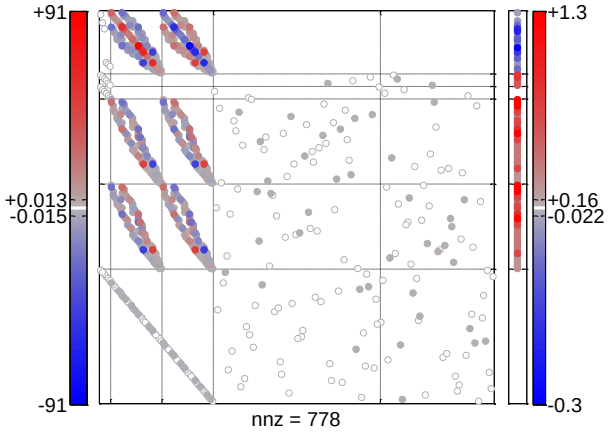

Figure 6.14: Nonzero elements of M' and b' in 30-bus system

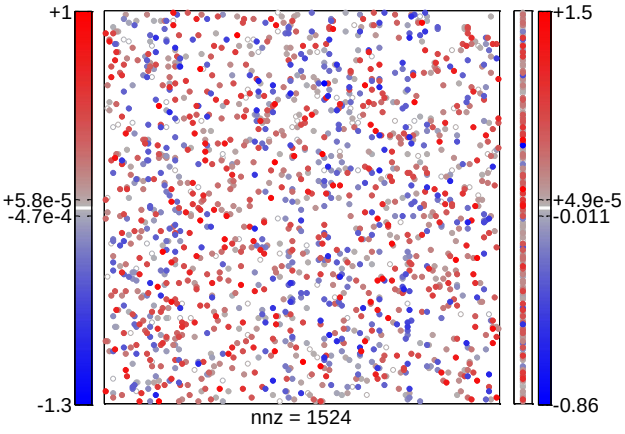

Figure 6.15: Nonzero elements of $M''=PM'T$ in 30-bus system

Further obscuring is performed by adding squared constraints to the cost function as outlined in Section 6.4-A. The fully masked DC OPF is passed to a quadratic program solver such as *quadprog* in MATLAB or *cplex* in GAMS [82], and the masked primal and dual variables are calculated. The original optimal primal variables can be solved with $\boldsymbol{T}z''^{*} = z'^{*} = \left[z^{*T}\ z_{sl}^{*T}\right]^{T}$ and $x^{*} = \boldsymbol{Q}(z^{*} - r)$. The original optimal objective value equals $f^{*} = \frac{1}{2}x^{*T}\boldsymbol{C}x^{*} + c^{T}x^{*}$, as in (6.7a). The primal variable vector $x^{T} = \left[P_g^{T} \quad \delta^{+T} \quad \delta^{-T}\right]$, so the final optimization variables are P_g and $\delta = \delta^{+} - \delta^{-}$. The original optimal dual variables can be solved with $\boldsymbol{P^{T}}u''^{*} = u'^{*} = \left[u^{*T} \quad v^{*T} \quad w^{*T}\right]^{T}$, and the LMPs are the Lagrange multipliers u.

The linear and quadratic cost coefficients of the IEEE 30-bus DC OPF are shown in Table 6.1.

Bus #	$\mathbf{c}_2$ [$/MWh²]	$\mathbf{c}_1$ [$/MWh]	$\mathbf{c}_0$ [$]
1	0.02	2	0
2	0.0175	1.75	0
13	0.025	3	0
22	0.0625	1	0
23	0.025	3	0
27	0.00834	3.25	0

Table 6.1: IEEE 30-bus Generator Costs

The binding branch flow constraints from the DC OPF are shown in Table 6.2 and the optimal power generations, bus angles and locational marginal prices are shown in Table 6.3. The recovered solution from the masked problem matches the solution from the unmasked problem as well as the optimal solution given by MATPOWER [83].

Optimal Cost: $f^* = \$567.25$

From Bus	To Bus	P_f $[MW]$	$\lvert P_{max} \rvert$ $[MW]$	μ $[\$/MWh]$
1	3	16.00	16.00	3.21
22	21	20.00	20.00	2.37

Table 6.2: DC OPF Binding Branch Flow Constraints

Bus #	Power Gen. P_g $[MW]$	Bus Angles δ	LMPs $[\$/MWh]$	Bus #	Power Gen. P_g $[MW]$	Bus Angles δ	LMPs $[\$/MWh]$
1	29.99	0°	3.20	16		-3.57°	4.39
2	56.19	-0.48°	3.72	17		-3.99°	4.48
3		-1.74°	4.77	18		-4.33°	4.32
4		-2.05°	4.43	19		-4.72°	4.38
5		-1.99°	3.99	20		-4.54°	4.41
6		-2.46°	4.27	21		-3.66°	5.3
7		-2.90°	4.16	22	17.23	-3.43°	3.15
8		-2.99°	4.26	23	18.48	-2.04°	3.92
9		-3.31°	4.43	24		-2.75°	3.54
10		-3.76°	4.51	25		-0.71°	3.78
11		-3.31°	4.43	26		-1.48°	3.78
12		-2.73°	4.30	27	41.27	1.00°	3.94
13	26.04	-0.64°	4.30	28		-2.21°	4.23
14		-3.44°	4.25	29		-0.45°	3.94
15		-3.27°	4.21	30		-1.39°	3.94

Table 6.3: DC OPF Optimal P_g, δ and LMPs

The nonlinear AC OPF for the 30-bus network can also be masked and solved for by following the steps in Section 6.5. The AC OPF formulation can be seen in [83]. A monomial $\boldsymbol{P}$ matrix was used in this AC OPF example, to better ensure locating the global minimum. The

results of the full AC OPF are listed below, and the solution from the masked problem matches the solution from the unmasked problem as well as the solution given by MATPOWER.

Optimal Cost: $f^* = \$603.12$

From Bus	To Bus	P_f $[MW]$	$\lvert P_{max} \rvert$ $[MW]$	μ $[\$/MWh]$
1	3	16.00	16.00	0.54
6	8	32.00	32.00	47.32
22	21	20.00	20.00	8.19
23	15	16.00	16.00	0.81
23	24	16.00	16.00	1.10
27	25	16.00	16.00	12.16

Table 6.4: AC OPF Binding Branch Flow Constraints

Bus #	Power Gen. P_g $[MW]$	Bus Voltage	LMPs $[\$/MWh]$	Bus #	Power Gen. P_g $[MW]$	Bus Voltage	LMPs $[\$/MWh]$
1	33.67	1.01∠0°	3.35	16		1.01∠-3.06°	5.14
2	48.38	1.01∠-0.55°	3.44	17		0.99∠-3.44°	5.64
3		1.00∠-1.69°	3.73	18		1.01∠-3.66°	5.35
4		1.00∠-1.99°	3.69	19		0.99∠-4.02°	5.57
5		0.99∠-1.92°	3.50	20		0.99∠-3.90°	5.63
6		0.99∠-2.30°	3.53	21		0.99∠-3.22°	8.70
7		0.98∠-2.68°	3.55	22	8.71	0.99∠-3.06°	2.09
8		0.98∠-2.71°	34.00	23	27.41	1.06∠-1.68°	4.37
9		0.99∠-2.96°	5.00	24		1.02∠-2.25°	5.66
10		0.99∠-3.30°	5.79	25		1.04∠-0.37°	12.04
11		0.99∠-2.96°	5.00	26		1.03∠-0.78°	12.30
12		1.04∠-2.41°	4.59	27	43.39	1.07∠ 1.02°	3.97
13	30.84	1.10∠-0.26°	4.54	28		1.00∠-2.16°	9.68
14		1.03∠-2.96°	4.81	29		1.05∠-0.12°	3.89
15		1.04∠-2.77°	4.93	30		1.04∠-0.91°	4.03

Table 6.5: AC OPF Optimal P_g, δ and LMPs

6.7 Conclusion

The optimal power flow (OPF) problem is a central problem in power systems optimization. The need to regularly solve this problem for large scale models motivates the use of cloud computing resources. However, concerns over the security of confidential power system models limits the potential use of cloud computing. In this chapter, existing methods of masking optimization problems was extended to the OPF problem. The masking process is appealing to system operators because it is a simple linear transformation that ensures a level of security without spending any dollars. The contributions made to this work include the procedure for extracting the Lagrange multipliers/ locational marginal prices from the masked dual problem, a method for preserving problem sparsity while ensuring a level of security in the masked problem, and a method for masking a quadratic cost function. The nonlinear AC OPF was examined as well, where its limitations were observed. In the nonlinear case, the numerical values of the model can be masked but there is a limitation on completely masking the system topology. The masking procedures in this work have additionally been extended by the authors in [84] to include contingency analysis.

The work in this chapter is valuable in any circumstance where the confidentiality of a power system model is important. For example, the procedure of masking power system problems for cloud computing could be extended for use in multi-party computation. Here each party contributes a piece of the entire problem to collaboratively solve the problem involving all parties. From a power systems perspective, each party uses the masking procedure to share their confidential part of the power system model to collectively solve problems such as generation dispatch and transmission planning, without revealing any sensitive details to one another.

The steps involved in the obscuring procedure in this chapter hint at an intriguing possibility of transforming one confidential power system model, into an entirely new and legitimate model (or family of models). This possibility would allow for increased sharing of confidential power system models for research purposes. Furthermore, this would allow for the secure sharing of equivalent models without divulging details of the actual system. Such a transformation

between models would imply that the need to use “true system models” for purposes of algorithmic development would become unnecessary. The results would suggest studies of any system of similar complexity are sufficient. It will be shown that this transformation exists for the DC OPF problem, which is presented next in Chapter 7.

Chapter 7

Structure Preserving Transformation

The electric power grid delivers essential energy to support almost all non-transportation energy needs. It is indisputably important to the functions of society and is considered part of the nation's critical infrastructure. There is considerable concern that the power grid may be vulnerable to and targeted for malicious cyber and physical attacks. These concerns raise conflicting needs in the area of advanced power system analysis and research. As critical energy infrastructure information (CEII), access to true power system data is restricted. At the same time, interest in advancing power system tools with a view toward security has increased. For a researcher working to advance this field, results must be shown on real models to prove credibility.

Unfortunately, these needs and expectations have resulted in the current state in which some researchers with access to CEII data can conduct relevant research using credible models; however, these models and results cannot be independently verified by peers in the field. This goes against traditional scientific principles that call for public verification of results. Correspondingly, there is a fundamental need for new standard publicly-available models that are provably related to credible but secure models. The purpose of this chapter is to show that such models can be developed through transformations that yield a new power system model that maintains the privacy of the original data. The transformation maps the solution of the new model to the original, establishing a strong and relevant connection between them.

In this chapter, the transformation techniques are applied to optimal power flow (OPF) models, as the OPF problem is the most important problem routinely solved in this industry. The OPF problem is known to be a non-convex problem and much of the current advanced research is directed towards developing efficient methods to find global optima. The analysis in this chapter begins with a linearized version of the OPF problem and show that it is possible to transform a given OPF model into a different OPF model that relates the optimal solutions through a transformation. It is not possible to infer the original model solely from the data of the new transformed model.

7.1 DC Optimal Power Flow Problem Overview

The standard DC OPF problem is shown below in (7.1). In Chapter 6, DC OPF problems were obscured for preserving confidentiality in cloud computing. However, the transformations in Chapter 6 do not preserve a typical power system structure, which precludes application of OPF specific solution techniques. The optimization variables in (7.1) are the powers generated P_g and bus angles δ.

$$\min_{P_g,\,\delta} \ \frac{1}{2} P_g^T \boldsymbol{D_g} P_g + c_g^T P_g \tag{7.1}$$

$$s.t. \qquad -P_g + \boldsymbol{B}\delta = -P_L$$

$$\delta_{ref} = 0$$

$$P_{g,min} \le P_g \le P_{g,max}$$

$$-P_{flow,max} \le diag(b_{br})\boldsymbol{A_{inc}}\delta \le P_{flow,max}$$

The constraints in (7.1) are power balance at each bus, the reference bus angle equal to 0°, power generation upper and lower limits, and power flow limits in both directions on transmission lines. Primarily, the analysis in this chapter will be on DC OPF problems with linear cost functions. Therefore the quadratic cost terms in (7.1) will be removed, and the problem will be rewritten in general form (7.2).

$$\min_{x} \ c^T x \tag{7.2}$$

$$s.t. \qquad \boldsymbol{M}x = b$$

$$\boldsymbol{I_{sl}}x \ge 0$$

In (7.2), P_g and δ are free variables. Slack variables x_{sl} are included to convert inequality constraints to equality constraints. The slack variables are nonnegative, $x_{sl} \ge 0$, and are enforced by $\boldsymbol{I_{sl}}x \ge 0$, where $\boldsymbol{I_{sl}} = [\boldsymbol{0} \ \ \boldsymbol{I}]$ and $\boldsymbol{I}$ is the identity matrix. The optimization variables are $x = \left[P_g^T \quad \delta^T \quad x_{sl}^T\right]^T$.

The constraint matrix $\boldsymbol{M}$ and vector b have typical topologies as seen in Figure 7.1 which displays the IEEE 14-bus test system [77]. Matrix $\boldsymbol{M}$ has $m = n_{bus} + 2n_{gen} + 2n_{line} + 1$ rows

and $n = n_{bus} + 3n_{gen} + 2n_{line}$ columns. Therefore there are $n - m = n_{gen} - 1$ more columns than rows in $\boldsymbol{M}$. It should be emphasized that the number of rows m is less than number of columns n, as it is critical for the transformation analysis. Note that the $\boldsymbol{M}$ matrix has full-rank in typical DC OPF problems and is therefore assumed to be full-rank.

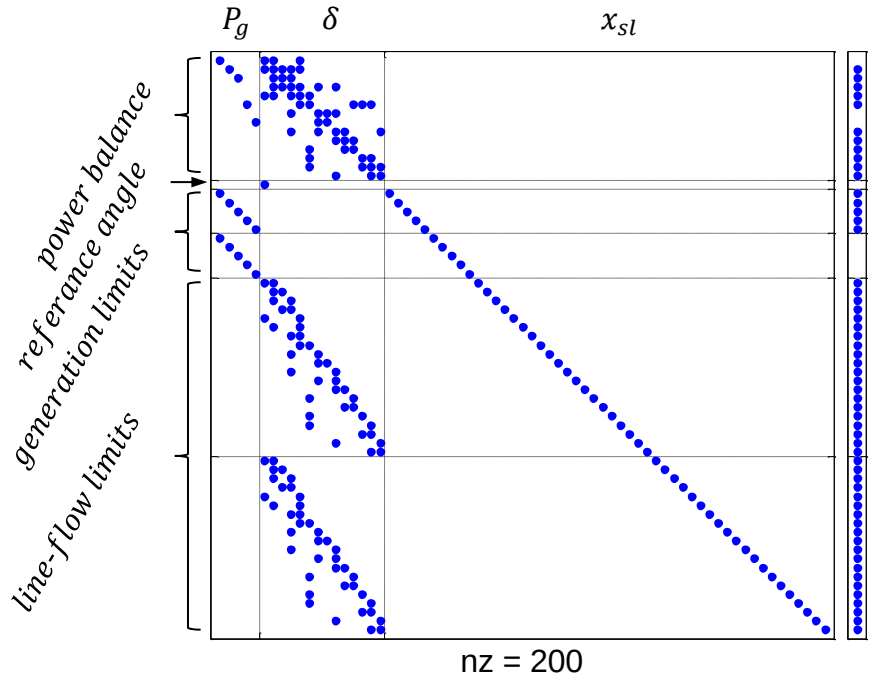

Figure 7.1: $\boldsymbol{M}$ matrix and $\boldsymbol{b}$ vector for IEEE 14-bus system

7.2 Transformation Problem

In this section, the procedure is detailed for transforming one DC OPF problem to a completely new problem. To begin with, two different DC OPF problems are constructed, with the first in (7.2) and the second in (7.3).

$$\min_{x'} \; c'^{T} x' \tag{7.3}$$

$$s.t. \quad \boldsymbol{M}'x' = b'$$

$$\boldsymbol{I_{sl}}x' \geq 0$$

For simplicity, it is assumed the two DC OPF problems possess the same number of generators, buses and lines. The two models differ by having different bus and line topologies as well as having numerically different component values such as line susceptances, generator cost coefficients, etc.

Next, a bidirectional linear transformation between (7.2) and (7.3) is constructed using similar linear transformations as those discussed in Chapter 6 for masking linear programs. The goal is to construct the transformation matrices $\boldsymbol{P}$ and $\boldsymbol{T}$ and vectors w and r such that the following relationships hold.

$$\boldsymbol{M'} = \boldsymbol{PMT} \tag{7.4a}$$

$$c' = \boldsymbol{T}^T c + \boldsymbol{M'}^T w \tag{7.4b}$$

$$b' = \boldsymbol{P}b + \boldsymbol{M'}r \tag{7.4c}$$

$$x^* = \boldsymbol{T}(x'^* - r) \tag{7.4d}$$

This structure preserving transformation was inspired by the confidentiality preserving OPF from Chapter 6. If the masking process from Chapter 6 were able to create an OPF structure rather than a random structure, it would allow several possibilities. It would allow secure exchanging of relatable power system models without divulging details of the original model. If the masking process could create an OPF structure, then an important statement can also be made about the security of the masked problem. If an adversary intercepted a masked OPF problem and attempted to reverse engineer an OPF structure out of the problem, then the structure preserving transformation would prove that many OPF structures could be backed out of the masked problem, thereby ensuring a high level of security.

Up to this point in time however, the transformation between OPF problems has not been identified within the restrictions of the masking process from Chapter 6. In particular, there are two sufficient conditions for the masking process of Chapter 6 that made identifying a transformation between OPF problems challenging, if not possible at all. In Chapter 6, the matrix $\boldsymbol{T}$ was required to be positive monomial (i.e. a matrix containing exactly one positive entry per row and

column, with the remainder of the entries being zero), and vector r was required to be nonnegative. These sufficient conditions were necessary for guaranteeing the optimal solutions x^* and x'^* mapped to one another by $x^* = \boldsymbol{T}(x'^* - r)$. The formulation of the DC OPF in this chapter is slightly different than in Chapter 6: free variables are no longer required to be split into positive and negative components and the constraints are dropped that enforce the powers generated and bus voltage angle components to be greater than or equal to zero. This simplified formulation can be observed in Figure 7.1 and in (7.3). With the simplified formulation, the two sufficient conditions are slightly less stringent. Define $x = [x_0^T \; x_{sl}^T]^T$ where $x_0 = [P_g^T \; \delta^T]^T$, and define $r = [r_0^T \; 0^T]^T$ and

$$\boldsymbol{T} = \begin{bmatrix} \boldsymbol{T_{11}} & \boldsymbol{T_{12}} \\ \boldsymbol{0} & \boldsymbol{T_{22}} \end{bmatrix}.$$

Observe the relationship between x^* and x'^*.

$$x = \boldsymbol{T}(x' - r)$$

$$\begin{bmatrix} x_0 \\ x_{sl} \end{bmatrix} = \begin{bmatrix} \boldsymbol{T_{11}} & \boldsymbol{T_{12}} \\ \boldsymbol{0} & \boldsymbol{T_{22}} \end{bmatrix} \left(\begin{bmatrix} x_0' \\ x_{sl}' \end{bmatrix} - \begin{bmatrix} r_0 \\ 0 \end{bmatrix} \right)$$

With the simplified formulation of the DC OPF, the sufficient conditions that guarantee the optimal solutions relate by $x^* = \boldsymbol{T}(x'^* - r)$ are that $\boldsymbol{T_{22}}$ must be positive monomial and r_0 is free to be any value. Even under these less stringent sufficient conditions, determining appropriate matrices $\boldsymbol{P}$ and $\boldsymbol{T}$ and vectors w and r such that relationships (7.4a)−(7.4c) hold has been challenging. There may be no reason to expect the transformation to exist at all under these sufficient conditions on $\boldsymbol{T}$ and r.

For the purpose of identifying whether the structure preserving transformation exists at all, these two sufficient conditions are relaxed so that $\boldsymbol{T}$ can be any nonsingular matrix and r can be any vector. This relaxation comes at a cost however, as now the two OPF problems are not guaranteed to map to each other by (7.4d), unless x^* and x'^* are solved apriori and (7.4d) is ex-

plicitly enforced. Therefore the optimal solutions x^* and x'^* to DC OPF problems (7.2) and (7.3) respectively, are determined apriori before proceeding with the transformation.

Relationships (7.4a)−(7.4d) all have specific purposes. Relationship (7.4a) enforces a linear transformation between the constraint matrix $\boldsymbol{M}$ in (7.2) and the constraint matrix $\boldsymbol{M'}$ in (7.3). This is done by multiplying $\boldsymbol{M}$ on the left and right by some transformation matrices $\boldsymbol{P}$ and $\boldsymbol{T}$. Relationship (7.4b) enforces a linear transformation between the cost coefficient vector c in (7.2) and the cost coefficient vector c' in (7.3). A linear combination of the rows in constraint matrix $\boldsymbol{M'}$ is added to $\boldsymbol{T}^T c$ in order to satisfy (7.4b). This is justified because adding $\boldsymbol{M'}^T w$ to the cost function does nothing more than add a constant $b'^T w$, $c'^T x' = (c^T \boldsymbol{T} + w^T \boldsymbol{M'}) x' = c^T \boldsymbol{T} x' + w^T b'$. Relationship (7.4c) enforces a linear transformation between the constraint vector b in (7.2) and the constraint vector b' in (7.3).

Relationship (7.4d) enforces a linear transformation between the optimal solution x^* in (7.2) and the optimal solution x'^* in (7.3). In [43] and [45], the matrix $\boldsymbol{T}$ was required to be a positive monomial matrix, and vector r was required to be nonnegative. By defining $\boldsymbol{T}$ and r this way, it can be shown that if $\boldsymbol{I_{sl}}\, x \geq 0$ then it is guaranteed that $\boldsymbol{I_{sl}}\, x' \geq 0$ as well. These requirements on $\boldsymbol{T}$ and r will be removed now, as they restrict the ability to transform to a new system with an OPF structure.

The transformation between the DC OPF problems (7.2) and (7.3) is developed by determining appropriate matrices $\boldsymbol{P}$ and $\boldsymbol{T}$ and vectors w and r such that the relationships (7.4a)−(7.4d) hold. Next, the relationships in (7.4a)−(7.4d) and the degrees of freedom inherent in the transformation (i.e. the null-spaces of the DC OPF problems' matrices) are exploited to describe an appropriate choice for these matrices and vectors.

The specifics of each transformation are detailed here. To begin, the relationship (7.4a) can be enforced by defining $\boldsymbol{T}$ as the sum of two terms, $\boldsymbol{T} = \boldsymbol{T_0} + \boldsymbol{NQ}$, where the $n \times (n-m)$ matrix $\boldsymbol{N}$ is the null-space of $\boldsymbol{M}$, i.e. $\boldsymbol{MN} = \boldsymbol{0}_{m \times (n-m)}$. Given any nonsingular $\boldsymbol{P}$, one can solve $\boldsymbol{T_0} = \boldsymbol{M}^{\dagger} \boldsymbol{P}^{-1} \boldsymbol{M'}$, where † denotes the Moore-Penrose pseudo-inverse [74]. This solution structure

on $\boldsymbol{T}$ satisfies (7.4a). It is necessary for $\boldsymbol{T}$ to be full-rank, therefore matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ must be full-rank.

Next, relationship (7.4b) is expanded by substituting $\boldsymbol{T} = \boldsymbol{T_0} + \boldsymbol{NQ}$ as shown in (7.5), where matrix $\boldsymbol{Q}$ and vector w are unknown.

$$c' = \boldsymbol{T_0}^T c + \boldsymbol{Q}^T \boldsymbol{N}^T c + \boldsymbol{M'}^T w \tag{7.5}$$

The procedure goes as follows, w is solved as a function of $\boldsymbol{Q}$ in (7.6) by taking the pseudoinverse of $\boldsymbol{M'}^T$.

$$w = \boldsymbol{M'}^{T\dagger}(c' - \boldsymbol{T_0}^T c - \boldsymbol{Q}^T \boldsymbol{N}^T c) \tag{7.6}$$

Next w is substituted back into (7.5) which can be rewritten as in (7.7).

$$\left(I - \boldsymbol{M'}^T \boldsymbol{M'}^{T\dagger}\right)(c' - \boldsymbol{T_0}^T c) = \left(I - \boldsymbol{M'}^T \boldsymbol{M'}^{T\dagger}\right)\boldsymbol{Q}^T \boldsymbol{N}^T c \tag{7.7}$$

Note that $\boldsymbol{M'}^T \boldsymbol{M'}^{T\dagger} \neq I_{n\times n}$ and $\boldsymbol{M'}^{T\dagger} \boldsymbol{M'}^T = I_{m\times m}$, therefore it follows that $\left(I - \boldsymbol{M'}^T \boldsymbol{M'}^{T\dagger}\right)\boldsymbol{M'}^T = \boldsymbol{0}_{n\times m}$. The $n\times n$ matrix $I - \boldsymbol{M'}^T \boldsymbol{M'}^{T\dagger}$ has rank of order $n-m$, and can be rewritten as $I - \boldsymbol{M'}^T \boldsymbol{M'}^{T\dagger} = \boldsymbol{N'} \boldsymbol{N'}^T$, where the $n\times(n-m)$ matrix $\boldsymbol{N'}$ is the null-space of $\boldsymbol{M'}$, i.e. $\boldsymbol{M'N'} = \boldsymbol{0}_{m\times(n-m)}$. Note that $\boldsymbol{N'}^{\dagger} = \boldsymbol{N'}^T$, $\boldsymbol{N'N'}^T \neq I_{n\times n}$ and $\boldsymbol{N'}^T \boldsymbol{N'} = I_{(n-m)\times(n-m)}$.

Before solving for $\boldsymbol{Q}$ by using (7.7) and the above observations, it should be emphasized that $\boldsymbol{T} = \boldsymbol{T_0} + \boldsymbol{NQ}$ must have full-rank of n. $\boldsymbol{T_0}$ will have rank m as long as $\boldsymbol{P}$ is full-rank m; therefore, $\boldsymbol{NQ}$ must have rank $n-m$. It becomes necessary to split $\boldsymbol{Q}$ into the sum of two $(n-m)\times n$ terms $\boldsymbol{Q} = \boldsymbol{Q_1} + \boldsymbol{Q_2}$. The first of these two terms $\boldsymbol{Q_1}$ will be solved by using (7.7) and the prior observations.

$$\boldsymbol{Q_1} = (c^T \boldsymbol{N})^{\dagger} c'^T \boldsymbol{N'} \boldsymbol{N'}^T \tag{7.8}$$

Setting $\boldsymbol{Q} = \boldsymbol{Q_1}$ now satisfies (7.7). With w solved as a function of $\boldsymbol{Q}$ using (7.6), relationship (7.4b) will be satisfied. As it turns out however, $\boldsymbol{NQ_1}$ is rank 1, and therefore $\boldsymbol{T}$ is still lack-

ing $n-m-1$ in its rank. Fortunately the remaining $n-m-1$ rank can be acquired by the second term $\boldsymbol{Q_2}$. Observe in (7.5) that $\boldsymbol{Q}^T$ multiplies an $(n-m)\times 1$ vector $\boldsymbol{N}^T c$. Define $\tilde{\boldsymbol{N}}$ as the $(n-m)\times(n-m-1)$ null-space of $c^T\boldsymbol{N}$, such that $c^T\boldsymbol{N}\tilde{\boldsymbol{N}} = 0_{1\times(n-m-1)}$. Define an $(n-m-1)\times n$ matrix $\boldsymbol{V}$ which can be any full-rank matrix. Finally define $\boldsymbol{Q_2} = \tilde{\boldsymbol{N}}\boldsymbol{V}$, which yields $\boldsymbol{T} = \boldsymbol{T_0} + \boldsymbol{N}(\boldsymbol{Q_1} + \boldsymbol{Q_2})$ to be a full-rank matrix.

The two vector relationships (7.4c) and (7.4d) have yet to be satisfied, but can be satisfied by appropriately determining the $n\times 1$ vector r. Define r as the sum of two terms, $r = r_0 + \boldsymbol{N'}q$. Rearrange (7.4c) to solve for r_0 as shown in (7.9).

$$r_0 = \boldsymbol{M'}^{\dagger}(b' - \boldsymbol{P}b) \tag{7.9}$$

Setting $r = r_0$ will now satisfy (7.4c), but not (7.4d). The last remaining variable to be solved for is q, which can be obtained by substituting $r = r_0 + \boldsymbol{N'}q$ and rearranging (7.4d).

$$q = \boldsymbol{N'}^{\dagger}(x'^{*} - \boldsymbol{T}^{-1}x^{*} - r_0) \tag{7.10}$$

Substitute q from (7.10) into $r = r_0 + \boldsymbol{N'}q$. This solution for r satisfies both (7.4c) and (7.4d). At this point, all four relationships (7.4a)–(7.4d) have been satisfied by appropriately determining the transformation matrix $\boldsymbol{T}$ and vectors w and r. The transformation solution is not unique, as there is some flexibility in choosing any full-rank matrix $\boldsymbol{P}$ and also any full-rank matrix $\boldsymbol{V}$ as described earlier.

7.3 Example

In this section, a numeric example of the transformation method described in Section 7.2 is presented. Two OPF problems derived from the IEEE 14-bus system are considered [77]. The first OPF problem has the same network topology and line susceptances (calculated as the reciprocal of the line reactance and neglecting line resistance) as the standard IEEE 14-bus system and has a 100 MVA base power. The one-line diagram for this system is shown in Figure 7.2

and a spy plot of matrix $\boldsymbol{M}$ and vector $\boldsymbol{b}$ is shown in Figure 7.3. To enable two-dimensional plotting of the feasible space of generator power injections (detailed in the Appendix Section 7.6), this example considers an OPF problem with only three generators, as opposed to five generators in the standard IEEE 14-bus system.

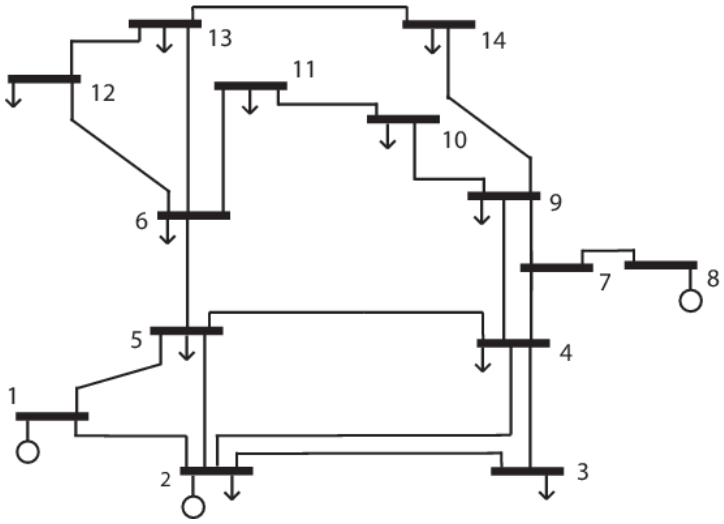

Figure 7.2: One-line diagram for OPF problem 1

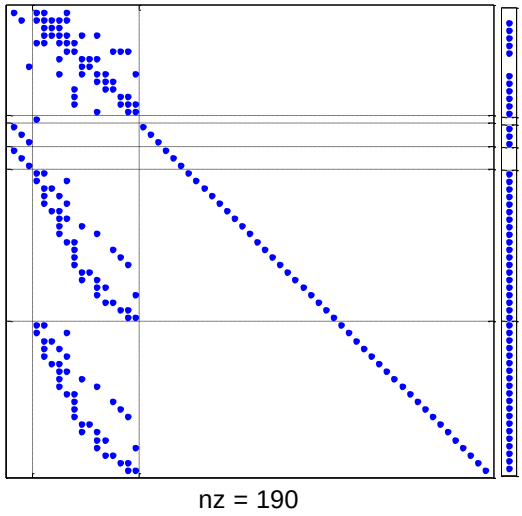

Figure 7.3: M matrix and b vector for OPF problem 1

The coefficients for the linear generator cost functions are given in Table 7.1. All generators have lower generation limits of zero and upper generation limits specified in Table 7.1. Line-flow limits of 100 MW are enforced on all lines. Load demands are the same as those specified for the standard IEEE 14-bus system.

Generator	Cost Coefficient $[\$/MWh]$	Upper Generation Limit $[MW]$
1	20	330
2	30	140
8	25	50

Table 7.1: Generator data for OPF problem 1

The second OPF problem has the same number of buses, lines and generators as the first OPF problem, but has different network topology, line susceptances, generator and line-flow

limits, load demands and generator costs. The one-line diagram for the second system is shown in Figure 7.4 and a spy plot of matrix $\boldsymbol{M}'$ and vector $\boldsymbol{b}'$ is shown in Figure 7.5.

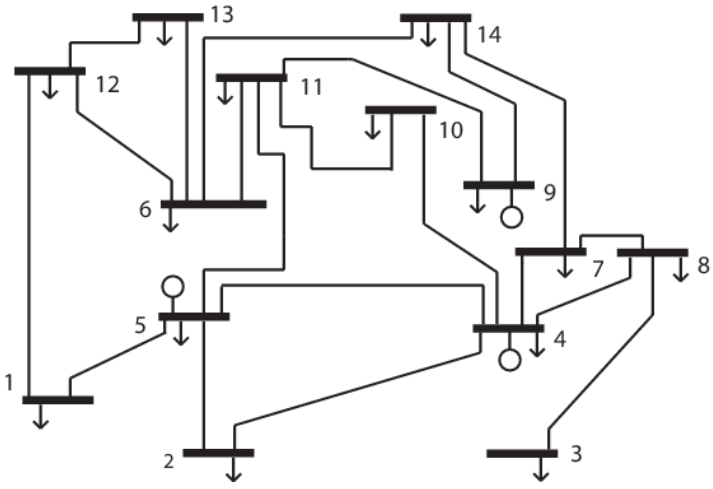

Figure 7.4: One-line diagram for OPF Problem 2

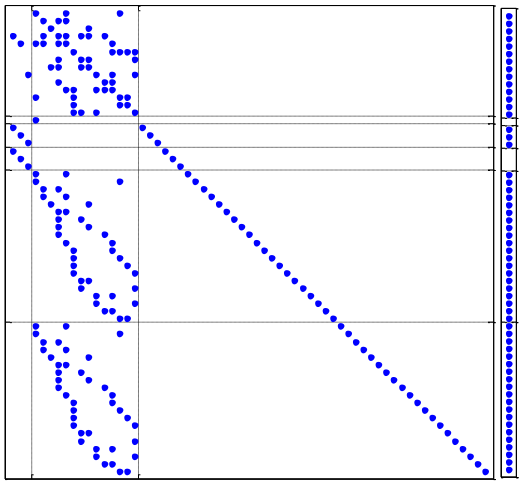

Figure 7.5: M' matrix and b' vector for OPF problem 2

The coefficients for the linear generator cost functions in the second OPF are given in Table 7.2, load demands are given in Table 7.3 and line susceptances are given in Table 7.4. All line-flows are limited to 90 MW.

Generator	Cost Coefficient $[\$/MWh]$	Upper Generation Limit $[MW]$
4	19.11	302
5	40.42	50
9	10.00	393

Table 7.2: Generator data for OPF problem 2

Bus	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Demand $[MW]$	31.5	47.6	64.5	5.9	25.4	37.4	12.5	35.2	58.9	16.1	17.3	35.7	94.5	21.7

Table 7.3: Load demands for OPF problem 2

From Bus	1	1	2	2	3	4	4	4	4	5
To Bus	5	12	4	5	8	5	7	8	10	11
Susceptance $[p.u.]$	18.00	5.41	4.21	4.87	5.82	6.85	19.81	4.74	1.70	4.14

From Bus	6	6	6	6	7	7	9	9	10	12
To Bus	11	12	13	14	8	14	11	14	11	13
Susceptance $[p.u.]$	6.04	3.30	9.45	5.49	8.12	10.82	3.80	6.10	4.75	2.57

Table 7.4: Line susceptances for OPF problem 2

The feasible spaces for generator power injections in the first and second OPF problems are shown in Figures (7.6) and (7.7), respectively. The feasible space in Figure 7.6 corresponds to the first OPF problem having the formulation in (7.2), and the feasible space in Figure 7.7 corresponds to the second OPF problem having the formulation in (7.3). The details for reducing these OPF problems to two optimization variables (generator power injections P_{g1} and P_{g2}) and plotting the optimization feasible space are outlined in the Appendix Section 7.6. The contour lines illustrate the linear cost function, with color blue representing lower cost. Using the procedure detailed in Section 7.2, a transformation (i.e. a set of matrices $\boldsymbol{P}$ and $\boldsymbol{T}$ and vectors w and r) is generated that satisfies relationships (7.4a)−(7.4d).

In Figure 7.8, the transformation is applied to the first OPF problem (7.2) and is plotted in the coordinate system consistent of the second OPF problem (7.3). That is to say, the purple colored polytope in Figure 7.8 corresponds to the feasible space of the problem formulation shown in (7.11b). The formulation in (7.11a) displays the transformation taking place on (7.2) to convert to (7.3). The formulations in (7.11a) and (7.11b) are equivalent, and the optimal solution x^* of (7.2) can be recovered from either (7.11a) or (7.11b). Notice the formulation in (7.11b) is nearly identical to the formulation in (7.3), excluding the inequality constraints $\boldsymbol{I_{sl}}\boldsymbol{T}x' \geq \boldsymbol{I_{sl}}\boldsymbol{T}r$. Relaxing these inequality constraints and replacing them with $\boldsymbol{I_{sl}}x' \geq 0$ completely transforms (7.2) to (7.3), which does not change the optimal solution x'^*. The feasible space of (7.3) is shown as the green colored polytope in Figure 7.8.

$$\min_{x} \quad (c^T \boldsymbol{T} + w^T \boldsymbol{PMT})(\boldsymbol{T}^{-1}x + r) \qquad (7.11a)$$

$$s.t. \quad \boldsymbol{PMT}(\boldsymbol{T}^{-1}x + r) = \boldsymbol{P}b + \boldsymbol{PMT}r$$

$$\boldsymbol{I_{sl}}x \geq 0$$

$$\Leftrightarrow$$

$$\min_{x'} \quad c'^T x' \qquad (7.11b)$$

$$s.t. \quad \boldsymbol{M}'x' = \boldsymbol{b}'$$

$$\boldsymbol{I_{sl}}\big(\boldsymbol{T}(x' - r)\big) \geq 0$$

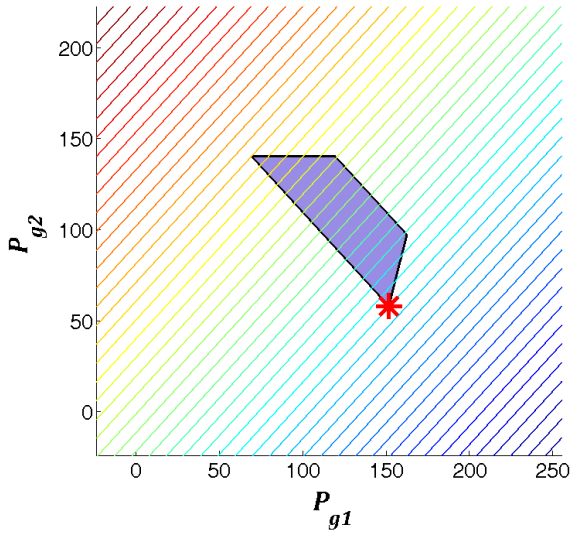

Figure 7.6: Feasible space of power generation for OPF problem 1 (Linear cost)

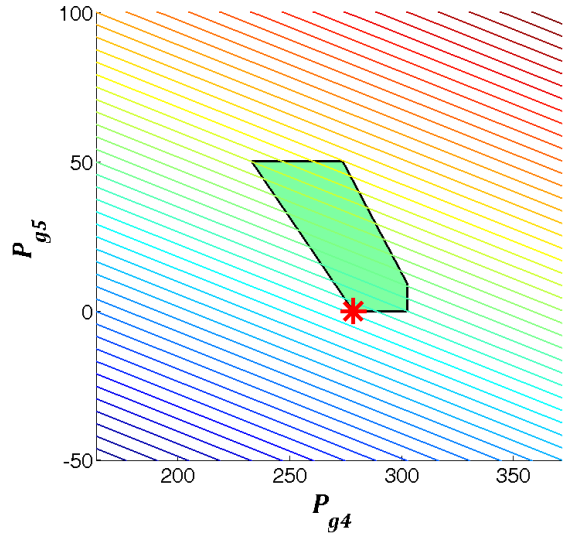

Figure 7.7: Feasible space of power generation for OPF problem 2 (Linear cost)

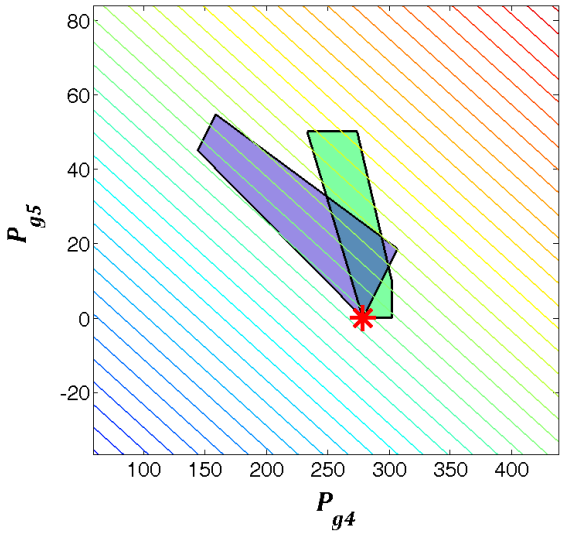

Figure 7.8: Feasible space of power generation for both OPF problems in coordinates of OPF problem 2 (Linear cost)

Both feasible spaces in Figure 7.8 return the same optimal solution, indicated by the red star in the figure. It is therefore true that the optimal solution x'^* to (7.3) can be used to recover the optimal solution to (7.2) by $x^* = \boldsymbol{T}(x'^* - r)$. The small-scale OPF in this example was chosen so that plotting the feasible space in two-dimensions was possible; however, the transformation procedure outlined in Section 7.2 is applicable to any larger DC OPF having linear cost function. This example has demonstrated the method for transforming one DC OPF problem to another while preserving the optimal solution. In the next section, DC OPF problems with quadratic and piecewise linear cost functions are analyzed.

7.4 Quadratic Cost and Piecewise-Linear Cost Functions

The DC OPF model used to develop the transformation in Section 7.2 requires linear cost functions on active power generation. Exploratory work has investigated extension of this transformation to quadratic and piecewise-linear cost functions. The details of the quadratic and piecewise-linear cost function transformations are shown next. The prior example of the two 14-bus systems is revisited as well, but with the cost function changed accordingly.

A. Quadratic Cost Function

Section 7.2 detailed the method for creating a transformation between two DC OPF problems with linear cost functions. In this section, a transformation between two DC OPF problems having quadratic cost functions in investigated. The requirements for the transformation are more strict than the linear cost function case, and it will be shown only systems meeting a certain required property can be transformed to one another. If two systems do not meet this requirement, they can only be transformed to one another by introducing an additional variable/degree of freedom that was not needed in the case of the linear cost function. Two DC OPF problems with quadratic cost functions are shown in (7.12) and (7.13).

$$\min_{x} \ \frac{1}{2}x^T \boldsymbol{D} x + c^T x \qquad (7.12)$$
$$s.t. \quad \boldsymbol{M}x = b$$
$$\boldsymbol{I_{sl}}x \geq 0$$

$$\min_{x'} \ \frac{1}{2}x'^T \boldsymbol{D'} x' + c'^T x' \qquad (7.13)$$
$$s.t. \quad \boldsymbol{M'}x' = b'$$
$$\boldsymbol{I_{sl}}x' \geq 0$$

The goal is to determine a set of transformation matrices and vectors that establish a bidirectional transformation between (7.12) and (7.13). Relationships (7.14a)-(7.14e) must hold in order for the transformation to be complete. Transformation matrices $\boldsymbol{P}$ and $\boldsymbol{T}$ and vectors r and w are analogous to those defined in Section 7.2. The matrix $\boldsymbol{W}$ in (7.14b) and (7.14c) results from the additional degrees of freedom in OPF problems with quadratic cost functions.

$$\boldsymbol{M'} = \boldsymbol{PMT} \qquad (7.14a)$$

$$\boldsymbol{D'} = \boldsymbol{T}^T \boldsymbol{DT} + 2\boldsymbol{M'}^T \boldsymbol{W} \boldsymbol{M'} \qquad (7.14b)$$

$$c' = \boldsymbol{T}^T c - \boldsymbol{T}^T \boldsymbol{DT} r - 2\boldsymbol{M'}^T \boldsymbol{W} b' + \boldsymbol{M'}^T w \qquad (7.14c)$$

$$b' = \boldsymbol{P}b + \boldsymbol{M'}r \qquad (7.14d)$$

$$x^* = \boldsymbol{T}(x'^* - r) \qquad (7.14e)$$

Relationships (7.14a), (7.14d) and (7.14e) are exactly the same as relationships (7.4a), (7.4c) and (7.4d) in the linear cost function case. There is now, however, an additional relation-

ship, (7.14b), which enforces a transformation between the quadratic cost matrix $\boldsymbol{D}$ in (7.12) and the quadratic cost matrix $\boldsymbol{D'}$ in (7.13). In (7.14b), a quadratic combination of the rows in constraint matrix $\boldsymbol{M'}$ is added to $\boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T}$. Adding $2\boldsymbol{M'}^T\boldsymbol{W}\boldsymbol{M'}$ to the quadratic cost matrix in (7.14b) and subtracting $2\boldsymbol{M'}^T\boldsymbol{W}b'$ from the linear cost vector in (7.14c), effectively just adds a constant to the cost function.

$$\begin{aligned} 0 &= (x'^T\boldsymbol{M}'^T - b'^T)\boldsymbol{W}(\boldsymbol{M}'x' - b') \\ &= x'^T\boldsymbol{M}'^T\boldsymbol{W}\boldsymbol{M}'x' - 2b'^T\boldsymbol{W}\boldsymbol{M}'x' + b'^T\boldsymbol{W}b' \end{aligned} \tag{7.15}$$

The cost function in (7.13) effectively has a constant $x'^T\boldsymbol{M}'^T\boldsymbol{W}\boldsymbol{M}'x' - 2b'^T\boldsymbol{W}\boldsymbol{M}'x' = -b'^T\boldsymbol{W}b'$ added to it, as evident in (7.15).

The relationship between the linear cost vector c in (7.12) and linear cost vector c' in (7.13), is shown in (7.14c). Similar to relationship (7.4b), a linear combination of the rows in constraint matrix $\boldsymbol{M'}$ is added as $\boldsymbol{M'}^T w$, which effectively adds a constant to the cost function. There is one additional term in (7.14c) that has yet to be explained, that being $-\boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T}r$. Consider substituting $x = \boldsymbol{T}(x' - r)$ into the quadratic cost term $\frac{1}{2}x^T\boldsymbol{D}x$ in (7.12).

$$\begin{aligned} \tfrac{1}{2}x^T\boldsymbol{D}x &= \tfrac{1}{2}(x'^T - r^T)\boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T}(x' - r) \\ &= \tfrac{1}{2}x'^T\boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T}x' - r^T\boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T}\,x' + \tfrac{1}{2}r^T\boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T}r \end{aligned} \tag{7.16}$$

As shown in (7.16), quadratic, linear and constant terms are created after making the substitution. Therefore $-\boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T}r$ in (7.14c) originates from the cross coupling terms of the quadratic cost function transformation. The constant term $\frac{1}{2}r^T\boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T}r$ is dropped altogether, as it has no impact on results of the optimization in (7.13).

The transformation procedure between two DC OPF problems having quadratic cost functions is detailed next. The procedure is started the same as the linear cost function case, where relationship (7.14a) is enforced by defining $\boldsymbol{T}$ as the sum of two terms, $\boldsymbol{T} = \boldsymbol{T_0} + \boldsymbol{N}\boldsymbol{Q}$, where $\boldsymbol{T_0} = \boldsymbol{M}^\dagger\boldsymbol{P}^{-1}\boldsymbol{M'}$ and the $n\times(n-m)$ matrix $\boldsymbol{N}$ is the null-space of $\boldsymbol{M}$, i.e. $\boldsymbol{M}\boldsymbol{N} = \boldsymbol{0}_{m\times(n-m)}$. With

relationship (7.14a) satisfied, next move on to relationship (7.14b). Rearrange (7.14b) by taking the pseudoinverse of $\boldsymbol{M'}^T$ and $\boldsymbol{M'}$ to solve for $\boldsymbol{W}$ as a function of $\boldsymbol{T}$ as shown in (7.17).

$$\boldsymbol{W} = \tfrac{1}{2}\boldsymbol{M'}^{T\dagger}(\boldsymbol{D'} - \boldsymbol{T}^T\boldsymbol{D}\boldsymbol{T})\boldsymbol{M'}^{\dagger} \tag{7.17}$$

Next, $\boldsymbol{W}$ from (7.17) and $\boldsymbol{T} = \boldsymbol{M}^{\dagger}\boldsymbol{P}^{-1}\boldsymbol{M'} + \boldsymbol{N}\boldsymbol{Q}$ are substituted back into (7.14b), which can be rewritten as in (7.18). Recall that $\boldsymbol{M'}^T\boldsymbol{M'}^{T\dagger} \neq I_{n\times n}$ and $\boldsymbol{M'}^{T\dagger}\boldsymbol{M'}^T = I_{m\times m}$.

$$\begin{aligned} &\boldsymbol{D'} - \boldsymbol{Q}^T\boldsymbol{N}^T\boldsymbol{D}\boldsymbol{N}\boldsymbol{Q} - \boldsymbol{M'}^T\boldsymbol{M'}^{T\dagger}(\boldsymbol{D'} - \boldsymbol{Q}^T\boldsymbol{N}^T\boldsymbol{D}\boldsymbol{N}\boldsymbol{Q})\boldsymbol{M'}^{\dagger}\boldsymbol{M'} \\ &= \quad \boldsymbol{M'}^T\boldsymbol{P}^{T^{-1}}\boldsymbol{M}^{T\dagger}\boldsymbol{D}\boldsymbol{N}\boldsymbol{Q}(I - \boldsymbol{M'}^{\dagger}\boldsymbol{M'}) \\ &+ \left(I - \boldsymbol{M'}^T\boldsymbol{M'}^{T\dagger}\right)\boldsymbol{Q}^T\boldsymbol{N}^T\boldsymbol{D}\boldsymbol{M}^{\dagger}\boldsymbol{P}^{-1}\boldsymbol{M'} \end{aligned} \tag{7.18}$$

Note that $I - \boldsymbol{M'}^{\dagger}\boldsymbol{M'} = I - \boldsymbol{M'}^T\boldsymbol{M'}^{T\dagger} = \boldsymbol{N'}\boldsymbol{N'}^T$, where the $n\times(n-m)$ matrix $\boldsymbol{N'}$ is the null-space of $\boldsymbol{M'}$, i.e. $\boldsymbol{M'}\boldsymbol{N'} = \boldsymbol{0}_{m\times(n-m)}$. Recall that $\boldsymbol{N'}^{\dagger} = \boldsymbol{N'}^T$, $\boldsymbol{N'}\boldsymbol{N'}^T \neq I_{n\times n}$ and $\boldsymbol{N'}^T\boldsymbol{N'} = I_{(n-m)\times(n-m)}$. Matrices $\boldsymbol{Q}$ and $\boldsymbol{P}^{-1}$ are unknown in (7.18). However, (7.18) has a special structure that allows solving for $\boldsymbol{P}^{-1}$ as a function of $\boldsymbol{Q}$. Define $\tilde{\boldsymbol{N}}$ as the $m\times(2m-n)$ null-space of $\boldsymbol{N}^T\boldsymbol{D}\boldsymbol{M}^{\dagger}$, such that $\boldsymbol{N}^T\boldsymbol{D}\boldsymbol{M}^{\dagger}\tilde{\boldsymbol{N}} = \boldsymbol{0}_{(n-m)\times(2m-n)}$. Define a $(2m-n)\times m$ matrix $\boldsymbol{V}$ which can be any full-rank matrix. A full-rank solution for $\boldsymbol{P}^{-1}$ as a function of $\boldsymbol{Q}$ is deduced from (7.18) and is shown in (7.19).

$$\boldsymbol{P}^{-1} = \tilde{\boldsymbol{N}}\boldsymbol{V} + (\boldsymbol{N}\boldsymbol{D}\boldsymbol{M}^{\dagger})^{\dagger}\boldsymbol{Q}^{T\dagger}\boldsymbol{N'}\boldsymbol{N'}^T(\boldsymbol{D'} - \boldsymbol{Q}^T\boldsymbol{N}^T\boldsymbol{D}\boldsymbol{N}\boldsymbol{Q})\boldsymbol{M'}^{\dagger} \tag{7.19}$$

Substituting the solution for $\boldsymbol{P}^{-1}$ in (7.19) into (7.18), reduces (7.18) to (7.20).

$$\boldsymbol{N'}\boldsymbol{N'}^T(\boldsymbol{D'} - \boldsymbol{Q}^T\boldsymbol{N}^T\boldsymbol{D}\boldsymbol{N}\boldsymbol{Q})\boldsymbol{N'}\boldsymbol{N'}^T = \boldsymbol{0}_{n\times n} \tag{7.20}$$

In (7.20), the only unknown variable is matrix $\boldsymbol{Q}$. The solution for $\boldsymbol{Q}$ in (7.20) is not unique, but is rather adjustable by any $(n-m)\times(n-m)$ orthogonal matrix $\boldsymbol{R}$, as shown in (7.21).

$$\boldsymbol{Q} = (\boldsymbol{N}^T\boldsymbol{D}\boldsymbol{N})^{-1/2}\boldsymbol{R}\,(\boldsymbol{N'}^T\boldsymbol{D'}\boldsymbol{N'})^{1/2}\boldsymbol{N'}^T \tag{7.21}$$

In (7.21), the exponent $1/2$ denotes a matrix square root, such that $\boldsymbol{Y} = \boldsymbol{X}^{1/2}$ and $\boldsymbol{YY} = \boldsymbol{X}$, and the exponent $-1/2$ denotes the inverse of the matrix square root. With $\boldsymbol{Q}$ from (7.21) and $\boldsymbol{T} = \boldsymbol{M}^{\dagger}\boldsymbol{P}^{-1}\boldsymbol{M}' + \boldsymbol{NQ}$, matrix $\boldsymbol{W}$ can be solved using (7.17), and relationship (7.14b) is satisfied.

Relationships (7.14d) and (7.14e) are exactly the same as (7.4c) and (7.4d) in the case of a linear cost function. As described in Section 7.2, the relationships are satisfied by defining r as the sum of two terms, $r = r_0 + \boldsymbol{N}'q$. It is necessary for $r_0 = \boldsymbol{M}'^{\dagger}(b' - \boldsymbol{P}b)$ and $q = \boldsymbol{N}'^{\dagger}(x'^{*} - \boldsymbol{T}^{-1}x^{*} - r_0)$ in order to satisfy (7.14d) and (7.14e). With proper cancelation, r can be rewritten in (7.22).

$$r = \boldsymbol{M}'^{\dagger}b' - \boldsymbol{T}^{-1}x^{*} + \boldsymbol{N}'\boldsymbol{N}'^{T}x'^{*} \tag{7.22}$$

The last remaining relationship to be satisfied is (7.14c), which equates the linear cost coefficients of DC OPF problems (7.12) and (7.13). Relationship (7.14c) will next be examined to determine what property is required for two systems, with quadratic cost functions, to be able to transform to one another. Rearrange (7.14c) to solve for w as a function of $\boldsymbol{Q}$ in (7.23) by taking the pseudoinverse of $\boldsymbol{M}'^{T}$.

$$w = \boldsymbol{M}'^{T\dagger}(c' - \boldsymbol{T}^{T}c + \boldsymbol{T}^{T}\boldsymbol{DT} + 2\boldsymbol{M}'^{T}\boldsymbol{W}b) \tag{7.23}$$

By substituting r from (7.22), substituting $\boldsymbol{T}^{T}\boldsymbol{DT} = \boldsymbol{D}' - 2\boldsymbol{M}'^{T}\boldsymbol{WM}'$ and substituting $\boldsymbol{T} = \boldsymbol{M}^{\dagger}\boldsymbol{P}^{-1}\boldsymbol{M}' + \boldsymbol{NQ}$, (7.14c) can be reduced to reveal a necessary property in order for two systems, having quadratic cost functions, to be able to transform to one another. This necessary property is shown in (7.24).

$$(\boldsymbol{N}^{T}\boldsymbol{DN})^{-1/2}\boldsymbol{N}^{T}(c + \boldsymbol{D}x^{*}) \;=\; \boldsymbol{R}\,(\boldsymbol{N}'^{T}\boldsymbol{D}'\boldsymbol{N}')^{-1/2}\boldsymbol{N}'^{T}(c' + \boldsymbol{D}'x'^{*}) \tag{7.24}$$

The two DC OPF problems in (7.12) and (7.13) must satisfy the given vector relationship (7.24) in order for (7.14c) to be satisfied. The only degree of freedom in (7.24) is the $(n-m)\times(n-m)$ orthogonal matrix $\boldsymbol{R}$. All other variables are predetermined by the DC OPF models in (7.12) and (7.13). It is acceptable to use a complex valued orthogonal matrix $\boldsymbol{R}$ if

needed. If no $\boldsymbol{R}$ matrix can be found, then an additional degree of freedom must be introduced. Quite simply, the last degree of freedom needed would be an $n\times1$ vector $\tilde{c}$, such that (7.25) is satisfied.

$$c' = \boldsymbol{T}^T c - \boldsymbol{T}^T \boldsymbol{D} \boldsymbol{T} r - 2\boldsymbol{M'}^T \boldsymbol{W} b' + \tilde{c} \tag{7.25}$$

At this point, the four relationships (7.14a), (7.14b), (7.14d) and (7.14e) have been satisfied by appropriately determining the transformation matrices $\boldsymbol{T}$ and $\boldsymbol{W}$ and vector r. If relationship (7.14c) cannot be solved by determining the orthogonal matrix $\boldsymbol{R}$, then an additional degree of freedom $\tilde{c}$ is introduced such that (7.25) is satisfied instead of (7.14c). With this approach, two DC OPF models with quadratic cost functions can be transformed to one another.

With the transformation defined for quadratic cost functions, the example from Section 7.3 is revisited. All parameters, excluding the quadratic cost coefficients, remain the same for both of the 14-bus systems. The generator data for both OPF problems are shown in Tables 7.5 and 7.6. The OPF problems are analogous to (7.12) and (7.13) which have quadratic cost functions.

Generator	Quadratic Cost Coefficient $[\$/MWh^2]$	Linear Cost Coefficient $[\$/MWh]$	Upper Generation Limit $[MW]$
1	0.043	20	330
2	0.250	30	140
8	0.010	25	50

Table 7.5: Generator data for OPF problem 1

Generator	Quadratic Cost Coefficient $[\$/MWh^2]$	Linear Cost Coefficient $[\$/MWh]$	Upper Generation Limit $[MW]$
4	0.144	19.11	302
5	0.964	40.42	50
9	0.480	10.00	393

Table 7.6: Generator data for OPF problem 2

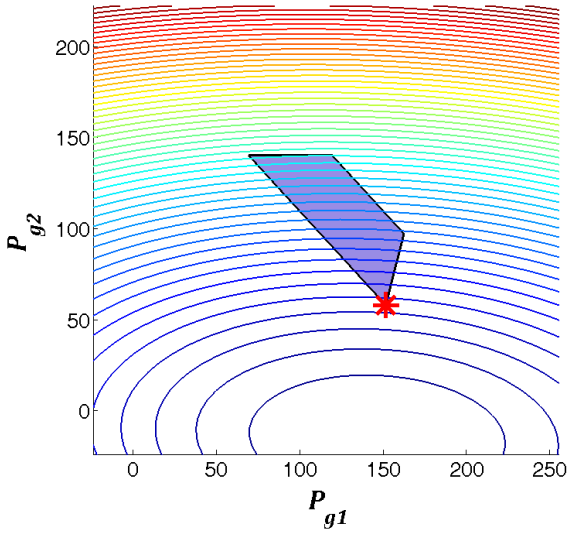

Figure 7.9: Feasible space of power generation for OPF problem 1 (Quadratic cost)

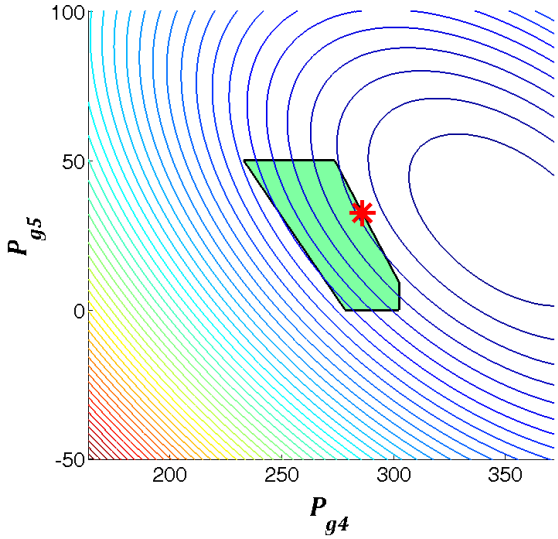

Figure 7.10: Feasible space of power generation for OPF problem 2 (Quadratic cost)

In Figures 7.9 and 7.10, the feasible space of power generation is shown for both OPF problems. The feasible space is the same as that shown for the linear cost cases in Figures 6.6 and 6.7, however the contours showing the cost are now quadratic. The change from linear to quadratic cost moved the optimal solution from a vertex in Figure 7.7 to an edge in Figure 7.10. The transformation procedure outlined in this section is applied.

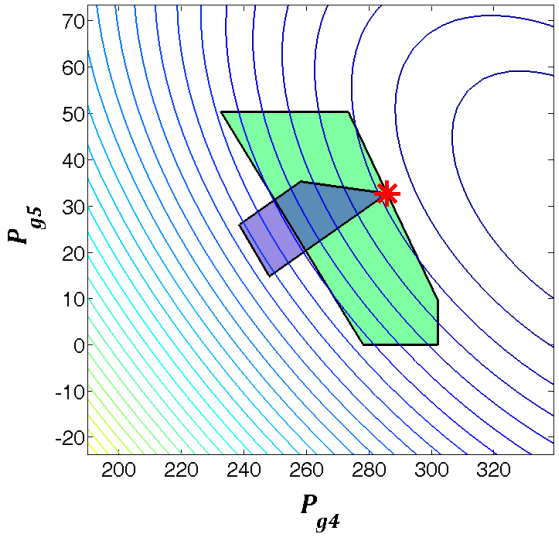

Figure 7.11: Feasible space of power generation for both OPF problems in coordinates of OPF problem 2 (Linear cost)

As was done in the linear cost example, Figure 7.11 displays the feasible space of OPF problem 1 as the purple polytope in the coordinate system of OPF problem 2. In the previous linear example, it was described that the feasible space of OPF problem 1 is created by inequalities $\boldsymbol{I_{sl}Tx' \geq I_{sl}Tr}$. With these inequalities relaxed and replaced by $\boldsymbol{I_{sl}x' \geq 0}$, the purple polytope in Figure 7.11 is completely transformed to the green polytope, which is the feasible space for OPF problem 2. As shown by Figure 7.11, the optimal solution is the same for both feasible spaces, indicated by the red star.

B. Piecewise-Linear Cost Function

Convex piecewise-linear cost functions are often used in DC OPF problems, particularly in electricity market contexts [83]. The piecewise-linear DC OPF is constructed as follows. Consider a piecewise-linear cost function for generator i with r_i linear segments specified by slopes $m_{i,1}, \ldots, m_{i,r_i}$ and breakpoints (a_{ij}, b_{ij}), $j = 1, \ldots, r_i$, where a_{ij} is the power generation coordinate and b_{ij} is the cost coordinate for the j^{th} breakpoint of generator i. With these specifications, the cost of power generation at the i^{th} generator becomes $C_g(P_{g,i})$ as shown in (7.26).

$$C_g(P_{g,i}) = \begin{cases} m_{i,1}(P_{g,i} - a_{i,1}) + b_{i,1}\,, & P_{g,i} \le a_{i,1} \\ m_{i,2}(P_{g,i} - a_{i,2}) + b_{i,2}\,, & a_{i,1} < P_{g,i} \le a_{i,2} \\ \vdots & \vdots \\ m_{i,r_i}(P_{g,i} - a_{i,r_i}) + b_{i,r_i}\,, & a_{i,r_i} \le P_{g,i} \end{cases} \tag{7.26}$$

Convex piecewise-linear cost functions can be implemented as a linear program using a set of linear inequality constraints. Specifically, define a scalar variable β_i for each generator. Then the piecewise-linear cost curves are implemented using the linear program in (7.27), which is the piecewise-linear modification of the formulation in (7.1).

Incorporation of this formulation for convex piecewise-linear cost functions does not change the fundamental characteristics of the DC OPF problem since inequality constraints are already allowed in the DC OPF formulation (7.1). The formulation for piecewise-linear cost functions has a linear objective, and therefore the transformation method described in Section 7.2 can be directly applied to this modified problem.

$$\min_{P_g,\,\delta,\,\beta} \sum_{i=1}^{n_{gen}} \beta_i \tag{7.27}$$

$$s.t. \quad \{\beta_i \ge m_{g,t}\,(P_{g,i} - a_{i,t}) + b_{i,t} \quad \forall\, t = 1,\ldots,r_i\} \quad \forall\, i = 1,\ldots,n_{gen}$$

$$-P_g + \boldsymbol{B}\delta = -P_L$$

$$\delta_{ref} = 0$$

$$P_{g,min} \le P_g \le P_{g,max}$$

$$-P_{flow,max} \le diag(b_{br})\boldsymbol{A_{inc}}\delta \le P_{flow,max}$$

The 14-bus systems used in the linear and quadratic cost examples will be revisited. For the piecewise-linear formulation of these problems, matrices $\boldsymbol{M}$ and $\boldsymbol{M'}$ displayed in Figures 7.3 and 7.5 must be modified to include the piecewise-linear constraints, i.e. the first set of constraints in (7.27). These inequality constraints are converted to equality constraints by introduc-

ing additional slack variables. Matrices $\boldsymbol{M}$ and $\boldsymbol{M'}$ are shown in Figures 7.12 and 7.13 with the piecewise-linear constraints enforced by the top sections in the matrices.

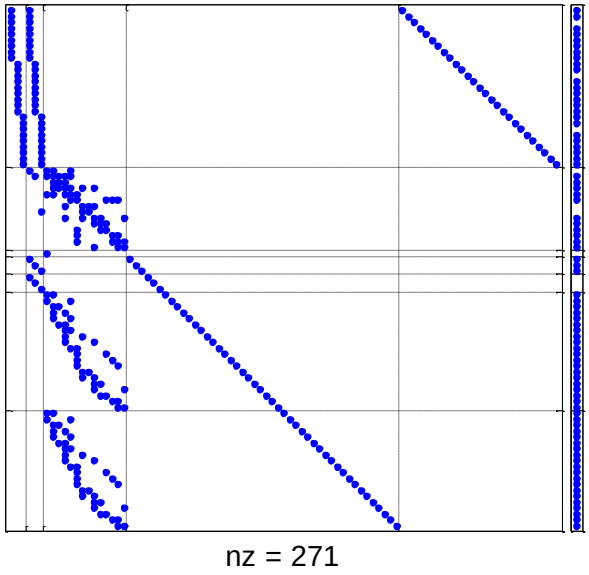

Figure 7.12: M matrix and b vector for OPF problem 1 (Piecewise-linear)

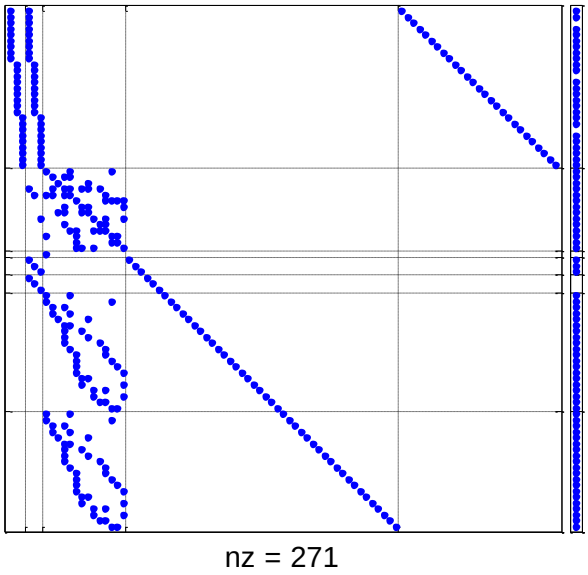

Figure 7.13: M' matrix and b' vector for OPF problem 2 (Piecewise-linear)

The piecewise-linear cost function was constructed in a way to approximate the quadratic cost function from the example in the prior section. The feasible spaces for the two OPF problems are shown in Figures 7.14 and 7.15.

The transformation method described in Section 7.2 can be directly applied to this linear problem. The piecewise-linear constraints add additional optimization variables β_i to the problem, and therefore plotting these 3 generator examples on a two-dimensional plot is not as straight-forward. The method determines matrices $\boldsymbol{P}$ and $\boldsymbol{T}$ and vectors r and w that allow a bi-directional transformation between the two piecewise-linear problems while mapping the optimal solutions from both OPF problems to one another. However, due to the complications from the additional optimization variables β_i, it is not possible to create a figure outright displaying the feasible space of power generation for both OPF problems in the coordinates of OPF problem 2 as was done for the linear and quadratic cost function examples.

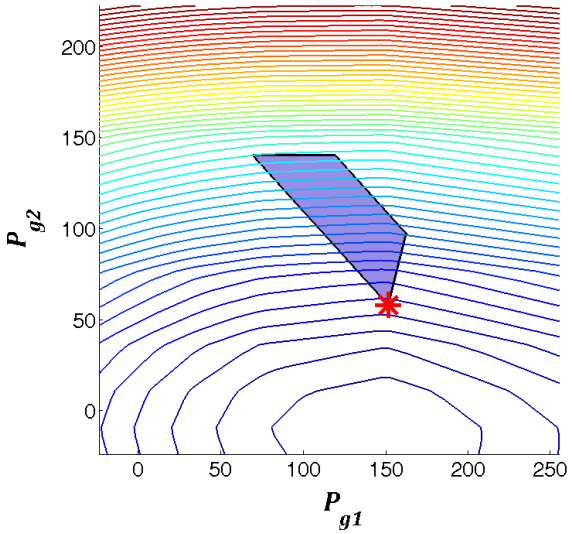

Figure 7.14: Feasible space of power generation for OPF problem 1 (Piecewise-linear)

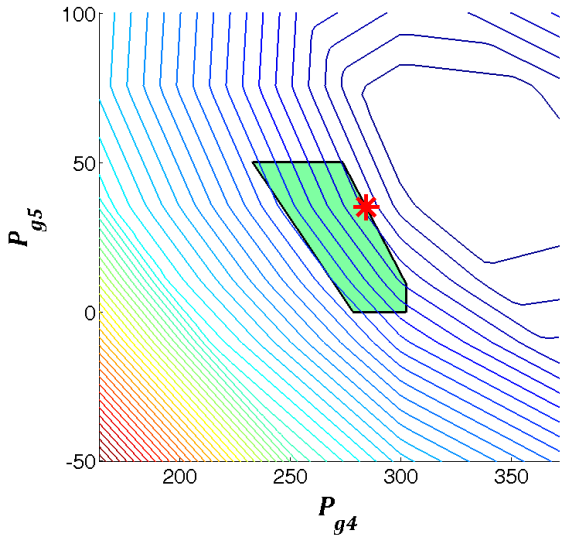

Figure 7.15: Feasible space of power generation for OPF problem 2 (Piecewise-linear)

7.5 Conclusion

This chapter has outlined a transformation method between two DC optimal power flow (OPF) problems, and by extension to a family of problems, which preserves a mapping between optimal solutions. The transformation method was first developed for DC OPF problems having linear cost functions, and the method was demonstrated on an example using a modified version of the IEEE 14-bus system. Next, the transformation method was developed for DC OPF prob-

lems having quadratic cost functions, and lastly was developed for DC OPF problems having piecewise-linear cost functions. The same IEEE 14-bus system was used as an example demonstrating the transformation for quadratic and piecewise-linear cost functions.

Chapter 6 examined transforming/masking OPF problems for purposes of preserving system confidentiality in cloud computing [45]. However, in that work the transformed/masked problem did not resemble a typical power system structure. For the purpose of cloud computing, it is not required for the model to have a power system structure.

The methods detailed in this chapter require the relaxation of two sufficient conditions on matrix $\boldsymbol{T}$ and vector r, in order to prove the existence of a constructively calculated transformation that preserves power system structure for two such solved systems. Ideally, the relaxation on sufficient conditions would be proven unnecessary and the transformation between OPF problems would exist within the restrictions of the sufficient conditions. If this were the case, then requiring the solutions x^* and x'^* to be solved apriori would no longer be necessary and the relationship $x^* = \boldsymbol{T}(x'^* - r)$ would inherently be true. Determining whether the transformation can be developed without solving both problems apriori remains a topic of future work.

The study of sensitive data, typically shared under non-disclosure agreements, is necessary for maintaining the reliable and secure operation of the electric power grid. However, the development of algorithms and the presentation of results using these models cannot be independently investigated and directly confirmed by others, as is the accepted practice in the scientific community. Therefore there is a need for commonly accepted power system models that can be shared broadly, that are accepted as equivalent to actual models that are not shared, and that are suitable for research purposes. This work in this chapter has shown that a method for transforming DC OPF models exists that preserves the confidentiality of the original model and the structure of a DC OPF. This work may serve to motivate the acceptance of synthetic models and to more freely allow sharing of realistic models among researchers, and thereby aid the process of algorithmic development for solving OPF problems.

7.6 Appendix – Plotting Feasible Space

In this Appendix, the procedure for reducing a quadratic or linear program by eliminating equality constraints is detailed. Additionally, the cost function is reconstructed for the reduced system, which can allow plotting of feasible spaces for small-scale problems.

Consider a quadratic program such as (7.12) with equality constraints $\boldsymbol{M}x = b$, where $\boldsymbol{M}$ has m rows and n columns and $m \leq n$. All m equality constraints and m of the optimization variables in x can be eliminated from the problem. Denote $\boldsymbol{M} = [\boldsymbol{M_1}\ \boldsymbol{M_2}]$, where $\boldsymbol{M_1}$ is $m\times(n-m)$ and $\boldsymbol{M_2}$ is $m\times m$; also denote $x = [x_1^T\ \ x_2^T]^T$ where x_1 is $(n-m)\times 1$ and x_2 is $m\times 1$. The optimization variables x_2 can be eliminated from the problem, shown in (7.28).

$$x_{2(x_1)} = \boldsymbol{M_2}^{-1}(b - \boldsymbol{M_1}x_1) \tag{7.28}$$

In (7.28), x_2 becomes a function of the remaining variables x_1. In a DC OPF problem, x_1 contains $n_{gen} - 1$ optimization variables. One choice of variables in x_1 and x_2 is shown below.

$$x_1 = \left[\ P_{g,1}\ \ P_{g,2}\ \ \ldots\ \ P_{g,n_{gen}-1}\ \right]^T$$

$$x_2(x_1) = \left[\ P_{g,slack}(x_1)\ \ \delta(x_1)^T\ \ x_{sl}(x_1)^T\ \right]^T$$

In other words, the power output by all generators, excluding the slack generator, determines the bus angles δ and operating point of the system. The quadratic cost matrix $\boldsymbol{D}$ and linear cost vector c in (7.12) can be similarly split into parts. In the case of linear programs, assume $\boldsymbol{D} = \boldsymbol{0}$ in the following derivation.

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{D_{11}} & \boldsymbol{D_{12}} \\ \boldsymbol{D_{21}} & \boldsymbol{D_{22}} \end{bmatrix} \text{ and } c = [c_1^T\ \ c_2^T]^T$$

With optimization variables x_2 eliminated, consider the reduced sized quadratic cost matrix $\widehat{\boldsymbol{D}}$ and linear cost vector $\hat{c}$.

$$\widehat{\boldsymbol{D}} = \boldsymbol{M_1^T M_2^{-1^T} D_{22} M_2^{-1} M_1 - M_1^T M_2^{-1^T} D_{21} - D_{12} M_2^{-1} M_1 + D_{11}}$$

$$\hat{c} = c_1^T - c_2^T \boldsymbol{M_2^{-1} M_1} - \tfrac{1}{2} b^T \boldsymbol{M_2^{-1^T}} (\boldsymbol{D_{22} + D_{22}^T}) \boldsymbol{M_2^{-1} M_1} + \tfrac{1}{2} b^T \boldsymbol{M_2^{-1^T} D_{12}^T} + \tfrac{1}{2} b^T \boldsymbol{M_2^{-1^T} D_{12}^T}$$

The quadratic program in (7.12) can be rewritten once more as shown in (7.29).

$$\min_{x_1} \ \frac{1}{2} x_1^T \widehat{\boldsymbol{D}} x_1 + \ \hat{c}^T x_1 \qquad (7.29)$$
$$s.t. \qquad x_{2(x_1)} \geq 0$$

The problem's feasible space in (7.29) is now clearly determined by $x_{2(x_1)} \geq 0$. If there are only two or three variables in x_1, then the feasible space can be plotted and visualized in two or three dimensions respectively.

Chapter 8

Conclusions, Contributions and Directions for Future Work

8.1 Conclusions

The work in this dissertation sheds insight and solutions onto fundamental power system problems involving modal analysis and model security. Advancements resulting from this work will better ensure the reliable and secure operation of the electric power system. Power system modeling and optimization form the underlying theme throughout this work.

Chapters 2-5 of this report analyzed a modal analysis problem involving oscillatory disturbance data. Disturbances in the electric grid tend to cause power oscillations that damp-out, sustain or continuously grow towards system collapse. Analysis of the oscillatory disturbance data provides power engineers information about the stability properties of the power system. Accurate estimation of modes can also be used for validating system models, used to make important planning and operating decisions. In addition, knowledge of the system's modes can be used for stabilizing technologies in the electric grid. For these reasons, better algorithms for determining modal content will help improve electric reliability and performance.

Chapter 2 introduced the modal analysis problem with discussion on model-based and measurement-based modal analysis methods such as Prony analysis. As a contribution, the measurement-based Matrix Pencil method and Eigensystem Realization Algorithm (ERA) were extended for multi-signal modal analysis. An example disturbance in the IEEE 14-bus system was used to compare the methods and to demonstrate the trade-offs between the methods' accuracy and computation time. Lastly, the metric of mode energy was introduced which is used for ranking the dominance of modes.

In Chapter 3, the modal analysis problem was recast as a nonlinear least-squares optimization problem which was solved using the variable projection method. The identification of the modal analysis problem as a nonlinear least-squares optimization was a key contribution in the dissertation and to the power system field. The method improves on solutions from traditional methods such as Prony analysis, matrix pencil and ERA, where these methods are used as initial

conditions in the optimization. The calculation of the gradient equation is detailed, which allows for the use of line-search or trust region methods for solving the optimization. Beyond achieving lower least-squares error to the data, the variable projection method offers several advantages over past used traditional methods. The method does not require uniformly-spaced data, it is insensitive to noise, it allows inclusion of polynomial detrending basis functions, allows constrained optimization and so on. Additionally, the method gives the user several degrees of flexibility, such as control over the number of real and complex modes and also using initial conditions and constraints to guide a solution for experienced users.

The theory of normal form analysis was presented in Chapter 4. Normal form analysis improves on small-signal analysis and linearization solutions, typically used for analyzing nonlinear state-space models. As the theory shows, high order combinations of the system's natural modes are observed in dynamic disturbances of nonlinear systems. This high order interaction between a system's natural modes, can be used as motivation for an extension to the variable projection method for modal analysis problems. A normal form solution structure is imposed on the optimization in order to mimic behavior known to be present in nonlinear disturbances. The recasting of modal analysis problems as optimizations, allows for enforcing the normal form solution structure, where this type of control is not possible with standard methods. A three-bus power system example was presented, that shows the variable projection method with normal form solution structure most accurately estimating the system's true eigenvalues. Lastly, in Chapter 4 some discussion and a basic example is provided for analyzing systems with repeated eigenvalues.

The culmination of research work in Chapters 2-4 is brought together with a modal analysis tool developed for industry use in Chapter 5. The chapter discusses using trust region methods for unconstrained optimization versus using interior-point methods for linear inequality constrained optimization, for solving the variable projection modal analysis problem. Various application requirements and user requests are discussed, as well as complete effective strategies for modal analysis. An example using real power system data is presented which showcases the

graphic user interface and the flexibility and user interactivity available by the tool and method, which concludes the research work related to modal analysis.

In Chapter 6, a power system model security problem is examined, regarding confidential optimal power flow problems. Recent methods developed for securely solving linear programs in unsecure cloud computing environments motivate the approach. These methods obscure or mask the problem with the intention of using powerful yet unsecure computing environments. This work is specifically tailored for masking a linear optimal power flow (OPF) problem, i.e. the DC OPF. The masking procedure allows solution of OPF problems through use of powerful remote cloud computers without the worry of leaking confidential system information if the data is compromised. Some of the contributions to this problem included a procedure for extracting the Lagrange multipliers and locational marginal prices from the masked dual problem, a method for preserving problem sparsity, a method for masking a quadratic cost function and masking nonlinear constraints for the full nonlinear AC OPF.

In Chapter 7, the model security research is examined from another angle in keeping with the theme of confidential systems. A method is presented which takes an existing power system model and transforms it into a completely new and valid power system model, which preserves a mapping between optimal solutions. One system's solution maps to another system's solution via a simple linear transformation for DC OPF problems with linear, quadratic and piecewise-linear cost functions. In contrast to the industry motivated methods in Chapter 6, this research has more of an academic motivation. As a starting point for this research, to prove the actual existence of such transformations, the optimal solution from both problems are determined a priori. Future work will investigate the limits of this transformation without the need to solve the problems ahead of time. The findings from this work are of interest to those doing academic research. For example, a real optimal power flow problem could be solved initially; then the original system could be transformed into a new power system with the optimal solution known. The newly transformed system could be passed along to researchers for the purpose of testing and demonstrating optimization methods. In this way, their optimization methods would assuredly be tested on power systems possessing realistic characteristics. More importantly, it shows that

an OPF system model can be transformed into a family of equivalent OPF models, and the need to use “true system models” for purposes of algorithmic development becomes unnecessary. The results would suggest studies of any system with similar complexity are sufficient

Valuable contributions were made in all chapters; however, several topics of research in this dissertation will be ongoing for future work.

8.2 Future Work

The modal analysis work in Chapters 2-5 is mostly complete, yet there remains some areas of interest. The variable projection Hessian matrix could be computed to allow quadratic convergence. Computational time was not too concerning in this work, but if the method were to be employed in real-time in large-scale power systems, then computation savings would become increasingly more necessary. Other metrics for ranking mode dominance other than the mode energy would be of value. For example, statistical analysis on the optimization variables could provide confidence intervals on the solutions. In industry practice, the data is often repeatedly analyzed by manually varying the time-window of the analysis. This provides the engineer a primitive method for sensitivity analysis. A more sophisticated automation and theoretically-backed means of sensitivity analysis would benefit the confidence in solutions obtained from the optimization.

The model security work in Chapter 6 used a process of linear transformations to mask optimal power flow problems for use in cloud computing. As was shown in the chapter, the number of nonzero elements in the problem was purposefully increased and the confidential structure was randomly distributed. The masking approach yielded quick computational solve times for the smaller IEEE test systems. However, for larger systems with thousands of buses, a dense $\boldsymbol{P}$ matrix would sufficiently mask the problem but would also detriment the computational solve time. In practice, it would be up to the engineer as to how obscured they would like the model to be, with a trade-off between security and solve time. Certainly any means of improving computation performance in a completely obscured problem would help to get the best of both worlds in regards to security and solve time.

In the model security work, it is assumed that the numerical values of the original OPF model are completely hidden in the fully masked problem. Though it is difficult to prove a negative, it would be valuable to assess what level of security the method has. According to the [43], the numerical values of the masked problem are secure. Beyond that, it is worth knowing if any sophisticated data cracking schemes exist that can extract little or some aspect of the problem, such as the number of system facilities and topological connections. Further investigation into the level of security in the fully masked OPF problem is thus warranted.

Future work on the model security problem should be extended to more scenarios where optimization and confidential models are needed. For example, the masking method could be formalized for use in multi-party power system problems. This would involve formally identifying the types of power engineering problems that involve multiple parties sharing confidential data. The work could also be examined from an angle on confidential optimal power flow auctions without the need for trusted third parties facilitating the electricity market.

The structure preserving transformation research in Chapter 7 is a relatively new concept. This work would largely benefit from developing the transformation for the full nonlinear AC OPF problem. The fully masked problem in Chapter 6 used a transformation to obscure the model, but the masked problem does not have a power system structure. The transformation in Chapter 7, does create a power system structure, but the two systems are solved *apriori* in order to identify the transformation. This approach was purely used as a starting point to prove the existence of the transformation. Two important sufficient conditions were relaxed in order to prove the existence of a transformation between OPF problems. If the transformation can be shown to exist within the restrictions of these sufficient conditions, then the need to solve both problems apriori becomes no longer necessary. Any connections drawn between the transformations in Chapters 6 and 7 would help provide more insight. Additional future work for the structure preserving transformation would seek to create a function that when given a confidential OPF model, will generate a comparable synthetic model that possesses similar physical features and computational complexity as the original confidential model. Up to this point, a transformation has been developed that relates two given models. In this future work, knowledge of the transfor-

mation will be used to generate synthetic models from a given confidential model. Any advancements in this research could potentially improve the prospects of sharing confidential models and efficiently solving secure confidential OPF models.

8.3 Publications

Several publications have resulted from the research detailed in this report.

[85] B.C. Lesieutre, D.K. Molzahn, A.R. Borden, C.L. DeMarco, "Examining the Limits of the Application of Semidefinite Programming to Power Flow Problems," in *49th Annual Allerton Conference on Communication, Control, and Computing 2011*, Sept. 28-30, 2011

Work from Chapter 2, 3 and 5 is discussed in the following publications, with a 2nd place Best Student Paper Award at the NAPS 2013 conference.

[63] A.R. Borden, B.C. Lesieutre, J. Gronquist, "Power System Modal Analysis Tool Developed for Industry Use," in *North American Power Symposium (NAPS) 2013*, Sept. 22-24, 2013

[87] A.R. Borden, B.C. Lesieutre, "Variable Projection Method for Power System Modal Identification," Power Systems, IEEE Transactions on, [under review]

Work from Chapter 4 was presented at the NAPS 2012 conference and won Best Student Paper Award.

[31] A.R. Borden, B.C. Lesieutre, "Determining Power System Modal Content of Data Motivated by Normal Forms," in *North American Power Symposium (NAPS) 2012*, Sept. 9-11, 2012

Work from Chapter 6 was presented at the Allerton 2012 conference.

[45] A.R. Borden, D.K. Molzahn, P. Ramanathan, B.C. Lesieutre, "Confidentiality-Preserving Optimal Power Flow for Cloud Computing," in *50th Annual Allerton Conference on Communication, Control, and Computing, 2012*, Oct. 1-5, 2012

Work from Chapter 7 was presented at the Allerton 2013 conference.

[86] A.R. Borden, D.K. Molzahn, P. Ramanathan, B.C. Lesieutre, "Power System Structure and Confidentiality Preserving Transformation of Optimal Power Flow Problem," in *51st Annual Allerton Conference on Communication, Control, and Computing, 2013*, Oct. 2-4, 2013

References

[1] North American Electric Reliability Corporation, "Balancing and Frequency Control, A Technical Document Prepared by the NERC Resources Subcommittee," NERC, Princeton, N.J., Jan. 26, 2011

[2] K.H. LaCommare, J. H. Eto, "Understanding the Cost of Power Interruptions to U.S. Electricity Consumers," Berkeley National Lab., Berkeley, CA, LBNL-55718, Sept. 2004

[3] K.H. LaCommare, J. H. Eto, "Cost of Power Interruptions to Electricity Consumers in the United States (U.S.)," Berkeley National Lab., Berkeley, CA, LBNL-55718, Sept. 2006

[4] S. Chandrabhan, M.G. Ganness, "Determination of Power System Voltage Stability Using Modal Analysis," *Power Engineering, Energy and Electrical Drives, 2007. POWERENG 2007. International Conference on*, pp.381-387, April 12-14, 2007

[5] B. Pal, B. Chaudhuri, "Power System Oscillations," in *Robust Control in Power Systems*, Springer, 2005, ch.2

[6] D.N. Kosterev, C.W. Taylor, W.A.Mittelstadt, "Model Validation for the August 10, 1996 WSCC System Outage," *Power Systems, IEEE Transactions on*, vol.14, no.3, pp.967-979, Aug. 1999

[7] V. Venkatasubramanian, Y. Li, "Analysis of 1996 Western American Electric Blackouts," *Bulk Power System Dynamics and Control-VI*, Cortina d'Ampezzo, Italy, Aug. 2004

[8] I. Dobson, B.A. Carreras, V.E. Lynch, D.E. Newmanm "Complex systems analysis of series of blackouts: Cascading failure, critical points and self-organization," *Chaos, An Interdisciplinary Journal on Nonlinear Science*, vol.17, 026103, June 2007

[9] L. Nettleton, R. Padilla, "WSCC Tutorial on Power System Stabilizers," *WSCC Control Work Group*, Sept. 1999

[10] A. Ragavendiran, R. Gnanadass, "Determination of location and performance analysis of power system stabilizer based on participation factor," *Electrical, Electronics and Computer Science (SCEECS), 2012 IEEE Students' Conference on*, pp.1-9, March 1-2, 2012

[11] P.H. Huang, T.H. Tseng, S.C. Ding, "Power system stabilizer using local and remote signals," *SICE Annual Conference, 2008*, pp.222-226, Aug. 20-22, 2008

[12] E.V. Larsen, D.A. Swann, "Applying Power System Stabilizers Part I: General Concepts," *Power Apparatus and Systems, IEEE Transactions on*, vol.PAS-100, no.6, pp.3017-3024, June 1981

[13] W. Du, H.F. Wang, R. Dunn, "Power system oscillation stability and control by FACTS and ESS — A survey," *Sustainable Power Generation and Supply, 2009. SUPERGEN '09. International Conference on* , April 6-7, 2009

[14] C.T. Chen, Linear System Theory and Design, 3rd ed., Oxford University Press, 1999

[15] I. Dobson, E. Barocio, "Scaling of normal form analysis coefficients under coordinate change," *Power Systems, IEEE Transactions on*, vol.19, no.3, pp.1438-1444, Aug. 2004

[16] Z. Wang, Q. Huang, C. Zhang, "Development of a tool for the analysis of power system higher order oscillation," *Power & Energy Society General Meeting, 2009. PES '09. IEEE*, pp.1-7, July 26-30, 2009

[17] Q. Huang, Z. Wang, C. Zhang, "Evaluation of the effect of modal interaction higher than 2nd order in small-signal analysis," *Power & Energy Society General Meeting, 2009. PES '09. IEEE*, pp.1-6, July 26-30, 2009

[18] S.K. Starrett, "Application of normal forms of vector fields to stressed power systems," Ph.D. dissertation, Dept. Elect. Eng., Iowa State Univ., Ames, IA, 1994

[19] G. Gaeta, "Poincare Normal and Renormalized Forms." *Acta Applicandae Mathematicae* 70.1-3 (2002): 113-31. *ABI/INFORM Complete; ABI/INFORM Complete; ProQuest Research Library*. May 31, 2012

[20] J.J. Sanchez-Gasca, V. Vittal, M.J. Gibbard, A.R. Messina, D.J. Vowles, S. Liu, U.D. Annakkage, "Inclusion of higher order terms for small-signal (modal) analysis: committee report-task force on assessing the need to include higher order terms for small-signal (modal) analysis," *Power Systems, IEEE Transactions on*, vol.20, no.4, pp. 1886- 1904, Nov. 2005

[21] S. Liu, A.R. Messina, V. Vittal, "Characterization of nonlinear modal interaction using normal forms and Hilbert analysis," *Power Systems Conference and Exposition, 2004. IEEE PES*, vol.2, pp. 1113-1118, Oct. 10-13, 2004

[22] Y. Fang, Y. Hong-Geng, "Saddle-node Bifurcation of Power Systems Analysis in the Simplest Normal Form," *Computer Distributed Control and Intelligent Environmental Monitoring (CDCIEM), 2012 International Conference on*, pp.613-616, March 5-6, 2012

[23] L.L. Scharf, C. Demeure, Statistical Signal Processing: *Detection, Estimation, and Time Series Analysis*. Reading, Mass.: Addison-Wesley Pub. Co., 1991

[24] D.J. Trudnowski, J.M. Johnson, J.F. Hauer, "Making Prony analysis more accurate using multiple signals," *Power Systems, IEEE Transactions on*, vol.14, no.1, pp.226-231, Feb 1999

[25] Y. Hua, T.K. Sarkar, "Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol.38, no.5, pp.814-824, May 1990

[26] L.L. Grant, M.L. Crow, "Comparison of Matrix Pencil and Prony methods for power system modal analysis of noisy signals," *North American Power Symposium (NAPS), 2011*, pp.1-7, Aug. 4-6, 2011

[27] G.H. Golub, V. Pereyra, "The Differentiation of Pseudo-Inverses and Nonlinear Least Squares Problems Whose Variables Separate," *SIAM Journal on Numerical Analysis*, vol.10, no.2, pp.413-432, Apr 1973

[28] G.H. Golub, V. Pereyra, "Separable nonlinear least squares: the variable projection method and its applications," *Inverse Problems* 19 (2003) R1-R26, Feb. 14, 2003

[29] R. Jaramillo, M. Lentini, "Stability Analysis of a Variant of the Prony Method," *Mathematical Problems in Engineering*, vol.2012, article ID 390645, Oct 2012

[30] M.R. Osborne, G.K. Smyth, "A Modified Prony Algorithm for Exponential Function Fitting," *SIAM Journal of Scientific Computing*, vol.16, pp.119-138, 1995

[31] A.R. Borden, B.C. Lesieutre, "Determining Power System Modal Content of Data Motivated by Normal Forms," in *North American Power Symposium (NAPS) 2012*, Sept. 9-11, 2012

[32] D. Kosterev, D Davies, "System model validation studies in WECC," *Power and Energy Society General Meeting, 2010 IEEE*, 25-29 July 2010

[33] S. Frank, I. Steponavice, S. Rebennack, "Optimal Power Flow: A Bibliographic Survey I: Formulations and Deterministic Methods," *Energy Systems*, vol.3, issue 3, pp.221-258, Sept 2012

[34] J. Carpentier, "Contribution to the Economic Dispatch Problem," Bull Soc Fmnc Efeci 8(3):431-447

[35] R. O'neill, "New Approaches to Transforming Wind, Rain and Fire into Electricity – Making a Smarter, Cleaner, Efficient Grid," *Weston Roundtable Series*, Madison, WI, March 8, 2012

[36] IBM, "Optimization Applications in the Energy and Power Industries, White Paper on" IBM Corp. Somers, NY. Aug 2010

[37] Y. Jadeja, K. Modi, "Cloud computing - concepts, architecture and challenges," *Computing, Electronics and Electrical Technologies (ICCEET), 2012 International Conference on*, pp.877-880, March 21-22, 2012

[38] M. Ferris, M. Mesnier, "NEOS and Condor: Solving Optimization Problems Over the Internet," *ACM Transactions on Mathematical Software*, vol.26, no.1, pp.1-18, March 2000

[39] R. Fourer, J.J. Moré, T. Munson, S. Leyffer, "Extending a cyberinfrastructure to bring high-performance computing and advanced web services to the optimization community," *Proposal to the National Science Foundatoin*, 2006

[40] M. Kantarcioglu, A. Bensoussan, H. SingRu, "Impact of security risks on cloud computing adoption," *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, Sept. 28-30, 2011

[41] G. Dàn, R.B. Bobba, G. Gross, R.H. Campbell, "Cloud Computing for the Power Grid: From Service Composition to Assured Clouds," *USENIX Workshop on Hot Topics in Cloud Computing*, 2013

[42] C. Cachin, M. Schunter, "A cloud you can trust," *Spectrum, IEEE*, vol.48, no.12, pp.28-51, December 2011

[43] J. Dreier, F. Kerschbaum, "Practical Privacy-Preserving Multiparty Linear Programming Based on Problem Transformation," *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pp.916-924, Oct. 9-11, 2011

[44] H. Shojaei, A. Davoodi, P. Ramanathan, "Confidentiality Preserving Integer Programming for Global Routing." *Proceedings of the 49th Annual Design Automation Conference*, pp. 709-716. ACM, 2012

[45] A.R. Borden, D.K. Molzahn, P. Ramanathan, B.C. Lesieutre, "Confidentiality-Preserving Optimal Power Flow for Cloud Computing," in *50th Annual Allerton Conference on Communication, Control, and Computing, 2012*, Oct. 1-5, 2012

[46] D.J. Hill, I.A. Hiskens, I.M.Y. Mareels, "Stability theory of differential/algebraic models of power systems", *Proceedings in Engineering Sciences of the Indian Academy of Sciences*, special issue on 'Nonlinear dynamics: Applications in power systems', Vol. 18, Part 5, September 1993, pp. 731-748.

[47] W.H. Hayt, J.E. Kemmerly, S.M. Durbin, *Engineering Circuit Analysis*, 8th ed., McGraw-Hill Education, 2011

[48] J.F. Hauer, C.J. Demeure, and L.L. Scharf, "Initial results in Prony analysis of power system response signals," *Power Systems, IEEE Transactions on*, vol.5, no.1, pp.80-89, Feb 1990

[49] M.A. Johnson, I.P. Zarafonitis, and M. Calligaris, "Prony analysis and power system stability-some recent theoretical and applications research," *Power Engineering Society Summer Meeting*, 2000. IEEE , vol.3, pp.1918-1923, 2000

[50] J.C.-H. Peng, N.-K.C. Nair, "Effects of sampling in monitoring power system oscillations using on-line Prony analysis," *Power Engineering Conference*, 2008. AUPEC '08. Australasian Universities, pp.1-5, 14-17 Dec. 2008

[51] J.-N. Juang, Applied System Identification, Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1993

[52] R.A. De Callafon, B.Moaveni, et al., "General Realization Algorithm for Modal Identification of Linear Dynamic Systems," *ASCE Journal of Engineering Mechanics*, vol.134, issue.9, pp.712-722, Sept. 2008

[53] F. Milano, "An Open Source Power System Analysis Toolbox," *Power Systems, IEEE Transactions on*, vol.20, no.3, pp.1199-1206, Aug. 2005

[54] D.P. O'Leary, B.W. Rust, "Variable projection for nonlinear least squares problems," *Computational Optimization and Applications*.

[55] Varpro*: http://www.cs.umd.edu/~oleary/software/varpro.m*

[56] L. Kaufman, A variable projection method for solving separable nonlinear least squares problems. BIT Numerical Mathematics 15(1), 49–57 (1975)

[57] J. Nocedal, S. J. Wright, *Numerical optimization*, 2nd ed., New York, USA: Springer, 2006

[58] N. Zhou, D. Trudnowski, J.W. Pierre, S. Sarawgi, and N. Bhatt, "An algorithm for removing trends from power-system oscillation data," *Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, 2008 IEEE, pp.1-7, 20-24 July 2008

[59] T. Guo, J.V. Milanovic, "Probabilistic Framework for Assessing the Accuracy of Data Mining Tool for Online Prediction of Transient Stability," *Power Systems, IEEE Transactions on*, vol.PP, no.99, pp.1,9

[60] G. Rogers, Power System Oscillation, Kluwer Academic Publishers, Norwell, USA, 1999

[61] J.H. Chow and K.W. Cheung, A toolbox for power system dynamics and control engineering education and research, *Power Systems, IEEE Transactions on*, vol.7, no.4, pp.1559-1564, Nov 1992.

[62] M.A. Branch, T.F. Coleman, Y. Li, "A Subspace, Interior, and Conjugate Gradient Method for Large-Scale Bound-Constrained Minimization Problems," SIAM Journal on Scientific Computing, vol.21, no.1., pp.1-23, 1999

[63] A.R. Borden, B.C. Lesieutre, J. Gronquist, "Power System Modal Analysis Tool Developed for Industry Use," in *North American Power Symposium (NAPS) 2013*, Sept. 22-24, 2013

[64] Western Electricty Coordinating Council, Joint Synchronized Information Subcommittee, "Oscillation Detection and Analysis Applications in WECC," Nov. 19, 2012: http://www.wecc.biz/committees/JSWG/01152013/Lists/Minutes/Forms/DispForm.aspx?ID=5

[65] Western Electricty Coordinating Council, Joint Synchronized Information Subcommittee, "WECC Guideline for Data Format Used in Engineering Analysis Applications of Disturbance and Simulated Data," Aug. 26, 2013: https://www.wecc.biz/committees/JSWG/Shared%20Documents/Forms/AllItems.aspx

[66] T.F. Coleman, Y. Li, "On the Convergence of Reflective Newton Methods for Large-Scale Nonlinear Minimization Subject to Bounds," *Mathematical Programming*, vol.67, no.2, pp. 189-224, Feb. 1994

[67] T.F. Coleman, Y. Li, "An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds," *SIAM Journal on Optimization*, vol.6, no.2., pp.418-445, May 1996

[68] J.J. Moré, D.C. Sorensen, "Computing a Trust Region Step," *SIAM Journal on Scientific and Statistical Computing*, vol.3, no.3, pp.553-572, Sept. 1983

[69] T. Steihaug, "The Conjugate Gradient Method and Trust Regions in Large Scale Optimization," *SIAM Journal on Scientific and Statistical Computing*, vol.20, no.3, pp.626-637, June 1983

[70] C.-J. Lin, J.J. Moré, "Newton's method for large-scale bound constrained problems," *SIAM Journal on Optimization*, vol.9, issue 4, pp.1100-1127, 1999

[71] R.H. Byrd, M.E. Hribar, J. Nocedal, "An Interior Point Algorithm for Large-Scale Nonlinear Programming," *SIAM Journal on Optimization*, vol.9, no.4, pp.877-900, 1999

[72] R.A. Waltz, J.L. Morales, J. Nocedal, D. Orban "An interior algorithm for nonlinear optimization that combines line search and trust region steps," *Mathematical Programming*, vol.107, no.3, pp.391-408, 2006

[73] J. Nocedal, "Updating Quasi-Newton Matrices with Limited Storage," *Mathematics of Computation*, vol.35, no.151, pp.773-782, July 1980

[74] G.H. Golub, C.F. Van Loan, Matrix Computations. 3rd Ed. JHU Press, 2012.

[75] M.C. Ferris, O.L. Mangasarian, S.J. Wright, *Linear Programming with MATLAB*, MPS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics and the Mathematical Programming Society, 2007

[76] J. Sun, L. Tesfatsion, DC Optimal Power Flow Formulation and Solution Using QuadProgJ. *ISU Economics Working Paper No. 06014*. Revised: March 1, 2010. Available at: http://www2.econ.iastate.edu/tesfatsi/dc-opf.jslt.pdf

[77] Power Systems Test Case Archive, University of Washington Department of Electrical Engineering. Available at: http://www.ee.washington.edu/research/pstca/

[78] E. Cuthill, J. McKee, "Reducing the Bandwidth of Sparse Symmetric Matrices," *Proceedings of the 1969 24th National Conference ACM '69*, pp.157-172, 1969

[79] A. George, J. Liu, "Computer Solution of Sparse Linear Systems," Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1981

[80] P. Bunus, P. Fritzson, "Methods for Structural Analysis and Debugging of Modelica Models," *2nd International Modelica Conference, Proceedings*, pp.157-165, March 18-19, 2002

[81] C. Ashcraft, J.W.H. Liu, "Applications of the Dulmage-Mendelsohn Decomposition and Network Flow to Graph Bisection Improvement," *SIAM J. Matrix Anal. Appl.*, vol.19, pp.325-354, 1998

[82] GAMS Development Corp. "GAMS – A User's Guide, Tutorial by R.E. Rosenthal," Washington, DC, Jan 2012

[83] R.D. Zimmerman, C.E. Murillo-Sánchez, R.J. Thomas, "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," *Power Systems, IEEE Transactions on*, vol.26, no.1, pp.12-19, Feb. 2011

[84] O. Vukovic, G. Dàn, R.B. Bobba, "Confidentiality-preserving Obfuscation for Cloud-based Power System Contingency Analysis," *IEEE Transactions on SmartGridComm*, Oct. 2013

[85] B.C. Lesieutre, D.K. Molzahn, A.R. Borden, C.L. DeMarco, "Examining the Limits of the Application of Semidefinite Programming to Power Flow Problems," in *49th Annual Allerton Conference on Communication, Control, and Computing 2011*, Sept. 28-30, 2011

[86] A.R. Borden, D.K. Molzahn, P. Ramanathan, B.C. Lesieutre, "Power System Structure and Confidentiality Preserving Transformation of Optimal Power Flow Problem," in *51st Annual Allerton Conference on Communication, Control, and Computing, 2013*, Oct. 2-4, 2013

[87] A.R. Borden, B.C. Lesieutre, "Variable Projection Method for Power System Modal Identification," *Power Systems, IEEE Transactions on*, [under review]