ALGEBRAIC SET PRESERVING MAPPINGS FOR ELECTRIC POWER GRID MODELS AND ITS APPLICATIONS

By

Dan Wu

A dissertation submitted in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

(Electrical Engineering)

at the

UNIVERSITY OF WISCONSIN–MADISON

2017

Date of final oral examination: June 14, 2017

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

Stephen J. Wright, Professor, Electrical and Computer Engineering

Michael C. Ferris, Professor, Computer Sciences

Nigel Boston, Professor, Mathematics

To my parents and my grandparents.

To my middle school mathematics teacher Mr. Shuang Liu.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to many people who have helped me throughout my graduate study. Among them I am most indebted to my advisor, Dr. Bernard Lesieutre, who guided my study with his extraordinary inspiration, great dedication, constant encouragement and wide knowledge. As an international student, I was blank of American social conventions. Bernie never felt offended when I unintentionally broke these conventions, but patiently taught me about them. He was also very tolerated on my course selections so that I learnt a lot from different areas simply for my curiosity. I will always remain my gratitude for his advising!

I am also very grateful for the supervising and assistance provided by my doctoral committee. Dr. Parameswaran Ramanathan's guidance of cryptography and parallel computing enriched my knowledge and skills of computer engineering. His assistance in publishing my first paper became invaluable for me. Dr. Christopher DeMarco guided me for my first year graduate study. He encouraged me to build a profound foundation of analysis. I also greatly benefited from Dr. Michael Ferris and Dr. Stephen Wright with their optimization classes. The optimization knowledge I learnt from them has become an essential underpinning in my doctoral research. I also learnt a lot of abstract algebra from Dr. Nigel Boston in our weekly discussions. I am much grateful for his great efforts in helping me understand engineering problems in an abstract algebraic way.

There are many other professors who have guided me in my graduate and undergraduate educations. Dr. Bob Barmish's linear system course helped me build a very concrete foundation for system theories and controls. His kind and wise mentoring enlightened my career choice. Dr. Lindsay Stovall's real analysis courses inspired my curiosity in mathematics and provided me with one of the most useful tools for my research. I am very grateful to Dr. Daniel Cobb who offered

me my first academic job (teaching assistant) and guided me throughout optimal control and variational methods. I am also very thankful to Dr. Sergey Bolotin for his class on nonlinear symbolic dynamics, and for his patience of my thousands of questions. I also appreciate the assistance from Dr. Steve Goldstein and Christina Koch for their help on my parallel computing implementations in CHTC platforms. When back to my undergraduate, I am always indebted to Dr. Jie Wu and Dr. Tao He for their dedication to my national mathematics competition. I further thank Dr. Zhiwei Bi for his first class in calculus.

I also very appreciate the advice and discussions from my friends and colleagues. Many of them are current and former graduate students at the University of Wisconsin-Madison. I would like to express my great gratitude to Dr. Daniel Molzahn for his kind discussions on my research, invaluable assistance on my paper writing, and great support on my job searching. I would also like to thank Dr. Honghao Zheng for his great inspiration on my research, and for teaching me how to drive. I am also very grateful to my friend Dr. Meihua Kuang for her insightful biologist's point of view in my graduate studies, and for her great patience and suggestions. My colleagues are also very helpful in my graduate studies. These include Chao Ren, Ziyi Xiu, Meizheng Chen, Zachary Charles, Alisha Zachariah, Byungkwon Park, Hillary Brown, Sowmya Acharya, Jonathan Snowdgrass, Jeffrey Wettstein, Zhongyi Xia, Ruihao Zhu, Ziqi Yang, Bozhao Qi, Di Han, Baoyun Ge, Yuying Shi, Wenbo Liu, Ying Zheng, Jiejian Dai, Hanyang Dai, Tiantian Nie.

I would like to express a special gratitude to my middle school mathematics teacher Mr. Shuang Liu who has passed away years ago. But his encouragement to me and rigorous attitude towards science has always been my treasure.

I would like to express my deepest gratitude for the love and support from my family. I feel very fortunate to grow up in a family of professors and teachers. I can never thank my mom, Jing Hu, enough for her unrequited love and care of me, and never thank my dad, Yaowu Wu, enough for his encouragement and scientific trainings. I am also very grateful for the love and support from my grandparents, granduncles and grandaunts. I am especially indebted to my grandpa, Bingsheng Hu, who built the important character of persistence in my life. Finally, I greatly appreciate the support from my uncles and aunts, cousins and nephews.

I gratefully acknowledge the support from the National Science Foundation under NSF Grant CNS-1329452 and the U.S. Department of Energy, Office of Science under grant DE-AC02-06CH11357 Argonne National Laboratory subcontract 3F-30222. I also really appreciate the award from the Global Stewards Sustainable Prize of Wisconsin Energy and Sustainability Challenge.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Chapter 1

Introduction

1.1 Overview

This thesis investigates “algebraic set preserving mappings” with application to three relevant but different topics in power engineering. These mappings convert the original problem to a new problem while preserving the solution. With careful design, such mappings are able to alter characteristics of the original problem without compromising the original solution. The problem representation idea is applied to the power flow problem, the optimal power flow problem and the multi-party optimal power flow problem respectively. It introduces favorable features that aid the design of new tools for these problems.

The first application attempts to efficiently compute multiple real-valued power flow solutions. Currently many mature routines and algorithms have been developed to solve the power flow problems for one particular solution, typically a “high-voltage” solution. However, it is important in some cases to obtain other solutions, typically ”low-voltage” solutions, but may also include other high voltage solutions. For example, the additional low-voltage solutions provide information about stability margins for voltage and transient stability analyses. In theory, it is possible to compute all complex-valued solutions and then sift out the real-valued solutions, however, the vast number of complex solutions makes this approach impractical. In their 1993 paper, Ma and Thorp observed that the number of real-valued solutions to the power flow problems occupies a small fraction of the total number of complex solutions. They analyzed an efficient tracing algorithm on

real manifolds that connects multiple solutions. In this thesis, an algebraic set preserving mapping is carefully designed to preserve the affine algebraic set of the power flow problem while changing the unbounded manifold of each power flow equation into a high-dimensional ellipsoid. Using this model, the traces are guaranteed to be bounded and return the solution set of the power flow problem. The revised tracing method proposed in this thesis is tested on all power flow problems for which the entire real solution sets are known, and further provides solution sets to several more benchmark systems which have never been completely solved. We further report two modified systems for which the numbers of real solutions increases temporarily with the increase of bus loads, and another modified system that has two "high-voltage" solutions.

The second application is directed to identify the global solution to the optimal power flow (OPF) problem. Typical nonlinear OPF algorithms, when they converge, provide only one solution for each initial point, and it is not guaranteed to be the global optimum due to the nonconvexity of the OPF problems. Recently, techniques for convex relaxation, especially semi-definite programming (SDP), have gained popularity because it can provide the global solutions to the OPF problems under certain conditions. However, when the SDP relaxation fails, it doesn't provide a physically-meaningful solution. In this thesis, we introduce a new way to locate multiple local extrema, in hope of enumerating the global solution. To our best knowledge, it is the only principled deterministic method to identify multiple local solutions to the OPF problems without any relaxations. Specifically, an algebraic set preserving mapping is carefully constructed to preserve the affine algebraic set of the Fritz John conditions of the OPF problem. It converts the unbounded manifold of each Fritz John equation into a high-dimensional ellipsoid. Then the branch tracing method is applied to locate the solution set of the Fritz John conditions. To enhance the searching efficiency, a monotone greedy search strategy enforces a non-increasing objective function at each step. The proposed method is tested on several hard ACOPF cases for which the first-order SDP relaxation fails to provide exact solutions. We also demonstrate that for a particular 39-bus system our proposed method found four additional local minima to the three known local minima. Finally, a small example shows that the proposed method is able to locate local minima which do not satisfy the KKT conditions.

The third application designs an encryption strategy to mask sensitive information in OPF data for shared computing techniques in a multi-party scenario. The idea is that OPF problem may be solved efficiently on a shared computing platform. However, power system data are confidential for security concerns, and in the multi-party scenario the shared computing platform also has to protect the commercial information for each participant while jointly solving multiple parties together. Traditional data encryption techniques require decryption before use, which may increase the risk of privacy loss in the shared platforms. The current fully homomorphic encryption technique, which does not require decryption before use, increases the computing time tremendously. To compromise both the security and the computational complexity, this thesis designs a strategy that enables each participant to construct its own encryption mapping to mask their sensitive information. A computing procedure is developed based on the masked data to obtain a masked solution. Finally, the solution is returned to each party for decryption by each participant.

Beyond the specific contributions to the above three topics, the basic idea should be emphasized: changing the original problem to a new one without compromising the solution can be beneficial for both understanding the problem and solving it. This is specifically done by the algebraic set preserving mappings in this thesis, but can be extended to a broader realm for other applications, including homology group preserving, fundamental group preserving, etc.

1.2 Background: the Power Flow Problem

The power flow equations describe the relation between power injections and bus voltages. Their derivations and mathematical descriptions in both polar coordinates and rectangular coordinates can be found in Chapter 2. These equations are nonlinear and admit multiple real-valued solutions [1, 2, 3, 4]. In practice a single solution (known as the high-voltage solution) is of primary interest, however, there are certain types of problems for which it is necessary to find multiple solutions. One example is the energy function method for angular stability analysis. The stability boundary of a stable equilibrium point (SEP) is characterized by the union of the stable manifolds of all the type-1 unstable equilibrium points (UEP) around that SEP. Thus, the application of energy function relies on finding these closest and controlling type-1 UEP's [5, 6, 7].

Another example involves a measure of energy between a stable high-voltage equilibrium point and a nearby unstable low-voltage equilibrium point as the voltage stability index [8, 9]. To ensure the needed equilibriums are found, the complete solution set of power flow equations should be provided. Another example focuses on the cascading stall of induction machines for fault-induced delay voltage recovery studies. This problem can be analyzed using bifurcation diagrams that trace every solution to a distribution network [10].

The study of multiple power flow solutions dates at least back to 1970's, when Tavora and Smith examined a 3-bus case which admits 0, 2, 4 or 6 real-valued solutions [11]. In their analysis, under the assumptions of lossless lines and PV buses, the number of distinct real-valued solutions to a power flow problem is determined by twice the number of the folds in its parameter range. The upper bound of the number of real-valued solutions is twice the number of distinct singular surfaces which is defined by the zero set of the determinant of the power flow Jacobian matrix.

In 1982 Baillieul and Byrnes derived a bound for the number of complex-valued solutions to any N bus lossless network for all PV buses [12]. Their bound is sharper than the bound given by Bezout's theorem. Bezout's bound is a general bound for any finite-degree polynomial systems, and to obtain such bound for the power flow equations, one must first homogenize the power flow equations. The degree of the homogenized equation is identical to the degree of the original power flow equation. By Bezout's theorem, the number of complex-valued solutions to the associated homogenized polynomials is the product of degrees of all the homogeneous polynomials, provided no continuum solution set. Thus, for power flow equations, the bound for the number of complex-valued solutions is given by 2^{2N-2}. However, Baillieul and Byrnes pointed out that the actual number of finite solutions to the power flow equations is less than Bezout's bound because Bezout's bound includes the number of solutions at infinity. Moreover, there is a subtle difficulty in applying Bezout's theorem for larger systems because there could be non-trivial parameter subsets which admit infinitely many solutions. To take out the solutions at infinity as well as to overcome the difficulty of infinitely many solutions, Baillieul and Byrnes applied results from intersection theory to reduce the bound to $\begin{pmatrix} 2N-2 \\ N-1 \end{pmatrix}$.

In 1989, Salam et al. applied a the probability-one homotopy method to find all the complex-valued solutions provably [13, 14]. The basic idea of this method is to trace homotopy curves defined by a homotopy function between two polynomial systems. Specifically, suppose polynomial system $F(x)$ is non-trivial. Design another polynomial system $Q(x)$ the solution set of which is trivial to compute and the number of solutions is identical to Bezout's bound (or Baillieul and Byrnes' bound). Then construct the homotopy function by $H(x,t) = tF(x) + (1-t)Q(x)$ with $t \in [0,\ 1]$. Starting at solutions of $H(x,0) = 0$, continuously change t from 0 to 1 and keep trace of every x, the solutions of $F(x) = 0$ will finally be reached at $H(x,1) = 0$. The number of the traces followed by this method is determined by the solution bound. For example, tracing a 30-bus system with Baillieul and Byrnes' bound requires about 3×10^{16} homotopy curves. Unless a very sharp bound appears, it is computationally very expensive to analyze a medium sized system using existing bounds.

To efficiently locate all the real-valued power flow solutions, Ma and Thorp introduced a tracing method, firstly appeared in [15], to connect real-valued solutions [16]. In this thesis, we will refer this tracing approach to the "branch tracing method" [17], while people sometimes refer it to "path following method" as well. Differing from tracing homotopy curves between two polynomial systems, the branch tracing method follows the curves defined by $N-1$ polynomials from total N polynomials each time. The curves all intersect as solutions, or looking from the solutions, the curves "branch" out from the solutions. An initial solution is required as a starting point which usually can be obtained by Newton's method. At each solution, trace every branch curve and collect the new solutions until no more new solutions are found. Since each trace generically admits more than one solutions (at least two), it only requires at most $N_{solu} \times (2N-1)$ many traces, where N_{solu} is the number of the real-valued solutions to the polynomial system. Given that N_{solu} is very small compared to the number of complex-valued solutions, this method is much more efficient than the bound-limited homotopy continuation methods.

In Ma and Thorp's original design, the power flow equations were cast in polar coordinates and the voltage equations on PV buses were neglected. Although they claimed that the branch tracing method found the complete real-valued solution set in their model, a counter-example was

presented in [18] showing that their method can fail to return a complete solution set. Given this counter-example, the only method that can provably find all the real-valued solutions is the bound-based homotopy continuation method.

To provably locate all the real-valued solutions for any network configurations, Mehta et al. in 2014 applied a polyhedral homotopy continuation method (similar to probability-one homotopy method) to a Bernstein-Khovanskii-Kushnirenko (BKK) bound. This bound is calculated by the mixed volume of Newton polytopes of given polynomials. It can be reached by some certain polynomial systems and it is sharper than Bezout's bound [19]. For example, the Bezout's bound for IEEE 14-bus system is 67,108,864, while the BKK bound is 49,283,072. To further explore a simpler bound, Chen and Mehta proposed the adjacency polytope (AP) bound in 2015, which is given by the normalized volume of a symmetric adjacency polytope [20]. It may be greater than the BKK bound, but more easily computable than BKK bound. Chen and Mehta's work also suggested that the topology of the network influences the number of the complex-valued solutions. To investigate this characteristic, Molzahn et al. evaluated the possible number of real-valued solutions to 4-bus systems and conjectured that the number of the real-valued solutions will be strictly less than the number of the complex-valued solutions for power flow equations with more than three buses [21]. That is, the number of real-valued solutions cannot equal the number of complex-valued solutions for large systems.

Although progressive, the homotopy continuation methods based on solution bounds still face several difficulties for large applications. The simple Bezout's bound is easy to calculate and easy to be implemented, but requires a huge number of traces to follow. The BKK bound and AP bound are smaller, but not easy to calculate.

Considering the easy implementation and high efficiency of the branch tracing method, this thesis applies it to solve all the real-valued solutions for all the existing cases in Chapter 4 Section 4.3. Innovatively, we apply the ellipsoidal formulation to the original polynomial system. The ellipsoidal formulation provides the boundedness for every 1-dimensional manifold, and includes the voltage equations for the PV buses which were omitted in Ma and Thorp's original design. Though without theoretical guarantee, experimentally we report that the proposed method found

the complete real solution sets to all the cases of which the complete real solution sets are known, and some real solution sets for several other benchmark systems.

1.3 Background: the Optimal Power Flow Problem

The optimal power flow (OPF) problem was first introduced in 1962 by Carpentier [22], and has since acquired an extensive amount of attention. Typically an OPF problem minimizes an economic cost function over a set of constraints which are derived from power balance relations and engineering limits. The cost function is often quadratic or piecewise linear with respect to the active power generation, and the power balance constraints are nonlinear with respect to bus voltage magnitudes and angles. This complicated nonlinear model is usually referred to as the AC optimal power flow (ACOPF) problem which is generally nonconvex [23, 24]. The nonconvexity further induces multiple local solutions to the problem [25, 26].

To solve ACOPF problems, several nonlinear optimization algorithms have been explored in the literature, including sequential linear programming, sequential quadratic programming, penalty function method, interior point method, etc [27, 28, 29]. These methods are not the subjects of this thesis because they are neither focused on the global solution nor focused on multiple solutions. Most of the work for ACOPF problems was primarily on enhancing the reliability and the efficiency of obtaining a local solution, leaving the global optimality elusive.

Recently, convex relaxation methods, especially semi-definite programming (SDP), succeeded in obtaining the global solution for many ACOPF cases [30, 31]. It was first proposed as a second-order cone programming (SOCP) problem in [32] and as an SDP problem in [31].

The SDP technique is derived from the equivalent relation as follow

$$\mathbf{x}^H\mathbf{M}\mathbf{x} = Trace(\mathbf{M}\mathbf{W})$$

where $\mathbf{x} \in \mathbb{C}^n$; $\mathbf{x}^H$ is the conjugate transpose of $\mathbf{x}$; $\mathbf{M}$ is a $n \times n$ Hermitian matrix; $\mathbf{W} = \mathbf{x}\mathbf{x}^H$. Note that $\mathbf{W}$ is a positive definite Hermitian rank-1 matrix.

A quadratic programming problem in the following form

$$\begin{aligned} \textbf{Minimize} \quad & \mathbf{x}^H \mathbf{D} \mathbf{x} \\ \textbf{Subject to}: \quad & \mathbf{x}^H \mathbf{M}_i \mathbf{x} \geq r_i \end{aligned}$$

can be cast in the SDP form as

$$\begin{aligned} \textbf{Minimize} \quad & Trace(\mathbf{DW}) \\ \textbf{Subject to}: \quad & Trace(\mathbf{M}_i \mathbf{W}) \succeq r_i \\ & \mathbf{W} \succeq 0 \\ & rank(\mathbf{W}) = 1 \end{aligned}$$

If we relax the rank-1 condition, the optimization problem in SDP form becomes convex. If this convex optimization problem is solved with rank-1 solution, then the original nonconvex problem is solved with global optimality (equivalently zero duality gap). However, Lesieutre et al. and others reported that the SDP relaxation can fail to provide a physically meaningful solution to the original nonconvex problem [33, 34, 35]. To ensure the SDP relaxation is exact, several sufficient conditions have been proposed [36, 37, 38, 39, 40, 41, 42, 43, 44], all of which require some certain graphic structures (tree, mesh with tunable phase shifters, etc). A review of recent progress for convex relaxation techniques in ACOPF can be found in [45, 46].

To obtain a solution when SDP relaxation fails, Mozahn and Hiskens developed the moment-based relaxations for ACOPF problems [47]. It is a generalized version of SDP with relaxations on higher order polynomials based on the Lasserre hierarchy in approximation algorithms [48]. It increases the dimensions of the problem significantly but a sparsity structure can be retained to reduce the computational complexity [47]. This approach is promising and research continues, however, it is recognized that it scales very poorly with the order of the relaxation. Despite the exciting breakthroughs in convex relaxation theories and applications, the algorithms are unable to provide a physically meaningful solution when the relaxation is not exact.

Another method that is capable of achieving global optimality for nonlinear OPF problems is the branch-and-bound method. This method was first proposed by Land and Doig in 1960 for solving the integer programming problems [49]. However, it turned out to be a global solver not

only for solving integer programming problems but also for solving continuous, discrete or mixed-integer nonlinear optimization problems. The searching framework includes three basic subroutines: branching, bounding and pruning [50]. At each step, the branching subroutine partitions the current feasible space into several regions, the bounding routine calculates the upper bound and the lower bound for each region, and the pruning routine prunes off the regions of which the lower bounds are greater than the upper bound of some region. This procedure is executed recursively until the difference between the upper bound and the lower bound of the remaining region is less than a given tolerance. The central part of this method relies on how to efficiently estimate the lower bound for each partitioned region. Several techniques have been proposed and are explained in [51]. In 2012, Phan introduced the Lagrangian duality function for ACOPF problems to estimate the lower bound on each partition [52]. Then Gopalakrishnan et al. applied the SDP relaxation to obtain the lower bound for each partition and compared it to the Lagrangian duality bound [53]. Their results indicated that the SDP relaxation may have a more reliable performance than the Lagrangian duality bound. Chen et al. in 2015 proposed a technique to tighten the bound of OPF variables which enhanced the performance of the branch-and-bound method for OPF problems [54]. Although the branch-and-bound method is promising to obtain the global solution for OPF problems without any conditions, the number of recursive steps may, in the worst case, grow exponentially with the growth of problem dimensions [55], despite that at each step a sub-optimization problem should be solved.

Another way to achieve global optimality is to enumerate all the local solutions. This approach seems less efficient than the convex relaxation, but does not require any relaxations, and can also provide multiple local solutions. The enumeration strategy usually faces the challenge that there is no deterministic way to identify all the local solutions. Many local solutions of ACOPF problems were merely generated by random initiation: scattering a huge number of random initial points and executing a standard nonlinear programming solver for every initial point [25]. This approach requires a large amount of initializations and can miss out solutions for large systems as we will demonstrate in Chapter 4.

In Chapter 4 Section 4.4 we introduce a principled deterministic algorithm for finding multiple OPF solutions, which has been modified to accommodate an efficient monotone search strategy to seek non-increasing local minima. We further compare it to the simplex method in linear programming and try to design more advanced search strategies to enhance efficiency. Such algorithm is able to be extended to a family of quadratic constrained quadratic programming (QCQP) problems.

1.4 Background: Multi-Party ACOPF Problems for Cloud Computing

The electric power grid is a critical infrastructure that is vulnerable to physical and cyber attacks [56, 57, 58, 59, 60, 61]. For security, it is important to minimize access to the actual power flow models. For economic competitiveness, the market participants are not inclined to share their sensitive information with other participants. It is a goal then to reveal as little as possible about the OPF models and their solutions. At present, a multi-party structure already exists in power grid managed by third-party Independent System Operators that manage transmission-coupled supply and distribution companies. With the emerging smart grid trends including Distribution System Operators (DSO) that will coordinate the operation of coupled micro-grids and interface with a larger grid, the individual micro-grids will likewise value privacy and security. Thus the multi-party encryption techniques are of interests to enable providers to preserve their sensitive information while allowing joint optimal operation of coupled grids and micro-grids.

For this third application, we pause to observe that the objective and form differs from previous two. Like those applications we will introduce a transformation that changes the form of the representation while preserving the solution. However in this case, the objective of the transformation is to mask the form of the original equations, and also to mask the actual value of the solution. The latter is a mild departure from the other approaches in that the solutions are mapped back to the original variables through a linear transformation.

There is extensive research on encryption techniques for exchanging information between two parties that prevents an eavesdropping third-party from deciphering the information [62]. These techniques require ”decryption-before-use”, i.e., the receiving party must first decrypt the encrypted information before it can make use of the information. These techniques are useful but

not exactly suitable to our purpose because the providers may be reluctant to decrypt their individual ACOPFs before solving the optimization problem.

More recently, there has been considerable progress on homomorphic encryption schemes where one can evaluate a function directly on the encrypted inputs to obtain encrypted results [63, 64, 65]. These techniques are useful for offloading computations to a third-party without revealing sensitive information; the offloading party encrypts the inputs and also decrypts the results. Although this work is inspired by this body of research, we do not apply any homomorphic encryption techniques from literature for the OPF because solving the optimization problem on homomorphically encrypted data is computationally very expensive [66]. Furthermore, these encryption schemes are not designed to hide sensitive electric grid information such as network topology, power generator limits, and line limits.

Secure multi-party computation techniques permit multiple parties to jointly evaluate a function without revealing each other's inputs to the other parties [67, 68, 69, 70]. One commonly used approach involves encrypting the function as a garbled logic circuit such that the output of the garbled circuit on encrypted inputs from all the parties is the output of the function [71, 69]. The approach in this thesis is a tailored realization of the secure multi-party computation for ACOPF problem. We specifically tailor the multi-party computation to ensure desirable properties such as preserving optimality and making efficient use of communication and storage resources.

There is also a body of research based on the notion of differential privacy. The idea was originally introduced in the context of databases where the objective is to prevent leakage of sensitive information in query responses [72]. The solution approaches for differential privacy perturb the database records to either ensure k-anonymity where an attacker can, at best, associate sensitive information to one of k possible records or they are designed to provide approximate, but statistically accurate, query results. In this thesis, we do not consider privacy approaches that provide approximate but statistically accurate solutions. Instead, our goal is to preserve the optimality of the solution, i.e., the power flow solution from the privacy-aware approach must be equivalent to the solution from a traditional non-privacy aware ACOPF solver. Notions of k-anonymity may be

useful in preserving the privacy of the grid topology. We take a different approach for preserving privacy of grid topology.

There are privacy solutions in the context of distributed controls. For instance, in [73], differential privacy approach prevents leakage of sensitive input information based on observation of the unencrypted outputs in a distributed control system. In contrast, in our approach, the outputs (i.e., solution of the optimization problem) remain obfuscated in such a way that each party can only unmask its part of the overall solution. Preserving the privacy of the solutions of the optimization is critical in the ACOPF context. Hence, we pursue solution approaches that preserve of the optimization outputs in the sense that each provider can only de-obfuscate his/her part of the solution. Furthermore, neither an eavesdropper nor the third-party that solved the optimization formulation can knowingly recover the solution corresponding to any provider. As a tradeoff, the optimization problem we solve does not retain the structure of ACOPF. Furthermore, there is also some degradation in the level of sparsity; albeit the amount of degradation can be controlled to some extent. As a result of these tradeoffs, the time required to solve the obfuscated optimization problem may be larger than that required for standard ACOPF.

[74, 75] proposed a solution approach to hide sensitive structural information in the single party DC Optimal Power Flow (DCOPF) problems. The DCOPF is usually formulated as an optimization problem with linear constraints and either a linear or quadratic objective. In this approach, the power system operator first obfuscates all the sensitive model information by applying a linear transformation that preserves the sparsity of the underlying formulation. The transformed problem is then sent to shared-computing platform such as a cloud computer for solving [76]. The authors argue that only the power system operator has the ability to recover the sensitive information from either the transformed problem and/or its solution. The solution is transferred back to power system operator who then applies a decipher transformation to recover the optimal solution to the original problem. They also showed that there is not much increase in the time required to solve the optimization.

In Chapter 5, as well as in [77], we focus on the nonlinear ACOPF problem with simultaneous multi-party obfuscation. Unlike the DCOPF, the constraints and the objective in ACOPF are nonlinear and the feasible region in ACOPF is usually non-convex [44, 24]. The coupled multiparty setting of our problem further motivates the need for third-party shared computing. In this setting each party will mask their own system models and parameters, and the third party will combine and solve the overall system model. To do this, we develop new methods to simultaneously accommodate the need for privacy of sensitive information while jointly solving the ACOPF problem. A preliminary presentation of the masking technique is found in [78] which considered a single-party model. Here we examine the more important multi-party problem, and apply different algorithmic techniques to enhance computational performance. Then, we return to consider linear programming (LP) models for another masking idea: embedding a small LP problem in a larger one for hiding information, and illustrate the feasibility of this idea with a small example.

1.5 Problem Representation

A fundamental premise of this thesis is that mathematical representations of a problem matter. It is possible for seemingly different problems to exhibit the same solutions, and one particular form of a problem may be more desirable for findings solutions. While these assertions may be obvious, it is useful to demonstrate with a simple example.

Consider the equations

$$x^2 - y^2 = -3$$
$$y^2 = 4$$

with four solutions $x = \pm 1$, $y = \pm 2$.

A graphical solution is shown on the left hand side of Figure 1.1. The typical computational approach to find all the solutions is to choose different initial conditions and solve using a Newton-Raphson algorithm.

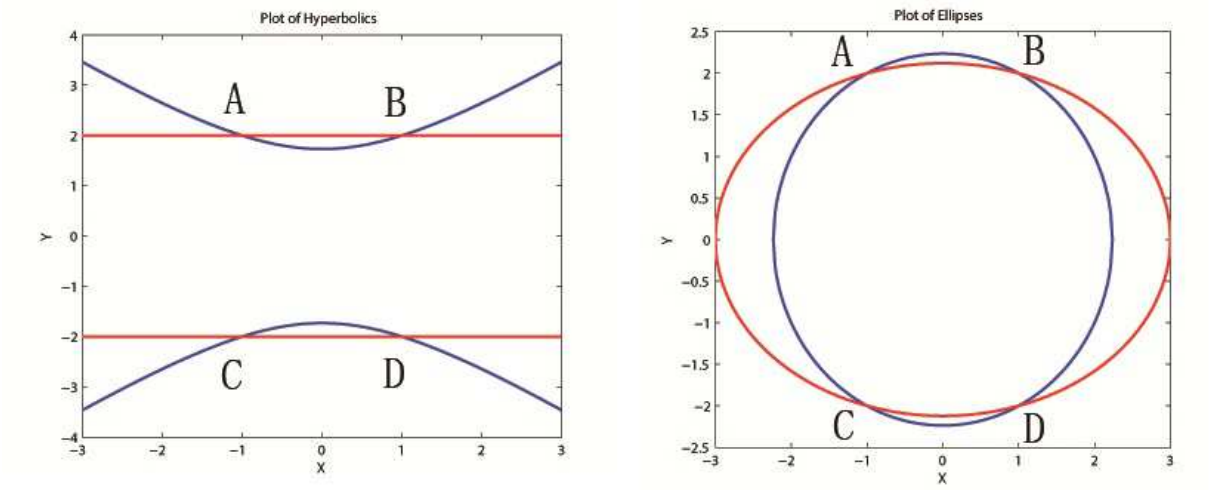


Figure 1.1: Plot of Hyperbolics (Left) and Ellipses (Right)

Alternatively, if we know one of the solutions, we can attempt to find additional solutions by tracing along one of the curves. For example, starting at point A, (-1,2), we can trace

$$(-1 + \Delta x)^2 - (2 + \Delta y)^2 = -3$$

. If we trace out values of Δy as a function of Δx, we may find solution B.

This tracing approach will only be guaranteed to find all solutions if they lie on connected curves, and if the curves are bounded. Neither of the conditions is satisfied in the form above.

Next, consider a linear combination of previous equations:

$$x^2 + y^2 = 5$$
$$x^2 + 2y^2 = 9$$

These equations form two intersecting ellipses shown on the right hand side of Figure 1.1, which will enable linking all the solutions on intersecting curves. The solution points at the intersections are the same for both representations.

This single example shows that different equations can admit the same solution set but may have different properties. In this case, all the real solutions are linked through intersecting ellipses. In this thesis we will show that the power flow equations and the OPF's first order conditions

intrinsically inherit ellipsoidal representations under mild conditions. We will explore this bounded structure to form intersecting curves for linking different real solutions.

A common phenomenon in problem solving is that the representation (description) of a problem can influence the complexity of solving it [79]. Such phenomenon has been recognized and investigated by the artificial intelligence and cognitive science scholars for a long time. They observed in reasoning systems that the alternative representations of a problem reveal different properties of it; each representation conceals some properties and highlights others [80, 81, 82]. The exact definition of problem representation differs from researchers [83, 84, 85, 86, 87], but one common intriguing feature is that as long as the solution to the problem retained, the representation of it can be altered for convenience. This methodological point of view is not uniquely beneficial for designing artificial intelligence, but can also be applied to many power engineering problems. For example, one can consider the power flow models in polar coordinates and in rectangular coordinates as two representations of the power flow problem. The polar representation may be superior to derive a DC approximation and have a better convergence performance [88], but the rectangular representation is more convenient to reveal the ellipsoidal structure [89] and, in the context of optimal power flow, more straightforward to derive the convex relaxations [45]. More interestingly, the ellipsoidal formulation of power flow equations is another alternative representation of the power flow problem in rectangular coordinates. This representation will help us eliminate unboundedness of the original power flow equations and enable a continuation method for searching power flow solutions more reliably in Chapter 4 Section 4.3.

The success of ellipsoidal representation for the power flow problem further inspires us to apply alternative representations to other problems in power engineering. A similar ellipsoidal formulation is constructed for the first order conditions of the traditional optimal power flow (OPF) model, and induces the first deterministic algorithm to efficiently identify multiple local solutions (and hopefully the global optimality) to the OPF problems in Chapter 4 Section 4.4. It will be highlighted that this algorithm can bridge disconnected feasible regions for searching local solutions.

1.6 Organization

The thesis is organized as follows. Chapter 2 introduces all the physical models and their modifications to accommodate specific forms. Chapter 3 describes all the necessary mathematical foundations. Chapter 4 firstly introduces the branch tracing method, then applies this method to both the power flow problem and the optimal power flow problem. Chapter 5 applies the encryption mapping to the multi-party ACOPF models to obfuscate the sensitive information for each party while solving them jointly. Chapter 6 concludes the thesis with contributions, open questions and proposed work.

1.7 Contributions

This thesis provides several theoretical and practical contributions that are worth of highlighting in the front.

Theoretically, we find an appropriate mathematical language to describe a certain type of problem representations, and apply them to three different topics in power engineering. Specifically, we rigorously define the concept of algebraic set preserving mapping for solution preserving representations (transformations); show that the algebraic set preserving mappings are closed under composition; show that two particular mappings, namely the induced affine mappings and the linear mappings, are algebraic set preserving mappings; prove (constructively) that the power flow problem of a power system without shunt elements can always be cast in the ellipsoidal formulation; derive a sufficient condition for the existence of the ellipsoidal formulation of the power flow problem of a power system with shunt elements; prove (constructively) that the first order conditions of the traditional ACOPF problem can always be cast in the ellipsoidal formulation; extend the ellipsoidal formulation to the first order conditions of arbitrary bounded quadratic constrained quadratic programming problems; prove that the set of ellipsoidal mappings is convex, and illustrate an equivalence between 1-dimensional curves defined by ellipsoidal representations and the original form.

Practically, we applied the branch tracing method to the ellipsoidal formulation of power flow equations, and found the complete solution sets for all the test cases which had been completely solved before. We further applied our proposed method for several more benchmark cases to obtain their real solution sets. Then, we extended our proposed method to ACOPF problems. It is the first principled deterministic algorithm to our best knowledge that can locate multiple local solutions for ACOPF problems. We successfully located all the local minima, including the global solutions, for a few test cases. Moreover, we identified four additional local minima to the three known ones for a 39-bus system, and identified the global solution which does not satisfy the KKT conditions for another particular test case with equality constrained model. Finally, we proposed an encryption mapping for the multi-party ACOPF problems for cloud computing which does not require decryption before use, and further discussed an embedding technique for masking linear programming problems.

Chapter 2

Algebraic Descriptions of Electric Power Grids

2.1 Overview

The detailed background for all the physical models treated in this thesis is presented in this chapter. This includes the AC power flow model in polar coordinates, the DC power flow model, the AC power flow model in rectangular coordinates, and a multi-party ACOPF model.

2.2 Power Flow Problem

The power flow problem describes the injected power at each node in terms of node voltages. For the basic power flow problem there are three types of nodes, called buses. A node with fixed complex power injection is called the PQ bus.

In practice, some nodes can adjust their reactive power injections to maintain the voltage magnitudes. This is typical for those nodes connected to generators or voltage regulators. Then the reactive power balancing equations on these nodes are replaced by the voltage magnitude equations. This kind of node is known as a PV bus.

A system angle reference needs to be assigned to some node. The reference node is called the slack bus, and is typically assigned to a generator node whose active power injection can be easily adjusted. Then the active power balancing equation on the slack bus is replaced by the angle reference equation, and the voltage magnitude is fixed.

In summary, the power flow problem is to solve the bus voltages that satisfy all the equations on PV buses, PQ buses and the slack bus.

2.2.1 AC Power Flow Model in Polar Coordinates

Given a connected power system with N_{bus} buses and N_{gen} generators. Without loss of generality, let's suppose that the 1-st bus is the slack bus, the 2-nd bus to the N_{gen}-th bus are the PV buses, and the rest are the PQ buses. By Ohm's law and the Kirchhoff's current law, the node current injections are related to bus voltage by

$$\mathbf{I} = \mathbf{Y}_{bus}\mathbf{V} \tag{2.1}$$

where $\mathbf{I} \in \mathbb{C}^{N_{bus}}$ is the node complex current injection vector; $\mathbf{Y}_{bus} \in \mathbb{C}^{N_{bus} \times N_{bus}}$ is the bus admittance matrix of which the real part is the bus conductance matrix $\mathbf{G}_{bus}$ and the imaginary part is the bus susceptance matrix $\mathbf{B}_{bus}$; $\mathbf{V} \in \mathbb{C}^{N_{bus}}$ is the node complex voltage vector. We omit the derivation of the complex voltages from the sinusoidal waves, as well as the construction of the bus admittance matrix from the network transmission line models. Interested readers can refer to any power system textbook (such as [90]) for details.

The complex power injection is given by

$$\mathbf{S} = \mathbf{V} \odot conj(\mathbf{I}) \tag{2.2}$$

where $\mathbf{S} \in \mathbb{C}^{N_{bus}}$ is the node complex power injection vector; $\odot$ is the Hadamard product, and $conj(\cdot)$ is the conjugate operator.

Combining Equation (2.1) and Equation (2.2) and then seperating the real part and the imaginary part of $\mathbf{S}$ we obtain the power flow equations (polar coordinates):

$$P_{gen,k} - P_{load,k} = V_k \sum_{n=1}^{N_{bus}} V_n \Big(G_{n,k} cos(\delta_k - \delta_n) + B_{n,k} sin(\delta_k - \delta_n) \Big) \tag{2.3a}$$

$$Q_{gen,k} - Q_{load,k} = V_k \sum_{n=1}^{N_{bus}} V_n \Big(G_{n,k} sin(\delta_k - \delta_n) - B_{n,k} cos(\delta_k - \delta_n) \Big) \tag{2.3b}$$

where $k = 1, \ldots, N_{bus}$; $P_{gen,k}$ and $P_{load,k}$ are the active power generation and the active power demand at the k-th bus; $Q_{gen,k}$ and $Q_{load,k}$ are the reactive power generation and the reactive power demand at the k-th bus; V_k is the voltage magnitude at the k-th bus; $G_{n,k}$ and $B_{n,k}$ are the (n,k)-th entry of $\mathbf{G}_{bus}$ and $\mathbf{B}_{bus}$; δ_k is the k-th bus voltage angle referred to the slack bus.

For the PV buses, namely $k = 2, \ldots, N_{gen}$, Equation (2.3b) is replaced by the voltage magnitude relation

$$V_k = V_{k,0} \tag{2.4}$$

where $V_{k,0}$ is fixed.

For the slack bus, namely $k = 1$, the active and the reactive power equations are replaced by

$$V_1 = V_{1,0} \tag{2.5a}$$

$$\delta_1 = 0 \tag{2.5b}$$

where $V_{1,0}$ is fixed.

Equation (2.3), (2.4), and (2.5) depict the entire AC power flow model in polar coordinates.

2.2.2 DC Power Flow Model

The power flow model in polar coordinates is usually used in practice because it highlights important $P - \delta$, $Q - V$ relations, and it enables some easy approximations for fast computing. The DC power flow model is one common simplification to the AC power flow model.

First, suppose that the electric power network is a high voltage transmission system. This assumption implies that the conductance is much smaller than the susceptance for the transmission lines, which further indicates that the entries of $\mathbf{G}_{bus}$ are much smaller than the entries of $\mathbf{B}_{bus}$. So we boldly approximate $\mathbf{G}_{bus}$ by a zero matrix.

Second, suppose that the power grid is lightly loaded. Empirically, the angle difference between two connected buses is usually small for light loading. Then, $sin(\delta_k - \delta_n)$ is approximated by $\delta_k - \delta_n$.

Finally, assume that each bus has enough reactive power sources to support the bus voltage magnitude. This assumption eliminates all the reactive power balancing equations, and usually sets every voltage magnitude to be 1 p.u.

With the above approximations, the simplified AC power flow model, known as the DC power flow, is described as

$$P_{gen,k} - P_{load,k} = \sum_{n=1}^{N_{bus}} B_{n,k}(\delta_k - \delta_n) \tag{2.6}$$

Note that Equation (2.6) is a linear equation with respect to the bus voltage angles δ_k for $k = 2, \ldots, N_{bus}$, therefore can be solved easily.

2.2.3 AC Power Flow Model in Rectangular Coordinates

To investigate the geometric features of the AC power flow model, it is convenient and insightful to reformulate it in rectangular coordinates. For the rest of the thesis, rectangular coordinates are used unless specifically stated.

Let

$$\mathbf{V} = \mathbf{V}_d + j\mathbf{V}_q \tag{2.7}$$

where $\mathbf{V}_d \in \mathbb{R}^{N_{bus}}$ and $\mathbf{V}_q \in \mathbb{R}^{N_{bus}}$ are the real part and the imaginary part of $\mathbf{V}$; $j = \sqrt{-1}$.

Consider the k-th entry of Equation (2.2)

$$S_k = (V_{d,k} + jV_{q,k})\, conj\Big[(\mathbf{G}_{k\cdot} + j\mathbf{B}_{k\cdot})(\mathbf{V}_d + j\mathbf{V}_q)\Big] \tag{2.8}$$

where S_k is the k-th entry of $\mathbf{S}$; $V_{d,k}$ and $V_{q,k}$ are the k-th entries of $\mathbf{V}_d$ and $\mathbf{V}_q$ respectively; $\mathbf{G}_{k\cdot}$ and $\mathbf{B}_{k\cdot}$ are the k-th rows of $\mathbf{G}_{bus}$ and $\mathbf{B}_{bus}$ respectively.

Separating the real part and the imaginary part of Equation (2.8), we have

$$\begin{aligned} P_{gen,k} - P_{load,k} = V_{d,k} \textstyle\sum_{n=1}^{N_{bus}} \Big(G_{n,k}V_{d,n} - B_{n,k}V_{q,n}\Big) + \\ V_{q,k} \textstyle\sum_{n=1}^{N_{bus}} \Big(G_{n,k}V_{q,n} + B_{n,k}V_{d,n}\Big) \end{aligned} \tag{2.9a}$$

$$\begin{aligned} Q_{gen,k} - Q_{load,k} = V_{q,k} \textstyle\sum_{n=1}^{N_{bus}} \Big(G_{n,k}V_{d,n} - B_{n,k}V_{q,n}\Big) - \\ V_{d,k} \textstyle\sum_{n=1}^{N_{bus}} \Big(G_{n,k}V_{q,n} + B_{n,k}V_{d,n}\Big) \end{aligned} \tag{2.9b}$$

To simplify, define

$$O(\cdot) := \frac{1}{2}\begin{bmatrix} \mathfrak{real}(\cdot) & -\mathfrak{imag}(\cdot) \\ \mathfrak{imag}(\cdot) & \mathfrak{real}(\cdot) \end{bmatrix} \tag{2.10a}$$

$$\mathbf{U} := \begin{bmatrix} \mathbf{V}_d \\ \mathbf{V}_q \end{bmatrix} \tag{2.10b}$$

$$\mathbf{M}_{v,k} := diag\left[\mathbf{e}_k^T \ \mathbf{e}_k^T\right] \tag{2.10c}$$

$$\mathbf{M}_{p,k} := \mathbf{M}_{v,k}O(\mathbf{Y}_{bus}) + O(\mathbf{Y}_{bus})^T\mathbf{M}_{v,k} \tag{2.10d}$$

$$\mathbf{M}_{q,k} := \mathbf{M}_{v,k}O(\mathbf{j}\mathbf{Y}_{bus}) + O(\mathbf{j}\mathbf{Y}_{bus})^T\mathbf{M}_{v,k} \tag{2.10e}$$

where $\mathbf{e}_k \in \mathbb{R}^{N_{bus}}$ is a column vector with a single unity element at the k-th entry; $diag[\cdot]$ is the diagonalization operator that add a vector to the diagonal of a corresponding zero matrix.

Then Equation (2.9) can be reformulated as

$$P_{gen,k} - P_{load,k} = \mathbf{U}^T\mathbf{M}_{p,k}\mathbf{U} \tag{2.11a}$$

$$Q_{gen,k} - Q_{load,k} = \mathbf{U}^T\mathbf{M}_{q,k}\mathbf{U} \tag{2.11b}$$

For the PV buses, namely $k = 2, \ldots, N_{gen}$, Equation (2.11b) is replaced by

$$\mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} = V_{k,0}^2 \tag{2.12}$$

where $V_{k,0}$ is fixed.

Finally, the slack bus equations are

$$\mathbf{U}^T\mathbf{M}_{v,1}\mathbf{U} = V_{1,0}^2 \tag{2.13a}$$

$$V_{q,1} = 0 \tag{2.13b}$$

where $V_{1,0}$ is fixed. In practice, since $V_{q,1} = 0$, we remove this variable and its associated columns and rows in $\mathbf{M}_{v,k}$, $\mathbf{M}_{p,k}$, and $\mathbf{M}_{q,k}$.

In summary, the AC power flow model in rectangular coordinates can be succinctly expressed by Equation (2.11), (2.12) and (2.13).

2.2.4 The ZIP Load Flow Model

Consider the active power power demand $P_{load,k}$ and the reactive power demand $Q_{load,k}$ in (2.11), they are usually constant terms for simplicity. However, the realistic load can change by the variation of the load bus voltage magnitude. A commonly used load model is called the "ZIP" model, where "Z" represents a constant load impedance, "I" represents a constant current load at constant power factor, and "P" represents a constant load power. Specifically,

$$P_{load,k} = P_{0,k} + P_{1,k}|V_k| + P_{2,k}|V_k|^2 \tag{2.14a}$$

$$Q_{load,k} = Q_{0,k} + Q_{1,k}|V_k| + Q_{2,k}|V_k|^2 \tag{2.14b}$$

where $P_{0,k}$ and $Q_{0,k}$ are the constant power terms, $P_{1,k}$ and $Q_{1,k}$ are the constant current terms, $P_{2,k}$ and $Q_{2,k}$ are the constant impedance terms, and $|V_k|$ is the voltage magnitude at bus-k.

Therefore, (2.11) can be written as

$$P_{gen,k} - P_{0,k} = \mathbf{U}^T\mathbf{M}_{p,k}\mathbf{U} + P_{1,k}|V_k| + P_{2,k}|V_k|^2 \tag{2.15a}$$

$$Q_{gen,k} - Q_{0,k} = \mathbf{U}^T\mathbf{M}_{q,k}\mathbf{U} + Q_{1,k}|V_k| + Q_{2,k}|V_k|^2 \tag{2.15b}$$

$$|V_k|^2 = V_{d,k}^2 + V_{q,k}^2 \tag{2.15c}$$

2.3 Optimal Power Flow Problem

The optimal power flow (OPF) problem is described by optimizing an objective function over a set of physical conditions and engineering limits of the power grid. It can be interpreted as a slightly relaxed version of the power flow problem within certain engineering ranges to reach an optimal operating state. Thus, the power flow model, as the core physical relations of the grid, is included in the OPF problem.

Typically the engineering limits include the bus voltage magnitude upper and lower bounds, the active and reactive power generation upper and lower bounds, and line flow upper bounds. Some limits may serve as proxy limits. For example, the bus voltage angles may be bounded for

dynamical concerns. Other kinds of constraints, continuous or discrete, can also be included in the OPF problem for different purposes. For generality and simplicity, this thesis only considers the most common engineering limits.

2.3.1 AC Optimal Power Flow Model

Again, let's consider a connected power system with N_{bus} many buses, N_{gen} many generators and N_{line} many transmission lines. Suppose that the 1-st bus is the slack bus, the 2-nd bus to the N_{gen}-th bus are the PV buses, and the rest are the PQ buses. Let the objective function to be a quadratic cost with respect to the active power generation. Then the OPF model is

Minimize $$\sum_{i=1}^{N_{gen}} d_i P_{gen,i}^2 + c_i P_{gen,i}$$

Subject to:

Generation Balance:
$$P_{gen,i} - \mathbf{U}^T \mathbf{M}_{p,i} \mathbf{U} = P_{load,i} \tag{2.16a}$$
$$Q_{gen,i} - \mathbf{U}^T \mathbf{M}_{q,i} \mathbf{U} = Q_{load,i} \tag{2.16b}$$

Load Balance:
$$\mathbf{U}^T \mathbf{M}_{p,m} \mathbf{U} = -P_{load,m} \tag{2.16c}$$
$$\mathbf{U}^T \mathbf{M}_{q,m} \mathbf{U} = -Q_{load,m} \tag{2.16d}$$

Voltage Limit:
$$\mathbf{U}^T \mathbf{M}_{v,k} \mathbf{U} \leq V_{max,k}^2 \tag{2.16e}$$
$$-\mathbf{U}^T \mathbf{M}_{v,k} \mathbf{U} \leq -V_{min,k}^2$$

Angle Reference:
$$V_{q,1} = 0 \tag{2.16f}$$

Generation Limit:
$$P_{gen,i} \leq P_{max,i} \tag{2.16g}$$
$$-P_{gen,i} \leq -P_{min,i}$$
$$Q_{gen,i} \leq Q_{max,i} \tag{2.16h}$$
$$-Q_{gen,i} \leq -Q_{min,i}$$

Line Current Limit:
$$|I_{in,n}|^2 \leq I_{max,n}^2 \tag{2.16i}$$
$$|I_{out,n}|^2 \leq I_{max,n}^2 \tag{2.16j}$$

Index:
$$i = 1, \ldots, N_{gen}$$

$$m = N_{gen} + 1, \ldots, N_{bus}$$
$$k = 1, \ldots, N_{bus}$$
$$n = 1, \ldots, N_{line}$$

where d_i and c_i are constants; $V_{min,k}$ and $V_{max,k}$ are the lower bound and the upper bound of the voltage magnitude for bus-k; $P_{min,i}$ and $P_{max,i}$ are the lower bound and the upper bound of the active power generation for bus-i; $Q_{min,i}$ and $Q_{max,i}$ are the lower bound and the upper bound of the reactive power generation for bus-i; $I_{in,n}$ and $I_{out,n}$ are the complex currents entering and leaving the n-th transmission line; $I_{max,n}$ is the current magnitude upper bound for the n-th transmission line.

First consider transmission line current limit model, define

$$\mathbf{M}_{I,n} := diag\left[\mathbf{t}_n^T\ \mathbf{t}_n^T\right] \tag{2.17a}$$
$$\mathbf{M}_{Iin,n} := O(\mathbf{Y}_f)^T \mathbf{M}_{I,n} O(\mathbf{Y}_f) \tag{2.17b}$$
$$\mathbf{M}_{Iout,n} := O(\mathbf{Y}_t)^T \mathbf{M}_{I,n} O(\mathbf{Y}_t) \tag{2.17c}$$
$$|I_{in,n}|^2 = \mathbf{U}^T \mathbf{M}_{Iin,n} \mathbf{U} \tag{2.17d}$$
$$|I_{out,n}|^2 = \mathbf{U}^T \mathbf{M}_{Iout,n} \mathbf{U} \tag{2.17e}$$

where $\mathbf{t}_n \in \mathbb{R}^{N_{line}}$ is a column vector with a single unity element at the n-th entry, $\mathbf{Y}_f$ and $\mathbf{Y}_t$ are the current entering and leaving admittance matrices such that $[I_{in,n}]_{n \in N_{line}} = \mathbf{Y}_f \mathbf{U}$ and $[I_{out,n}]_{n \in N_{line}} = \mathbf{Y}_t \mathbf{U}$, $\mathbf{M}_{Iin,n},\ \mathbf{M}_{Iout,n} \in \mathbb{R}^{N_{bus} \times N_{bus}}$ are constant matrices.

Substitute Equation (2.17), (2.16f) and (2.16b) into Equation (2.16) we have

Minimize $$\sum_{i=1}^{N_{gen}} d_i P_{gen,i}^2 + c_i P_{gen,i}$$

Subject to:

Generation Balance: $$P_{gen,i} - \mathbf{U}^T \mathbf{M}_{p,i} \mathbf{U} = P_{load,i} \tag{2.18a}$$

Load Balance: $$\mathbf{U}^T \mathbf{M}_{p,m} \mathbf{U} = -P_{load,m} \tag{2.18b}$$
$$\mathbf{U}^T \mathbf{M}_{q,m} \mathbf{U} = -Q_{load,m} \tag{2.18c}$$

$$\begin{aligned}
&\text{Voltage Limit:} && \mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} \le V_{max,k}^2 && (2.18d)\\
& && -\mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} \le -V_{min,k}^2 &&\\
&\text{Generation Limit:} && P_{gen,i} \le P_{max,i} && (2.18e)\\
& && -P_{gen,i} \le -P_{min,i} &&\\
& && \mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} \le Q_{max,i} && (2.18f)\\
& && -\mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} \le -Q_{min,i} &&\\
&\text{Line Current Limit:} && \mathbf{U}^T\mathbf{M}_{Iin,n}\mathbf{U} \le I_{max,n}^2 && (2.18g)\\
& && \mathbf{U}^T\mathbf{M}_{Iout,n}\mathbf{U} \le I_{max,n}^2 && (2.18h)\\
&\text{Index:} && i = 1,\ldots,N_{gen} &&\\
& && m = N_{gen}+1,\ldots,N_{bus} &&\\
& && k = 1,\ldots,N_{bus} &&\\
& && n = 1,\ldots,N_{line} &&
\end{aligned}$$

where the unknowns in Equation (2.18) are $P_{gen,i}$ and $\mathbf{U}$. They appear only in linear and quadratic forms.

One can also consider a transmission line apparent power limit model. Define connection matrices $\mathbf{C}_f$ and $\mathbf{C}_t \in \mathbb{R}^{N_{line}\times N_{bus}}$ for lines and buses, respectively. The (n,k)-th entry of $\mathbf{C}_f$ is 1 if the n-th line is connected to the k-th "from" bus and the (n,r)-th entry of $\mathbf{C}_t$ is 1 if the n-th line is connected to the r-th "to" bus.

Further, define

$$\mathbf{T}_{f,n} := \mathbf{C}_f^T(n,:)\ \mathrm{conj}(\mathbf{Y}_f(n,:)) \quad (2.19a)$$

$$\mathbf{T}_{t,n} := \mathbf{C}_t^T(n,:)\ \mathrm{conj}(\mathbf{Y}_t(n,:)) \quad (2.19b)$$

$$\mathbf{M}_{Pline,f,n} := O(\mathbf{T}_{f,n} + \mathbf{T}_{f,n}^T) \quad (2.19c)$$

$$\mathbf{M}_{Qline,f,n} := O(j\mathbf{T}_{f,n} + j\mathbf{T}_{f,n}^T) \quad (2.19d)$$

$$\mathbf{M}_{Pline,t,n} := O(\mathbf{T}_{t,n} + \mathbf{T}_{t,n}^T) \quad (2.19e)$$

$$\mathbf{M}_{Qline,t,n} := O(j\mathbf{T}_{t,n} + j\mathbf{T}_{t,n}^T) \quad (2.19f)$$

where $\mathbf{C}_f(n,:)$ is the n-th row of $\mathbf{C}_f$, $\mathbf{Y}_f(n,:)$ is the n-th row of $\mathbf{Y}_f$, and $\mathrm{conj}\,(\cdot)$ is the complex conjugate.

Limits on apparent power flows are formulated as

$$P_{line,f,n} - \mathbf{U}^T \mathbf{M}_{Pline,f,n} \mathbf{U} = 0 \tag{2.20a}$$

$$Q_{line,f,n} - \mathbf{U}^T \mathbf{M}_{Qline,f,n} \mathbf{U} = 0 \tag{2.20b}$$

$$P^2_{line,f,n} + Q^2_{line,f,n} \leq S^2_{max,n} \tag{2.20c}$$

$$P_{line,t,n} - \mathbf{U}^T \mathbf{M}_{Pline,t,n} \mathbf{U} = 0 \tag{2.20d}$$

$$Q_{line,t,n} - \mathbf{U}^T \mathbf{M}_{Qline,t,n} \mathbf{U} = 0 \tag{2.20e}$$

$$P^2_{line,t,n} + Q^2_{line,t,n} \leq S^2_{max,n} \tag{2.20f}$$

where P denotes active power; Q denotes reactive power; S denotes apparent power; subscripts "f" and "t" denote the "from" and "to" terminals, respectively; and subscript "n" indicates the line index.

One can replace (2.18g) and (2.18h) with (2.20) to formulate the ACOPF problem with flow limits in terms of apparent power, adding $4 \times N_{line}$ constraints to the problem. Throughout this thesis, we apply the line current model unless specifically remarked. There are several reasons that we favor the line current model.

1. The line current model has $4 \times N_{line}$ less constraints to the problem.

2. The line current limits represent the transmission line thermal limits, which are more meaningful in practice than apparent power limits.

3. The line current equations are purely quadratic with respect to the voltage variables $\mathbf{U}$.

2.3.2 Equality Constrained AC Optimal Power Flow Model

The challenge of applying a linear mapping in Chapter 3 to the first order conditions of the ACOPF model in Equation (2.18) is that the first order conditions include inequality relations. A linear combination of inequalities can change their semi-algebraic set (intersection set), thus

it is not an algebraic set preserving mapping. Hence, it is necessary to convert inequalities into equalities. We achieve this by introducing free slack variables.

Consider the voltage limit inequalities in Equation (2.18d). They are reformulated as

$$\mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} + s^2_{v,max,k} = V^2_{max,k} \tag{2.21a}$$

$$s^2_{v,min,k} - \mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} = -V^2_{min,k} \tag{2.21b}$$

where $s_{v,max,k}$ and $s_{v,min,k}$ are the free slack variables.

The line current inequalities in Equation (2.18g) are reformulated as

$$\mathbf{U}^T\mathbf{M}_{Iin,n}\mathbf{U} + s^2_{I,max,in,n} = I^2_{max,in} \tag{2.22a}$$

$$\mathbf{U}^T\mathbf{M}_{Iout,n}\mathbf{U} + s^2_{I,max,out,n} = I^2_{max,out} \tag{2.22b}$$

where $s_{I,max,in,n}$ and $s_{I,max,out,n}$ are the free slack variables.

The reactive power inequalities in Equation (2.18f) are reformulated as

$$\mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} + s^2_{q,max,i} = Q_{max,i} \tag{2.23a}$$

$$s^2_{q,min,i} - \mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} = -Q_{min,i} \tag{2.23b}$$

where $s_{q,max,i}$ and $s_{q,min,i}$ are the free slack variables.

The active power inequalities in Equation (2.18e) are considered differently. Firstly, we assume that the active power generation is non-negative. This is usually the situation for realistic power plants since they are designed to deliver active power. If the cost function is linear, namely $d_i = 0$, we treat the active power generation limits as below.

$$p^2_{gen,i} + s^2_{p,max,i} = P_{max,i} \tag{2.24a}$$

$$s^2_{p,min,i} - p^2_{gen,i} = -P_{min,i} \tag{2.24b}$$

where $p_{gen,i} := \sqrt{P_{gen,i}}$; $s_{p,max,i}$ and $s_{p,min,i}$ are the free slack variables.

If the objective function is quadratic, the active power generation limits are reformulated as

$$P^2_{gen,i} + s^2_{p,max,i} = P^2_{max,i} \tag{2.25a}$$

$$s^2_{p,min,i} - P^2_{i,gen} = -P^2_{min,i} \tag{2.25b}$$

where $s_{p,max,i}$ and $s_{p,min,i}$ are the free slack variables.

If the active power lower bound is allowed to be negative, we can shift the active power variable by its negative lower bound. Then the shifted active power retains non-negativity. Consider (2.18e) with $P_{min,i} < 0$. This is equivalent to

$$0 \leq P_{gen,i} - P_{min,i} \leq P_{max,i} - P_{min,i} \tag{2.26}$$

If the cost function is linear, then we let $p^2_{gen,i} := P_{gen,i} - P_{min,i}$. Hence, (2.18a) can be written as

$$p^2_{gen,i} - \mathbf{U}^T \mathbf{M}_{P,i} \mathbf{U} = \hat{P}_{load,i} \tag{2.27}$$

where $\hat{P}_{load,i} = P_{load,i} - P_{min,i}$ is a constant.

The linear cost function can be expressed as

$$J = \sum_{i=1}^{N_{gen}} c_i p^2_{gen,i} + C \tag{2.28}$$

where $C = \sum_{i=1}^{N_{gen}} c_i P_{min,i}$ is a constant.

If the cost function is quadratic, we let $\hat{P}_{gen,i} := P_{gen,i} - P_{min,i}$ and $p^2_{gen,i} := \hat{P}_{gen,i}$. Thus, the cost function is

$$J = \sum_{i=1}^{N_{gen}} d_i P^2_{gen,i} + c_i P_{gen,i} \tag{2.29a}$$

$$= \sum_{i=1}^{N_{gen}} d_i (\hat{P}_{gen,i} + P_{min,i})^2 + c_i (\hat{P}_{gen,i} + P_{min,i}) \tag{2.29b}$$

$$= \sum_{i=1}^{N_{gen}} d_i \hat{P}^2_{gen,i} + e_i \hat{P}_{gen,i} + a_i \tag{2.29c}$$

$$= \sum_{i=1}^{N_{gen}} d_i \hat{P}^2_{gen,i} + e_i p^2_{gen,i} + a_i \tag{2.29d}$$

where $e_i = 2 d_i P_{min,i} + c_i$ and $a_i = d_i P^2_{min,i} + c_i P_{min,i}$ are constants.

Therefore, the equality constrained ACOPF model with a linear objective function is described by

$$\textbf{Minimize} \qquad \sum_{i=1}^{N_{gen}} c_i(\mathbf{U}^T\mathbf{M}_{p,i}\mathbf{U} + P_{load,i}) + C$$

Subject to:

Load Balance:
$$\mathbf{U}^T\mathbf{M}_{p,m}\mathbf{U} = -P_{load,m} \tag{2.30a}$$
$$\mathbf{U}^T\mathbf{M}_{q,m}\mathbf{U} = -Q_{load,m} \tag{2.30b}$$

Volt. Limit:
$$\mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} + s^2_{v,max,k} = V^2_{max,k} \tag{2.30c}$$
$$s^2_{v,min,k} - \mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} = -V^2_{min,k} \tag{2.30d}$$

Gen. Limit:
$$(\mathbf{U}^T\mathbf{M}_{p,i}\mathbf{U} + P_{load,i}) + s^2_{p,max,i} = P_{max,i} \tag{2.30e}$$
$$s^2_{p,min,i} - (\mathbf{U}^T\mathbf{M}_{p,i}\mathbf{U} + P_{load,i}) = -P_{min,i} \tag{2.30f}$$
$$\mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} + s^2_{q,max,i} = Q_{max,i} \tag{2.30g}$$
$$s^2_{q,min,i} - \mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} = -Q_{min,i} \tag{2.30h}$$

Curr. Limit:
$$\mathbf{U}^T\mathbf{M}_{Iin,n}\mathbf{U} + s^2_{I,max,in,n} = I^2_{max,in} \tag{2.30i}$$
$$\mathbf{U}^T\mathbf{M}_{Iout,n}\mathbf{U} + s^2_{I,max,out,n} = I^2_{max,out} \tag{2.30j}$$

Index:
$$i = 1, \ldots, N_{gen}$$
$$m = N_{gen} + 1, \ldots, N_{bus}$$
$$k = 1, \ldots, N_{bus}$$
$$n = 1, \ldots, N_{line}$$

The equality constrained ACOPF model with a quadratic objective function is slightly different than Equation (2.30), and is summarized below.

$$\textbf{Minimize} \qquad \sum_{i=1}^{N_{gen}} d_i P^2_{gen,i} + c_i p^2_{gen,i} + a_i$$

Subject to:

Gen. Balance:
$$p^2_{gen,i} - \mathbf{U}^T\mathbf{M}_{p,i}\mathbf{U} = P_{load,i} \tag{2.31a}$$

Load Balance: $$\mathbf{U}^T\mathbf{M}_{p,m}\mathbf{U} = -P_{load,m} \quad (2.31b)$$

$$\mathbf{U}^T\mathbf{M}_{q,m}\mathbf{U} = -Q_{load,m} \quad (2.31c)$$

Volt. Limit: $$\mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} + s^2_{v,max,k} = V^2_{max,k} \quad (2.31d)$$

$$s^2_{v,min,k} - \mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} = -V^2_{min,k} \quad (2.31e)$$

Gen. Limit: $$P^2_{gen,i} + s^2_{p,max,i} = P^2_{max,i} \quad (2.31f)$$

$$s^2_{p,min,i} - P^2_{gen,i} = -P^2_{min,i} \quad (2.31g)$$

$$\mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} + s^2_{q,max,i} = Q_{max,i} \quad (2.31h)$$

$$s^2_{q,min,i} - \mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} = -Q_{min,i} \quad (2.31i)$$

Curr. Limit: $$\mathbf{U}^T\mathbf{M}_{Iin,n}\mathbf{U} + s^2_{I,max,in,n} = I^2_{max,in} \quad (2.31j)$$

$$\mathbf{U}^T\mathbf{M}_{Iout,n}\mathbf{U} + s^2_{I,max,out,n} = I^2_{max,out} \quad (2.31k)$$

New Relation: $$p^2_{gen,i} - P_{gen,i} = 0 \quad (2.31l)$$

Index: $$i = 1, \ldots, N_{gen}$$

$$m = N_{gen} + 1, \ldots, N_{bus}$$

$$k = 1, \ldots, N_{bus}$$

$$n = 1, \ldots, N_{line}$$

2.3.3 Multi-Party AC Optimal Power Flow Model

The modern power grid is operated by Independent System Operators (ISO) that manage transmission-coupled supply and distribution companies. Each ISO supervises a part of the continental power grid while coordinating with others. The newly emerged smart grid trends will include Distribution System Operators (DSO) whose duty is to coordinate the operation of the micro-grids in sub-areas, and interface with a larger area. The individual grid participants value their privacy in the multi-party scenario.

Suppose a power grid is operated by several independent parties. Each party is only in charge of a sub-area. Some sub-areas are interconnected by a few transmission lines so that the entire

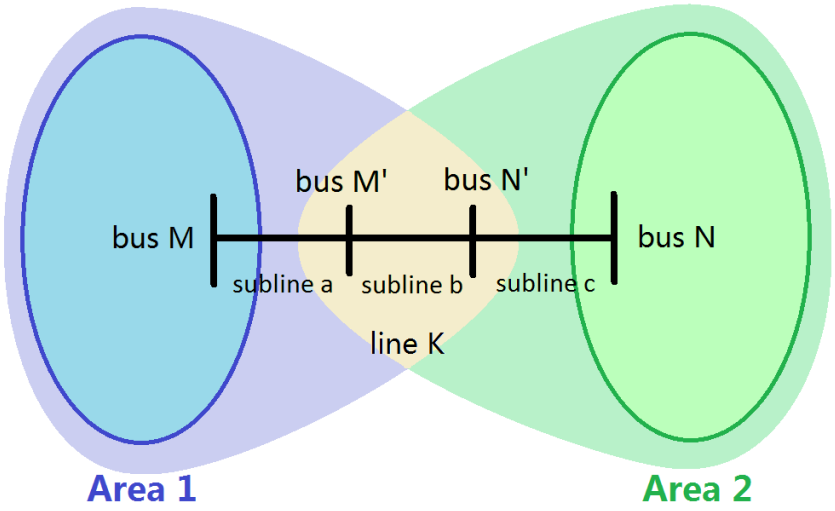


Figure 2.1: 30-Bus System with Two Parties

grid is connected. To interface with the entire area, each party has to request the information of the voltages on its adjacent buses from other parties, and needs to share the information of the voltages on the buses that are adjacent to other parties.

To reduce the need of exchanging information between connected sub-areas, we introduce two fictitious buses on each transmission line that connects two different sub-areas. For instance, Figure 2.1 depicts a transmission line between **Area-1** and **Area-2**. We introduce two fictitious buses M' and N' on the shared transmission line K and divide it into three sub-lines a, b and c. The new **Area-1** includes bus M' and N' with sub-line a and b, while the new **Area-2** includes buses N' and M' with sub-line c and b. Although these fictitious buses and sub-lines are not needed to model multi-party ACOPF, it is convenient for modeling purpose to include these fictitious buses to eliminate sharing of any direct information between the two areas. Therefore, the compliance between **Area-1** and **Area-2** is that the voltages in the intersection area, namely bus M', bus N' and sub-line b, should be in consistent for both area's models.

Suppose a power grid is divided into N_{area} many areas. Let $SB_{\alpha,\beta}$ denote the set of shared (fictitious) buses between **Area-**α and **Area-**β. Let the voltage vector of **Area-**α be $\mathbf{U}_\alpha$. Define a sharing matrix $\mathbf{e}^T_{\alpha,\beta}$ of dimension $|SB_{\alpha,\beta}| \times |\mathbf{U}_\alpha|$ containing 0's and 1's. Each row of $\mathbf{e}^T_{\alpha,\beta}$ has

exactly two 1's corresponding to a voltage's real part and imaginary part on one of the shared fictitious buses between **Area-**α and **Area-**β. Then the multi-party ACOPF model is described by

$$\textbf{Minimize} \qquad \sum_{\alpha=1}^{N_{area}} \sum_{i=1}^{N_{gen}} d_{\alpha,i} P_{gen,\alpha,i}^2 + c_{\alpha,i} P_{gen,\alpha,i}$$

Subject to:

$$\text{Gen. Balance:} \qquad P_{gen,\alpha,i} - \mathbf{U}_\alpha^T \mathbf{M}_{p,\alpha,i} \mathbf{U}_\alpha = P_{load,\alpha,i} \tag{2.32a}$$

$$\text{Load Balance:} \qquad \mathbf{U}_\alpha^T \mathbf{M}_{p,\alpha,m} \mathbf{U}_\alpha = -P_{load,\alpha,m} \tag{2.32b}$$

$$\mathbf{U}_\alpha^T \mathbf{M}_{q,\alpha,m} \mathbf{U}_\alpha = -Q_{load,\alpha,m} \tag{2.32c}$$

$$\text{Volt. Limit:} \qquad V_{min,\alpha,k}^2 \leq \mathbf{U}_\alpha^T \mathbf{M}_{v,\alpha,k} \mathbf{U}_\alpha \leq V_{max,\alpha,k}^2 \tag{2.32d}$$

$$\text{Gen. Limit:} \qquad P_{min,\alpha,i} \leq P_{gen,\alpha,i} \leq P_{max,\alpha,i} \tag{2.32e}$$

$$Q_{min,\alpha,i} \leq \mathbf{U}_\alpha^T \mathbf{M}_{q,\alpha,i} \mathbf{U}_\alpha \leq Q_{max,\alpha,i} \tag{2.32f}$$

$$\text{Curr. Limit:} \qquad \mathbf{U}_\alpha^T \mathbf{M}_{Iin,\alpha,n} \mathbf{U}_\alpha \leq I_{max,\alpha,n}^2 \tag{2.32g}$$

$$\mathbf{U}_\alpha^T \mathbf{M}_{Iout,\alpha,n} \mathbf{U}_\alpha \leq I_{max,\alpha,n}^2 \tag{2.32h}$$

$$\text{Compliance:} \qquad \mathbf{e}_{\alpha,\beta}^T \mathbf{U}_\alpha - \mathbf{e}_{\beta,\alpha}^T \mathbf{U}_\beta = \mathbf{0} \tag{2.32i}$$

$$\text{Index:} \qquad i = 1, \ldots, N_{gen}$$

$$m = N_{gen} + 1, \ldots, N_{bus}$$

$$k = 1, \ldots, N_{bus}$$

$$n = 1, \ldots, N_{line}$$

$$\alpha = 1, \ldots, N_{area}$$

where the subscript α indicates the α-th party's quantity.

Note that the multi-party ACOPF model of Equation (2.32) can also be cast in an equality constrained model:

$$\textbf{Minimize} \qquad \sum_{\alpha=1}^{N_{area}} \sum_{i=1}^{N_{gen}} d_{\alpha,i} P_{gen,\alpha,i}^2 + c_{\alpha,i} p_{gen,\alpha,i}^2$$

Subject to:

$$\text{Gen. Balance:} \quad p^2_{gen,\alpha,i} - \mathbf{U}^T_\alpha \mathbf{M}_{p,\alpha,i} \mathbf{U}_\alpha = P_{load,\alpha,i} \tag{2.33a}$$

$$\text{Load Balance:} \quad \mathbf{U}^T_\alpha \mathbf{M}_{p,\alpha,m} \mathbf{U}_\alpha = -P_{load,\alpha,m} \tag{2.33b}$$

$$\mathbf{U}^T_\alpha \mathbf{M}_{q,\alpha,m} \mathbf{U}_\alpha = -Q_{load,\alpha,m} \tag{2.33c}$$

$$\text{Volt. Limit:} \quad \mathbf{U}^T_\alpha \mathbf{M}_{v,\alpha,k} \mathbf{U}_\alpha + s^2_{v,max,\alpha,k} = V^2_{max,\alpha,k} \tag{2.33d}$$

$$s^2_{v,min,\alpha,k} - \mathbf{U}^T_\alpha \mathbf{M}_{v,\alpha,k} \mathbf{U}_\alpha = -V^2_{min,\alpha,k} \tag{2.33e}$$

$$\text{Gen. Limit:} \quad P^2_{gen,\alpha,i} + s^2_{p,max,\alpha,i} = P^2_{max,\alpha,i} \tag{2.33f}$$

$$s^2_{p,min,\alpha,i} - P^2_{gen,\alpha,i} = -P^2_{min,\alpha,i} \tag{2.33g}$$

$$\mathbf{U}^T_\alpha \mathbf{M}_{q,\alpha,i} \mathbf{U}_\alpha + s^2_{q,max,\alpha,i} = Q_{max,\alpha,i} \tag{2.33h}$$

$$s^2_{q,min,\alpha,i} - \mathbf{U}^T_\alpha \mathbf{M}_{q,\alpha,i} \mathbf{U}_\alpha = -Q_{min,\alpha,i} \tag{2.33i}$$

$$\text{Curr. Limit:} \quad \mathbf{U}^T_\alpha \mathbf{M}_{Iin,\alpha,n} \mathbf{U}_\alpha + s^2_{I,max,in,\alpha,n} = I^2_{max,in} \tag{2.33j}$$

$$\mathbf{U}^T_\alpha \mathbf{M}_{Iout,\alpha,n} \mathbf{U}_\alpha + s^2_{I,max,out,\alpha,n} = I^2_{max,out} \tag{2.33k}$$

$$\text{New Relation:} \quad p^2_{gen,\alpha,i} - P_{gen,\alpha,i} = 0 \tag{2.33l}$$

$$\text{Compliance:} \quad \mathbf{e}^T_{\alpha,\beta} \mathbf{U}_\alpha - \mathbf{e}^T_{\beta,\alpha} \mathbf{U}_\beta = \mathbf{0} \tag{2.33m}$$

$$\text{Index:} \quad i = 1, \ldots, N_{gen}$$

$$m = N_{gen} + 1, \ldots, N_{bus}$$

$$k = 1, \ldots, N_{bus}$$

$$n = 1, \ldots, N_{line}$$

2.4 Conclusion

This Chapter introduced the power flow models used in this thesis with emphasis on the power flow model in rectangular coordinates. This representation of the power flow model is described by quadratic polynomials. The optimal power flow model in rectangular coordinates was discussed, and was further reformulated as the equality constrained ACOPF model with free slack variables. Finally, we introduced the multi-party optimal power flow model. The power flow and the optimal

power flow models in rectangular coordinates will be used throughout this thesis, and the quadratic form will be specifically inspected in the next Chapter.

Chapter 3

Affine Algebraic Set Preserving Mappings

3.1 Overview

This Chapter establishes the theoretical foundations for this thesis. It first recalls some basic concepts and definitions in algebraic geometry. Then it elaborates the core idea of the thesis by defining the affine algebraic set preserving mapping, or simply set mapping. Specifically, the induced affine mapping and the linear mapping over polynomial rings are studied. They are shown to be useful in the construction of the ellipsoidal formulation of the power flow model in rectangular coordinates from Chapter 2, and is extended to the ellipsoidal formulation of the Fritz John conditions of the optimal power flow model. Finally, an encryption mapping is designed for the multi-party ACOPF model. Notations in Section 3.2 and 3.3 are distinct from the notations in the rest of the thesis, allowing common variables used in the mathematical literatures. For example, the variables $\mathbf{U}$ and $\mathbf{V}$ do not correspond to the voltage variables as in other parts of the thesis.

3.2 Background

Although an algebraically closed field is usually assumed in the context of algebraic geometry, this thesis only considers the real closed field denoted by $\mathbb{R}$.

Definition 3.2.1. *Let $\mathbb{R}$ be a real closed field. A real affine n-space over $\mathbb{R}$, denoted by $\mathbb{A}^n_{\mathbb{R}}$, or simply $\mathbb{A}^n$, is the set of all n-tuples of elements in $\mathbb{R}$. The element $\mathbf{a} \in \mathbb{A}^n$ is called a point. If $\mathbf{a} = (a_1, \ldots, a_n)$ for $a_i \in \mathbb{R}$, then a_i's are called the coordinates of $\mathbf{a}$.*

Definition 3.2.2. *Let $\mathbb{R}[x_1, \ldots, x_n]$ be the polynomial ring in n variables over $\mathbb{R}$. A subset $\mathbf{S} \subset \mathbb{A}^n$ is called a real algebraic set, or simply algebraic set throughout this thesis, if there exsits a subset $\mathbf{F} \subset \mathbb{R}[x_1, \ldots, x_n]$ such that*

$$\mathbf{S} = \Big\{ \mathbf{X} \in \mathbb{A}^n \mid f(\mathbf{X}) = 0,\ \forall\ f \in \mathbf{F} \Big\}.$$

Denote the algebraic set of a subset $\mathbf{F}$ in the polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$ by $\mathcal{Z}(\mathbf{F})$.

Lemma 3.2.1. *An algebraic set in $\mathbb{A}^n$ given by a finite subset $\mathbf{F}$ of the polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$ can always be defined by a single polynomial in $\mathbb{R}[x_1, \ldots, x_n]$.*

Proof. Consider $\mathbf{F} = \{f_i(\mathbf{X}) \in \mathbb{R}[x_1, \ldots, x_n], i \in \mathbf{I}\}$. The algebraic set of $\mathbf{F}$ is given by

$$\mathcal{Z}(\mathbf{F}) = \Big\{ \mathbf{X} \in \mathbb{A}^n \mid f_i(\mathbf{X}) = 0, i \in \mathbf{I} \Big\}$$

Let

$$\mathbf{Z}^* = \Big\{ \mathbf{X} \in \mathbb{A}^n \mid \sum_{i \in \mathbf{I}} f_i^2(\mathbf{X}) = 0 \Big\}$$

It is easy to show that for any $\mathbf{X}^* \in \mathcal{Z}(\mathbf{F})$, $\mathbf{X}^* \in \mathbf{Z}^*$, and vice versa. Therefore, $\mathcal{Z}(\mathbf{F}) = \mathcal{Z}(\sum_{i \in \mathbf{I}} f_i^2(\mathbf{X}))$. □

Theorem 3.2.1. *Given two algebraic sets $\mathbf{S}_1$ and $\mathbf{S}_2$ described by two finite subsets $\mathbf{F}_1$ and $\mathbf{F}_2$ respectively in the polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$, then*

$$\mathbf{S}_1 \cap \mathbf{S}_2 = \mathcal{Z}(\mathbf{F}_1, \mathbf{F}_2)$$
$$\mathbf{S}_1 \cup \mathbf{S}_2 = \mathcal{Z}(\mathbf{F}_1 \mathbf{F}_2)$$

where $\mathbf{F}_1\mathbf{F}_2$ denotes the set containing all the products of $f_i \in \mathbf{F}_1$ and $g_j \in \mathbf{F}_2$.

Proof. It is trivial for the intersection case. Let's consider the union case. Suppose $\mathbf{F}_1 = \{f_i(\mathbf{X}), i \in \mathbf{I}\}$ and $\mathbf{F}_2 = \{g_j(\mathbf{X}), j \in \mathbf{J}\}$, then $\mathbf{S}_1 = \mathcal{Z}(\sum_{i \in \mathbf{I}} f_i^2(\mathbf{X}))$, and $\mathbf{S}_2 = \mathcal{Z}(\sum_{j \in \mathbf{J}} g_j^2(\mathbf{X}))$. It is trivial to show that $\mathbf{S}_1 \cup \mathbf{S}_2 = \mathcal{Z}(\sum_{i \in \mathbf{I}} f_i^2(\mathbf{X}) \sum_{j \in \mathbf{J}} g_j^2(\mathbf{X}))$. Since $\sum_{i \in \mathbf{I}} f_i^2(\mathbf{X}) \sum_{j \in \mathbf{J}} g_j^2(\mathbf{X}) = \sum_{i \in \mathbf{I}} \sum_{j \in \mathbf{J}} \Big(f_i(\mathbf{X}) g_j(\mathbf{X}) \Big)^2$, by Lemma 3.2.1 we are done. □

Proposition 3.2.1. *For any subsets $\mathbf{F}_1 \subseteq \mathbf{F}_2 \subset \mathbb{R}[x_1, \ldots, x_n]$, $\mathcal{Z}(\mathbf{F}_1) \supseteq \mathcal{Z}(\mathbf{F}_2)$.*

Proof. Ref. [91]. □

Proposition 3.2.2. *The union of finitely many algebraic sets is an algebraic set. The intersection of any family of algebraic sets is an algebraic set. The empty set $\emptyset$ and the whole space $\mathbb{A}^n$ are algebraic sets.*

Proof. By Theorem 3.2.1. □

By Proposition 3.2.2, the collection of algebraic sets of $\mathbb{A}^n$ is closed under intersection operation, thus induces a topology on $\mathbb{A}^n$.

Definition 3.2.3. *The Zariski topology on $\mathbb{A}^n$ is defined by taking the closed set as the algebraic set on $\mathbb{A}^n$ generated by a finite subset of the polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$. The open set is defined by taking the complementary set of the closed set.*

Definition 3.2.4. *A nonempty subset $\mathbf{U}$ of a topological space $\mathbf{\Omega}$ is said to be irreducible if $\mathbf{U}$ cannot be expressed as the union of two proper subsets of $\mathbf{U}$ which are closed in $\mathbf{U}$.*

Definition 3.2.5. *An affine algebraic variety, simply algebraic variety, is defined by an irreducible closed subset of $\mathbb{A}^n$ with the Zariski topology.*

Definition 3.2.6. *Given a subset $\mathbf{S} \subset \mathbb{A}^n$, let*

$$\mathit{I}(\mathbf{S}) = \left\{ f \in \mathbb{R}[x_1, \ldots, x_n] \mid \forall \mathbf{X} \in \mathbf{S},\ f(\mathbf{X}) = 0 \right\}$$

be the ideal of the polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$ vanishing on $\mathbf{S}$.

Theorem 3.2.2. *An algebraic set $\mathbf{S} \subset \mathbb{A}^n$ is irreducible if and only if the ideal $\mathit{I}(\mathbf{S})$ of the polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$ vanishing on $\mathbf{S}$ is prime, i.e., if $a, b \in \mathbb{R}[x_1, \ldots, x_n]$ and $ab \in \mathit{I}(\mathbf{S}) \neq \mathbb{R}[x_1, \ldots, x_n]$, then $a \in \mathit{I}(\mathbf{S})$ or $b \in \mathit{I}(\mathbf{S})$.*

Proof. Ref. [92] □

Proposition 3.2.3. *If $\mathbf{S}_1 \subseteq \mathbf{S}_2 \subset \mathbb{A}^n$, then $\mathit{I}(\mathbf{S}_1) \supseteq \mathit{I}(\mathbf{S}_2)$.*

Proof. Ref. [91]. □

Proposition 3.2.4. *If $\mathbf{S}_1$ and $\mathbf{S}_2 \subset \mathbb{A}^n$, then $\mathit{I}(\mathbf{S}_1 \bigcup \mathbf{S}_2) = \mathit{I}(\mathbf{S}_1) \bigcap \mathit{I}(\mathbf{S}_2)$.*

Proof. Ref. [91]. □

3.3 Affine Algebraic Set Preserving Mappings

With the basic definitions and background stated, we can define a class of mappings for our purposes, and explore their properties.

Definition 3.3.1. *Consider a mapping $\phi : \mathbb{R}[x_1, \ldots, x_n] \to \mathbb{R}[x_1, \ldots, x_n]$. If there exist two polynomials f and $g \in \mathbb{R}[x_1, \ldots, x_n]$ such that*
1) $\phi(f) = g$,
2) $\mathcal{Z}(f)$ and $\mathcal{Z}(g)$ are irreducible,
3) $\mathcal{Z}(f)$ and $\mathcal{Z}(g)$ are homeomorphic,
then ϕ is called the affine algebraic variety preserving mapping, or shortly variety mapping, between f and g.

The variety mapping is defined from one polynomial to another, however, it also includes the situation that a finite set of polynomials is mapped to another finite set of polynomials because Lemma 3.2.1 ensures a set of polynomials is equivalent to a single polynomial. The core idea behind this definition, as well as the next definition of the affine algebraic set preserving mappings, is that although the algebraic variety (set) is preserved by the homeomorphism, some other features may be altered. Let's first provide a few examples to show that such kinds of mapping exist.

Let $f(x) = x - 1$, consider $\mathbb{M}(f) = f^2$. It is easy to show that $\mathbb{M}$ is an variety mapping between $x - 1$ and $x^2 - 2x + 1$.

Consider $h \in \mathbb{R}[x_1, \ldots, x_n]$ such that $\mathcal{Z}(h) = \emptyset$. Let $\phi(f) = hf$, then ϕ is a variety mapping between f and hf with the corresponding homeomorphism being the identity map. Generally, between any two elements of an arbitrary prime ideal of the polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$ there exists a variety mapping.

Definition 3.3.2. *Consider a mapping $\phi : \mathbb{R}[x_1, \ldots, x_n] \to \mathbb{R}[x_1, \ldots, x_n]$. If there exist two polynomials f and $g \in \mathbb{R}[x_1, \ldots, x_n]$ such that*
1) $\phi(f) = g$,
2) $\mathcal{Z}(g) = \bigcup_{k \in \mathbf{K}} \mathbf{V}_k$ with each $\mathbf{V}_k$ being an affine algebraic variety in $\mathcal{Z}(g)$,

3) $\mathcal{Z}(f)$ *is homeomorphic to* $\bigcup_{k\in\mathbf{L}} \mathbf{V}_k$ *for some* $\mathbf{L} \subseteq \mathbf{K}$*,*
then ϕ *is called the affine algebraic set preserving mapping, or shortly set mapping, between* f *and* g*.*

It should be pointed out that the affine algebraic variety preserving mappings are also the affine algebraic set preserving mappings when set $\mathbf{K}$ in Definition 3.3.2 is a singleton.

Theorem 3.3.1. *Affine algebraic set preserving mappings are closed under composition.*

Proof. Let ϕ_1 be a set mapping from f_1 to f_2 with $\mathcal{Z}(f_2)$ being the union of varieties $\bigcup_{i\in\mathbf{I}} \mathbf{V}_i$. Suppose $\mathcal{Z}(f_1)$ is homeomorphic to $\bigcup_{i\in\mathbf{J}} \mathbf{V}_i$ for $\mathbf{J} \subseteq \mathbf{I}$. Similarly, ϕ_2 is a set mapping from f_2 to f_3 with $\mathcal{Z}(f_3)$ being the union of varieties $\bigcup_{n\in\mathbf{N}} \mathbf{U}_n$. $\mathcal{Z}(f_2)$ is homeomorphic to $\bigcup_{n\in\mathbf{M}} \mathbf{U}_n$ for $\mathbf{M} \subseteq \mathbf{N}$. It suffices to show that the composition of two set mappings is still a set mapping.

First, we have $\phi(f_1) = \phi_2 \circ \phi_1(f_1) = \phi_2\Big(\phi_1(f_1)\Big) = \phi_2(f_2) = f_3$. Then we need to show that there exists a subset $\mathbf{H} \subseteq \mathbf{M}$ such that $\bigcup_{i\in\mathbf{J}} \mathbf{V}_i$ is homeomorphic to $\bigcup_{n\in\mathbf{H}} \mathbf{U}_n$.

Since $\bigcup_{i\in\mathbf{J}} \mathbf{V}_i$ is closed, its image under a homeomorphism should also be closed. Considering $\mathbf{U}_n$'s are irreducible, the image of $\bigcup_{i\in\mathbf{J}} \mathbf{V}_i$ should be $\bigcup_{n\in\mathbf{H}} \mathbf{U}_n$ for some $\mathbf{H} \subseteq \mathbf{N}$. Since $\bigcup_{i\in\mathbf{J}} \mathbf{V}_i \subseteq \mathcal{Z}(f_2)$, $\mathbf{H} \subseteq \mathbf{M}$.

Therefore, we have $\mathcal{Z}(f_1)$ is homeomorphic to $\bigcup_{i\in\mathbf{J}} \mathbf{V}_i$, which is homeomorphic to $\bigcup_{n\in\mathbf{H}} \mathbf{U}_n$ with $\mathbf{H} \subseteq \mathbf{M} \subseteq \mathbf{N}$. Thus, ϕ is a variety mapping. □

Corollary 3.3.1. *Affine algebraic variety preserving mappings are closed under composition.*

Proof. By Theorem 3.3.1 it is trivial. □

With the above results, we are going to discuss two specific mappings, and show that they are set mappings. These two mappings, and their composition, will eventually be applied in this chapter to our power flow models, ACOPF models and the multi-party ACOPF models.

3.3.1 Induced Affine Mapping over A Polynomial Ring

The first set mapping we are going to investigate is induced by the affine maps between affine spaces. Geometrically, this kind of set mappings does not change any structures of the algebraic

set, but defines a class of representations of the same algebraic set. An intuitive interpretation could be describing the same object in different coordinate systems. It plays an important role in masking the appearance of the original problem, and may relate symmetries to some algebraic set.

Definition 3.3.3. *Let* $\mathbf{V}$ *and* $\mathbf{U}$ *be two vector spaces over a field* $\mathbb{F}$*, a map* $\mathcal{L} : \mathbf{V} \to \mathbf{U}$ *is called a linear map if for any* α *and* $\beta \in \mathbb{F}$*,* $\mathbf{v} \in \mathbf{V}$ *and* $\mathbf{u} \in \mathbf{U}$ *we have* $\mathcal{L}(\alpha\mathbf{v} + \beta\mathbf{u}) = \alpha\mathcal{L}(\mathbf{v}) + \beta\mathcal{L}(\mathbf{u})$.

Definition 3.3.4. *Let* $\mathbb{A}$ *and* $\mathbb{B}$ *be two affine spaces, a map* $\mathcal{A} : \mathbb{A} \to \mathbb{B}$ *is called an affine map if there exists a linear map* $\mathcal{L}$ *such that for any* $\mathbf{a} \in \mathbb{A}$ *and* $\mathbf{b} \in \mathbb{B}$ *we have* $\mathcal{A}(\mathbf{a}) - \mathcal{A}(\mathbf{b}) = \mathcal{L}(\mathbf{a} - \mathbf{b})$.

Now let's consider a special map $\mathcal{A} : \mathbb{A}^n \to \mathbb{A}^n$ such that $\mathbf{X} \mapsto \mathbf{T}\mathbf{X} + \gamma$, where $\mathbf{T} \in \mathbb{A}^{n\times n}$ and $\gamma \in \mathbb{A}^n$ are fixed. It is apparently an affine map. Applying it to the coordinates of the polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$ we get an induced mapping over $\mathbb{R}[x_1, \ldots, x_n]$.

Definition 3.3.5. *Given an affine map* $\mathcal{A} : \mathbb{A}^n \to \mathbb{A}^n$*,* $\mathbf{X} \mapsto \mathbf{T}\mathbf{X} + \gamma$*, where* $\mathbf{T}$ *is nonsingular. It induces an natural mapping* $\mathcal{A}^* : \mathbb{R}[x_1, \ldots, x_n] \to \mathbb{R}[x_1, \ldots, x_n]$*, called the induced affine mapping, over the polynomial ring* $\mathbb{R}[x_1, \ldots, x_n]$ *such that* $\mathcal{A}^*\Big(f(\mathbf{X})\Big) = f(\mathbf{T}\mathbf{X} + \gamma)$ *for any* $f \in \mathbb{R}[x_1, \ldots, x_n]$.

Theorem 3.3.2. *An induced affine mapping* $\mathcal{A}^*$ *is an affine algebraic set preserving mapping for any* f *and* g *in the polynomial ring* $\mathbb{R}[x_1, \ldots, x_n]$ *such that* $g(\mathbf{X}) = f(\mathbf{T}\mathbf{X} + \gamma)$.

Proof. First notice that $\mathcal{A}^*(f) = g$ for $g(\mathbf{X}) = f(\mathbf{T}\mathbf{X} + \gamma)$.

Next, consider $\mathcal{A} : \mathcal{Z}(f) \to \mathcal{Z}(g)$. $\forall \mathbf{Y} \in \mathcal{Z}(g)$, $g(\mathbf{Y}) = f(\mathbf{T}\mathbf{Y} + \gamma) = f(\mathcal{A}(\mathbf{Y})) = 0$. So $\mathcal{A}$ is surjective from $\mathcal{Z}(f)$ to $\mathcal{Z}(g)$. Since $\mathbf{T}$ is nonsingular, $\forall \mathbf{Y}_1$ and $\mathbf{Y}_2 \in \mathcal{Z}(g)$, $\mathcal{A}(\mathbf{Y}_1) - \mathcal{A}(\mathbf{Y}_2) = \mathbf{T}(\mathbf{Y}_1 - \mathbf{Y}_2) = 0$ implies $\mathbf{Y}_1 = \mathbf{Y}_2$. So $\mathcal{A}$ is injective. Thus, $\mathcal{A}$ is a bijection. Finally, by the continuity of $\mathcal{A}$ and $\mathcal{A}^{-1}$, $\mathcal{A}$ is a homeomorphism between $\mathcal{Z}(f)$ and $\mathcal{Z}(g)$. □

Proposition 3.3.1. *Consider two subsets of polynomials* $\mathbf{F} = \{f_i,\ i \in \mathbf{I}\}$ *and* $\mathbf{G} = \{g_i,\ i \in \mathbf{I}\}$ *in the polynomial ring* $\mathbb{R}[x_1, \ldots, x_n]$*. An induced affine mapping* $\mathcal{A}^*$ *is an affine algebraic set preserving mapping beween* $\mathbf{F}$ *and* $\mathbf{G}$ *if* $g_i(\mathbf{X}) = f_i(\mathbf{T}\mathbf{X} + \gamma)$ *for all* $i \in \mathbf{I}$.

Proof. By Lemma 3.2.1, $\mathbf{F}$ and $\mathbf{G}$ can be represented by two single polynomials respectively. Then Theorem 3.3.2 concludes the proposition. □

Proposition 3.3.1 indicates that for an arbitrary polynomial subset $\mathbf{F}(\mathbf{X}) \subset \mathbb{R}[x_1, \ldots, x_n]$, an induced affine mapping $\mathcal{A}^*$ serves as the affine algebraic set preserving mapping to a new polynomial set $\mathbf{F}\Big(\mathcal{A}(\mathbf{X})\Big)$.

3.3.2 Linear Mapping over Polynomial Rings

Definition 3.3.6. *Consider a vector space $\mathbb{R}[x_1, \ldots, x_n]^n$ given by n-tuples of polynomial ring $\mathbb{R}[x_1, \ldots, x_n]$ over the closed real field $\mathbb{R}$. A map $\mathcal{L}^* : \mathbb{R}[x_1, \ldots, x_n]^n \to \mathbb{R}[x_1, \ldots, x_n]^n$ is called a linear mapping over polynomial rings if*

$$\mathcal{L}^*(\mathbf{P}) = \mathbf{RP}$$

where $\mathbf{P} \in \mathbb{R}[x_1, \ldots, x_n]^n$ and $\mathbf{R} \in \mathbb{R}^{n \times n}$ is invertible.

This class of mappings, as we will show below, is also an affine algebraic set preserving mapping.

Theorem 3.3.3. *A linear mapping $\mathcal{L}^* : \mathbb{R}[x_1, \ldots, x_n]^n \to \mathbb{R}[x_1, \ldots, x_n]^n$, $\mathbf{P} \mapsto \mathbf{RP}$ with $\mathbf{R} \in \mathbb{R}^{n \times n}$ invertible, is an affine algebraic set preserving mapping.*

Proof. Since a finite linear combination of polynomials is still a polynomial, it suffices to show that there exists a homeomorhpism between $\mathcal{Z}(\mathbf{P})$ and $\mathcal{Z}(\mathbf{RP})$.

For any $\mathbf{X} \in \mathcal{Z}(\mathbf{P})$, $\mathbf{P}(\mathbf{X}) = \mathbf{0}$, so $\mathbf{RP}(\mathbf{X}) = \mathbf{0}$. Thus, $\forall \mathbf{X} \in \mathcal{Z}(\mathbf{P}),\ \mathbf{X} \in \mathcal{Z}(\mathbf{RP})$. On the other hand, for any $\mathbf{Y} \in \mathcal{Z}(\mathbf{RP})$, $\mathbf{RP}(\mathbf{Y}) = 0$. Since $\mathbf{R}$ is invertible, $\mathbf{P}(\mathbf{Y}) = 0$. Thus $\forall \mathbf{Y} \in \mathcal{Z}(\mathbf{RP}),\ \mathbf{Y} \in \mathcal{Z}(\mathbf{P})$. Therefore, $\mathcal{Z}(\mathbf{P}) = \mathcal{Z}(\mathbf{RP})$, hence homeomorphic to each other by the identity map. □

It is worth comparing the differences between the induced affine mapping and the linear mapping. Although they are different mappings, one intrinsic difference should be revealed, and it turns out to be useful for the following sections.

Definition 3.3.7. *Consider two polynomial sets $\mathbf{F} = \{f_i,\ i \in \mathbf{I}\}$ and $\mathbf{G} = \{g_j,\ j \in \mathbf{J}\}$. If $\mathbf{F} \subseteq \mathbf{G}$, then $\mathcal{Z}(\mathbf{F})$ is called a natural embedding of $\mathcal{Z}(\mathbf{G})$.*

Theorem 3.3.4. *Consider two polynomial sets* $\mathbf{F}$ *and* $\mathbf{G}$*, if* $\mathcal{Z}(\mathbf{F})$ *is a natural embedding of* $\mathcal{Z}(\mathbf{G})$*, then* $\mathcal{Z}(\mathbf{F}) \supseteq \mathcal{Z}(\mathbf{G})$.

Proof. For any $\mathbf{X} \in \mathcal{Z}(\mathbf{G})$, we have $\mathbf{G}(\mathbf{X}) = \mathbf{0}$. Since $\mathbf{F} \subseteq \mathbf{G}$, $\mathbf{F}(\mathbf{X}) = \mathbf{0}$. Thus, $\mathcal{Z}(\mathbf{F}) \supseteq \mathcal{Z}(\mathbf{G})$. □

Definition 3.3.8. *Consider an affine algebraic set preserving mapping* ϕ *between two polynomial sets* $\mathbf{F}$ *and* $\mathbf{G}$*. If for any subset* $\mathbf{F}_i \subset \mathbf{F}$*, there exists a subset* $\mathbf{G}_i \subset \mathbf{G}$ *such that* ϕ *is an affine algebraic set preserving mapping between* $\mathbf{F}_i$ *and* $\mathbf{G}_i$*, then* ϕ *is called the natural embedding preserving mapping.*

Theorem 3.3.5. *The induced affine mapping is a natural embedding preserving mapping.*

Proof. By Theorem 3.3.2 the induced affine mapping maps each element f_i of $\mathbf{F}$ to an element g_j of $\mathbf{G}$ independently. Thus it maps any subset of $\mathbf{F}$ to the corresponding subset of $\mathbf{G}$. □

However, the linear mapping is not an natural embedding preserving mapping. We show this by an example. Consider

$$\mathbf{F} = \left\{ f_1 = y^2 - \frac{1}{4}x^2 - 1,\ f_2 = x^2 - \frac{1}{4}y^2 - 1 \right\}$$

Let

$$\mathbf{R} = \begin{bmatrix} 1 & 1 \\ \frac{1}{2} & 1 \end{bmatrix}$$

Then

$$\mathbf{G} = \mathbf{RF} = \left\{ g_1 = \frac{3}{4}y^2 + \frac{3}{4}x^2 - 2,\ g_2 = \frac{7}{8}x^2 + \frac{1}{4}y^2 - \frac{3}{2} \right\}$$

Although $\mathcal{Z}(\mathbf{F}) = \mathcal{Z}(\mathbf{G}) = \left\{ (\frac{2\sqrt{3}}{3}, \frac{2\sqrt{3}}{3}),\ (-\frac{2\sqrt{3}}{3}, \frac{2\sqrt{3}}{3}),\ (\frac{2\sqrt{3}}{3}, -\frac{2\sqrt{3}}{3}),\ (-\frac{2\sqrt{3}}{3}, -\frac{2\sqrt{3}}{3}) \right\}$, $\mathcal{Z}(f_1)$ cannot be homeomorphic to $\mathcal{Z}(g_1)$ or $\mathcal{Z}(g_2)$, because $\mathcal{Z}(f_1)$ is unbounded but $\mathcal{Z}(g_1)$ and $\mathcal{Z}(g_2)$ are both bounded. The following figure depicts both of them in pairs. It shows that the blue curves on the left hand side are unbounded, but the red curves on the right hand side are bounded.

Now let's make a stage conclusion that the induced affine mapping preserves every natural embedding of the algebraic set. However, the linear mapping only preserves the algebraic set,

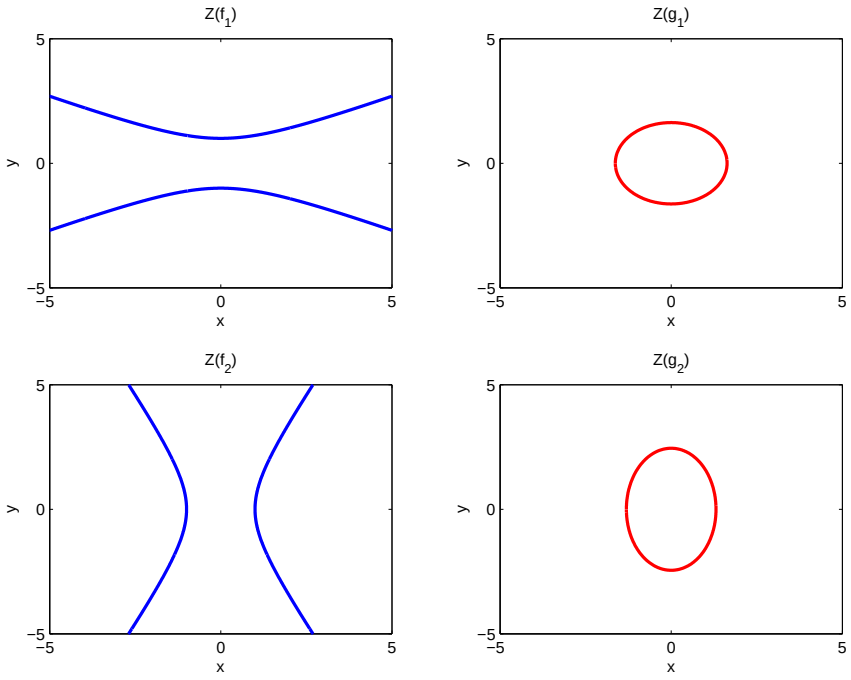


Figure 3.1: Plots of Individual Natural Embeddings

changing all its natural embeddings when necessary. The flexibility of the linear mapping to modify the natural embeddings of an algebraic set is crucial in the next discussions. It enables the ellipsoidal formulation of the power flow equations as well as of the Fritz John conditions for the ACOPF problem. Consequently, it enables the reliable and efficient application of the branch tracing method for both of them. Afterwards, the induced affine mappings will be applied to the multi-party ACOPF data encryption, and will be revisited towards another view of how to identify multiple solutions to these ellipsoidal formulations.

3.4 Ellipsoidal Formulation of Power Flow Equations

In the previous section we have shown that the linear mapping over the polynomial rings can change the natural embeddings of a certain algebraic set, converting the unbounded manifolds to bounded ones. This particular aim is achieved and proved to be feasible under mild assumptions

in this section, as well as in the next section, by the careful designs of the $\mathbf{R}$ matrix. Before that, some preliminary properties of the power flow equations should be stated.

Definition 3.4.1. *Consider a quadratic polynomial in n variables given by $\mathbf{X}^T\mathbf{M}\mathbf{X} + \mathbf{B}^T\mathbf{X} - 1$, where $\mathbf{X} \in \mathbb{R}^n$ for $n \geq 1$; $\mathbf{M} \in \mathbb{SR}^{n\times n}$ and $\mathbf{B} \in \mathbb{R}^n$ are constant matrix and vector. If $\mathbf{M}$ is positive definite, then $\mathcal{Z}(\mathbf{X}^T\mathbf{M}\mathbf{X} + \mathbf{B}^T\mathbf{X} - 1)$ is called a real $(n-1)$-dimensional ellipsoid, or shortly, an ellipsoid.*

Theorem 3.4.1. *A n-dimensional ellipsoid is homeomorphic to an n-dimensional sphere.*

Proof. Without loss of generality, let the center point of an ellipsoid be the origin. Consider an affine map $\mathcal{A}(\mathbf{X}) = \mathbf{V}^{-T}\Lambda^{-\frac{1}{2}}\mathbf{X}$ where Λ is the diagonal eigenvalue matrix of $\mathbf{M}$, $\mathbf{V}$ are the eigenvectors of $\mathbf{M}$. Its induced affine mapping $\mathcal{A}^*$ sends $\mathbf{X}^T\mathbf{M}\mathbf{X} - 1$ to $\mathbf{X}^T\mathbf{X} - 1$ whose algebraic set is a sphere. □

Proposition 3.4.1. *A n-dimensional ellipsoid is a connected compact manifold.*

Proof. Since a sphere is a connected compact manifold, by the homeomorphism we are done. □

Lemma 3.4.1. *Suppose $\mathbf{C} = \mathbf{A} + j\mathbf{B}$ with $\mathbf{A}$ and $\mathbf{B} \in \mathbb{R}^{n\times n}$, $\mathbf{A} = \mathbf{A}^T$, and $\mathbf{B} = -\mathbf{B}^T$. Then there is a one-to-one map f such that*

$$f(\mathbf{C}) = \hat{\mathbf{C}} = \begin{bmatrix} \mathbf{A} & -\mathbf{B} \\ \mathbf{B} & \mathbf{A} \end{bmatrix}.$$

Furthermore, if $\mathbf{C}$ has an eigenvalue λ with the corresponding eigenvector $\mathbf{v} = \mathbf{v}_d + j\mathbf{v}_q$, where $\mathbf{v}_d$ and $\mathbf{v}_q \in \mathbb{R}^n$, then $\begin{bmatrix} \mathbf{v}_d \\ \mathbf{v}_q \end{bmatrix}$ and $\begin{bmatrix} -\mathbf{v}_q \\ \mathbf{v}_d \end{bmatrix}$ are the eigenvectors of $\hat{\mathbf{C}}$ with the corresponding repeated eigenvalue λ.

Proof. Ref. [93]. □

Theorem 3.4.2. *Consider $\mathbf{C} = \mathbf{A} + j\mathbf{B}$ with $\mathbf{A}$ and $\mathbf{B} \in \mathbb{R}^{n\times n}$. Let $\mathbf{H} = conj(\mathbf{C}^T)\mathbf{e}_k\mathbf{e}_k^T + \mathbf{e}_k\mathbf{e}_k^T\mathbf{C}$, where $\mathbf{e}_k$ is a column vecter only has the unity on its k-th entry, then $\mathbf{H}$ only has two non-zero*

eigenvalues

$$\lambda_{\pm} = A_{k,k} \pm \sqrt{A_{k,k}^2 + \sum_{i \neq k} |C_{k,i}|^2}$$

where $A_{k,k}$ *is the* (k,k)*-th entry of* $\mathbf{A}$*,* $C_{k,i}$ *is the* (k,i)*-th entry of* $\mathbf{C}$*.* λ_+ *is non-negative,* λ_- *is non-positive. Their corresponding eigenvectors are*

$$\mathbf{v}_{\pm} = conj(\mathbf{C}_{\cdot,k}^T) + (\lambda_{\pm} - conj(C_{k,k}))\mathbf{e}_k$$

where $\mathbf{C}_{\cdot,k}^T$ *is the* k*-th column of* $\mathbf{C}^T$*.*

Proof. Ref. [93]. □

Proposition 3.4.2. *The quadratic matrix* $\mathbf{M}_{p,k}$*, as well as* $\mathbf{M}_{q,k}$*, in* Equation (2.11) *only has four non-zero eigenvalues: two repeated positive ones and two repeated negative ones.*

Proof. According to Theorem 3.4.2, let $\mathbf{C} = \mathbf{Y}_{bus}$, then $\mathbf{H} = (\mathbf{G}_{bus} - j\mathbf{B}_{bus})^T \mathbf{e}_k \mathbf{e}_k^T + \mathbf{e}_k \mathbf{e}_k^T (\mathbf{G}_{bus} + j\mathbf{B}_{bus})$ has two eigenvalues, one is positive, the other is negative. By Lemma 3.4.1, $\mathbf{H}$ can be one-to-one mapped to $\hat{\mathbf{H}}$. Thus for each eigenvalue of $\mathbf{H}$, it becomes two repeated eigenvalues of $\hat{\mathbf{H}}$. Therefore, $\hat{\mathbf{H}}$ has four eigenvalues, two repeated positive ones and two repeated negative ones. Finally, $\hat{\mathbf{H}}$ coincides with the quadratic matrix $\mathbf{M}_{p,k}$ in Equation (2.11). The argument is similar for $\mathbf{M}_{q,k}$. □

Corollary 3.4.1. *A power balancing equation defines an unbounded manifold.*

Proof. By Proposition 3.4.2, the algebraic set of any power balancing equation is homeomorphic to the algebraic set of $x_1^2 + x_2^2 - x_3^2 - x_4^2 - 1$, which defines an unbounded manifold. □

3.4.1 Network Model without Shunt Elastance

We have seen from Corollary 3.4.1 that in the power flow model, the power balancing equations are unbounded. Although the algebraic set of the power flow model is bounded, it is very likely that its natural embeddings are unbounded, which may cause difficulty when applying branch tracing method. To address this difficulty, we apply the linear mapping to the power flow model, changing each unbounded manifold into a bounded one. To show the existence of such linear mapping, we

need some mild assumptions and a lemma which is stated below. Throughout this chapter, we assume that the transmission line is modeled in Π equivalent circuit.

Lemma 3.4.2. *Consider a connected electric power grid with N_{bus} buses. Assume there are no phase shifting transformers and no shunt elastance on the transmission lines. Then the bus susceptance matrix $\mathbf{B}_{bus} \in \mathbb{SR}^{n\times n}$ is negative semi-definite. Furthermore, it has a single zero eigenvalue with the corresponding eigenvector $\mathbf{v}^T = [1\ 1\ \cdots\ 1]^T$.*

Proof. By Ref. [94] the bus admittance matrix is given by $\mathbf{Y}_{bus} = \mathbf{A}^T diag(\mathbf{y})\mathbf{A}$ where $\mathbf{A}$ is the bus incidence matrix, $\mathbf{y}$ is the line admittance vector. Thus we have $\mathbf{B}_{bus} = \mathbf{A}^T diag(\mathbf{b})\mathbf{A}$ where $\mathbf{b}$ is the line susceptance vector.

Since the value of line susceptance is typically negative (due to the inductive transmission line design), then $diag(\mathbf{y})$ must be negative definite.

Since the incidence matrix $\mathbf{A}$ only has one zero eigenvalue with the corresponding eigenvector $[1\ 1\ \cdots\ 1]^T$, then $\mathbf{B}_{bus}[1\ 1\ \cdots\ 1]^T = \mathbf{0}$. Thus $\mathbf{B}_{bus}$ has a single zero eigenvalue with the same eigenvector. □

The next theorem claims that for a power grid with no shunt elastance, the power flow model has an ellipsoidal formulation.

Theorem 3.4.3 (A Sufficient Condition for Ellipsoidal Formulation of Power Flow Model without Shunt Elements)**.** *Consider a connected electric power grid with N_{bus} buses and N_{gen} generators. Without loss of generality, suppose the 1-st bus is the slack bus, the 2-nd bus to the N_{gen}-th bus are PV buses, and the rest are PQ buses. Let $n = 2N_{bus}$. Assume further that there are no phase shifting transformers and no shunt elastance on the transmission lines. The power flow model $\mathbf{F}_{PF}$ in quadratic form has an ellipsoidal formulation. That is, there exists a linear mapping $\mathcal{L}^* : \mathbb{R}[x_1, \ldots, x_n]^n \to \mathbb{R}[x_1, \ldots, x_n]^n$, $\mathbf{F}_{PF} \mapsto \mathbf{F}_E = \mathbf{R}\mathbf{F}_{PF}$, where each element in $\mathbf{F}_E$ defines a $(n-1)$-dimensional ellipsoid.*

To show the theorem is equivalent to show that there exists a constant matrix $\mathbf{R}$ such that each polynomial in $\mathbf{F}_E = \mathbf{R}\mathbf{F}_{PF}$ has a positive definite quadratic matrix, which will be sufficed by

showing that there exists a linear combination of power flow equations that represents an ellipsoid. This is done by the following lemma.

Lemma 3.4.3. *Under the assumptions of Theorem 3.4.3, there exists a constant $\gamma \in \mathbb{R}$ such that*

$$\gamma \sum_{i=1}^{N_{gen}} \mathbf{M}_{v,i} - \sum_{j=N_{gen}+1}^{N_{bus}} \mathbf{M}_{q,j} \succ 0 \tag{3.1}$$

$$\gamma \sum_{i=1}^{N_{gen}} V_{i,0}^2 + \sum_{j=N_{gen}+1}^{N_{bus}} Q_{load,j} > 0 \tag{3.2}$$

Proof. Since $V_{i,0}^2 > 0$ for high voltage magnitude, condition (3.2) is trivial. Let's show condition (3.1) by induction.

Firstly, consider $N_{gen} = 1$. The quadratic matrix is given by

$$\begin{aligned} \gamma \sum_{i=1}^{N_{gen}} \mathbf{M}_{v,i} - \sum_{j=N_{gen}+1}^{N_{bus}} \mathbf{M}_{q,j} &= \gamma \mathbf{M}_{v,1} + \mathbf{M}_{q,1} - \sum_{j=1}^{N_{bus}} \mathbf{M}_{q,j} \\ &= \gamma \mathbf{M}_{v,1} + \mathbf{M}_{q,1} + \frac{1}{2} \begin{bmatrix} -\mathbf{B}_{bus} & \mathbf{0} \\ \mathbf{0} & -\mathbf{B}_{bus} \end{bmatrix} \end{aligned} \tag{3.3}$$

Let $\hat{\mathbf{B}}_{bus} := \frac{1}{2} \begin{bmatrix} -\mathbf{B}_{bus} & \mathbf{0} \\ \mathbf{0} & -\mathbf{B}_{bus} \end{bmatrix}$. According to Lemma 3.4.2, $\hat{\mathbf{B}}_{bus} \succeq 0$ with two repeated zero eigenvalues and their corresponding eigenvectors $[0, \cdots, 0, 1, \cdots, 1]^T$ and $[1, \cdots, 1, 0, \cdots, 0]^T$.

Since $\gamma \mathbf{M}_{v,1} + \mathbf{M}_{q,1}$ only has the 1-st and the $(N_{bus}+1)$-th rows and columns nonzero, its eigenvalues can be dominated by the 1-st and the $(N_{bus}+1)$-th entries on the diagonal. Since $\mathbf{M}_{v,1}$ only has nonzero (specifically positive) elements at the the 1-st and the $(N_{bus}+1)$-th entries, $\gamma \mathbf{M}_{v,1}$ will dominate the eigenvalues by some large γ. So $\gamma \mathbf{M}_{v,1} + \mathbf{M}_{q,1}$ is positive semi-definite. Thus, Equation (3.3) is at least positive semi-definite for some large γ. Next, we are going to show it is strictly potive definite.

Note that the zero eigenvector of $\hat{\mathbf{B}}_{bus}$ takes the form of

$$\mathbf{v} = a[0, \cdots, 0, 1, \cdots, 1]^T + b[1, \cdots, 1, 0, \cdots, 0]^T$$

where $a^2 + b^2 \neq 0$. Choose any zero eigenvector $\mathbf{v}$ such that $|\mathbf{v}| = 1$, we have $\mathbf{v}^T\Big(\gamma\mathbf{M}_{v,1} + \mathbf{M}_{q,1}\Big)\mathbf{v} = \gamma(a^2 + b^2) + \mathbf{v}^T\mathbf{M}_{q,1}\mathbf{v}$. Since $\mathbf{v}^T\mathbf{M}_{q,1}\mathbf{v}$ is bounded, then a large γ will dominate $\mathbf{v}^T\Big(\gamma\mathbf{M}_{v,1} + \mathbf{M}_{q,1}\Big)\mathbf{v}$ and makes it positive. Hence, it is impossible to coincide the zero eigenvectors of $\gamma\mathbf{M}_{v,1} + \mathbf{M}_{q,1}$ and $\hat{\mathbf{B}}_{bus}$, which means Equation (3.3) is positive definite for some large γ.

Secondly, suppose for $N_{gen} = n$ the positive definiteness holds. Let's consider $N_{gen} = n + 1$. The quadratic matrix for $N_{gen} = n + 1$ is given by

$$\gamma \sum_{i=1}^{n+1} \mathbf{M}_{v,i} - \sum_{j=(n+1)+1}^{N_{bus}} \mathbf{M}_{q,j} = \gamma \sum_{i=1}^{n} \mathbf{M}_{v,i} - \sum_{j=n+1}^{N_{bus}} \mathbf{M}_{q,j} + \gamma\mathbf{M}_{v,n+1} - \mathbf{M}_{q,n+1} \tag{3.4}$$

By the assumption, $\gamma \sum_{i=1}^{n} \mathbf{M}_{v,i} - \sum_{j=n+1}^{N_{bus}} \mathbf{M}_{q,j} \succ 0$. With the similar arguments we made for $N_{gen} = 1$, $\gamma\mathbf{M}_{v,n+1} - \mathbf{M}_{q,n+1} \succeq 0$ for some γ. Therefore, $\gamma \sum_{i=1}^{n+1} \mathbf{M}_{v,i} - \sum_{j=(n+1)+1}^{N_{bus}} \mathbf{M}_{q,j} \succ 0$ for some large γ, which concludes the lemma. □

Lemma 3.4.3 implies that for a connected electric power grid without shunt elastance a linear combination of the voltage equations for the PV buses and the reactive power equations for the PQ buses can always provide us a $(2N_{bus} - 1)$-dimensional ellipsoid, which is called the *base ellipsoid*. Adding the scaled base ellipsoid to other polynomials in the power flow model $\mathbf{F}_{PF}$, we can make all of them to be ellipsoids, which concludes Theorem 3.4.3.

3.4.2 Network Model with Shunt Elastance

Theorem 3.4.3 guarantees the ellipsoidal formulation of the quadratic power flow model for power grids without shunt elastance. However, to describe a power grid more accurately, shunt elastance matrix is usually considered for the bus admittance matrix, which will weaken the above result.

In this subsection, we will obtain a sufficient condition to guarantee the ellipsoidal formulation with shunt elastance. Although one can intentionally design a power grid that fails our sufficient condition, in practice however, real power system models are very likely to satisfy it. On the other

hand, even though the sufficient condition fails, one may still construct an ellipsoidal formulation for the power flow problem.

Theorem 3.4.4 (A Sufficient Condition for Ellipsoidal Formulation of Power Flow Model with Shunt Elastance)**.** *Consider a connected electric power grid with N_{bus} buses and N_{gen} generators. Suppose the 1-st bus is the slack bus, the 2-nd bus to the N_{gen}-th bus are PV buses, and the rest are PQ buses. Assume that there are no phase shifting transformers. Let the bus shunt elastance matrix be $\mathbf{S}_{shunt}$, where $\mathbf{S}_{shunt} \succeq 0$ is a diagonal matrix. Define the imaginary part of the bus admittance matrix $\mathbf{Y}_{bus}$ to be $\mathbf{B}_{bus} := \mathbf{B}_0 + \mathbf{S}_{shunt}$, where $\mathbf{B}_0$ is the bus susceptance matrix of the grid without shunt elastance. Separate $\mathbf{B}_0$ and $\mathbf{S}_{shunt} \in \mathbb{R}^{N_{bus}\times N_{bus}}$ into sub-matrices*

$$\mathbf{B}_0 = \begin{bmatrix} \mathbf{B}_{1,1} & \mathbf{B}_{1,2} \\ \mathbf{B}_{2,1} & \mathbf{B}_{2,2} \end{bmatrix}$$

$$\mathbf{S}_{shunt} = \begin{bmatrix} \mathbf{S}_{1,1} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_{2,2} \end{bmatrix}$$

where $\mathbf{B}_{1,1},\ \mathbf{S}_{1,1} \in \mathbb{R}^{N_{gen}\times N_{gen}}$, $\mathbf{B}_{1,2} \in \mathbb{R}^{N_{gen}\times N_{load}}$, $\mathbf{B}_{2,1} \in \mathbb{R}^{N_{load}\times N_{gen}}$ and $\mathbf{B}_{2,2},\ \mathbf{S}_{2,2} \in \mathbb{R}^{N_{load}\times N_{load}}$ for $N_{load} = N_{bus} - N_{gen}$.

The power flow model $\mathbf{F}_{PF}$ in quadratic form has a base ellipsoid formulation described by

$$\mathbf{U}^T\Big(\gamma \sum_{i=1}^{N_{gen}} \mathbf{M}_{v,i} - \sum_{j=N_{gen}+1}^{N_{bus}} \mathbf{M}_{q,j}\Big)\mathbf{U} = \gamma \sum_{i=1}^{N_{gen}} V_{i,0}^2 + \sum_{j=N_{gen}+1}^{N_{bus}} Q_{load,j} > 0 \tag{3.5}$$

provided that $-\mathbf{B}_{2,2} - \mathbf{S}_{2,2} \succ 0$.

To prove Theorem 3.4.5, a few preliminary lemmas and propositions are needed.

Lemma 3.4.4. *Every principal submatrix of $\mathbf{B}_0$ (not include $\mathbf{B}_0$ itself) is negative definite.*

Proof. By Lemma 3.4.2, $\mathbf{B}_0$ is negative semi-definite with a single zero eigenvalue and the corresponding eigenvector $[1, \cdots, 1]^T$. So every strict principal submatrix of $\mathbf{B}_0$ is also negative semi-definite. Since adding all the columns of $\mathbf{B}_0$ results in a zero vector, then take out at least one column and one row of $\mathbf{B}_0$ will make it non-singular. Thus, every principal submatrix of $\mathbf{B}_0$ (not include $\mathbf{B}_0$ itself) is negative definite. □

Lemma 3.4.5. *Consider a connected power grid without phase shifting transformers and shunt elastance. Denote the quadratic matrix of the reactive power balancing equation in* Equation (2.11b) *by* $\mathbf{M}^0_{q,j}$. *Let* $\mathbf{T}^0(\gamma) := \gamma \sum_{i=1}^{N_{gen}} \mathbf{M}_{v,i} - \sum_{j=N_{gen}+1}^{N_{bus}} \mathbf{M}^0_{q,j}$. *Then the minimum eigenvalue of* $\mathbf{T}^0(\gamma)$, *denoted by* $\lambda_{min}(\mathbf{T}^0(\gamma))$, *is a non-decreasing function with respect to* γ.

Proof. Consider $z := \mathbf{v}^T\mathbf{T}^0(\gamma)\mathbf{v}$ for $|\mathbf{v}|^2 = 1$. Then z is bounded below by $\lambda_{min}(\mathbf{T}^0(\gamma))$, and z can achieve $\lambda_{min}(\mathbf{T}^0(\gamma))$ when $\mathbf{v}$ is the eigenvector of $\mathbf{T}^0(\gamma)$ associated with $\lambda_{min}(\mathbf{T}^0(\gamma))$. For any fixed vector $\mathbf{v}$, $\mathbf{v}^T\mathbf{T}^0(\gamma)\mathbf{v} = \mathbf{v}^T(\gamma \sum \mathbf{M}_{v,i})\mathbf{v} - \mathbf{v}^T(\sum \mathbf{M}^0_{q,j})\mathbf{v}$ with $\sum \mathbf{M}_{v,i} \succeq 0$, then $\mathbf{v}^T(\gamma \sum \mathbf{M}_{v,i})\mathbf{v}$ is non-decreasing with respect to γ at $\mathbf{v}$. Now let's show the Lemma by contradiction.

Suppose there exist $\gamma_1 < \gamma_2 \in \mathbb{R}$ such that $\lambda_{min}(\mathbf{T}^0(\gamma_1)) > \lambda_{min}(\mathbf{T}^0(\gamma_2))$. Then, $\mathbf{v}_1^T\mathbf{T}^0(\gamma_1)\mathbf{v}_1 > \mathbf{v}_2^T\mathbf{T}^0(\gamma_2)\mathbf{v}_2$, where $\mathbf{v}_1$ is the eigenvector associated with $\lambda_{min}(\mathbf{T}^0(\gamma_1))$ and $\mathbf{v}_2$ is the eigenvector associated with $\lambda_{min}(\mathbf{T}^0(\gamma_2))$. Since $\gamma_1 < \gamma_2$, we have $\mathbf{v}_2^T\mathbf{T}^0(\gamma_2)\mathbf{v}_2 \geq \mathbf{v}_2^T\mathbf{T}^0(\gamma_1)\mathbf{v}_2$ by the non-decreasing property at fixed $\mathbf{v}_2$. Therefore, $\mathbf{v}_1^T\mathbf{T}^0(\gamma_1)\mathbf{v}_1 > \mathbf{v}_2^T\mathbf{T}^0(\gamma_1)\mathbf{v}_2$. However, since $\mathbf{v}_1^T\mathbf{T}^0(\gamma_1)\mathbf{v}_1 = \lambda_{min}(\mathbf{T}^0(\gamma_1))$ which is the minimal value for all $|\mathbf{v}|^2 = 1$, we have the contradiction from $\mathbf{v}_1^T\mathbf{T}^0(\gamma_1)\mathbf{v}_1 \leq \mathbf{v}_2^T\mathbf{T}^0(\gamma_1)\mathbf{v}_2$.

Therefore, for any $\gamma_1 < \gamma_2 \in \mathbb{R}$ we have $\lambda_{min}(\mathbf{T}^0(\gamma_1)) \leq \lambda_{min}(\mathbf{T}^0(\gamma_2))$. □

Proposition 3.4.3. $0 < \lim_{\gamma\to\infty} \lambda_{min}(\mathbf{T}^0(\gamma)) < \infty$.

Proof. By the bus ordering in Theorem 3.4.5, the structure of $\mathbf{T}^0(\gamma) \in \mathbb{R}^{2N_{gen}\times 2N_{gen}}$ is

$$\mathbf{T}^0(\gamma) = \begin{bmatrix} \gamma\mathbf{I} & \mathbf{T}_{1,2} & \mathbf{T}_{1,3} & \mathbf{T}_{1,4} \\ \mathbf{T}_{2,1} & -\mathbf{B}_{2,2} & \mathbf{T}_{2,3} & \mathbf{T}_{2,4} \\ \mathbf{T}_{3,1} & \mathbf{T}_{3,2} & \gamma\mathbf{I} & \mathbf{T}_{3,4} \\ \mathbf{T}_{4,1} & \mathbf{T}_{4,2} & \mathbf{T}_{4,3} & -\mathbf{B}_{2,2} \end{bmatrix}$$

where $\mathbf{I} \in \mathbb{R}^{N_{gen}\times N_{gen}}$ is the identity matrix.

As $\gamma \to \infty$, $\gamma\mathbf{I}$ will dominate the corresponding rows and columns. Thus, $\lambda_{min}(\mathbf{T}^0(\gamma)) = \lambda_{min}(-\mathbf{B}_{2,2}) < \infty$. Since $\mathbf{B}_{2,2}$ is a leading principal sub-matrix of $\mathbf{B}_0$, by Lemma 3.4.4, $-\mathbf{B}_{2,2}$ is positive definite. Hence, $\lambda_{min}(\mathbf{T}^0(\gamma)) > 0$. □

Corollary 3.4.2. $max(\lambda_{min}(\mathbf{T}^0(\gamma))) \to \lambda_{min}(-\mathbf{B}_{2,2})$ *as* $\gamma \to \infty$.

Proof. By the proof of Proposition 3.4.3. □

Proof. (for Theorem 3.4.5) According to Corollary 3.4.2 and Lemma 3.4.5, the maximum value of the minimum eigenvalue of $\mathbf{T}^0(\gamma)$ is the minimum eigenvalue of the sub-matrix $-\mathbf{B}_{2,2}$. Now let's consider a power grid with shunt elastance. Let $\mathbf{T}(\gamma) := \gamma \sum_{i=1}^{N_{gen}} \mathbf{M}_{v,i} - \sum_{j=N_{gen}+1}^{N_{bus}} \mathbf{M}_{q,j}$. Then

$$\mathbf{T}(\gamma) = \mathbf{T}^0(\gamma) - \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_{2,2} \end{bmatrix} \tag{3.6}$$

Since $\mathbf{S}_{2,2}$ is fixed, we have $max(\lambda_{min}(\mathbf{T}(\gamma))) \to \lambda_{min}(-\mathbf{B}_{2,2} - \mathbf{S}_{2,2})$ as $\gamma \to \infty$. Therefore, if $-\mathbf{B}_{2,2} - \mathbf{S}_{2,2} \succ 0$, there exists some $\gamma < \infty$ such that $\lambda_{min}(\mathbf{T}(\gamma)) > 0$, which indicates that $\mathbf{T}(\gamma) \succ 0$. □

3.4.3 Network Model with Shunt Conductance and Elastance

Theorem 3.4.4 claims a sufficient condition to guarantee an ellipsoidal formulation of a power flow problem with shunt elastance. However, when the sufficient condition fails, one can still have a chance to construct an ellipsoidal formulation for the power flow problem.

This subsection presents a more general condition to ensure an ellipsoidal formulation of a power grid with both shunt conductance and elastance. Theorem 3.4.3 and Theorem 3.4.4 can be considered as two special cases in this general condition. Again, one can artificially design a power grid that fails such condition, nevertheless, real power system models are very likely to satisfy it. On the other hand, even though the sufficient condition fails, one may still construct an ellipsoidal formulation for the power flow problem.

Theorem 3.4.5 (A Sufficient Condition for Ellipsoidal Formulation of Power Flow Model with Shunt Conductance and Elastance)**.** *Consider a connected electric power grid with* N_{bus} *buses and* N_{gen} *generators. Suppose the* 1*-st bus is the slack bus, the* 2*-nd bus to the* N_{gen}*-th bus are PV buses, and the rest are PQ buses. Assume that there are no phase shifting transformers. Let the bus shunt conductance matrix be* $\mathbf{C}_{shunt}$*, where* $\mathbf{C}_{shunt} \succeq 0$ *is diagonal. Let the bus shunt*

elastance matrix be $\mathbf{S}_{shunt}$*, where* $\mathbf{S}_{shunt} \succeq 0$ *is diagonal. Consider the bus admittance matrix* $\mathbf{Y}_{bus} = \mathbf{G}_{bus} + j\mathbf{B}_{bus}$*, where* $\mathbf{G}_{bus} := \mathbf{G}_0 + \mathbf{C}_{shunt}$ *and* $\mathbf{B}_{bus} := \mathbf{B}_0 + \mathbf{S}_{shunt}$*,* $\mathbf{G}_0$ *is the bus conductance matrix without shunt conductance,* $\mathbf{B}_0$ *is the bus susceptance matrix without shunt elastance. Separate* $\mathbf{G}_0$*,* $\mathbf{C}_{shunt}$*,* $\mathbf{B}_0$ *and* $\mathbf{S}_{shunt} \in \mathbb{R}^{N_{bus} \times N_{bus}}$ *into sub-matrices*

$$\mathbf{G}_0 = \begin{bmatrix} \mathbf{G}_{1,1} & \mathbf{G}_{1,2} \\ \mathbf{G}_{2,1} & \mathbf{G}_{2,2} \end{bmatrix}$$

$$\mathbf{C}_{shunt} = \begin{bmatrix} \mathbf{C}_{1,1} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{2,2} \end{bmatrix}$$

$$\mathbf{B}_0 = \begin{bmatrix} \mathbf{B}_{1,1} & \mathbf{B}_{1,2} \\ \mathbf{B}_{2,1} & \mathbf{B}_{2,2} \end{bmatrix}$$

$$\mathbf{S}_{shunt} = \begin{bmatrix} \mathbf{S}_{1,1} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_{2,2} \end{bmatrix}$$

where $\mathbf{G}_{1,1}$*,* $\mathbf{C}_{1,1}$*,* $\mathbf{B}_{1,1}$ *and* $\mathbf{S}_{1,1} \in \mathbb{R}^{N_{gen} \times N_{gen}}$*,* $\mathbf{G}_{1,2}$ *and* $\mathbf{B}_{1,2} \in \mathbb{R}^{N_{gen} \times N_{load}}$*,* $\mathbf{G}_{2,1}$ *and* $\mathbf{B}_{2,1} \in \mathbb{R}^{N_{load} \times N_{gen}}$*,* $\mathbf{G}_{2,2}$*,* $\mathbf{C}_{2,2}$*,* $\mathbf{B}_{2,2}$ *and* $\mathbf{S}_{2,2} \in \mathbb{R}^{N_{load} \times N_{load}}$ *for* $N_{load} = N_{bus} - N_{gen}$*. If one of the following conditions is true,*

1. $\mathbf{G}_{2,2} + \mathbf{C}_{2,2} \succ 0$*,*
2. $-\mathbf{B}_{2,2} - \mathbf{S}_{2,2} \succ 0$*,*

then the power flow problem has an ellipsoidal formulation.

Proof. The proof of Theorem 3.4.4 ensures the second condition.

To prove the first condition, we let

$$\mathbf{T}(\gamma) := \gamma \sum_{i=1}^{N_{gen}} \mathbf{M}_{v,i} + \sum_{j=N_{gen}+1}^{N_{bus}} \mathbf{M}_{p,j} \tag{3.7}$$

and follow the similar arguments for the proof of Theorem 3.4.4. Here we provide a sketch for the proof.

Firstly, show that any principal submatrix of $\mathbf{G}_{bus}$ is positive semi-definite. Secondly, show that $max(\lambda_{min}(\mathbf{T}(\gamma))) \to \lambda_{min}(\mathbf{G}_{2,2} + \mathbf{C}_{2,2})$ as $\gamma \to \infty$. Finally, show that the constant term $\gamma \sum_{i=1}^{N_{gen}} V_{i,0}^2 - \sum_{j=N_{gen}+1}^{N_{bus}} P_{load,j} > 0$ for some large γ.

Then, condition 1 will induce an ellipsoidal formulation. □

Note that if $\mathbf{S}_{shunt} = \mathbf{0}$, Theorem 3.4.5 is reduced to Theorem 3.4.3.

3.4.4 Ellipsoidal Formulation for The ZIP Load Flow Model

Consider the "ZIP" load flow model in (2.15), we regard the voltage magnitude $|V_k|$ as new a variable. Since in (2.15a) and (2.15b) the voltage magnitude $|V_k|$ is independent to $\mathbf{U}$, we can construct the ellipsoidal formulation in the same way as (3.5) or (3.7), provided the constant coefficients $P_{2,k}$ and $Q_{2,k}$ are non-negative.

However, one should be cautious about applying the "ZIP" model to the power flow problem for solving low-voltage solutions because the "ZIP" model is a simplified load response designated to accommodate the change of the high-voltage magnitude. Hence, the low voltage solutions to the "ZIP" model may not be physically meaningful. For example, if a load bus voltage is zero, the load power should also be zero. This can be true only when the constant power terms $P_{0,k}$ and $Q_{0,k}$ are zeros in (2.15).

3.4.5 Boundedness of PV and QV Curves

The PV and QV curves are examined to identify the total transfer capability (TTC) at a certain bus, or the maximum loadability of a power grid, and to locate voltage static stability boundary points. These curves are usually calculated by continuation power flow algorithms which follow some paths defined by continuously increasing active power or reactive power at certain buses. Typical PV and QV curves are given in Figure 3.2.

However, for some power grid models, the PV and QV curves can be unbounded. We illustrate this unboundedness via a small example which is modified from the "case5Salam" test case [14]. Specifically, we reduce the line resistance of "case5Salam" to zero, and follow QV curves by

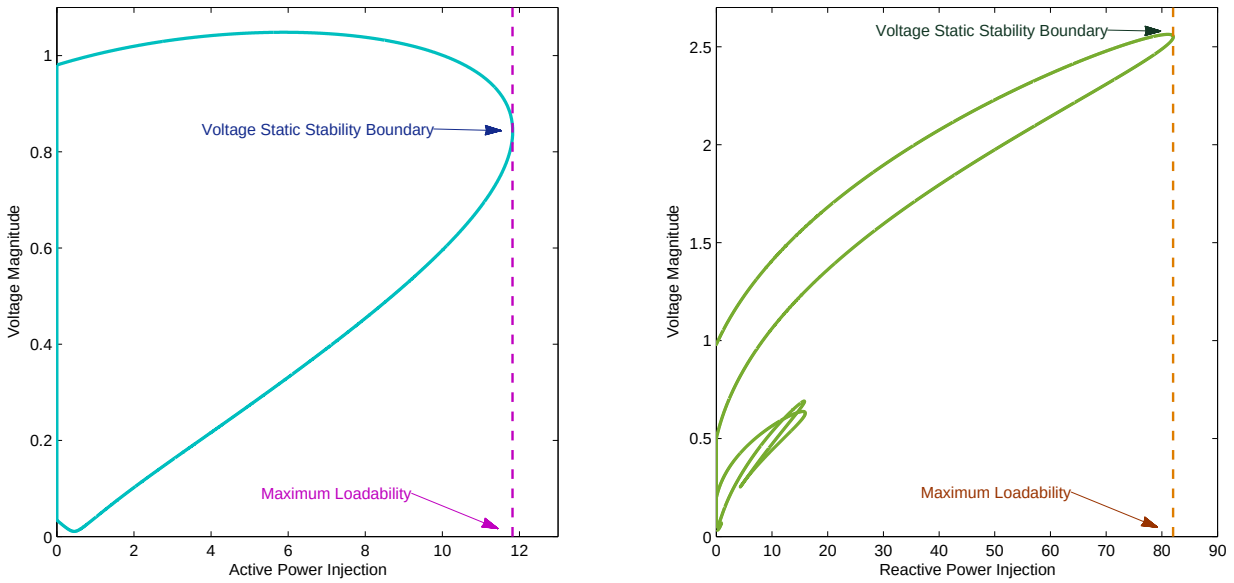


Figure 3.2: PV Curve (Left) and QV Curve (Right)

continuously increasing the reactive power at all the PQ buses. The QV curves are shown in Figure 3.3.

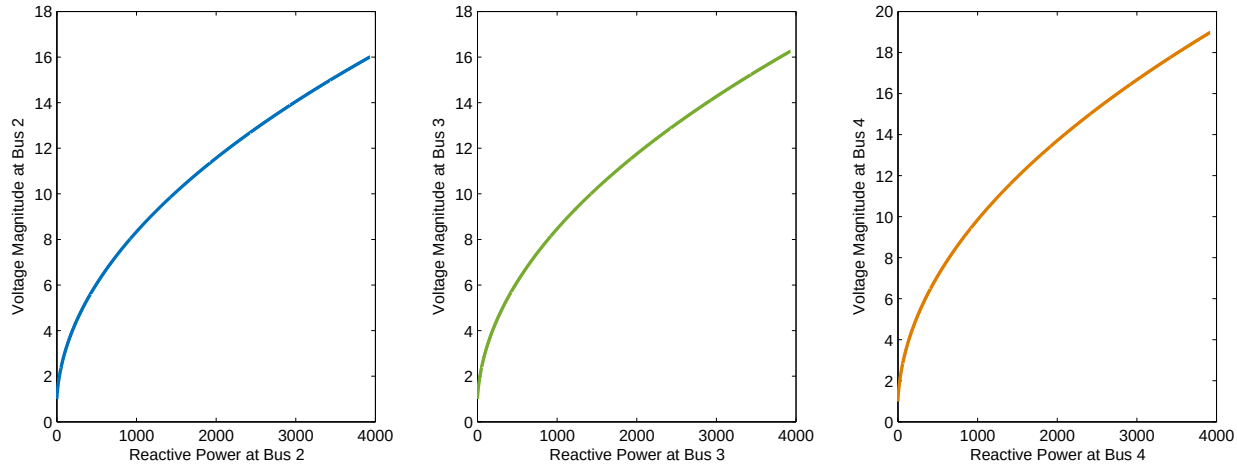


Figure 3.3: Unbounded QV Curves on Bus 2 (Left), Bus 3 (Middle), and Bus 4 (Right)

To ensure bounded PV and QV curves, we provide a sufficient condition as follow.

Theorem 3.4.6. *Consider a power flow model* $\mathbf{F}_{PF} \in \mathbb{R}[x_1, \ldots, x_n]^n$. *Let* $\mathbf{F}_{PF,i} \in \mathbb{R}[x_1, \ldots, x_n]^{n-1}$ *be a subset of* $\mathbf{F}_{PF}$ *withouth the* i*-th element. The real solution set* $\mathcal{Z}(\mathbf{F}_{PF,i})$ *is bounded if* $\mathbf{F}_{PF,i}$ *has an ellipsoidal formulation.*

Proof. If $\mathbf{F}_{PF,i}$ has an ellipsoidal formulation, there exists an invertible mapping $\mathbf{M}_i : \mathbb{R}[x_1, \ldots, x_n]^{n-1} \to \mathbb{R}[x_1, \ldots, x_n]^{n-1}$ such that $\mathbf{M}_i(\mathbf{F}_{PF,i}) = \mathbf{E}_{PF,i}$, where each element of $\mathbf{E}_{PF,i}$ represents a $(n-1)$-dimensional ellipsoid. Since any ellipsoid is bounded, the intersection of $(n-1)$ of them is still bounded. □

Corollary 3.4.3. *QV curves are bounded if the network is lossy everywhere.*

Proof. If the network is lossy everywhere, Condition 1 in Theorem 3.4.5 is satisfied. Thus, by Theorem 3.4.5 we can construct a base ellipsoid independent of reactive power balance equations, which further suggests that any QV curve can be represented by an intersection of a bunch of ellipsoids. □

Corollary 3.4.3 is somehow counter-intuitive because reactive power is directly related to the transmission line susceptance. However, our analysis indicates that the upper bound of reactive power injection can be influenced by the transmission line conductance. Specifically, if a power grid is lossy everywhere, then bus reactive power injection must be bounded.

Corollary 3.4.4. *PV curves are bounded if Condition 2 in Theorem 3.4.5 is satisfied.*

Proof. If Condition 2 in Theorem 3.4.5 is true, there exists a base ellipsoid which is independent of active power balance equations. Therefore, a PV curve can be formulated as an intersection of a bunch of ellipsoids. □

Corollary 3.4.4 states that the upper bound of bus active power injection can be influenced by the transmission line susceptance and shunt capacitance. This result is also counter-intuitive since the active power loss is determined by the conductance on transmission lines. However, Condition 2 in Theorem 3.4.5 provides a sufficient condition for the boundedness of bus active power injection which only relates to the susceptance. It suggests that as long as we have at least

one PV bus (or the slack bus), the active power injection of a power grid must be bounded under Condition 2, even without line conductance.

3.5 Ellipsoidal Formulation of the First Order Conditions of ACOPF

This section applies our linear mapping to the first order conditions of the equality constrained ACOPF model, turning all the unbounded manifolds into bounded ones, namely the high dimensional ellipsoids. Then the ellipsoidal formulation enables the reliable implementation of the branch tracing method for the next chapter to identify multiple local solutions to the ACOPF problem. We briefly introduce the KKT conditions of the equality constrained ACOPF and discuss the difficulties for an ellipsoidal formulation. Thus, the Fritz John conditions are elaborated with one extra constraint to eventually reach its ellipsoidal formulation.

3.5.1 Karush-Kuhn-Tucker Conditions for Equality Constrained ACOPF

Recall the equality constrained ACOPF model with quadratic objective function in Equation (2.31), the objective function and the constraints are all in quadratic form without linear terms except the New Relation in Equation (2.31l). Particularly, if we only consider the linear objective function model in Equation (2.30), then every polynomial in the model is quadratic without linear terms. Without loss of generality, let's consider the quadratic objective function model for this section, then Equation (2.31) can be concisely written as

$$\begin{aligned} \textbf{Minimize} \quad & \mathbf{X}^T\mathbf{D}\mathbf{X} \\ \textbf{Subject to}: \quad & \mathbf{X}^T\mathbf{M}_j\mathbf{X} - r_j = 0 \\ & \mathbf{X}^T\mathbf{M}_i\mathbf{X} + \mathbf{B}_i^T\mathbf{X} - r_i = 0 \\ & j = 1, ..., 3N_{gen} + 4N_{bus} + 2N_{line} \\ & i = 1, ..., N_{gen} \end{aligned} \tag{3.8}$$

where $\mathbf{X} \in \mathbb{R}^{6N_{gen}+4N_{bus}+2N_{line}}$ is the variable vector by stacking $P_{gen,i}$, $p_{gen,i}$, $\mathbf{U}$, $s_{p,max,k}$, $s_{p,min,k}$, $s_{q,max,k}$, $s_{q,min,k}$, $s_{v,max,k}$, $s_{v,min,k}$, $s_{I,max,in,n}$ and $s_{I,max,out,n}$ into a column.

The Lagrange function of this optimization problem in Equation (3.8) is

$$\begin{aligned}\mathcal{L}(\mathbf{X},\lambda,\mu) = \quad & \mathbf{X}^T\mathbf{D}\mathbf{X} - \textstyle\sum_{j=1}^{3N_{gen}+4N_{bus}+2N_{line}} \lambda_j\Big(\mathbf{X}^T\mathbf{M}_j\mathbf{X} - r_j\Big) \\ & - \textstyle\sum_{i=1}^{N_{gen}} \mu_i\Big(\mathbf{X}^T\mathbf{M}_i\mathbf{X} + \mathbf{B}_i^T\mathbf{X} - r_i\Big)\end{aligned} \tag{3.9}$$

where λ and μ are the Lagrangian multiplier vectors to the constraints; λ_j is the j-th entry of λ; μ_i is the i-th entry of μ.

Suppose $\mathbf{X}_*$ is a local solution to problem (3.8), and the linear independent constraint qualification (LICQ) holds. By the Karush-Kuhn-Tucker (KKT) Conditions there exist λ_* and μ_* such that the following conditions are satisfied

$$\bigtriangledown_{\mathbf{X}}\mathcal{L}(\mathbf{X}_*,\lambda_*,\mu_*) = 0 \tag{3.10a}$$

$$\mathbf{X}_*^T\mathbf{M}_j\mathbf{X}_* - r_j = 0 \tag{3.10b}$$

$$\mathbf{X}_*^T\mathbf{M}_i\mathbf{X}_* + \mathbf{B}_i^T\mathbf{X}_* - r_i = 0 \tag{3.10c}$$

Note that conditions (3.10) are not equivalent to the KKT conditions of the original problem (2.18) because the original problem has "less-or-equal-to" inequality constraints for which the multipliers should be non-positive:

$$\lambda_k^* \leq 0 \tag{3.11}$$

where k is in the index set of the active inequality constraints of problem (2.18). However, the complementarity conditions for problem (2.18) are intrinsically included in the conditions of (3.10a). Specifically, they are equivalent to the partial derivatives of $\mathcal{L}(\mathbf{X},\lambda,\mu)$ with respect to the slack variables. Thus, conditions (3.10) with (3.11) are equivalent to the KKT conditions of problem (2.18). This relation implies that solutions to (3.10) should include solutions of the KKT conditions for problem (2.18). To distinguish different solutions and models, we refer "KKT conditions" to the KKT conditions for the original problem (2.18), and refer "KKT solutions (points)" to the solutions that satisfy the KKT conditions for the original problem (2.18). Also, we use "eq-KKT conditions" to represent the KKT conditions for the equality constrained problem (2.31) or (2.30), and use "eq-KKT solutions (points)" to represent the solutions that satisfy the eq-KKT conditions (3.10) for those equality constrained problems.

The eq-KKT conditions comprise only equalities which are more favorable than inequalities for constructing ellipsoidal formulations and applying the branch tracing method. Hence, we use eq-KKT conditions throughout our algorithm design and leave condition (3.11) to be checked after obtaining multiple eq-KKT solutions.

Even with the help of equalities in the eq-KKT conditions, however, we still lack an important feature to construct an ellipsoidal formulation; and the dependence on the constraint qualifications for both the KKT and the eq-KKT conditions can cause potential difficulties for branch tracing methods. To illustrate these restrictions, let's examine Equation (3.10a) for details.

$$\bigtriangledown_{\mathbf{x}} \mathcal{L}(\mathbf{X}, \lambda, \mu) = 2\mathbf{D}\mathbf{X} - \sum_j 2\lambda_j \mathbf{M}_j \mathbf{X} - \sum_i \mu_i \Big(2\mathbf{M}_i \mathbf{X} + \mathbf{B}_i\Big) = \mathbf{0} \tag{3.12}$$

Since λ and μ are also unknowns when we unsolve the first order conditions, Equation (3.12) is quadratic with respect to $\mathbf{X}$, λ and μ, with some linear terms $\mathbf{D}\mathbf{X}$ and $\mu_i \mathbf{B}_i$. However, since λ and μ only appear in the cross-product terms with $\mathbf{X}$, there are no univariate quadratic terms of λ or μ in Equation (3.12). It implies a difficulty of designing a positive-definite quadratic polynomial with respect to $\mathbf{X}$, λ and μ that retains the eq-KKT solutions.

Another issue occurs from the strong reliance on the constraint qualifications. It is possible that an optimization problem has a local solution that does not satisfy the constraint qualifications, which may result in no eq-KKT and KKT solutions. For example, try to minimize x subject to $x^2 = 0$. The solution is trivial but neither eq-KKT nor KKT solutions exist. For our branch tracing method, it traces the first order conditions by continuously changing some parameters. It is possible that there are some particular parameter values that fail the constraint qualifications, and hence cause numerical issues.

3.5.2 Sphere Confined Fritz John Conditions and Its Ellipsoidal Formulation

To deal with the above two difficulties in the KKT conditions, we employ the Fritz John (FJ) conditions and restrict them on a sphere. The detailed comparison between the Fritz John conditions and the KKT conditions can be found in [95]. Briefly, the Fritz John conditions are a generalized version of the KKT conditions, which do not require any constraint qualifications. However, it could in return induce 'fake' local solutions [95], but this issue can be easily addressed.

First let's define a scalar function $\mathcal{F}(\mathbf{X}, \lambda, \mu, \delta)$ by

$$\begin{aligned}\mathcal{F}(\mathbf{X}, \lambda, \mu, \delta) = \quad & \delta \mathbf{X}^T \mathbf{D} \mathbf{X} - \textstyle\sum_{j=1}^{3N_{gen}+4N_{bus}+2N_{line}} \lambda_j \Big(\mathbf{X}^T \mathbf{M}_j \mathbf{X} - r_j \Big) \\ & - \textstyle\sum_{i=1}^{N_{gen}} \mu_i \Big(\mathbf{X}^T \mathbf{M}_i \mathbf{X} + \mathbf{B}_i^T \mathbf{X} - r_i \Big)\end{aligned} \tag{3.13}$$

where δ is a scalar multiplier to the objective function; λ and μ are the multiplier vectors to the constraints; λ_j is the j-th entry of λ; μ_i is the i-th entry of μ.

Suppose $\mathbf{X}_*$ is a local solution to problem (3.8), by the Fritz John Conditions there exist δ_*, λ_* and μ_* such that the following conditions are satisfied

$$\bigtriangledown_{\mathbf{X}} \mathcal{F}(\mathbf{X}_*, \lambda_*, \mu_*, \delta_*) = 0 \tag{3.14a}$$

$$\mathbf{X}_*^T \mathbf{M}_j \mathbf{X}_* - r_j = 0 \tag{3.14b}$$

$$\mathbf{X}_*^T \mathbf{M}_i \mathbf{X}_* + \mathbf{B}_i^T \mathbf{X}_* - r_i = 0 \tag{3.14c}$$

$$(\delta_*, \lambda_*, \mu_*) \neq (0, \mathbf{0}, \mathbf{0}) \tag{3.14d}$$

Note that if LICQ holds, then $\delta_* \neq 0$ and the strict complementary condition holds.

Again, the FJ conditions (3.14) for problem (3.8) are not equivalent to the FJ conditions for the original problem (2.18) because the multipliers for active inequalities of (2.18) are further constrained by their signs:

$$\begin{aligned}\lambda_{*k} \leq 0, \text{ if } \delta_* \geq 0 \\ \lambda_{*k} \geq 0, \text{ if } \delta_* \leq 0\end{aligned} \tag{3.15}$$

where k is in the index set of the active inequality constraints of Equation (2.18).

Conditions (3.14) with (3.15) are equivalent to the FJ conditions of the original problem (2.18). This relation implies that solutions to (3.14) should include solutions of the FJ conditions for problem (2.18). To distinguish different solutions and models, we refer "FJ conditions" to the FJ conditions for the original problem (2.18), and refer "FJ solutions (points)" to the solutions that satisfy the FJ conditions for the original problem (2.18). Also, we use "eq-FJ conditions" to represent the FJ conditions for the equality constrained problem (2.31) or (2.30), and use "eq-FJ solutions (points)" to represent the solutions that satisfy the eq-FJ conditions (3.14) for those equality constrained problems.

Consider Equation (3.14a) for details.

$$\bigtriangledown_{\mathbf{x}}\mathcal{F}(\mathbf{X},\lambda,\mu,\delta) = 2\delta\mathbf{D}\mathbf{X} - \sum_j 2\lambda_j\mathbf{M}_j\mathbf{X} - \sum_i \mu_i\Big(2\mathbf{M}_i\mathbf{X} + \mathbf{B}_i\Big) = \mathbf{0} \tag{3.16}$$

If $\mathbf{X}_*$, δ_*, λ_* and μ_* is a solution to Equation (3.16), then $\mathbf{X}_*$, $K\delta_*$, $K\lambda_*$ and $K\mu_*$ is also a solution for any $K \neq 0$. To avoid such a situation, we set a spherical constraint to the multipliers.

$$\delta^2 + \lambda^T\lambda + \mu^T\mu = 1 \tag{3.17}$$

This extra condition eliminates the nonzero conditions of Equation (3.14d) and provides us the univariate quadratic terms of all the multipliers, enabling us to construct the ellipsoidal formulation.

Thus, the sphere confined eq-FJ conditions are

$$\bigtriangledown_{\mathbf{x}}\mathcal{F}(\mathbf{X}_*,\lambda_*,\mu_*,\delta_*) = 0 \tag{3.18a}$$

$$\mathbf{X}_*^T\mathbf{M}_j\mathbf{X}_* - r_j = 0 \tag{3.18b}$$

$$\mathbf{X}_*^T\mathbf{M}_i\mathbf{X}_* + \mathbf{B}_i^T\mathbf{X}_* - r_i = 0 \tag{3.18c}$$

$$\delta^2 + \lambda^T\lambda + \mu^T\mu = 1 \tag{3.18d}$$

For conditions (3.18), a base ellipsoid can be constructed as

$$\begin{aligned}
&\textstyle\sum_i^{N_{gen}} \Big[(P_{gen,i}^2 + s_{p,max,i}^2 - P_{max,i}^2) + \frac{1}{2}(s_{p,min,i}^2 - P_{gen,i}^2 + P_{min,i}^2) \\
&+(\mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} + s_{q,max,i}^2 - Q_{max,i}) + (s_{q,min,i}^2 - \mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} + Q_{min,i}) \\
&+(p_{gen,i}^2 - P_{load,i} - \mathbf{U}^T\mathbf{M}_{p,i}\mathbf{U})\Big] + \textstyle\sum_n^{N_{line}} \Big[(\mathbf{U}^T\mathbf{M}_{Iin,n}\mathbf{U} + s_{I,max,in,n}^2 - I_{max,in}^2) \\
&+(\mathbf{U}^T\mathbf{M}_{Iout,n}\mathbf{U} + s_{I,max,out,n}^2 - I_{max,out}^2)\Big] + \gamma_0 \textstyle\sum_k^{N_{bus}} \Big[(\mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} + s_{v,max,k}^2 - V_{max,k}^2) \\
&+\tfrac{1}{2}(s_{v,min,k}^2 - \mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} + V_{min,k}^2)\Big] + (\delta^2 + \lambda^T\lambda + \mu^T\mu - 1) = 0
\end{aligned} \tag{3.19}$$

where $\gamma_0 \in \mathbb{R}$ is a constant.

Theorem 3.5.1. *As γ_0 large enough, the algebraic set of* Equation (3.19) *is an ellipsoid.*

Proof. The quadratic matrix of Equation (3.19) has most of its nonzero (positive) elements on the diagonal except for $\mathbf{M}_{q,i}$, $\mathbf{M}_{p,i}$, $\mathbf{M}_{Iin,n}$ and $\mathbf{M}_{Iout,n}$. These types of matrices only related to the voltage vector $\mathbf{U}$, thus can be dominated by the matrices $\gamma_0\mathbf{M}_{v,k}$'s. Since $\mathbf{M}_{v,k}$ has nonzero (positive) elements on the diagonal, a large γ_0 guarantees the positive definiteness of the entire quadratic matrix. □

With the base ellipsoid constructed, the entire sphere confined Fritz John conditions can be mapped to ellipsoids under linear mappings. Note that there are multiple ways to construct a base ellipsoid. Equation (3.19) is one particular construction.

3.5.3 Ellipsoidal Formulations for Bounded Quadratic Constrained Quadratic Programming Problems

Through the discussion of previous subsections one may note that the existence of ellipsoidal formulation of eq-FJ conditions is mildly dependent on problem formulations. It is possible that any bounded quadratic constrained quadratic programming (BQCQP) problems can have ellipsoidal formulations for their eq-FJ conditions. This subsection will provide a positive answer.

Without loss of generality, consider a QCQP problem

$$\begin{aligned} \textbf{Minimize} \quad & \mathbf{x}^T\mathbf{D}\mathbf{x} + \mathbf{c}^T\mathbf{x} \\ \textbf{Subject to}: \quad & \mathbf{x}^T\mathbf{M}_j\mathbf{x} + \mathbf{b}_j^T\mathbf{x} - r_j \leq 0 \\ & \mathbf{x}^T\mathbf{H}_k\mathbf{x} + \mathbf{p}_k^T\mathbf{x} - v_k = 0 \\ & j = 1, ..., s \\ & k = 1, ..., t \end{aligned} \tag{3.20}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the decision variable vector, $\mathbf{D}$, $\mathbf{M}_j$ and $\mathbf{H}_k \in \mathbb{R}^{n\times n}$, $\mathbf{c}$, $\mathbf{b}_j$ and $\mathbf{p}_k \in \mathbb{R}^n$, r_j and $v_k \in \mathbb{R}$.

We apply our square term trick from subsection 2.3.2 to make (3.20) completely equality constrained.

$$\begin{aligned} \textbf{Minimize} \quad & \mathbf{x}^T\mathbf{D}\mathbf{x} + \mathbf{c}^T\mathbf{x} \\ \textbf{Subject to}: \quad & \mathbf{x}^T\mathbf{M}_j\mathbf{x} + \mathbf{b}_j^T\mathbf{x} + \tau_j^2 - r_j = 0 \\ & \mathbf{x}^T\mathbf{H}_k\mathbf{x} + \mathbf{p}_k^T\mathbf{x} - v_k = 0 \\ & j = 1, ..., s \\ & k = 1, ..., t \end{aligned} \tag{3.21}$$

where $\tau_j \in \mathbb{R}$ is the free slack variable. To recover the first order solutions to the original problem (3.20), one only needs to check the sign conditions on the multipliers associated with the binding inequality constraints.

Let's first construct the scalar function for the sphere confined eq-FJ conditions of (3.21):

$$\begin{aligned} \mathcal{F}(\mathbf{x}, \tau, \lambda, \mu, \delta) \quad & = \delta(\mathbf{x}^T\mathbf{D}\mathbf{x} + \mathbf{c}^T\mathbf{x}) - \textstyle\sum_{j=1}^{s} \lambda_j \Big(\mathbf{x}\mathbf{M}_j\mathbf{x} + \mathbf{b}_j^T\mathbf{x} + \tau_j^2 - r_j\Big) \\ & \quad - \textstyle\sum_{k=1}^{t} \mu_k(\mathbf{x}^T\mathbf{H}_k\mathbf{x} + \mathbf{p}_k^T\mathbf{x} - v_k) \end{aligned} \tag{3.22}$$

The sphere confined eq-FJ conditions are presented as

$$\bigtriangledown_{\mathbf{x},\tau}\mathcal{F}(\mathbf{x}_*, \tau_*, \lambda_*, \mu_*, \delta_*) = \mathbf{0} \tag{3.23a}$$

$$\mathbf{x}_*\mathbf{M}_j\mathbf{x}_* + \mathbf{b}_j^T\mathbf{x}_* + \tau_{j,*}^2 - r_j = 0 \tag{3.23b}$$

$$\mathbf{x}_*^T\mathbf{H}_k\mathbf{x}_* + \mathbf{p}_k^T\mathbf{x}_* - v_k = 0 \tag{3.23c}$$

$$\delta_*^2 + \lambda_*^T\lambda_* + \mu_*^T\mu_* = 1 \tag{3.23d}$$

One should note that the boundedness of the feasible space of (3.20) may not induce a trivial construction of a base ellipsoid for (3.23). To overcome this difficulty, we introduce artificial constraints to (3.20) as follow.

$$\begin{aligned} \textbf{Minimize} \quad & \mathbf{x}^T\mathbf{D}\mathbf{x} + \mathbf{c}^T\mathbf{x} \\ \textbf{Subject to}: \quad & \mathbf{x}^T\mathbf{M}_j\mathbf{x} + \mathbf{b}_j^T\mathbf{x} - r_j \leq 0 \\ & \mathbf{x}^T\mathbf{H}_k\mathbf{x} + \mathbf{p}_k^T\mathbf{x} - v_k = 0 \\ & X_\alpha^2 \leq \zeta_\alpha^2 \\ & j = 1, ..., s \\ & k = 1, ..., t \\ & \alpha = 1, ..., n \end{aligned} \tag{3.24}$$

where $\zeta_\alpha \in \mathbb{R}$.

As long as (3.20) is bounded, we can always choose some large ζ_α such that $X_\alpha^2 \leq \zeta_\alpha^2$ will never be binding for all α. This indicates that (3.24) should retain the same feasible space of (3.20) and the same first order solutions as well. Therefore, the equality constrained QCQP problem is written

$$\begin{aligned} \textbf{Minimize} \quad & \mathbf{x}^T\mathbf{D}\mathbf{x} + \mathbf{c}^T\mathbf{x} \\ \textbf{Subject to}: \quad & \mathbf{x}^T\mathbf{M}_j\mathbf{x} + \mathbf{b}_j^T\mathbf{x} + \tau_j^2 - r_j = 0 \\ & \mathbf{x}^T\mathbf{H}_k\mathbf{x} + \mathbf{p}_k^T\mathbf{x} - v_k = 0 \\ & X_\alpha^2 + \eta_\alpha^2 = \zeta_\alpha^2 \\ & \alpha = 1, ..., n \end{aligned} \tag{3.25}$$

where η_α is the free slack variable at index α.

The scalar function for the sphere confined eq-FJ conditions of (3.25) is

$$\begin{aligned} \mathcal{F}(\mathbf{X}, \lambda, \mu, \delta) \quad & = \delta(\mathbf{x}^T\mathbf{D}\mathbf{x} + \mathbf{c}^T\mathbf{x}) - \textstyle\sum_{j=1}^{s} \lambda_j \Big(\mathbf{x}\mathbf{M}_j\mathbf{x} + \mathbf{b}_j^T\mathbf{x} + \tau_j^2 - r_j\Big) \\ & \quad - \textstyle\sum_{k=1}^{t} \mu_k(\mathbf{x}^T\mathbf{H}_k\mathbf{x} + \mathbf{p}_k^T\mathbf{x} - v_k) - \sum_{\alpha=1}^{n} \theta_\alpha(X_\alpha^2 + \eta_\alpha^2 - \zeta_\alpha^2) \end{aligned} \tag{3.26}$$

The sphere confined eq-FJ conditions with artificial constraints are presented as

$$\bigtriangledown_{\mathbf{x},\tau}\mathcal{F}(\mathbf{x}_*, \tau_*, \lambda_*, \mu_*, \theta_*, \delta_*) = \mathbf{0} \tag{3.27a}$$

$$\mathbf{x}_*\mathbf{M}_j\mathbf{x}_* + \mathbf{b}_j^T\mathbf{x}_* + \tau_{j,*}^2 - r_j = 0 \tag{3.27b}$$

$$\mathbf{x}_*^T\mathbf{H}_k\mathbf{x}_* + \mathbf{p}_k^T\mathbf{x}_* - v_k = 0 \tag{3.27c}$$

$$X_{\alpha,*}^2 + \eta_{\alpha,*}^2 - \zeta_\alpha^2 = 0 \tag{3.27d}$$

$$\delta_*^2 + \lambda_*^T\lambda_* + \mu_*^T\mu_* + \theta_*^T\theta_* = 1 \tag{3.27e}$$

With the help of (3.27d) we can construct an ellipsoidal formulation for problem (3.21).

Theorem 3.5.2. *The first order solutions to* (3.21) *can be represented by the intersection of a set of high-dimensional ellipsoids, provided the feasible space of* (3.20) *is bounded.*

Proof. It suffices to show that there exists a base ellipsoid as a linear combination of (3.27).

Let's consider

$$\begin{aligned}&\gamma_0 \textstyle\sum_{\alpha=1}^{n}(X_\alpha^2 + \eta_\alpha^2 - \zeta_\alpha^2) + \sum_{j=1}^{s}(\mathbf{x}\mathbf{M}_j\mathbf{x} + \mathbf{b}_j^T\mathbf{x} + \tau_j^2 - r_j)\\ &+(\delta^2 + \lambda^T\lambda + \mu^T\mu + \theta^T\theta - 1) = 0\end{aligned} \tag{3.28}$$

Note that (3.28) includes all the decision variables, slack variables and multipliers. As long as γ_0 is large enough, the square terms of each decision variable X_α in the first summation will dominate the second summation, henceforth, providing a positive definite quadratic matrix. On the other hand, since the constant term ζ_α^2 is positive (we can make it not equal to 0), a large γ_0 will ensure a positive constant term. In summary, as γ_0 large enough, (3.28) represents a high-dimensional ellipsoid. □

3.6 Ellipsoidal Mappings and Its First Order Embedding Mappings

Previous subsections have discussed a few linear mappings that can construct ellipsoidal formulations for some power flow problems, optimal power flow problems, and BQCQP problems. Generally, they map a set of quadratic polynomials to another set of quadratic polynomials in

which each individual polynomial represents a high dimensional ellipsoid. Such mappings have special geometric structures which will be discussed in this subsection.

Definition 3.6.1. *Consider n quadratic polynomials in n variavles given by $\mathbf{B}(\mathbf{X}) \in \mathbb{R}[x_1, \ldots, x_n]^n$, where $\mathbf{X} \in \mathbb{R}^n$ for $n \geq 1$. If there exists a constant vector $\mathbf{p} \in \mathbb{R}^n$ such that $\{\mathbf{X} \in \mathbb{R}^n | \mathbf{p}^T\mathbf{B}(\mathbf{X}) = 0\}$ represents a real $(n-1)$-dimensional ellipsoid, then $\mathbf{B}(\mathbf{X})$ is called an ellipsoidal base.*

Corollary 3.6.1. *Theorem 3.4.3, 3.4.4, and 3.4.5 indicate that power flow equations under certain conditions form an ellipsoidal base.*

Corollary 3.6.2. *Theorem 3.5.1 indicates that the sphere confined eq-FJ conditions of optimal power flow problems form an ellipsoidal base.*

Corollary 3.6.3. *Theorem 3.5.2 indicates that the sphere confined eq-FJ conditions of BQCQP problems form an ellipsoidal base.*

Definition 3.6.2. *Given an ellipsoidal base $\mathbf{B}(\mathbf{X}) \in \mathbb{R}[x_1, \ldots, x_n]^n$, if there exists a linear mapping $\mathcal{E} : \mathbb{R}[x_1, \ldots, x_n]^n \to \mathbb{R}[x_1, \ldots, x_n]^n$ such that $\mathcal{E}\mathbf{B}(\mathbf{X}) = \mathbf{E}(\mathbf{X})$, and each element of $\mathbf{E}(\mathbf{X})$ represents a $(n-1)$-dimensional ellipsoid, then $\mathcal{E}$ is called an ellipsoidal mapping associated with $\mathbf{B}(\mathbf{X})$.*

Lemma 3.6.1. *The set of ellipsoidal mappings associated with the same ellipsoidal base are closed under addition.*

Proof. Given two ellipsoidal mappings $\mathcal{E}_1$ and $\mathcal{E}_2$ associated with the ellipsoidal base $\mathbf{B}(\mathbf{X})$. Let $\mathcal{E}_1\mathbf{B}(\mathbf{X}) = \mathbf{E}_1(\mathbf{X})$ and $\mathcal{E}_2\mathbf{B}(\mathbf{X}) = \mathbf{E}_2(\mathbf{X})$, then $(\mathcal{E}_1 + \mathcal{E}_2)\mathbf{B}(\mathbf{X}) = \mathbf{E}_1(\mathbf{X}) + \mathbf{E}_2(\mathbf{X})$. Since each entry of $\mathbf{E}_1(\mathbf{X})$ or $\mathbf{E}_2(\mathbf{X})$ is an ellipsoid, its quadratic matrix must be positive definite, and the constant scalar must be positive. Adding two positive definite matrix together will result in a positive definite matrix, so is for positive scalars. Therefore, each entry of $\mathbf{E}_1(\mathbf{X}) + \mathbf{E}_2(\mathbf{X})$ has a positive definite quadratic matrix and a positive constant scalar. Thus, $\mathcal{E}_1 + \mathcal{E}_2$ is also an ellipsoidal mapping. □

Theorem 3.6.1. *The set of ellipsoidal mappings $\Theta_B := \{\mathcal{E} \mid \mathcal{E}\mathbf{B}(\mathbf{X}) = \mathbf{E}(\mathbf{X})\ ellipsoids\}$ for a fixed ellipsoidal base $\mathbf{B}(\mathbf{X})$ is convex.*

Proof. Let $\mathcal{E}_1$ and $\mathcal{E}_2 \in \Theta_B$. For $0 \leq \lambda \leq 1$, consider $\mathcal{E}_\lambda := (1-\lambda)\mathcal{E}_1 + \lambda\mathcal{E}_2$, we have

$$\begin{aligned} \mathcal{E}_\lambda \mathbf{B}(\mathbf{X}) &= \Big((1-\lambda)\mathcal{E}_1 + \lambda\mathcal{E}_2\Big)\mathbf{B}(\mathbf{X}) && (3.29a) \\ &= (1-\lambda)\mathcal{E}_1\mathbf{B}(\mathbf{X}) + \lambda\mathcal{E}_2\mathbf{B}(\mathbf{X}) && (3.29b) \\ &= (1-\lambda)\mathbf{E}_1(\mathbf{X}) + \lambda\mathbf{E}_2(\mathbf{X}) && (3.29c) \end{aligned}$$

where $\mathbf{E}_1(\mathbf{X})$ and $\mathbf{E}_2(\mathbf{X})$ are the images of $\mathbf{B}(\mathbf{X})$ under $\mathcal{E}_1$ and $\mathcal{E}_1$, respectively.

By Lamma 3.6.1, $(1-\lambda)\mathbf{E}_1(\mathbf{X}) + \lambda\mathbf{E}_2(\mathbf{X})$ is a vector for which each entry represents an ellipsoid. Therefore, $\mathcal{E}_\lambda$ is also an ellipsoidal mapping at $\mathbf{B}(\mathbf{X})$, which concludes the proof. □

Definition 3.6.3. *Consider an ellipsoidal base $\mathbf{B}(\mathbf{X})$ and its associated ellipsoidal mapping $\mathcal{E}$. Let $\mathcal{E}_{-k}$ be a mapping given by $\mathcal{E}$ without the k-th row. We call $\mathcal{E}_{-k}$ the first order embedding mapping at k. Denote the image of $\mathbf{B}(\mathbf{X})$ under $\mathcal{E}_{-k}$ as $\mathbf{E}_{-k}(\mathbf{X})$.*

Corollary 3.6.4. *The zero set of $\mathbf{E}_{-k}(\mathbf{X})$, denoted by $\mathcal{Z}(\mathbf{E}_{-k})$, is a natural embedding of the zero set of $\mathbf{E}(\mathbf{X})$, denoted by $\mathcal{Z}(\mathbf{E})$, namely, $\mathcal{Z}(\mathbf{E}_{-k}) \supseteq \mathcal{Z}(\mathbf{E})$.*

Proof. Since $\mathbf{E}_{-k}(\mathbf{X}) \subset \mathbf{E}(\mathbf{X})$, by definition 3.3.7 we are done. □

Note that $\mathbf{E}_{-k}(\mathbf{X})$ includes $n-1$ polynomials in $\mathbb{R}^n$. The zero set $\mathcal{Z}(\mathbf{E}_{-k})$ is comprised by 1-dimensional smooth curves generically.

Theorem 3.6.2. *$\mathcal{Z}(\mathbf{E}_{-k})$ is identical to the set $\mathcal{Z}(\mathbf{B}; \mathcal{E}_{-k}) := \{\mathbf{X} \in \mathbb{R}^n \mid \mathbf{B}(\mathbf{X}) = Null(\mathcal{E}_{-k}) \times \alpha,\ \forall\alpha \in \mathbb{R}\}$.*

Proof. Firstly, for any $\mathbf{X}^* \in \mathcal{Z}(\mathbf{E}_{-k})$, we have

$$\mathcal{E}_{-k}\mathbf{B}(\mathbf{X}^*) = \mathbf{E}_{-k}(\mathbf{X}^*) = \mathbf{0} \tag{3.30}$$

Thus, $\mathcal{Z}(\mathbf{E}_{-k}) \subseteq \mathcal{Z}(\mathbf{B}; \mathcal{E}_{-k})$.

Secondly, for any $\mathbf{X}^* \in \mathcal{Z}(\mathbf{B}; \mathcal{E}_{-k})$, we have

$$\mathbf{E}_{-k}(\mathbf{X}^*) = \mathcal{E}_{-k}\mathbf{B}(\mathbf{X}^*) = \mathbf{0} \tag{3.31}$$

Hence, $\mathcal{Z}(\mathbf{B}; \mathcal{E}_{-k}) \subseteq \mathcal{Z}(\mathbf{E}_{-k})$. Therefore, $\mathcal{Z}(\mathbf{B}; \mathcal{E}_{-k}) = \mathcal{Z}(\mathbf{E}_{-k})$. □

Theorem 3.6.2 indicates that the 1-dimensional curves defined by the zero set of $\mathbf{E}_{-k}(\mathbf{X})$ is identical to the relaxation of the zero set of original $\mathbf{B}(\mathbf{X})$ at a certain direction. This direction is provided by the null space of the first order embedding mapping $\mathcal{E}_{-k}$.

3.7 Encryption Mapping for Multi-Party ACOPF

Recall Theorem 3.3.1 that the set mappings are closed under compositions, thus we can define an encryption mapping which is the composition of affine mapping and linear mapping, for instance $\Gamma^* := \mathcal{L}^* \circ \mathcal{A}^*$, and Γ^* is still an affine algebraic set preserving mapping. In this section, we are going to apply this kind of mappings for each party of the power grid, and show that, although privately obfuscated by each party's encryption mapping Γ^*_α, the overall encrypted problem is equivalent to the unmasked problem.

Consider the equality constrained multi-party ACOPF model in Equation (2.33). The model can be concisely written as

$$\begin{aligned}
&\textbf{Minimize} && \sum_{\alpha=1}^{N_{area}} J_\alpha(\mathbf{X}_\alpha) && \\
&\textbf{Subject to:} && && \\
&\text{Non-sharing:} && \mathbf{F}_\alpha(\mathbf{X}_\alpha) = \ \mathbf{0} && (3.32a)\\
&\text{Compliance:} && \mathbf{G}_{\alpha,\beta}(\mathbf{X}_\alpha) - \mathbf{G}_{\beta,\alpha}(\mathbf{X}_\beta) = \ \mathbf{0} && (3.32b)\\
& && \alpha \ = \ 1, \ldots, N_{area} && \\
& && \beta \ (\neq \alpha) \ = \ 1, \ldots, N_{area} &&
\end{aligned}$$

where the subscript α indicates the α-th party, β indicates the β-th party; $J_\alpha(\mathbf{X}_\alpha)$ is the objective function; $\mathbf{F}_\alpha(\mathbf{X}_\alpha)$ is the constraint set which is not shared with any other parties; $\mathbf{G}_{\alpha,\beta}(\mathbf{X}_\alpha) - \mathbf{G}_{\beta,\alpha}(\mathbf{X}_\beta) = \mathbf{0}$ is the intersection area compliance relation between the α-th party and the β-th party.

Now let's encrypt each party's model by its own mapping Γ^*_α:

$$\Gamma^*_\alpha\Big(J_\alpha(\mathbf{X}_\alpha)\Big) = J_\alpha(\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha) \tag{3.33a}$$

$$\Gamma_{\alpha}^{*}\Big(\mathbf{F}_{\alpha}(\mathbf{X}_{\alpha})\Big) = \mathbf{R}_{\alpha}\mathbf{F}_{\alpha}(\mathbf{T}_{\alpha}\mathbf{Y}_{\alpha} + \gamma_{\alpha}) \tag{3.33b}$$

$$\Gamma_{\alpha}^{*}\Big(\mathbf{G}_{\alpha}(\mathbf{X}_{\alpha})\Big) = \mathbf{G}_{\alpha}(\mathbf{T}_{\alpha}\mathbf{Y}_{\alpha} + \gamma_{\alpha}) \tag{3.33c}$$

where $\mathbf{X}_{\alpha} = \mathbf{T}_{\alpha}\mathbf{Y}_{\alpha} + \gamma_{\alpha}$ is a random invertible affine map; $\mathbf{R}_{\alpha}$ is a random invertible linear combination.

We are supposed to show that these random encryption mappings Γ_{α}^{*} work together, keeping the algebraic set of the original problem unaltered.

From the global view of the entire power grid let's define the variable vector

$$\mathbf{Z} := (\mathbf{X}_1^T,\ \mathbf{X}_2^T,\ \cdots,\ \mathbf{X}_{N_{area}}^T)^T \tag{3.34}$$

Then Equation (3.32) can be written in terms of $\mathbf{Z}$

$$\begin{aligned} \textbf{Minimize} \quad & \sum_{\alpha=1}^{N_{area}} J_{\alpha}(\mathbf{Z}) \\ \textbf{Subject to:} \quad & \begin{bmatrix} \mathbf{F}_1(\mathbf{Z}) \\ \vdots \\ \mathbf{F}_{N_{area}}(\mathbf{Z}) \\ \mathbf{G}_1(\mathbf{Z}) - \mathbf{G}_i(\mathbf{Z}) \\ \vdots \\ \mathbf{G}_{N_{area}}(\mathbf{Z}) - \mathbf{G}_j(\mathbf{Z}) \end{bmatrix} = \mathbf{0} \end{aligned} \tag{3.35}$$

Let's define an affine map $\mathcal{A}$ as follows:

$$\mathcal{A} : \mathbf{Z} = \mathbf{T}\mathbf{W} + \gamma \tag{3.36}$$

where $\mathbf{W} := (\mathbf{Y}_1^T,\ \mathbf{Y}_2^T,\ \cdots,\ \mathbf{Y}_{N_{area}}^T)^T$; $\mathbf{T}$ and γ are specified by

$$\mathbf{T} = \begin{bmatrix} \mathbf{T}_1 & & \\ & \ddots & \\ & & \mathbf{T}_{N_{area}} \end{bmatrix} \tag{3.37}$$

$$\gamma = \begin{bmatrix} \gamma_1, \cdots, \gamma_{N_{area}} \end{bmatrix}^T \tag{3.38}$$

Also define a linear mapping $\mathcal{L}$ with its matrix $\mathbf{R}$ by

$$\mathbf{R} = \begin{bmatrix} 1 & & & & \\ & \mathbf{R}_1 & & & \\ & & \ddots & & \\ & & & \mathbf{R}_{N_{area}} & \\ & & & & \mathbf{I}_G \end{bmatrix} \tag{3.39}$$

where $\mathbf{I}_G$ is the identity matrix with its dimension being the total number of the compliance equations.

Let $\Gamma^* := \mathcal{L}^* \circ \mathcal{A}^*$, and apply Γ^* to all the polynomials in Equation (3.35) we will have

$$\begin{bmatrix} 1 & & & & \\ & \mathbf{R}_1 & & & \\ & & \ddots & & \\ & & & \mathbf{R}_{N_{area}} & \\ & & & & \mathbf{I}_G \end{bmatrix} \begin{bmatrix} \sum_{\alpha=1}^{N_{area}} J_\alpha(\mathbf{TW}+\gamma) \\ \mathbf{F}_1(\mathbf{TW}+\gamma) \\ \vdots \\ \mathbf{F}_{N_{area}}(\mathbf{TW}+\gamma) \\ \begin{pmatrix} \mathbf{G}_1(\mathbf{TW}+\gamma) - \mathbf{G}_i(\mathbf{TW}+\gamma) \\ \vdots \\ \mathbf{G}_{N_{area}}(\mathbf{TW}+\gamma) - \mathbf{G}_j(\mathbf{TW}+\gamma) \end{pmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{\alpha=1}^{N_{area}} J_\alpha(\mathbf{T}_\alpha\mathbf{Y}_\alpha+\gamma_\alpha) \\ \mathbf{R}_1\mathbf{F}_1(\mathbf{T}_1\mathbf{Y}_1+\gamma_1) \\ \vdots \\ \mathbf{R}_{N_{area}}\mathbf{F}_{N_{area}}(\mathbf{T}_{N_{area}}\mathbf{Y}_{N_{area}}+\gamma_{N_{area}}) \\ \mathbf{G}_1(\mathbf{T}_1\mathbf{Y}_1+\gamma_1) - \mathbf{G}_i(\mathbf{T}_i\mathbf{Y}_i+\gamma_i) \\ \vdots \\ \mathbf{G}_{N_{area}}(\mathbf{T}_{N_{area}}\mathbf{Y}_{N_{area}}+\gamma_{N_{area}}) - \mathbf{G}_j(\mathbf{T}_j\mathbf{Y}_j+\gamma_j) \end{bmatrix} \tag{3.40}$$

Since the right hand side of Equation (3.40) coincides with the corresponding equation of Equation (3.33) and the compliance relations retain, it indicates that applying Γ^* to the overall

problem is equivalent to applying each party's encryption to its own part. Since Γ^* is a composition of a linear mapping and an induced affine mapping, it is an affine algebraic set preserving mapping. Consequently, every party's encryption works together will preserve the affine algebraic set of the overall problem.

3.8 Conclusion

In this chapter we first reviewed some basic definitions in algebraic geometry, defined "affine algebraic set preserving mapping", and showed that the affine mappings and linear mappings are both affine algebraic preserving mappings. We proved that there exists a linear mapping that can convert the power flow equations into ellipsoids for power grids without shunt elastance and phase shifting transformers, and also showed a sufficient condition for the ellipsoidal formulation of power grids with shunt elements. For the ACOPF problem, two kinds of first order conditions, the KKT conditions and the Fritz John conditions, were discussed, and an ellipsoidal formulation for the sphere confined Fritz John conditions was introduced. We further discussed an ellipsoidal formulation for a general bounded quadratic constrained quadratic programming problem. Then, we defined ellipsoidal mappings and its convexity. We also examined an equivalent relation between two different 1-dimensional embeddings. Finally, we designed an encryption mapping by the composition of affine mappings and linear mappings for the multi-party ACOPF models, which was shown equivalent to the unmasked problem.

Chapter 4

Branch Tracing Method in Identifying Multiple Solutions for the Power Flow Problem and for the Optimal Power Flow Problem

4.1 Overview

This chapter discusses how to implement the branch tracing method in detail. It introduces core techniques for following curves including the predictor-corrector algorithm, initialization, step length control, bad conditioning control, solution identification and termination criteria. We also discuss how to accelerate the branch tracing method by exploiting the sparse structure of the Jacobian matrix for ACOPF problems. Then, the branch tracing method is applied to the ellipsoidal formulation of the power flow problem to identify multiple real-valued solutions. We report the performance on all the existing cases for which the complete real-valued solution sets are known, and several more examples which have never been completely solved before. After that, the branch tracing method is applied to the ellipsoidal formulation of the eq-FJ conditions of optimal power flow problems to locate multiple local solutions. Two basic searching strategies are introduced, and their performances are compared. We also lead a discussion on how to design advanced tracing strategies to enhance the search performance. A few known numerical examples which fail the SDP relaxation are solved using this method, and a special modified example is presented to show some advantageous features of our proposed method.

4.2 Branch Tracing Method

The branch tracing method [15, 16] can be regarded as one type of continuation method [17] as it applies a parameterization to continuously follow 1-dimensional curves. It differs from the common homotopy continuation method [19, 14] since there is no homotopy construct to trace.

Let's consider a real polynomial system $F(\mathbf{X})$ with N polynomials and N unknowns.

$$F(\mathbf{X}) := \begin{bmatrix} f_1(\mathbf{X}) \\ \vdots \\ f_N(\mathbf{X}) \end{bmatrix} \tag{4.1}$$

where $\mathbf{X} \in \mathbb{R}^N$.

Define a polynomial sub-system $F_i(\mathbf{X})$ by removing the i-th polynomial $f_i(\mathbf{X})$ of Equation (4.1). $F_i(\mathbf{X}) \subset F(\mathbf{X})$. By Theorem 3.3.4, the algebraic set of $F(\mathbf{X})$, denoted by $\mathcal{Z}(F)$, is embedded in the algebraic set of $F_i(\mathbf{X})$, denoted by $\mathcal{Z}(F_i)$.

Generically, suppose $\mathcal{Z}(F)$ consists of finitely many isolated points (0-dimensional) and $\mathcal{Z}(F_i)$ consists of finitely many curves (1-dimensional). The central idea of branch tracing method is to continuously follow those curves in $\mathcal{Z}(F_i)$ for all the i's and enumerate $\mathcal{Z}(F)$. There are two obstacles in this strategy:

1. Some components of $\mathcal{Z}(F_i)$ can be unbounded,
2. $\bigcup_{i=1}^{N} \mathcal{Z}(F_i)$ can be disconected.

The boundedness is guaranteed in our ellipsoidal formulation from Chapter 3, however, the connectedness is still under investigation. In this chapter we only assume every $\mathcal{Z}(F_i)$ is bounded but make no assumptions on the connectedness of the union of $\mathcal{Z}(F_i)$'s. The connectedness will not affect the algorithm. It only provides a theoretical guarantee for enumerating every point in $\mathcal{Z}(F)$.

4.2.1 Predictor-Corrector Algorithm

Let's consider $\mathcal{Z}(F_1)$ for illustration. Instead of removing $f_1(\mathbf{X})$ in Equation (4.1) to obtain subset F_1, we parameterize it by a free parameter α.

$$F_1(\mathbf{X}, \alpha) := \begin{bmatrix} f_1(\mathbf{X}) - \alpha \\ f_2(\mathbf{X}) \\ \vdots \\ f_N(\mathbf{X}) \end{bmatrix} \tag{4.2}$$

Since α is free to change, pick any point $\mathbf{X}_* \in \mathcal{Z}(F_1)$, there always exists an $\alpha_* \in \mathbb{R}$ such that $F_1(\mathbf{X}_*, \alpha_*) = \mathbf{0}$. On the other hand, choose any point $(\mathbf{X}_*, \alpha_*)$ with $F_1(\mathbf{X}_*, \alpha_*) = \mathbf{0}$, $\mathbf{X}_*$ must be in the set of $\mathcal{Z}(F_1)$. Therefore, the 1-dimensional curve $\mathcal{Z}(F_1)$ is geometrically equivalent to the zero set of parameterized polynomial sub-system $F_1(\mathbf{X}, \alpha)$.

4.2.1.1 Initiate the First Step

To initiate the very first step, one must obtain a first known root $\mathbf{X}_0$ to Equation (4.1). This is typically done using routine algorithms, i.e., the Newton's method. Then, it is trivial that $F_1(\mathbf{X}_0, 0) = \mathbf{0}$.

Given the starting point $\mathbf{X}_0$, the first step $\mathbf{X}_1$ is attained by setting α to a small nonzero value α_1 and solve $F_1(\mathbf{X}_1, \alpha_1) = \mathbf{0}$. This is also executed by the Newton's method.

4.2.1.2 Linear Predictor

Based on local information at $\mathbf{X}_0$ and $\mathbf{X}_1$, one can predict the next point $\tilde{\mathbf{X}}_2$ by the secant slope $(\mathbf{X}_1 - \mathbf{X}_0)/(\alpha_1 - 0)$. This is called the linear predictor.

Suppose at the k-th step $\mathbf{X}_k$, $\mathbf{X}_{k-1}$, α_k and α_{k-1} are provided. Then the linear predictor is obtained by

Algorithm 1 Linear Prediction

1: **procedure** PREDICTE NEXT STEP BY LINEAR FUNCTIONS
2: $\Delta s_k \leftarrow ||[\mathbf{X}_k^T \ \ \alpha_k] - [\mathbf{X}_{k-1}^T \ \ \alpha_{k-1}]||$ ▷ Update arc length Δs
3: $\tilde{\mathbf{X}}_{k+1} \leftarrow \mathbf{X}_k + (\mathbf{X}_k - \mathbf{X}_{k-1})\frac{\Delta r}{\Delta s_k}$ ▷ Update $\mathbf{X}$
4: $\tilde{\alpha}_{k+1} \leftarrow \alpha_k + (\alpha_k - \alpha_{k-1})\frac{\Delta r}{\Delta s_k}$ ▷ Update α

In our simulation, the linear predictor is only applied for predicting the second step $\tilde{\mathbf{X}}_2$, and the rest steps are predicted by the quadratic predictor which will be introduced next.

4.2.1.3 Quadratic Predictor

Once the second step $\mathbf{X}_2$ is linearly predicted and corrected (the correction part will be discussed after this subsection), one can implement the quadratic predictor for all the future steps. It typically performs better than the linear predictor because the quadratic predictor function at each entry of $\mathbf{X}_k$ captures the curvature of a curve within a certain range.

Suppose at the k-th step $\mathbf{X}_k$, $\mathbf{X}_{k-1}$, $\mathbf{X}_{k-2}$, α_k, α_{k-1} and α_{k-2} are provided. Then, we apply quadratic function to predict future steps

Algorithm 2 Quadratic Predictor

procedure PREDICT NEXT STEP BY QUADRATIC FUNCTIONS

$\Delta s_k \leftarrow |||[\mathbf{X}_k^T \ \ \alpha_k] - [\mathbf{X}_{k-1}^T \ \ \alpha_{k-1}]||$ ▷ Compute arc length Δs_k

$\Delta s_{k-1} \leftarrow |||[\mathbf{X}_{k-1}^T \ \ \alpha_{k-1}] - [\mathbf{X}_{k-2}^T \ \ \alpha_{k-2}]||$ ▷ Compute arc length Δs_{k-1}

$\Delta s_{k+1} \leftarrow h_{k+1}\Delta r$ ▷ Compute arc length Δs_{k+1}

for $i = 1,\ 2,\ \cdots,\ N$ **do**

$c \leftarrow \mathbf{X}_{k,i}$ ▷ Update the coefficients c, b and a

$$\begin{bmatrix} b \\ a \end{bmatrix} \leftarrow \begin{bmatrix} \Delta s_k & \Delta s_k^2 \\ \Delta s_k + \Delta s_{k-1} & (\Delta s_k + \Delta s_{k-1})^2 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{X}_{k-1,i} - c \\ \mathbf{X}_{k-2,i} - c \end{bmatrix}$$

$\tilde{\mathbf{X}}_{k+1,i} \leftarrow a\Delta s_{k+1}^2 + b\Delta s_{k+1} + c$ ▷ Predict the i-th entry of $\mathbf{X}_{k+1}$

$c \leftarrow \alpha_k$ ▷ Update the coefficients c, b and a

$$\begin{bmatrix} b \\ a \end{bmatrix} \leftarrow \begin{bmatrix} \Delta s_k & \Delta s_k^2 \\ \Delta s_k + \Delta s_{k-1} & (\Delta s_k + \Delta s_{k-1})^2 \end{bmatrix}^{-1} \begin{bmatrix} \alpha_{k-1} - c \\ \alpha_{k-2} - c \end{bmatrix}$$

$\tilde{\alpha}_{k+1} \leftarrow a\Delta s_{k+1}^2 + b\Delta s_{k+1} + c$ ▷ Predict α_{k+1}

Note that h_{k+1} is the step length ratio for controlling the step size. It will be fully discussed in the subsection 4.2.3 of step length control.

4.2.1.4 Corrector

After implementing predictor at each step, a corrector step is applied to force our sequence back on the 1-dimensional curve that we follow. This corrector step is commonly undertaken by the Newton's method, and we refer to it as the "Phase I" corrector. A typical implementation at step $k+1$ would consider solving

$$F_1(\mathbf{X}_{k+1}, \alpha_{k+1}) = 0 \tag{4.3}$$

where $\mathbf{X}_{k+1}$ is unknown but α_{k+1} is given.

Assume that $F_1(\cdot, \alpha_{k+1})$ is non-degenerate at $\mathbf{X}_{k+1}$, by Taylor's expansion we have

$$F_1(\mathbf{X}_{k+1}, \alpha_{k+1}) = F_1(\tilde{\mathbf{X}}_{k+1}, \alpha_{k+1}) + \bigtriangledown_{\mathbf{X}} F_1(\tilde{\mathbf{X}}_{k+1}, \alpha_{k+1})\Delta\mathbf{X}_{k+1} + O(\Delta\mathbf{X}_{k+1}^T\Delta\mathbf{X}_{k+1}) \tag{4.4}$$

Neglecting higher order terms $O(\Delta\mathbf{X}_{k+1}^T\Delta\mathbf{X}_{k+1})$ and substituting (4.3) in (4.4) we have

$$0 = F_1(\tilde{\mathbf{X}}_{k+1}, \alpha_{k+1}) + \bigtriangledown_{\mathbf{X}} F_1(\tilde{\mathbf{X}}_{k+1}, \alpha_{k+1})\Delta\mathbf{X}_{k+1} \tag{4.5}$$

Therefore

$$\Delta\mathbf{X}_{k+1} = -\bigtriangledown_{\mathbf{X}} F_1(\tilde{\mathbf{X}}_{k+1}, \alpha_{k+1})^{-1} F_1(\tilde{\mathbf{X}}_{k+1}, \alpha_{k+1}) \tag{4.6}$$

and

$$\mathbf{X}_{k+1} = \tilde{\mathbf{X}}_{k+1} + \Delta\mathbf{X}_{k+1} \tag{4.7}$$

Note that it may take a few Newton's iterations to make $\tilde{\mathbf{X}}_{k+1}$ converge to a certain tolerance. The number of Newton's iterations can serve as an important index for adjusting the step size which will be discussed in subsection 4.2.3.

To ensure searching efficiency of our branch tracing method and to allow α go backwards automatically, we add one extra equation to Equation (4.2) and thus regard α as an unknown as well.

$$G_1(\mathbf{X}, \alpha) := \begin{bmatrix} F_1(\mathbf{X}, \alpha) \\ (\mathbf{X} - \mathbf{X}_p)^T(\mathbf{X} - \mathbf{X}_p) + (\alpha - \alpha_p)^2 - \Delta r^2 \end{bmatrix} \tag{4.8}$$

where Δr is a constant.

Note that Equation (4.8) has $N+1$ unknowns and polynomials. The newly added polynomial can be interpreted as an analog to a trust region. Instead of setting an inequality region, we set a sphere surface centered at a previous step $(\mathbf{X}_p, \alpha_p)$ with adjustable radius Δr. It can also be interpreted as setting the arc length step size for the curve trace.

Let's consider the Taylor's expansion of (4.8) at step $k+1$:

$$\begin{aligned} G_1(\mathbf{X}_{k+1}, \alpha_{k+1}) &= G_1(\tilde{\mathbf{X}}_{k+1}, \tilde{\alpha}_{k+1}) \\ &+ \begin{bmatrix} \bigtriangledown_{\mathbf{X}} F_1(\tilde{\mathbf{X}}_{k+1}, \tilde{\alpha}_{k+1}) & \bigtriangledown_{\alpha} F_1(\tilde{\mathbf{X}}_{k+1}, \tilde{\alpha}_{k+1}) \\ 2(\tilde{\mathbf{X}}_{k+1} - \mathbf{X}_k)^T & 2(\tilde{\alpha}_{k+1} - \alpha_k) \end{bmatrix} \begin{bmatrix} \Delta\mathbf{X}_{k+1} \\ \Delta\alpha_{k+1} \end{bmatrix} \quad (4.9a) \\ &+ O(\Delta\mathbf{X}_{k+1}^T \Delta\mathbf{X}_{k+1} + \Delta\alpha_{k+1}^2) \quad (4.9b) \end{aligned}$$

By neglecting the higher-order terms $O(\Delta\mathbf{X}_{k+1}^T \Delta\mathbf{X}_{k+1} + \Delta\alpha_{k+1}^2)$ we have

$$\begin{bmatrix} \Delta\mathbf{X}_{k+1} \\ \Delta\alpha_{k+1} \end{bmatrix} = - \begin{bmatrix} \bigtriangledown_{\mathbf{X}} F_1(\tilde{\mathbf{X}}_{k+1}, \tilde{\alpha}_{k+1}) & \bigtriangledown_{\alpha} F_1(\tilde{\mathbf{X}}_{k+1}, \tilde{\alpha}_{k+1}) \\ 2(\tilde{\mathbf{X}}_{k+1} - \mathbf{X}_k)^T & 2(\tilde{\alpha}_{k+1} - \alpha_k) \end{bmatrix}^{-1} G_1(\tilde{\mathbf{X}}_{k+1}, \tilde{\alpha}_{k+1}) \quad (4.10)$$

Thus,

$$\begin{bmatrix} \mathbf{X}_{k+1} \\ \alpha_{k+1} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{X}}_{k+1} \\ \tilde{\alpha}_{k+1} \end{bmatrix} + \begin{bmatrix} \Delta\mathbf{X}_{k+1} \\ \Delta\alpha_{k+1} \end{bmatrix} \quad (4.11)$$

A typical implementation algorithm is provided below, where τ_F and $\tau_x \in \mathbb{R}$ are small constants given for tolerance.

4.2.2 Sparse Structure in Jacobian Matrix for ACOPF

The corrector step is the most computationally intensive part in our tracing method because solving the incremental $\Delta\mathbf{X}_{k+1}$ and $\Delta\alpha_{k+1}$ requires solving a set of linear equations, which in general is expensive. For example, if apply the LU factorization to solve a linear system with a dense $n \times n$ matrix, the computational complexity is $O(n^3)$. However, a special technique can be applied to the ellipsoidal formulation, retaining the sparsity of the Jacobian matrix. Furthermore, a sparse pattern in the Jacobian matrix of the ACOPF problem is observed, and it can be used for accelerating the LU factorization.

Algorithm 3 Phase I Corrector

1: **procedure** NEWTON'S STEP
2: $[\mathbf{L}, \mathbf{U}, \mathbf{P}] \leftarrow lu(-\mathbf{J}_k)$ ▷ LU factorization of $\mathbf{J}_k$
3: $\Delta\mathbf{X}_k \leftarrow \mathbf{L}\backslash\mathbf{U}\backslash\mathbf{P}\ \mathbf{F}(\mathbf{X}_k)$ ▷ Compute incremental $\Delta\mathbf{X}_k$
4: $\mathbf{X}_{k+1} \leftarrow \mathbf{X}_k + \Delta\mathbf{X}_k$ ▷ Compute $\mathbf{X}_{k+1}$
5: **procedure** CHECK CONVERGENCE
6: **if** $|\mathbf{F}(\mathbf{X}_{k+1})| < \tau_F$ and $|\Delta\mathbf{X}_k| < \tau_x$ **then** ▷ Check convergence
7: Break!

4.2.2.1 Retaining Sparsity

First, recall that in Chapter 3 we constructed a base ellipsoid in Equation (3.19) and added the scaled base ellipsoid to all the other equations for the eq-FJ conditions. Therefore, the Jacobian matrix $\mathbf{J}$ of the ellipsoidal formulation of the eq-FJ conditions can be expressed as

$$\mathbf{J} = \mathbf{J}_0 + \mathbf{a}\mathbf{b}^T \tag{4.12}$$

where $\mathbf{J}_0$ is the Jacobian matrix of the original FJ conditions; $\mathbf{b}$ is the gradient of the base ellipsoid; $\mathbf{a}$ is the vector of scaling factors of the base ellipsoid for each equation.

Note that $\mathbf{J}_0$ is sparse due to low degrees of power network at each bus. But $\mathbf{a}\mathbf{b}^T$ is usually a dense matrix because the base ellipsoid is constructed by adding constraints together. Therefore, solving $\Delta\mathbf{X}$ for the following equation is expensive.

$$\mathbf{J}\Delta\mathbf{X} = (\mathbf{J}_0 + \mathbf{a}\mathbf{b}^T)\Delta\mathbf{X} = -\mathbf{F} \tag{4.13}$$

To retain the sparsity of the Jacobian matrix, let's consider an augmented linear system:

$$\begin{bmatrix} \mathbf{J}_0 & \mathbf{a} \\ \mathbf{b}^T & -1 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{X} \\ r \end{bmatrix} = - \begin{bmatrix} \mathbf{F} \\ 0 \end{bmatrix} \tag{4.14}$$

When substituting the second row of (4.14) into the first row, we will recover (4.13). The benefit of (4.14) is that $\mathbf{J}_0$ is sparse. The left plot of Figure 4.1 shows a dense Jacobian matrix for a

39-bus system, while the right plot shows the corresponding augmented sparse Jacobian matrix. Although (4.14) and (4.13) are equivalent, their computational complexity is quite different. Figure 4.2 demonstrates a time report for solving one Newton's step for the dense Jacobian matrix and the corresponding sparse matrix. The linear solver is the Matlab backslash "\" specified with "umfpack" package [96]. One can see that as a system size increases, solving a dense linear system requires increasingly more time than solving a sparse linear system.

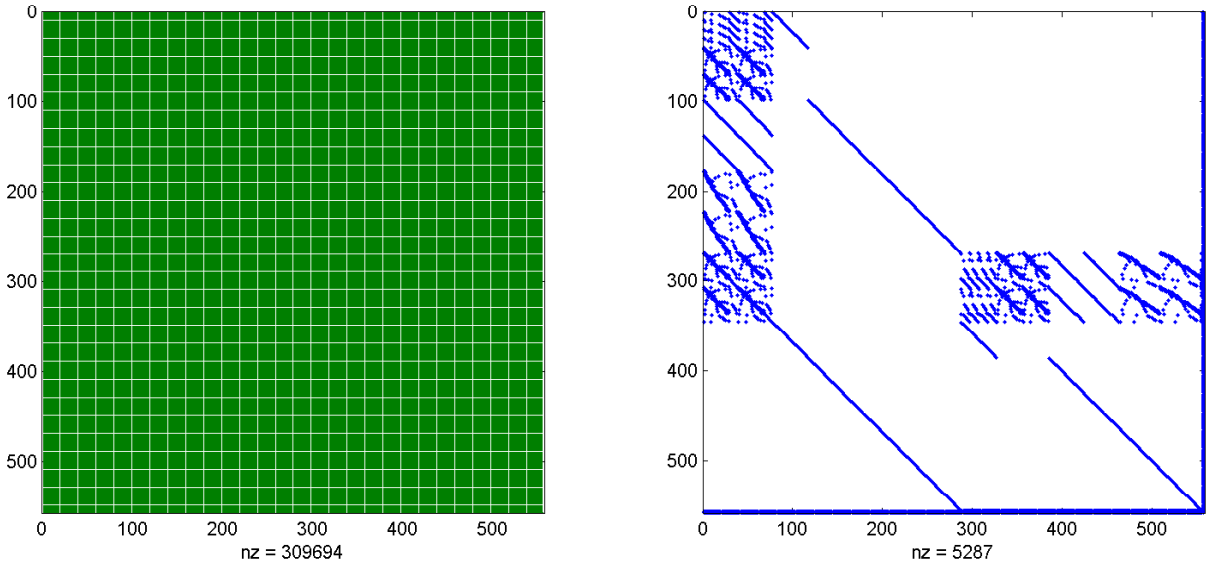


Figure 4.1: Plot of Dense Jacobian Matrix (Left) and Augmented Sparse Jacobian Matrix (Right)

4.2.2.2 Exploring Sparse Structure

One may notice that the sparse Jacobian matrix in the right plot of Figure 4.1 demonstrates a few patterns. Some sub-matrices are very likely to be diagonal; other "dense" parts are very limited and concentrated. A natural thought would be to rearrange the sparse Jacobian matrix to make the nonzero patterns more favorable for solving linear equations. One specific idea is to make the Jacobian matrix as diagonal as possible. A diagonal centralized matrix is very favorable for the

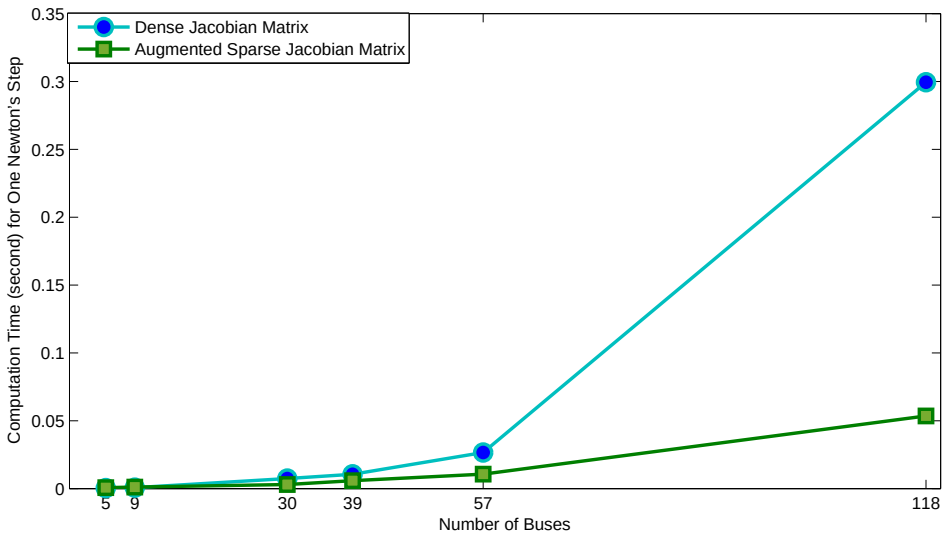


Figure 4.2: Computation Time for Solving One Newton's Step with Dense and Sparse Jacobian Matrices

LU factorization since one can use the diagonal elements to trianglize the matrix by subtracting off-diagonal elements. Base on this particular idea, we rearrange the sphere confined eq-FJ conditions in (3.18) for the linear cost case with line current model.

$$(\mathbf{U}^T\mathbf{M}_{p,i}\mathbf{U} + P_{load,i}) + s^2_{p,max,i} = P_{max,i} \tag{4.15a}$$

$$s^2_{p,min,i} - (\mathbf{U}^T\mathbf{M}_{p,i}\mathbf{U} + P_{load,i}) = -P_{min,i} \tag{4.15b}$$

$$\mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} + s^2_{q,max,i} = Q_{max,i} \tag{4.15c}$$

$$s^2_{q,min,i} - \mathbf{U}^T\mathbf{M}_{q,i}\mathbf{U} = -Q_{min,i} \tag{4.15d}$$

$$\mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} + s^2_{v,max,k} = V^2_{max,k} \tag{4.15e}$$

$$s^2_{v,min,k} - \mathbf{U}^T\mathbf{M}_{v,k}\mathbf{U} = -V^2_{min,k} \tag{4.15f}$$

$$\mathbf{U}^T\mathbf{M}_{Iin,n}\mathbf{U} + s^2_{I,max,in,n} = I^2_{max,in} \tag{4.15g}$$

$$\mathbf{U}^T\mathbf{M}_{Iout,n}\mathbf{U} + s^2_{I,max,out,n} = I^2_{max,out} \tag{4.15h}$$

$$\bigtriangledown_{\mathbf{S}}\mathcal{F}(\mathbf{U},\mathbf{S},\lambda,\delta) = 0 \tag{4.15i}$$

$$\mathbf{U}^T\mathbf{M}_{p,m}\mathbf{U} = -P_{load,m} \tag{4.15j}$$

$$\mathbf{U}^T\mathbf{M}_{q,m}\mathbf{U} = -Q_{load,m} \tag{4.15k}$$

$$\bigtriangledown_{\mathbf{U}}\mathcal{F}(\mathbf{U},\mathbf{S},\lambda,\delta) = 0 \tag{4.15l}$$

$$\delta^2 + \lambda^T\lambda = 1 \tag{4.15m}$$

where $\mathbf{S}$ represents the slack variable vector.

We also rearrange the decision variables as follow

$$\mathbf{X} := [\mathbf{S}^T\ \lambda_{Pmax}^T\ \lambda_{Pmin}^T\ \lambda_{Qmax}^T\ \lambda_{Qmin}^T\ \lambda_{Vmax}^T\ \lambda_{Vmin}^T\ \ \lambda_{If}^T\ \ \lambda_{It}^T\ \ \lambda_P^T\ \ \lambda_Q^T\ \ \mathbf{U}^T\ \ \delta]^T \tag{4.16}$$

where $\lambda_\cdot$'s are multipliers associated with subscript-corresponding constraints, subscript $Pmax$ represents active power upper bound, $Pmin$ represents active power lower bound, $Qmax$ represents reactive power upper bound, $Qmin$ represents reactive power lower bound, $Vmax$ represents voltage magnitude upper bound, $Vmin$ represents voltage magnitude lower bound, If represents line current upper bound at "from" side, It represents line current upper bound at "to" side, P represents PQ bus active power balance, and Q represents PQ bus reactive power balance.

After the rearrangements of (4.15) and (4.16), the Jacobian matrix is more favorable for the LU factorization. The left plot of Figure 4.3 is the original Jacobian matrix for the 39-bus system, while the right plot is the corresponding permutated Jacobian matrix.

Solving a Newton's step with permutated Jacobian matrix is approximately two times faster than that with the original Jacobian matrix. Figure 4.4 illustrates a time report for solving one Newton's step for the original sparse Jacobian matrix and the corresponding permutated Jacobian matrix. The linear solver for the original sparse Jacobian matrix is the Matlab backslash "\" specified with "umfpack" package [96], while for the permutated Jacobian we use the Matlab LU factorization routine "$[U, L, P] = lu(J, thresh)$" with $thresh = 0.01$. The execution time for the permutated Jacobian shown pink in Figure 4.4 includes both the LU factorization time and the backward substitution time.

From the dense Jacobian matrix to the permutated sparse Jacobian matrix, the acceleration rate for solving a single Newton's step is depicted in Figure 4.5, where the rate is calculated by the time ratio T_{dense}/T_{perm}.

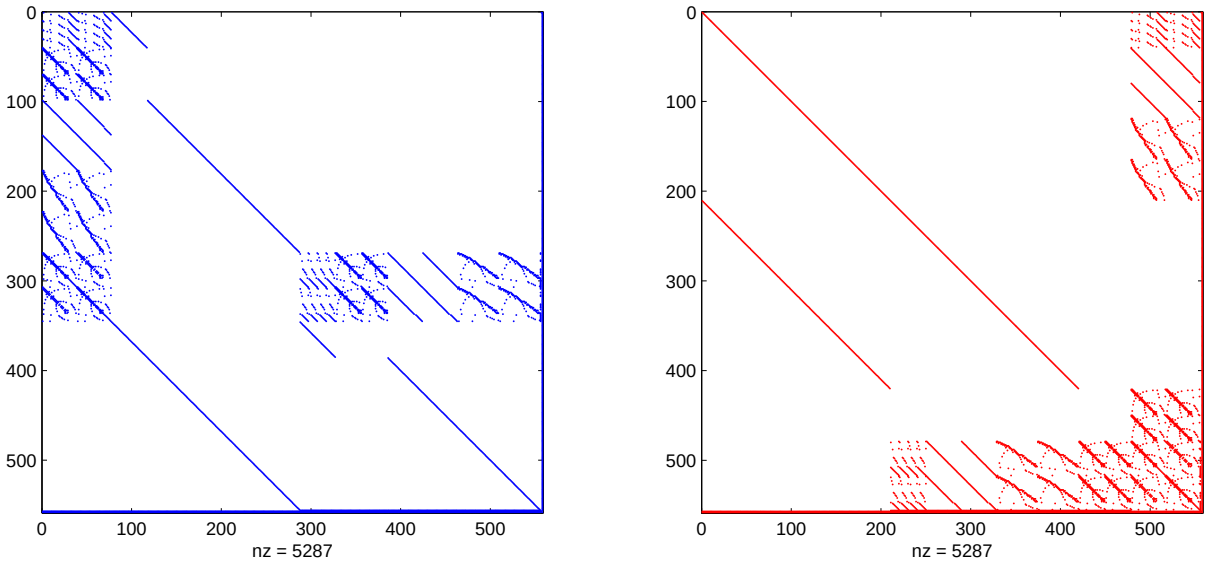


Figure 4.3: Plot of Original Jacobian Matrix (Left) and Permutated Jacobian Matrix (Right)

4.2.3 Step Length Control

In Algorithm 2 we introduced the step ratio h_{k+1}. This section will explain the details.

The purpose of the step length ratio h_{k+1} is to control the step size such that our prediction is reliable as well as progressive. The reliability is characterized by the number of the Newton's iterations for each correction step. Specifically, a target number of convergent iterations is set for reference. If at a certain step the Newton's method takes more iterations than the target number, we reduce the step ratio. If the Newton's method takes less iterations than the target number, we increase the step ratio until the iteration number equals the target number. This process is proposed in Algorithm 4.

Given the target number t_0 (we usually use $t_0 = 3$) the minimal ratio h_{min} and the maximal ratio h_{max}, the step length control algorithm is

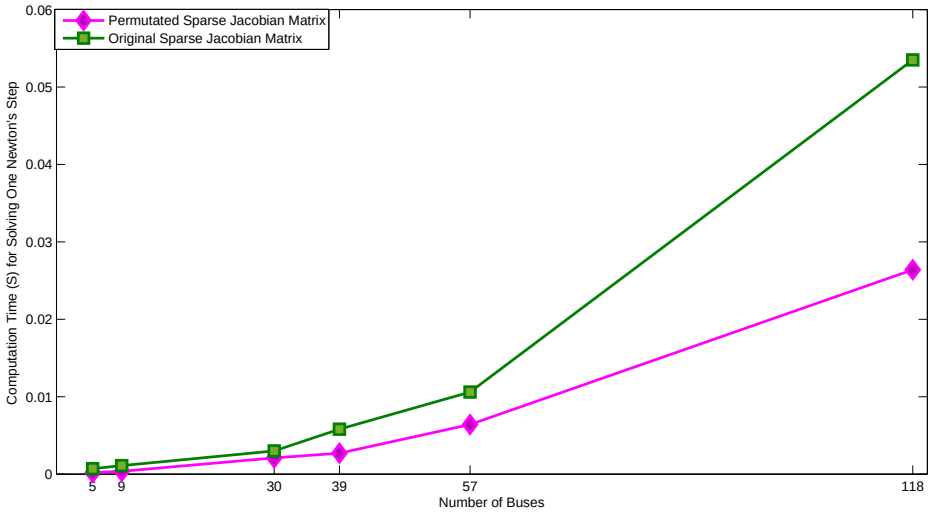


Figure 4.4: Computation Time for Solving One Newton's Step with Permutated and Original Sparse Jacobian Matrices

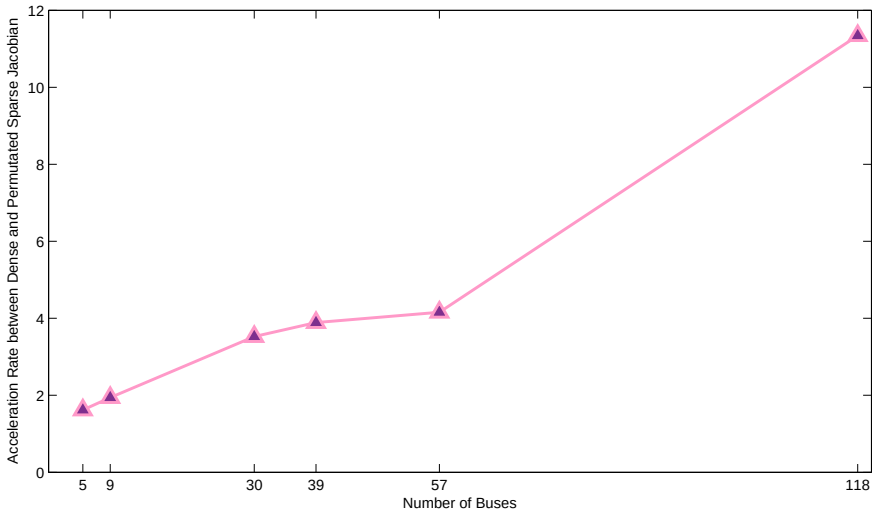


Figure 4.5: Acceleration Rate for Solving One Newton's Step between Dense and Permutated Sparse Jacobian Matrices

4.2.4 Bad Conditioning Control

Recall (4.2) that our 1-dimensional curves are tracing using free variable α. As α continuously changes, the curve we are following may encounter singular or near singular points. For a

Algorithm 4 Step Length Control

1: **procedure** CONTROL STEP LENGTH
2: Execute corrector step, terminates at iteration i ▷ Obtain the iteration number i
3: $h_{k+1} \leftarrow h_k(1 + (t_0 - i)/100)$ ▷ Update h
4: $h_{k+1} \leftarrow min\{h_{k+1},\ h_{max}\}$ ▷ Upper bound h
5: $h_{k+1} \leftarrow max\{h_{k+1},\ h_{min}\}$ ▷ Lower bound h

linearly independent polynomial vector $F_1(\mathbf{X}, \alpha)$, its singular set is generically finite. Thus, the predictor-corrector algorithm can almost always skip over such singular points. A more troublesome difficulty comes from a possible bad conditioning neighborhood around a singular point, or just a long arc of badly conditioned curve. For example, Figure 4.6 shows two sample curves we obtained from our simulation on a 39-bus system. On the left is the multiplier associated with the active power balance equation at bus 1; on the right is the multiplier associated with the reactive power balance equation at bus 1. It is common for multipliers to have sharp changes and steep arcs in their 1-dimensional curves. One may mistakenly regard the steep arcs in Figure 4.6 as non-smooth pulses, but they are actually smooth curves. Figure 4.7 zooms in these pulse-like arcs and shows that each of them represents two smooth but steep peaks. Their slopes are around 3×10^9. If not treated properly, such long arcs with bad conditioning will make the Newton's step inaccurate which further results in a tiny step size that makes the tracing progress slowly, or even fails to converge within a given tolerance.

To overcome this numerical difficulty, we introduce "Phase II" algorithm aiming at dealing with badly conditioned situations. The basic idea behind this algorithm is to rescale decision variables for Newton's iterations. An appropriate rescaling of variables can draw the very steep slopes back to a given range, which will enhance the numerical stability for solving Newton's steps.

In Algorithm 5 $condn \in \mathbb{R}$ is a given threshold for assessing good conditioning[1], $B_0 \in \mathbb{R}$ is a given constant for assessing bad conditioning[2].

[1] In our simulations we usually take $condn = 2 \times 10^5$.
[2] In our simulations we usually take $B_0 = 0.022 \times length(\mathbf{X})$.

Algorithm 5 Phase II Corrector for Bad Conditioning

procedure COMPUTE CONDITIONING

$rY \leftarrow abs(\frac{\mathbf{X}_k - \mathbf{X}_{k-1}}{\alpha_k - \alpha_{k-1}})$ ▷ Compute secant slope for each entry of $\mathbf{X}$ with respect to α

$Hd \leftarrow rY > condn$ ▷ Identify the index of slope greater than $condn$

$Bad \leftarrow Hd^T log_{10}(rY/condn)$ ▷ Compute bad scaling values in log

if $phase2 = 0$ **then** ▷ Check starting iteration

$\mathbf{T} \leftarrow diag\Big(Hd \odot rY/condn + (\mathbf{1} - Hd)\Big)$ ▷ Compute rescaling matrix

$phase2 \leftarrow phase2 + 1$ ▷ Count Phase II numbers

procedure RESCALE DECISION VARIABLES AND JACOBIAN MATRIX

$\hat{\mathbf{X}}_k \leftarrow \mathbf{T} \backslash \mathbf{X}_k$ ▷ Rescale decision variables

$\hat{\mathbf{J}}_k \leftarrow \mathbf{J}_k \mathbf{T}$ ▷ Rescale Jacobian matrix

procedure NEWTON'S STEP

$[\hat{\mathbf{L}}, \hat{\mathbf{U}}, \hat{\mathbf{P}}] \leftarrow lu(-\hat{\mathbf{J}}_k)$ ▷ LU factorization of $\hat{\mathbf{J}}_k$

$\Delta\hat{\mathbf{X}}_k \leftarrow \hat{\mathbf{L}} \backslash \hat{\mathbf{U}} \backslash \hat{\mathbf{P}}\ \mathbf{F}(\mathbf{X}_k)$ ▷ Compute incremental $\Delta\hat{\mathbf{X}}_k$

$\mathbf{X}_{k+1} \leftarrow \mathbf{T}(\hat{\mathbf{X}}_k + \Delta\hat{\mathbf{X}}_k)$ ▷ Compute $\mathbf{X}_{k+1}$

procedure CHECK CONVERGENCE AND CONDITIONING

if $|\mathbf{F}(\mathbf{X}_{k+1})| < \tau_F$ and $|\Delta\hat{\mathbf{X}}_k| < \tau_x$ **then** ▷ Check convergence

if $Bad \leq B_0$ **then** ▷ Check conditioning

$phase2 \leftarrow 0$

Break!

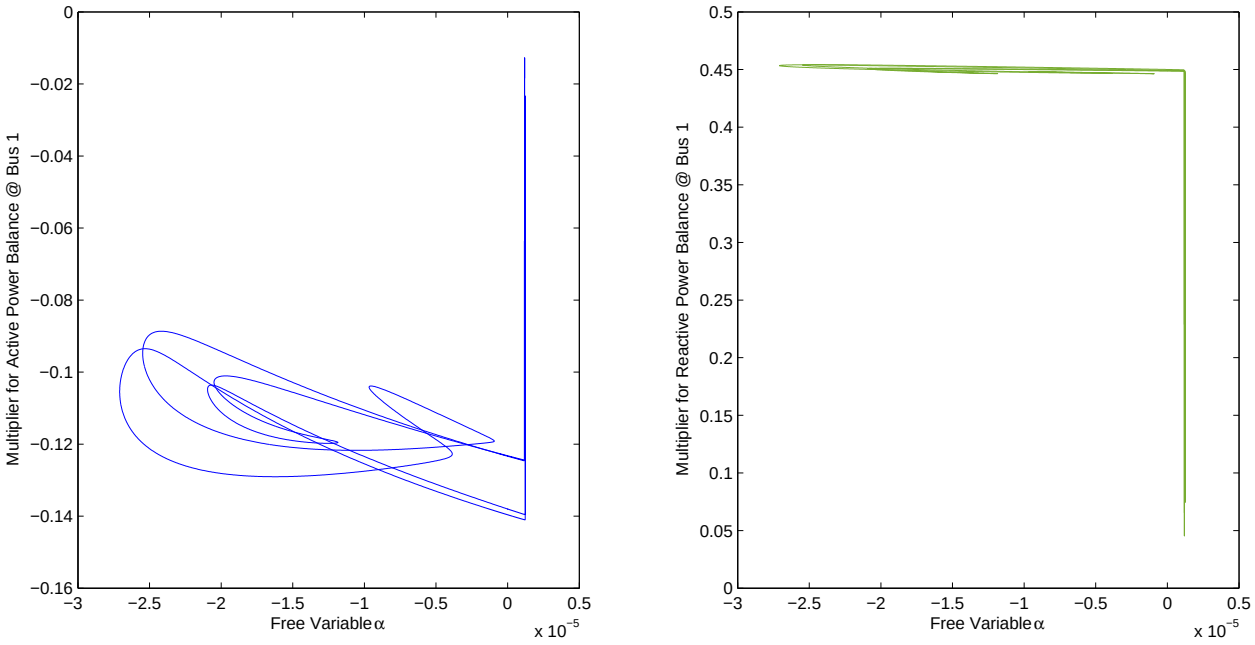


Figure 4.6: Multipliers for Active Power Balance Constraint (Left) and for Reactive Power Balance Constraint (Right) at Bus 1

4.2.5 Solution Identification and Termination Criteria

Recall Equation (4.2) that we add one parameter α to the original polynomial system Equation (4.1). Along with our trace, each time α across 0 indicates we find a solution to Equation (4.1). By comparing the solution found to the starting point we can determine if our trace returns to the starting point.

Note that ϵ is a given small number.

4.2.6 Summary

This section explained how to implement the branch tracing method in detail. It addressed several important techniques:

1. How to initiate tracing,

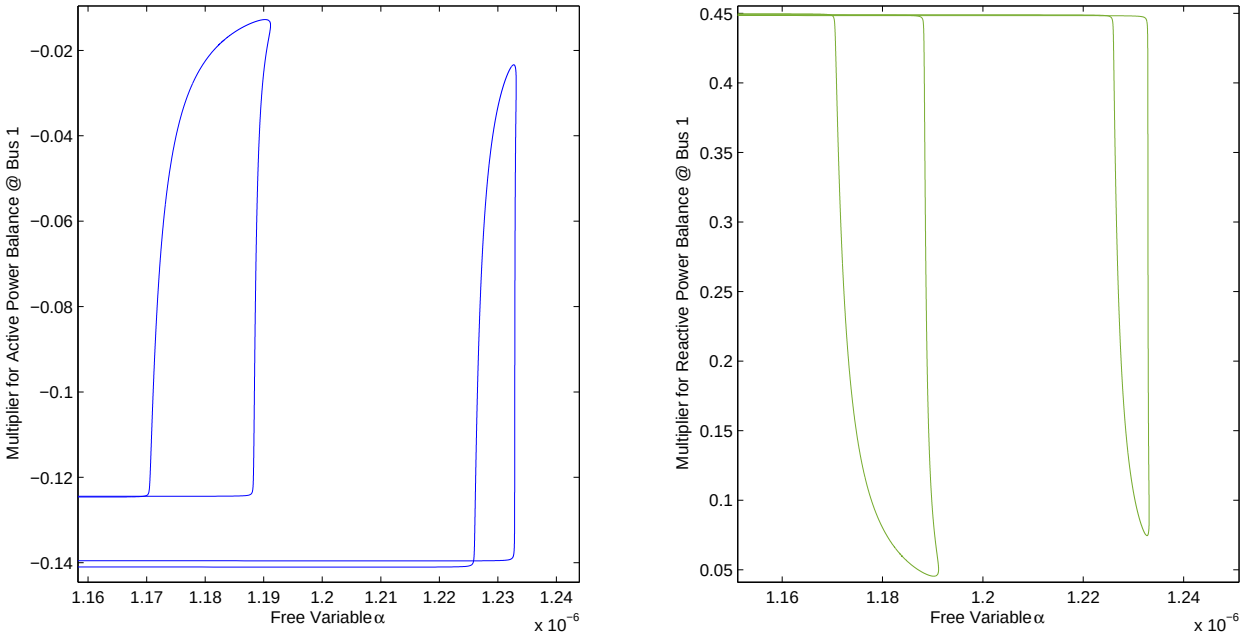


Figure 4.7: Zoom of Multipliers for Active Power Balance Constraint (Left) and for Reactive Power Balance Constraint (Right) at Bus 1

Algorithm 6 Obtain Solutions and Check Termination

1: **procedure** OBAIN SOLUTIONS
2: **if** $\alpha_k \alpha_{k-1} \leq 0$ **then** ▷ Indicates solution crossing
3: $\alpha_k \leftarrow 0$ ▷ Update α_k at the solution
4: Correct $\mathbf{X}_k$ ▷ Find the solution
5: **procedure** CHECK TERMINATION
6: **if** $|\mathbf{X}_k - \mathbf{X}_0| \leq \epsilon$ **then** ▷ Stopping Criteria
7: Break! ▷ Terminate current trace

2. How to predict the next step and correct prediction,

3. How to retain sparsity of Jacobian matrix,

4. How to explore specific sparse structure for further accelerating tracing,

5. How to control step length,

6. How to control bad conditioning,

7. How to identify new solutions and when to terminate tracing.

Equipped with the above algorithms and techniques, the overall tracing procedure is summarized in Algorithm 7. It will serve as a technical support for the following sections to identify

Algorithm 7 Overall Branch Tracing Procedure

procedure OBAIN THE STARTING POINT
    Run routine algorithm to obtain a starting point $\mathbf{X}_0$ ▷ Obtain the starting point
procedure INITIATE THE FIRST STEP
    Send α to α_1 and correct $\mathbf{X}_1$ ▷ Compute the first step
procedure INITIATE THE SECOND STEP
    Algorithm 1 ▷ Linearly predict the second step
    Algorithm 2 ▷ Quadratically predict the next step
    if Numerically Stable **then**
        Algorithm 3 ▷ Call Phase I Corrector for Newton's method
    else
        Algorithm 5 ▷ Call Phase II Corrector for rescaling and Newton's method
procedure COMPUTE THE NEXT STEP
    Algorithm 2 ▷ Quadratically predict the next step
    if Numerically Stable **then**
        Algorithm 3 ▷ Call Phase I Corrector for Newton's method
    else
        Algorithm 5 ▷ Call Phase II Corrector for rescaling and Newton's method
    Algorithm 4 ▷ Control the step length
    Algorithm 6 ▷ Obain solutions and check termination criteria

multiple real-valued solutions to the power flow problems and to locate multiple local solutions to the optimal power flow problems.

4.3 Identify Multiple Real-Valued Solutions to the Power Flow Problem

The first application of our algebraic set preserving mapping relates to finding multiple real-valued solutions of the power flow problem. Though without theoretical guarantee at present, our simulation currently identified the complete solutions sets to all the existing test cases for which the entire solution sets are known. We also report on solution sets of several other standard test cases which have never been completely solved before. Finally, we introduce three modified power flow examples to show that

1. The number of real solutions to the power flow problem can increase temporarily with an increasing active power load,

2. The number of real solutions to the power flow problem can increase temporarily with an increasing reactive power load,

3. A power flow model with PQ buses can also have multiple high-voltage solutions.

This section is organized as follows. First, we introduce how to construct the ellipsoidal formulation for the power flow equations. Then, an enumeration search strategy is explained. Finally, we report simulation results on a lot of benchmark systems.

4.3.1 Computation Procedure and Algorithms

The computation procedure for identifying multiple solutions to the power flow problem consists of three sub-blocks: finding a starting point, constructing ellipsoids and searching for multiple solutions.

The first sub-block calls Newton's method (or similar) to provide a starting solution to the power flow problem. In this thesis we use the Matpower 5.0 package under default settings to return an initial solution. The second sub-block is for constructing an ellipsoidal formulation of

Algorithm 8 Overall Procedure for Solving Power Flow Solutions

1: **procedure** SUB-BLOCKS
2: Finding A Starting Solution ▷ Newton's method or the similar
3: Constructing Ellipsoids ▷ Check and construct ellipsoidal formulation
4: Searching Solutions ▷ Follow 1-d curves

the power flow equations. The third sub-block is implemented by the branch tracing method from Section 4.2.

4.3.1.1 Constructing an Ellipsoidal Formulation

Chapter 3 addressed an existence condition for a particular type of base ellipsoid. If a base ellipsoid exists, we continue to construct the ellipsoidal formulation for the power flow problem; otherwise we cease the program. (The base ellipsoids exist for all standard cases we examined.)

Given a power grid with N_{bus} buses and N_{gen} generators. Without loss of generality, let the first bus be the slack bus, and the second bus to the N_{gen}-th bus be the generator buses. If Condition 1 of Theorem 3.4.5 is satisfied, let

$$\mathbf{M}_{base}(\gamma) := \gamma \sum_{i=1}^{N_{gen}} \mathbf{M}_{v,i} - \sum_{j=N_{gen}+1}^{N_{bus}} \mathbf{M}_{q,j}. \tag{4.17}$$

If Condition 2 of Theorem 3.4.5 is satisfied, let

$$\mathbf{M}_{base}(\gamma) := \gamma \sum_{i=1}^{N_{gen}} \mathbf{M}_{v,i} + \sum_{j=N_{gen}+1}^{N_{bus}} \mathbf{M}_{p,j} \tag{4.18}$$

.

The procedure to construct the ellipsoidal formulation is summarized in Algorithm 9. The constant γ can be chosen in different ways, but will not affect the construction of ellipsoids. To control bad conditioning, we start to try a small value of γ_0. If it fails to provide an base ellipsoid, we gradually increase γ_0 until reaching a base ellipsoid. However, if γ is beyond a certain large value, we stop the program and print "No base ellipsoid generated within a certain range." If a base ellipsoid is successfully generated, we may choose to revise the base ellipsoid by combining

other power flow equations into the base ellipsoid. This procedure proves particularly useful when solving hard power flow problems such as "Molzahn Lesieutre 5-bus" [18]. It helps 1-dimensional curves link with each other and returns a complete real solution set for "Molzahn Lesieutre 5-bus". It indicates that the construction of base ellipsoid can influence the connectivity of our 1-dimensional curves. How to design an ellipsoidal formulation to ensure connectedness of these curves is still elusive, and requires further investigation.

Once the base ellipsoid is generated, one can scale it and add it to each power flow equation to make an ellipsoidal formulation. The constructing algorithm is provided in Algorithm 9.

4.3.1.2 Enumeration Search Strategy

Starting with an initial solution, we apply the branch tracing method in Algorithm 7 to follow every 1-dimensional curve defined by $N_{bus} - 1$ ellipsoids out of total N_{bus} ellipsoids. We will set newly found solutions as different starting points and follow the curves again until no more new solutions are found. This recursive enumeration search strategy is shown in Algorithm 10.

4.3.2 Numerical Examples

In this subsection we first apply Algorithm 10 to all the cases for which the entire solution sets are known. The results demonstrate that the proposed algorithm finds all the real solutions for all these test cases [89]. Next, we test the proposed algorithm on a few more benchmark systems, the solution sets of which have never been reported. Then, we introduce a particular 7-bus example to show how the number of real solutions increases as load increases. Finally, we use a 5-bus example with PQ buses to illustrate that there can exist multiple high-voltage solutions.

4.3.2.1 Existing Solved Power Flow Test Cases

The existing cases for which the solution sets are known include the Tavara and Smith's 3-bus system [11], Bailleuil and Brynes's 4-bus system [12], Molzahn and Lesieutre's 5-bus system [18], Salam's 5-bus and 7-bus systems [14] and the IEEE 14-bus system [97].

Algorithm 9 Constructing Ellipsoids For Power Flow Equations

Check existence of base ellipsoid ▷ Check Theorem 3.4.5
if Yes **then**
    procedure CONSTRUCTING BASE ELLIPSOID
        $\gamma \leftarrow 1$ ▷ Initialize the scaling number
        $\mathbf{M}_{base} \leftarrow \mathbf{M}_{base}(\gamma)$ ▷ Initialize the base ellipsoid matrix
        $\lambda_{min} \leftarrow -1$ ▷ Initialize the smallest eigenvalue
        while $\lambda_{min} \leq 0$ or $b_{base} \leq 0$ **do**
            $\gamma \leftarrow 2\gamma$ ▷ Update the scaling factor
            $\mathbf{M}_{base} \leftarrow \mathbf{M}_{base}(\gamma)$ ▷ Update the base ellipsoid matrix
            $b_{base} \leftarrow b_{base}(\gamma)$ ▷ Update the base ellipsoid constant
            λ_{min} ▷ Compute the minimum eigenvalue of $\mathbf{M}_{base}$
    procedure REVISE BASE ELLIPSOID
        $\lambda_{min} \leftarrow -1$ ▷ Initialize the smallest eigenvalue
        $k \leftarrow 0$ ▷ Initialize the scaling factor
        while $\lambda_{min} \leq 0$ or $b_{base} \leq 0$ **do**
            $k \leftarrow k + 1$ ▷ Update the scaling factor
            $\mathbf{M}_{base} \leftarrow k\mathbf{M}_{base} + \sum_{i=1}^{N_{bus}} \mathbf{M}_i$ ▷ Update the base ellipsoid matrix
            $b_{base} \leftarrow kb_{base} + \sum_{i=1}^{N_{bus}} b_i$ ▷ Update the base ellipsoid constant
            λ_{min} ▷ Compute the minimum eigenvalue of $\mathbf{M}_{base}$
    procedure CONSTRUCTING ELLIPSOIDS
        for $i = 1,\ 2,\ \cdots,\ N_{bus}$ **do**
            $\lambda_{i,min} = -1$ ▷ Initialize the smallest eigenvalue
            while $\lambda_{i,min} \leq 0$ or $b_i \leq 0$ **do**
                $\mathbf{M}_i \leftarrow \mathbf{M}_i + \mathbf{M}_{base}$ ▷ Update the i-th equation matrix
                $b_i \leftarrow b_i + b_{base}$ ▷ Update the i-th ellipsoid constant
                $\lambda_{i,min}$ ▷ Compute the minimum eigenvalue of $\mathbf{M}_i$
else
    Stop!

Algorithm 10 Enumeration Search For Power Flow Problems

1: **procedure** SEARCH POWER FLOW SOLUTIONS
2: $S \leftarrow x_1$ ▷ Initialize the solution set
3: $N_{solu} \leftarrow Count(S)$ ▷ Initialize the number of solutions
4: $k \leftarrow 1$ ▷ Initialize counting number
5: **while** $k \neq N_{solu}$ **do**
6: $k \leftarrow k + 1$ ▷ Update counting number
7: $x_0 \leftarrow x_k$ ▷ Update starting solution
8: **for** $i = 1,\ 2,\ \cdots,\ N_{bus}$ **do**
9: Algorithm 7 ▷ Trace the curve defined by relaxing the i-th equation
10: **return** S_{new} ▷ Return the newly found solutions
11: $S \leftarrow S \cup S_{new}$ ▷ Update the solution set
12: $N_{solu} \leftarrow Count(S)$ ▷ Update the number of the solutions

The simulation results are listed in the Table 4.1. We highlight the efficiency of our proposed method by comparing the number of traces we ran to the number of traces needed for solution-bound-based homotopy continuation methods. It can be seen from Table 4.1 that the number of traces required for solution-bound-based homotopy method is tremendous. Unless an extreme sharp bound can be found, solution-bound-based methods are less practical than the branch tracing method to calculate multiple solutions for large scale power systems. Figure 4.8 presents all the solutions and their linking curves for the "Molzahn Lesieutre 5-bus" projected onto a 3-dimensional subspace. The equation-solution relations for this test case is provided in Table 4.2.

4.3.2.2 Newly Solved Power Flow Test Cases

Here we provide a few more power systems most of which have never been completely solved before: case3, case4gs [98], case6ww [99], case9, IEEE 30-bus system [100], case33bw [101],

Table 4.1: Summary of Existing Solved Power Flow Test Cases

Cases	# Solutions	# Traces	Bailleuil Brynes's Bound	Bezout's Bound
Tavara Smith 3-bus	6	8	6	16
Bailleuil Brynes 4-bus	12	18	20	64
Molzahn Lesieutre 5-bus	10	35	70	256
Salam 5-bus	10	28	70	256
Salam 7-bus	4	21	924	4096
IEEE 14-bus	30	297	10400600	67108864

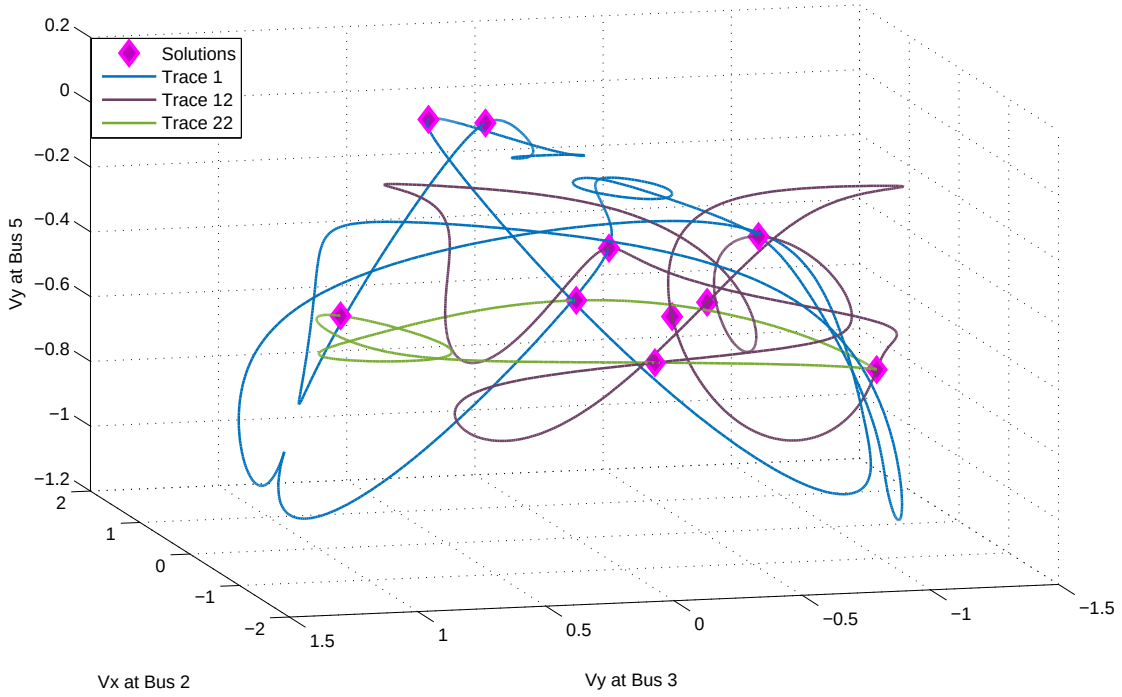


Figure 4.8: Solutions and Linking Curves for Molzahn Lesieutre 5-bus System

IEEE 39-bus system [102] and IEEE 118-bus system[3] [103]. A comparison of the number of real solution, number of traces and complex solution bounds is presented in Figure 4.9.

[3]This case hasn't been completely enumerated, but only partly followed for its curves.

Table 4.2: Equation-Solution Relation for Molzahn Lesieutre 5-bus Example

	Traces by Equations								
Solution #	Eqn 1	Eqn 2	Eqn 3	Eqn 4	Eqn 5	Eqn 6	Eqn 7	Eqn 8	Eqn 9
1	1	2	3	4	5	6	7	8	9
2	1	2	3	4	5	6	7	8	9
3	1	10	11	12	13	14	15	16	17
4	1	10	11	12	13	14	15	16	17
5	18	19	20	12	21	22	23	24	25
6	18	19	20	12	21	26	27	24	25
7	18	19	20	12	28	26	27	24	25
8	18	19	20	12	28	22	23	24	25
9	29	19	30	31	32	22	33	34	35
10	29	19	30	31	32	22	33	34	35

Table 4.3: Summary of Newly Solved Power Flow Test Cases

Cases	# Solutions	# Traces	Bailleuil Brynes's Bound	Bezout's Bound
case3	6	8	6	16
case4gs	6	12	20	64
case6ww	6	25	252	1024
case9	8	51	12870	65536
IEEE 30-bus	472	9780	5.9×10^{16}	2.88×10^{17}
case33bw	16	365	1.83×10^{18}	1.84×10^{19}
IEEE 39-bus	176	5270	6.89×10^{21}	7.56×10^{22}
IEEE 57-bus	606	29142	3.91×10^{32}	5.19×10^{33}
IEEE 118-bus	20000+	?	1.44×10^{69}	2.76×10^{70}

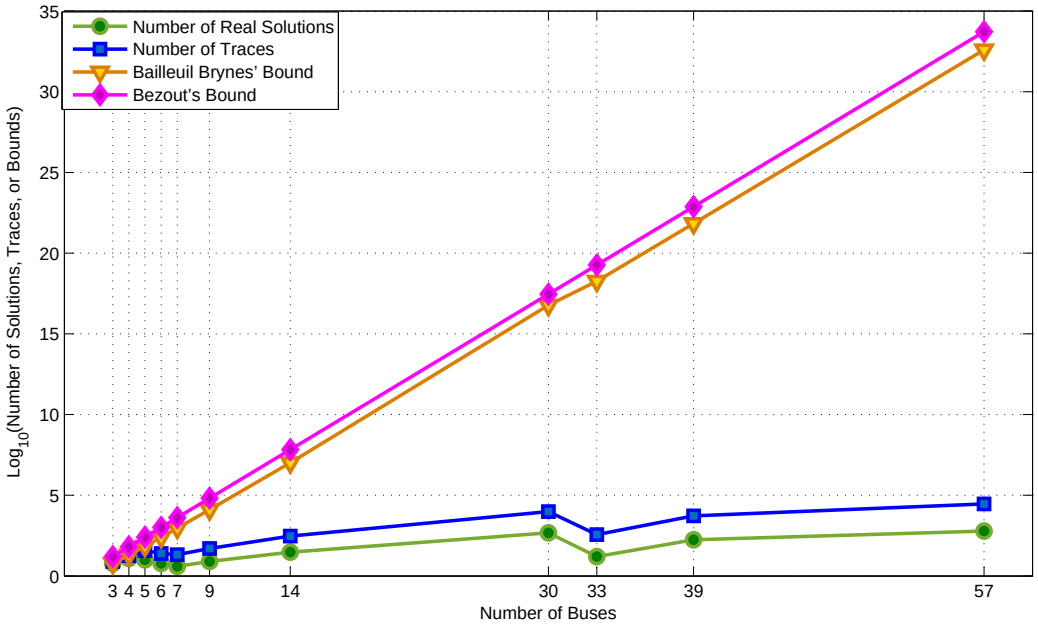


Figure 4.9: Comparison of Number of Real Solutions, Number of Traces, Bailleuil Brynes's Bound and Bezout's Bound in Logarithm

For small test cases such as "case3", "case4gs", "case6ww", and "case9", a traditional homotopy continuation method or a Grobner bases method can solve them completely with a tolerable execution time. Figure 4.10 makes a comparison of execution time required to a few small systems for both the proposed method and the homotopy continuation method. The proposed method was coded in Matlab and executed on a PC with 2-core i7 2800M Hz CPU and 4Gb RAM. The homotopy continuation method was implemented by the existing solver "PHCpack" [104] on the same PC. The "PHCpack" ran over 16,000 seconds to solve "case9" system, and required more than several days for IEEE 14-bus system. For a larger test case, say, 30-bus system, it becomes difficult to solve with limited computation resources. For example, consider the 30-bus system. We report 472 real-valued solutions by tracing only 9780 curves, however, the homotopy continuation method requires as many as 10^{16} many traces to enumerate all the complex-valued solutions. For the 39-bus system we find 176 real-valued solutions by tracing 5270 curves, but the number of complex solutions is as many as 10^{21}. It is worth of mentioning that the real-valued solutions

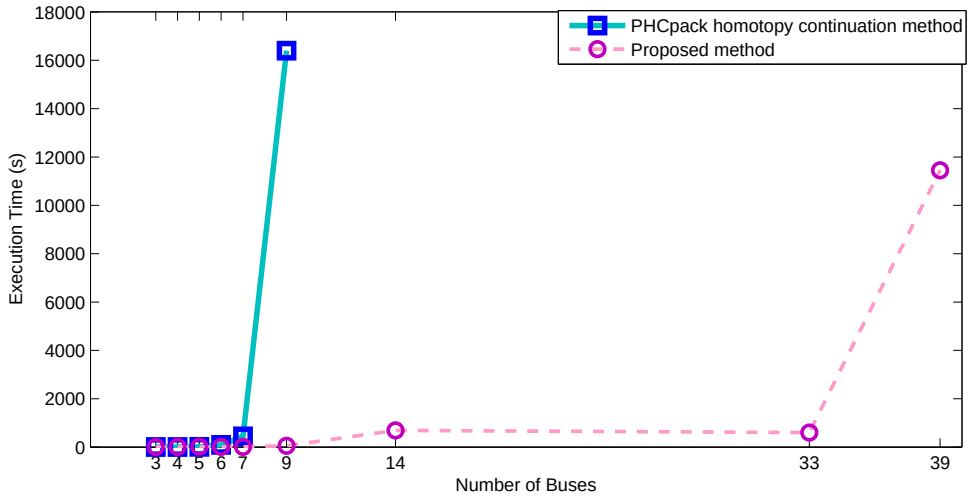


Figure 4.10: Comparison of Execution Time for Proposed Method and Homotopy Continuation Method

seem to occupy a very small proportion of the complex-valued solutions. Power system special structures may play an important role in limiting the number of real solutions. However, no validated theoretical results can be found at present to explain this phenomenon. Figure 4.11 depicts the ratio between the number of real solutions and the number of complex solutions. It empirically indicates that the ratio asymptotically goes to zero as the number of buses goes to infinity.

4.3.3 Number of Real Solutions by Change of Load

According to the PV-curves and QV-curves, one may observe that beyond a certain load level a power grid can lose its solution. On the other hand, the "Tavara Smith 3-bus" case shows that with PV buses, zero load and lossless transmission lines a power grid can have multiple high-voltage solutions. So a natural question would be, whether the number of real solutions to the power flow problem decreases monotonically[4] with the increase of load. In this subsection we provide a negative answer to this question by two counter examples. These examples are solved both by the proposed method and by homotopy continuation methods from *PHCpack* [104] and *Bertini* [105].

[4]For "decrease monotonically" we mean decrease but not strictly decrease.

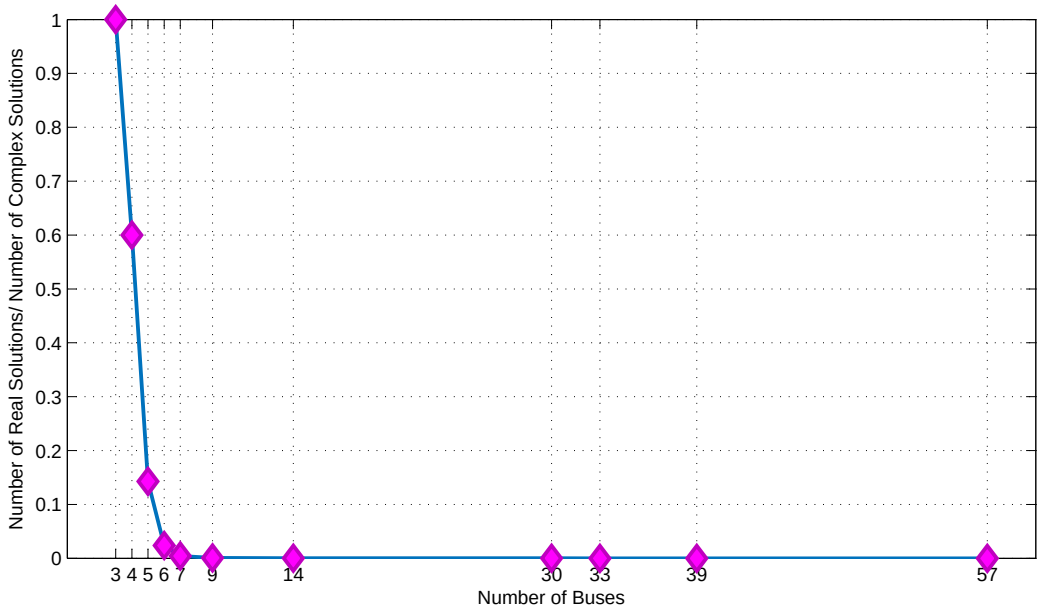


Figure 4.11: Ratio between Number of Real Solutions and Number of Complex Solutions

Consider a 7-bus power grid case called "case7Salam Mod1" which is modified from "case7Salam". See Appendix A for detailed data. Its power flow equations admit two real solutions in Table A.2. We first increase the active power load of "case7Salam Mod1" at Bus 3 by 50 MW, and call it "case7Salam Mod2". By solving the power flow equations we report four real solutions of "case7Salam Mod2" in Table B.1, i.e. two more real solutions to "case7Salam Mod1". It suggests that the number of real solutions can be non-monotonic with respect to the increase of active power load. Figure 4.12 shows PV curves at bus 3 for "case7Salam Mod1" and "case7Salam Mod2". Intersections of $\alpha = 0$ and the curve are different real solutions. We can see that "case7Salam Mod1" only has two real solutions, while "case7Salam Mod2" has four.

Next, we increase the reacitive power load of "case7Salam Mod1" at Bus 4 by 60 MVar, and call it "case7Salam Mod3". The power flow equations admit four real solutions for "case7Salam Mod3" in Table B.2, i.e. two more real solutions to "case7Salam Mod1". It indicates that the number of real solutions can be non-monotonic with respect to the increase of reactive power load. Figure 4.13 shows QV curves at bus 4 for "case7Salam Mod1" and "case7Salam Mod3".

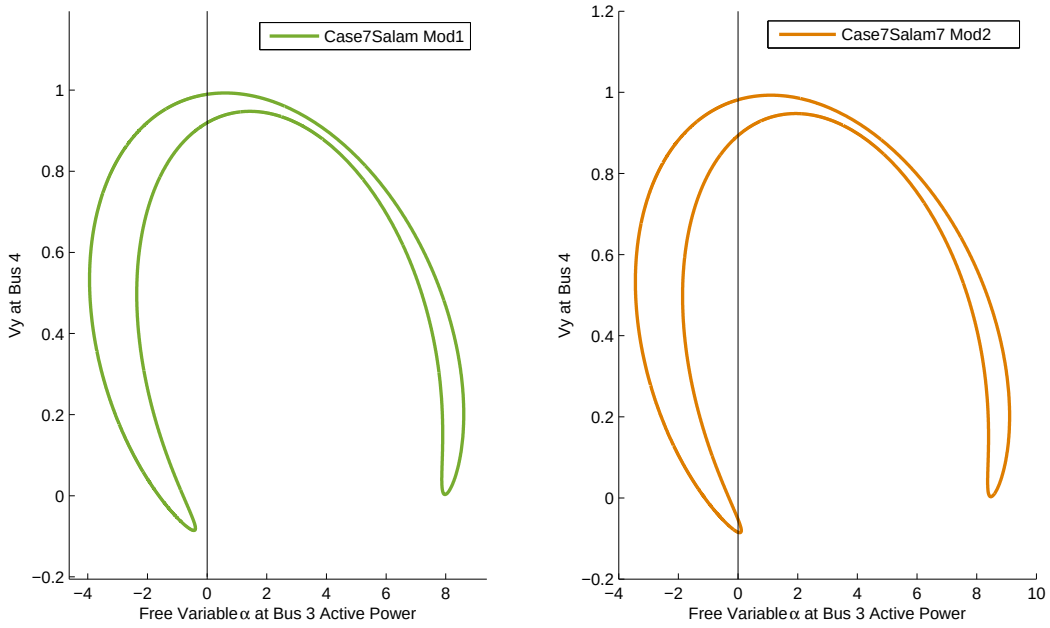


Figure 4.12: PV Curves at Bus 3 for case7Salam Mod1 (Left) and case7Salam Mod2 (Right)

Intersections of $\alpha = 0$ and the curve are different real solutions. We can see that "case7Salam Mod1" only has two real solutions, while "case7Salam Mod3" has four.

4.3.4 Multiple High-Voltage Solutions for a Particular Power System

It is well known that power systems with PV buses and loop structures can admit multiple high-voltage solutions [106, 11]. A recent investigation also reveals that a radial network with PQ buses can admit multiple high-voltage solutions [107]. In this subsection, we introduce a new power system called "case5Salam Mod" which has meshed network topology with both PV and PQ buses, and admits multiple high voltage solutions. This example is generated from "case5Salam" test case by reducing the reactive power to -585 MVar at bus 2. We report twelve real solutions to "case5Salam Mod" in Appendix C in which two of them are high voltage solutions, namely, the first one and the last one in Table C.1. Further investigation shows that the power flow problem at the first high voltage has a positive-definite Jacobian matrix in polar coordinates, while the power flow problem at the second high voltage has an eigenvalue with negative real part for its Jacobian

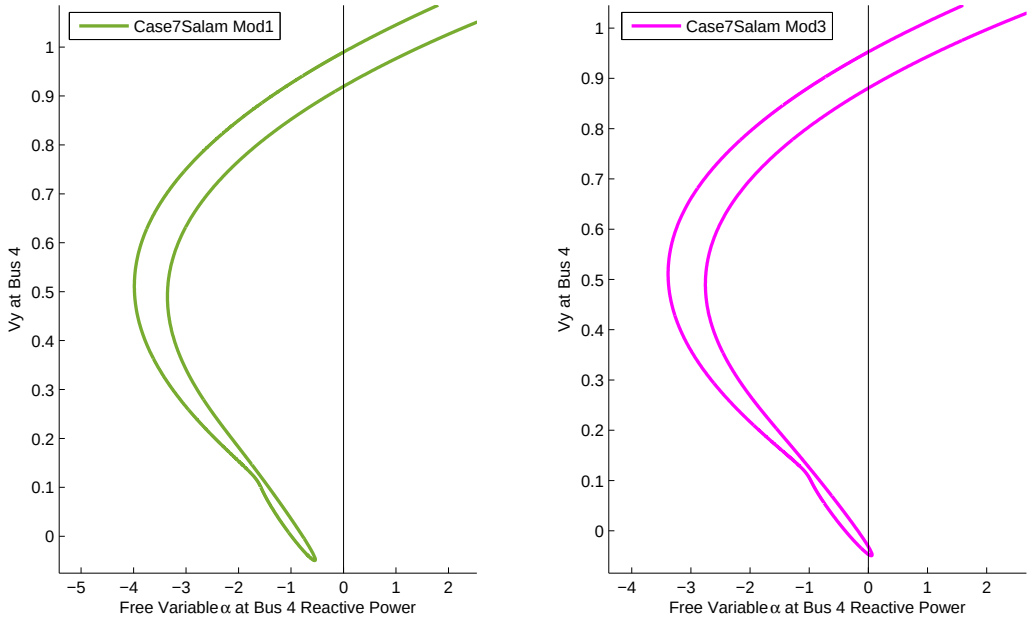


Figure 4.13: QV Curves at Bus 4 for case7Salam Mod1 (Left) and case7Salam Mod3 (Right)

matrix. Figure 4.14 illustrates low voltage solutions (green diamond), high voltage solutions (pink square) and a linking curve (blue curve) obtained from our proposed method.

4.3.5 Summary

In this section we specifically explained how to construct an ellipsoidal formulation for the power flow problem, and then applied the branch tracing method to search for multiple power flow solutions. The proposed method was verified by all the cases which had been completely solved before. Then, the proposed method was further applied to other benchmark test systems, some of which are too large to compute with existing methods. We highlight the efficiency of our proposed method comparing to traditional homotopy continuation method. Finally, we showed by examples that the number of real solutions for a power flow problem can change non-monotonically with the increase of load; and designed a new system that admits multiple high-voltage solutions.

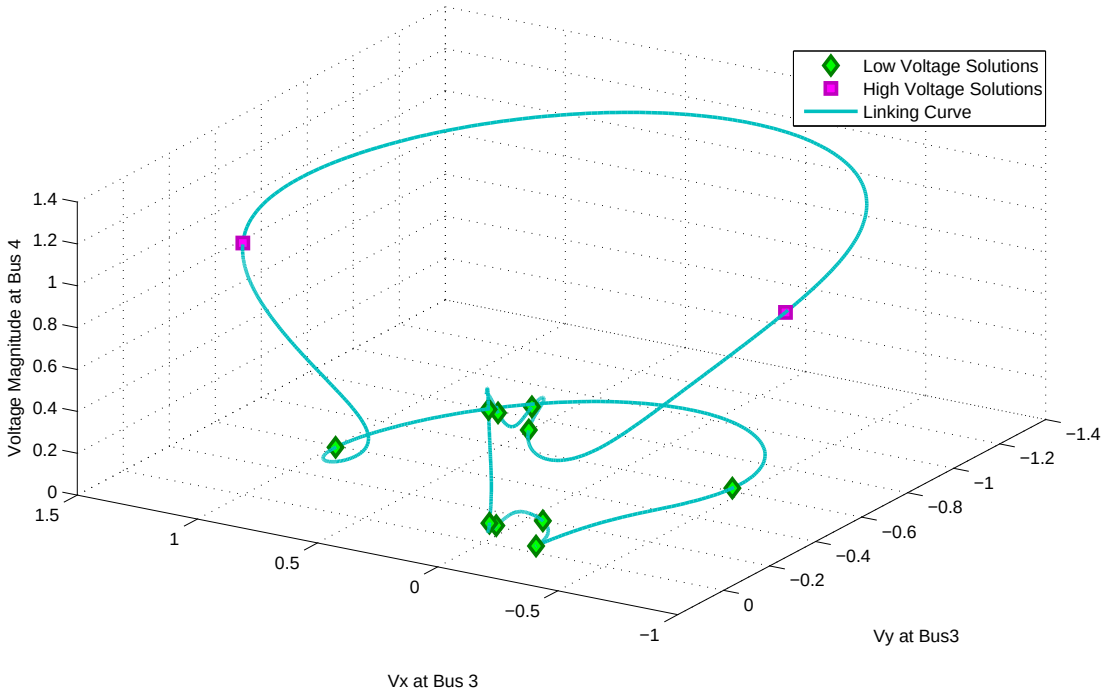


Figure 4.14: Solutions and Linking Curves for Case5Salam Mod

4.4 Identify Multiple Local Solutions to the Optimal Power Flow Problem

This section serves as the second application of our algebraic set preserving mappings to the AC optimal power flow problem. We specifically design an ellipsoidal formulation for the sphere confined eq-FJ conditions and apply the branch tracing method to the ellipsoidal formulation to locate multiple local solutions to the ACOPF problem. We first introduce two search strategies to locate multiple solutions: enumeration search and monotone search, and apply both of them to a few hard[5] numerical examples for illustration. Then, we apply an enumeration search strategy to find four additional local solutions for a published 39-bus system which has been known for admitting at least three local solutions. We also report that the proposed method can identify local solutions which do not satisfy the KKT conditions under equality constrained ACOPF model by a specific 5-bus example. Finally, we lead a discussion on designing more advanced search strategies

[5]These examples are hard for semi-definite programming relaxation to locate exact global solutions.

based on the proposed method. We would like to emphasize important capabilities of our proposed method for bridging disconnected feasible spaces to explore new solutions, and for locating local solutions which do not satisfy the KKT conditions.

4.4.1 Computation Procedure and Algorithms

The procedure for identifying multiple local solutions to the ACOPF problem can be partitioned into three sub-blocks, including finding a starting point, constructing an ellipsoidal formulation for the sphere confined eq-FJ conditions and searching for multiple solutions.

Algorithm 11 Overall Procedure for Solving Multiple Solutions of ACOPF

1: **procedure** SUB-BLOCKS
2: Finding A Starting Solution ▷ Call a standard nonlinear solver
3: Constructing Ellipsoids ▷ Add base ellipsoid to all the equations
4: Searching Solutions ▷ Apply enumeration search or monotone search

The first sub-block calls some standard nonlinear programming solver to provide a starting solution for the first order conditions. In this thesis we use the Matpower 5.0 package which, by default, applies the "MIPS" primal-dual interior point solver in the Matlab environment to return a first order solution. The second sub-block is developed to construct an ellipsoidal formulation of the sphere confined eq-FJ conditions to the ACOPF problem. The third sub-block can be implemented by one of two different strategies: enumeration search and monotone search, both of which apply the branch tracing method described in Section 4.2. Numerical examples demonstrate that the monotone search could be more efficient than the enumeration search for finding a sequence of lower-cost solutions in our examples.

4.4.1.1 Constructing an Ellipsoidal Formulation

In Chapter 3 we have shown that a base ellipsoid of the sphere confiend eq-FJ conditions can be constructed by Equation (3.19), where γ_0 has to be some large number. Equation (3.19) can be

succinctly written as

$$\mathbf{X}^T\mathbf{M}_c\mathbf{X} + b_c + \gamma_0(\mathbf{X}^T\mathbf{M}_0\mathbf{X} + b_0) = 0 \tag{4.19}$$

where $\mathbf{M}_c$ and b_c are the sums of matrices and scalars independent of γ_0 in Equation (3.19), $\mathbf{M}_0$ and b_0 are the sums of matrices and scalars associated with γ_0.

The procedure to construct the ellipsoidal formulation is summarized in Algorithm 12. The constant γ_0 can be chosen in different ways, but will not affect the construction of ellipsoids as long as γ_0 is greater than some particular value. To control bad conditioning, we start to try a small value of γ_0. If it fails to provide a base ellipsoid, we gradually increase γ_0 until reaching a base ellipsoid. According to Theorem 3.5.2, a base ellipsoid always exists as long as the problem is bounded. When a base ellipsoid is generated, we scale it and add it to all other equations to form an ellipsoidal formulation.

Once all the equations in the sphere confined eq-FJ conditions have been converted into ellipsoids, we come to the last sub-block of Algorithm 11: searching solutions. We first introduce the enumeration search strategy, and modify it to accommodate a more efficient monotone search strategy.

4.4.1.2 Enumeration Search Strategy

The enumeration search strategy in this section is similar to the enumeration search strategy for the power flow problem. That is, starting at a solution point, following every 1-dimensional curve defined by $n-1$ ellipsoids out of total n ellipsoids, recording new solutions found along these curves and setting them as new starting points for next round tracing. The enumeration terminates when every found solution has been set as a starting point and traced all the corresponding curves, and no more solutions appear. A pseudo-code is provided in Algorithm 13.

Once the enumeration terminates, it returns the solution set S including all the connected solutions to the sphere confined eq-FJ conditions. Note that each solution to the sphere confined eq-FJ conditions is not uniquely determined because flipping the signs of slack variables will not alter the solution. We call these solutions "non-distinguishable solutions". To identify multiple local extrema, we first remove the non-distinguishable solutions from S, then check the inequalities of

Algorithm 12 Constructing Ellipsoids For Fritz John Conditions

procedure CONSTRUCTING BASE ELLIPSOID
  $k \leftarrow 0$ ▷ Initialize the count number
  $\gamma_0 \leftarrow 1$ ▷ Initialize the scaling number
  $\mathbf{M} \leftarrow \mathbf{M}_c + \gamma_0 \mathbf{M}_0$ ▷ Initialize the base ellipsoid matrix
  $b \leftarrow b_c + \gamma_0 b_0$ ▷ Initialize the base ellipsoid constant
  $\lambda_{min} \leftarrow -1$ ▷ Initialize the smallest eigenvalue
  while $\lambda_{min} \leq 0$ or $b \leq 0$ **do**
    $k \leftarrow k + 1$ ▷ Update the count number
    $\gamma_0 \leftarrow 3^k$ ▷ Update the scaling factor
    $\mathbf{M} \leftarrow \mathbf{M}_c + \gamma_0 \mathbf{M}_0$ ▷ Update the base ellipsoid matrix
    $b \leftarrow b_c + \gamma_0 b_0$ ▷ Update the base ellipsoid constant
    λ_{min} ▷ Compute the minimum eigenvalue of $\mathbf{M}$
procedure REVISE BASE ELLIPSOID
  $\lambda_{min} \leftarrow -1$ ▷ Initialize the smallest eigenvalue
  $k \leftarrow 0$ ▷ Initialize the scaling factor
  while $\lambda_{min} \leq 0$ or $b \leq 0$ **do**
    $k \leftarrow k + 1$ ▷ Update the scaling factor
    $\mathbf{M} \leftarrow k\mathbf{M} + \sum_{i=1}^{N_{eqn}} \mathbf{M}_i$ ▷ Update the base ellipsoid matrix
    $b \leftarrow kb + \sum_{i=1}^{N_{eqn}} b_i$ ▷ Update the base ellipsoid constant
    λ_{min} ▷ Compute the minimum eigenvalue of $\mathbf{M}$
procedure CONSTRUCTING ELLIPSOIDS
  for $i = 1,\ 2,\ \cdots,\ N_{eqn}$ **do**
    $\lambda_{i,min} = -1$ ▷ Initialize the smallest eigenvalue
    while $\lambda_{i,min} \leq 0$ or $b_i \leq 0$ **do**
      $b_i \leftarrow b_i + b$ ▷ Update the i-th ellipsoid constant
      $\mathbf{M}_i \leftarrow \mathbf{M}_i + \mathbf{M}$ ▷ Update the i-th ellipsoid matrix
      $\lambda_{i,min}$ ▷ Compute the minimum eigenvalue of $\mathbf{M}_i$

Algorithm 13 Enumeration Search for ACOPF

procedure ENUMERATE ACOPF SOLUTIONS
  $S \leftarrow x_1$ ▷ Initialize the solution set
  $N_{solu} \leftarrow Count(S)$ ▷ Initialize the number of solutions
  $k \leftarrow 1$ ▷ Initialize counting number
  while $k \neq N_{solu}$ **do**
    $k \leftarrow k + 1$ ▷ Update counting number
    $x_0 \leftarrow x_k$ ▷ Update starting solution
    for $i = 1,\ 2,\ \cdots,\ N_{bus}$ **do**
      Algorithm 7 ▷ Trace the curve defined by relaxing the i-th equation
      return S_{new} ▷ Return the newly found solutions
      $S \leftarrow S \cup S_{new}$ ▷ Update the solution set
      $N_{solu} \leftarrow Count(S)$ ▷ Update the number of the solutions

multipliers associated with active inequality constraints. Finally, the second order sufficient condition [108] is applied to classify these solutions and to distinguish local extrema. To capture the best solution (can be the global one), we substitute these solutions to the objective function and compare their corresponding objective values.

4.4.1.3 Monotone Search Strategy

The enumeration search strategy described in Algorithm 13 can be time consuming because the problem size increases with the number of equations. This means the number of curves also increases with the problem size. However, the enumeration search strategy is completely suitable for parallel computing since each curve is independent to other curves. Later in subsection 4.4.2.3, we implemented the enumeration search for a 39-bus system in parallel for searching additional local solutions. On the other hand, if we are only interested in identifying the best (global) solution, a more efficient search strategy called "monotone search" can be designed. It is modified from

the enumeration search strategy by maintaining a non-increasing[6] objective function value while following the curves from the sphere confined eq-FJ conditions. This is done by introducing one extra constraint to the ACOPF problem in Equation (2.31) or (2.30).

Suppose we have a known solution $(\mathbf{P}_{gen,*},\ \mathbf{U}_*)$ to the sphere confined eq-FJ conditions of Equation (2.31). At current solution the objective function value is $J_* = \sum_{i=1}^{N_{gen}} d_i P_{gen,i,*}^2 + c_i p_{gen,i,*}^2$. We use this value to set a threshold for the objective function:

$$\sum_{i=1}^{N_{gen}} d_i P_{gen,i}^2 + c_i p_{gen,i}^2 + t^2 = J_* + \epsilon^2 \tag{4.20}$$

where t is a free slack variable, ϵ is a small constant.

Since $t^2 \geq 0$, the objective function is restricted to be less than $J_* + \epsilon^2$. The value of J_* will be updated each time when we find a better solution (has a lower objective value) until we reach the best solution. The pseudo-code for the monotone search strategy is provided in Algorithm 14.

Algorithm 14 Monotone Search

```
1:  procedure MONOTONE SEARCH STRATEGY
2:      x_0 ← x_1                                   ▷ Set a starting solution
3:      J* = J(x_1) + ε^2                           ▷ Set current threshold
4:      for i = 1, 2, ··· , N_eqn do
5:          Algorithm 7          ▷ Follow curves by relaxing the i-th ellipsoid
6:          return x_new                            ▷ Return new solutions
7:          if t_new ∈ x_new such that |t_new| > ε then
8:              x_1 ← x_new                         ▷ Update starting solution
9:              go to 2
10:         else
11:             return x_new                        ▷ The best (global) solution
```

Note that we usually set $\epsilon^2 = 0.1 J_*$ in case a better solution can only be reached by a solution that has a higher objective value than current objective value. As the objective value decreases, the

[6] In practice we allow the objective value to be not strictly non-increasing to accommodate various situations.

threshold value for the objective function cut the feasible space more and more, which is possible to induce more severe bad conditioning.

Another issue occurs when we introduce the threshold Equation (4.20) for each starting point. Since our method cannot distinguish different types of first order solutions, the threshold will introduce additional eq-FJ points which are fictitious for the original problem. This issue can be solved by discarding solutions binding at the objective threshold.

4.4.2 Numerical Examples

This section provides several examples for which the proposed approach yields multiple local minima and the global optima but the SDP relaxation of [44] is not exact. The first three examples, i.e. WB5, case9mod and case39mod4, were proposed and studied in [25] with random initial points to locate multiple local solutions. See [26] for the test case data and the results from the approach of [25]. The feasible spaces of the 5-bus (WB5) and the 9-bus (case9mod) examples were investigated in [109] and shown to be disconnected. While many local algorithms cannot bridge disconnected feasible spaces to search for multiple solutions, our proposed approach finds all local minima and the global optima for these two problems. Moreover, we also report seven local minima (including the global minimum) to the 39-bus (case39mod4) example, which include four new local minima in addition to the three reported in [25, 26]. Finally, we modified the 5-bus (WB5) system and showed that our method successfully identified the global solution for this example which does not satisfy the eq-KKT conditions under equality constrained ACOPF model (2.31). Numerical experiments showed that some off-shelf commercial solvers (IPOPT, KNITRO, CONOPT, SNOPT and BARON) failed to locate the global solution for this example.

Our tracing program was coded in Matlab and executed on a PC with a 2.8GHz Intel i-7 CPU and 4GB RAM for the 5- and 9-bus systems. The initial conditions are solved by MATPOWER 5.0 [110] with the MIPS solver under default settings. An incomplete enumeration for the 39-bus system was performed in parallel using the computing resources and assistance of the Center for High

Throughput Computing (CHTC) [7] at the University of Wisconsin–Madison Department of Computer Sciences. Each trace was assigned to an autonomous computer with one CPU, 2 GB of memory, and 4 GB of disk space.

4.4.2.1 9-Bus System with Quadratic Cost Function

Our first example is the ”case9mod” case in [26]. It has nine buses, nine transmission lines and three generators. The objective function of this example is a quadratic function. The specific data can be found in [26] as well as in Appendix D. The network one-line diagram is shown in Figure 4.15. [109] shows that the feasible space of this case consists of three disconnected components.

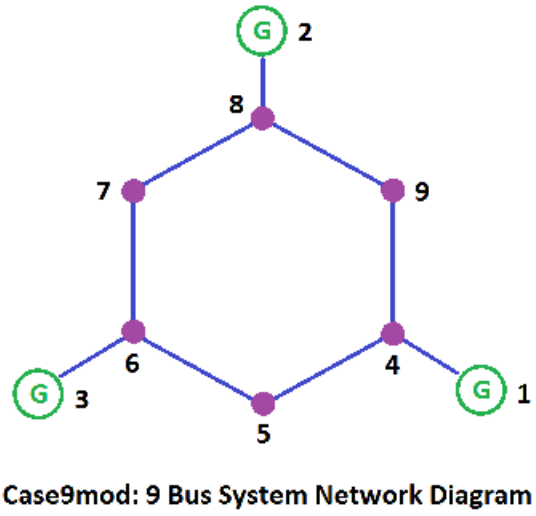


Figure 4.15: Case9mod 9-Bus Example One-Line Diagram

We first applied the enumeration search strategy to find multiple eq-FJ points for this problem. The enumeration search strategy terminates after 2489 traces and returns 27 stationary solutions, among which four are verified to be local minima, five are local maxima, and the rest are saddle

[7] The CHTC is supported by UW-Madison, the Advanced Computing Initiative, the Wisconsin Alumni Research Foundation, the Wisconsin Institutes for Discovery, and the National Science Foundation, and is an active member of the Open Science Grid, which is supported by the National Science Foundation and the U.S. Department of Energy’s Office of Science.

points [111]. These four local minima, which are listed in Table D.3, match those in [26]. Solution 4 in the last column of Table D.3 is the global optimum as verified both by exhaustive search of the feasible space [109] and by the second-order moment relaxation [112].These four local minima are identical to the four local solutions found in [26], and they are listed in Table D.3. The overall execution time is 2.78×10^4 seconds (7.7 hours) with mean and median times for each trace of 11.2 seconds and 4.3 seconds, respectively. A histogram plot of the execution times for each trace (on a log scale) is shown in Figure 4.16. Although the traces were computed sequentially, parallel computing techniques could be used to reduce the computation time. Note that most of the traces terminated within a few seconds, but a small fraction of the traces were much slower. Reasons for the slow traces include

1. Many sharp corners on a trace resulting in a small step size.

2. Many portions of the trace with poor numeric conditioning.

3. Many solutions are indistinguishable except for a change in sign of the slack variables.

The monotone search strategy is next applied to the same problem. Projections of the monotone search traces and solutions are depicted in Fig. 4.17. The gray regions in the Figure 4.17 illustrate a projection of the disconnected feasible space generated using the method described in [109]. Starting at an initial local solution (black dot) obtained by MATPOWER 5.0 [110], the monotone search strategy locates a second eq-FJ point (light blue dot labeled 2) via the fifth trace (light blue curve). The fourth trace (green curve) starting from the second eq-FJ point yields the global optimum (the green star labeled 3). The monotone search approach uses 9 traces to locate the global solution, which is significantly fewer than the 2489 traces used in the enumeration strategy. The overall execution time was 214.6 seconds with mean and median times for each trace of 23.8 seconds and 11.6 seconds, respectively. The overall execution time is thus 130 times faster than the full enumeration [111]. The non-increasing solutions are listed in Appendix D.

We can see from Table D.4 that the objective function value decreased at each solutions. Note that the intermediate none-increasing solutions obtained by the monotone search are not necessarily the local minima. For instance, the second solution in Table D.4 is a saddle point.

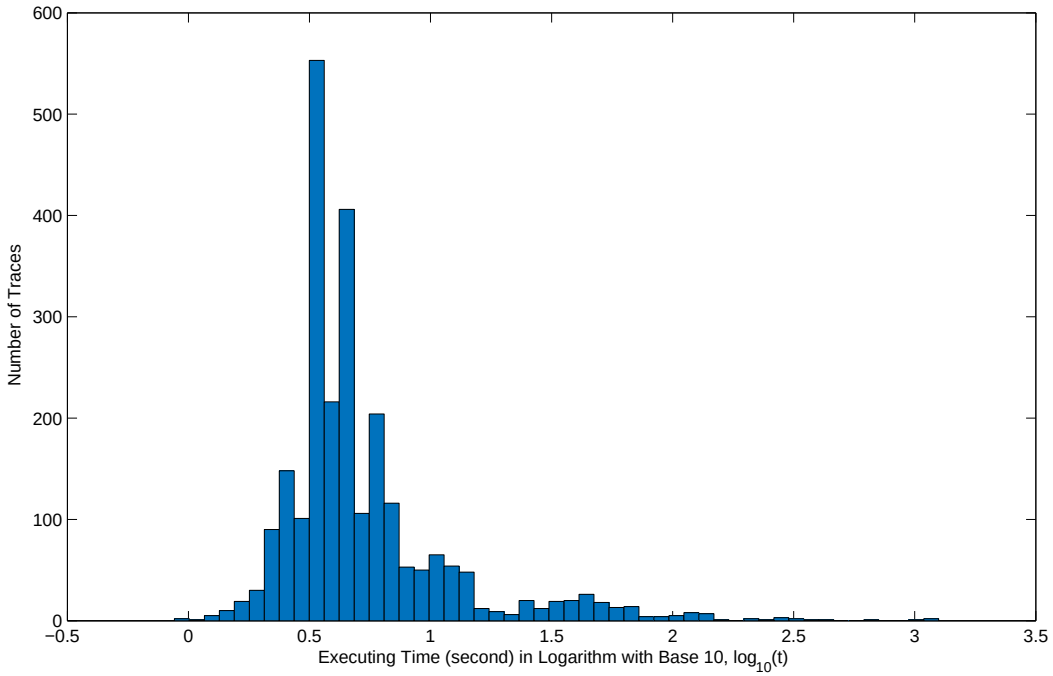


Figure 4.16: Case9mod 9-Bus Execution Time

4.4.2.2 5-Bus System with Linear Cost Function

The second example, "WB5", has five buses, six transmission lines, and two generators [26]. Its one-line diagram is depicted in Figure 4.18. As shown in Fig. 4.20, which was created using the approach in [109], the feasible space for this problem has two disconnected regions. The enumeration search strategy computes 628 traces to yield 12 eq-FJ points, among which two are local minima matching those reported in [25]. The overall execution time was 1959 seconds with mean and median times for each trace of 3.1 seconds and 2.5 seconds, respectively, with the distribution shown in Fig. 4.19. The monotone search strategy locates the global solution in the eleventh trace from the initial starting point. Fig. 4.20 depicts all the stationary solutions identified by the enumeration strategy as well as one of the traces in the enumeration strategy that connects to the

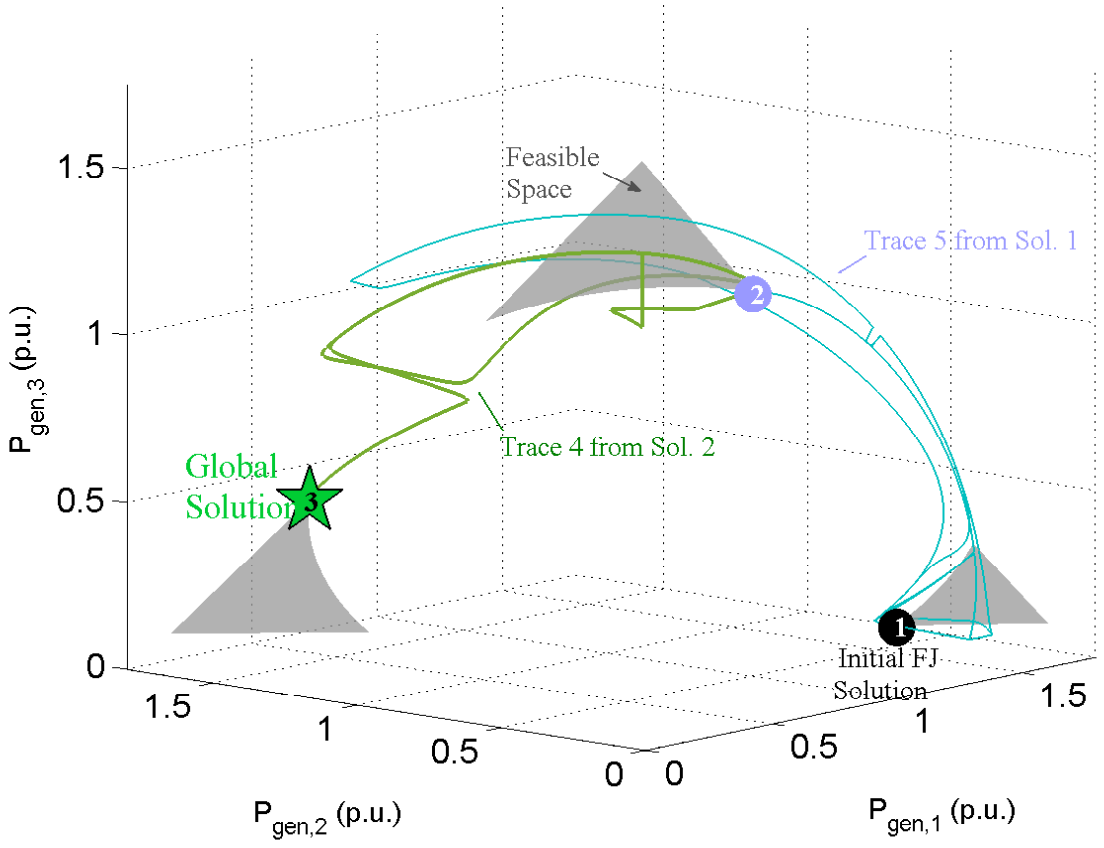


Figure 4.17: Selected Monotone Search Traces for case9mod

global solution.[8] The overall execution time was 44 seconds (a factor of 44 faster than the full enumeration) with mean and median times for each trace of 4.0 seconds and 3.0 seconds, respectively [111].

The black dot is the initial FJ solution given by MATPOWER, the blue diamonds denote all the stationary points identified by enumeration, and the green star is the best solution identified by the enumeration approach. As verified both by exhaustive search of the feasible space [109] and by the second-order moment relaxation [112], the green star is the global optimum.

[8]The trace shown in Fig. 4.20 connects the initial solution to the global solution, but does not connect all the eq-FJ points found by other traces.

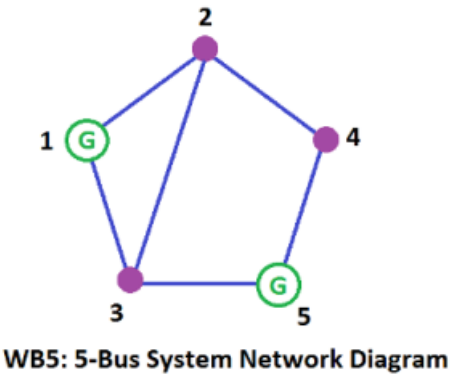


Figure 4.18: Executing Time of Enumeration Strategy for WB5

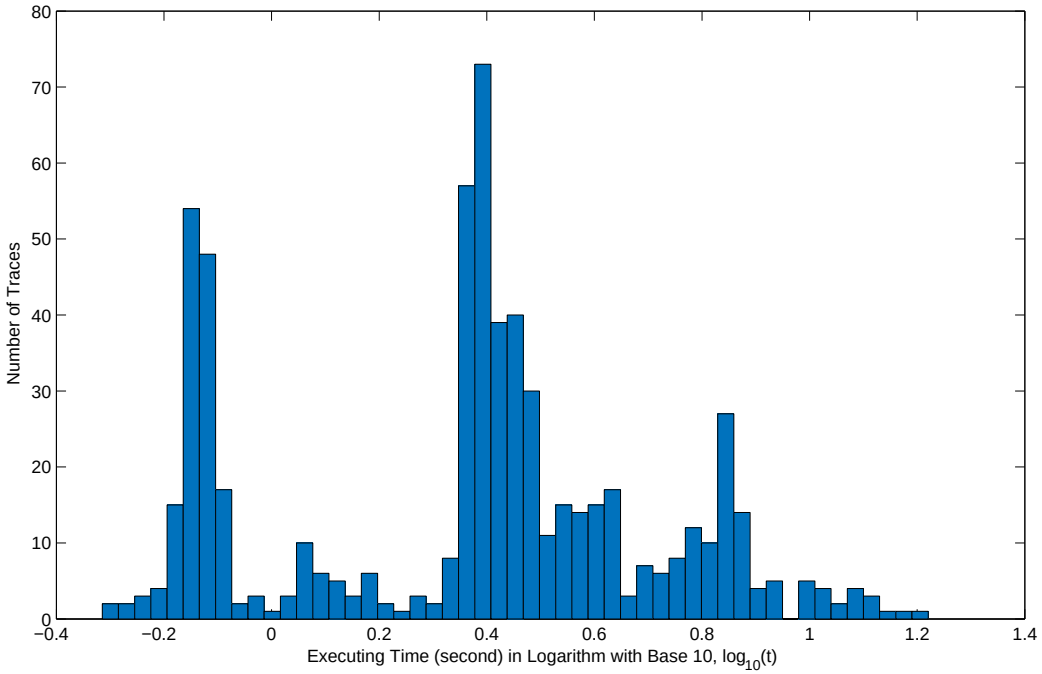


Figure 4.19: Executing Time of Enumeration Strategy for WB5

4.4.2.3 case39mod4 39-Bus System

The third example is a 39-bus system called “case39mod4” with 46 transmission lines and 10 generators. Three local minima have been reported for this system [26]. The proposed tracing

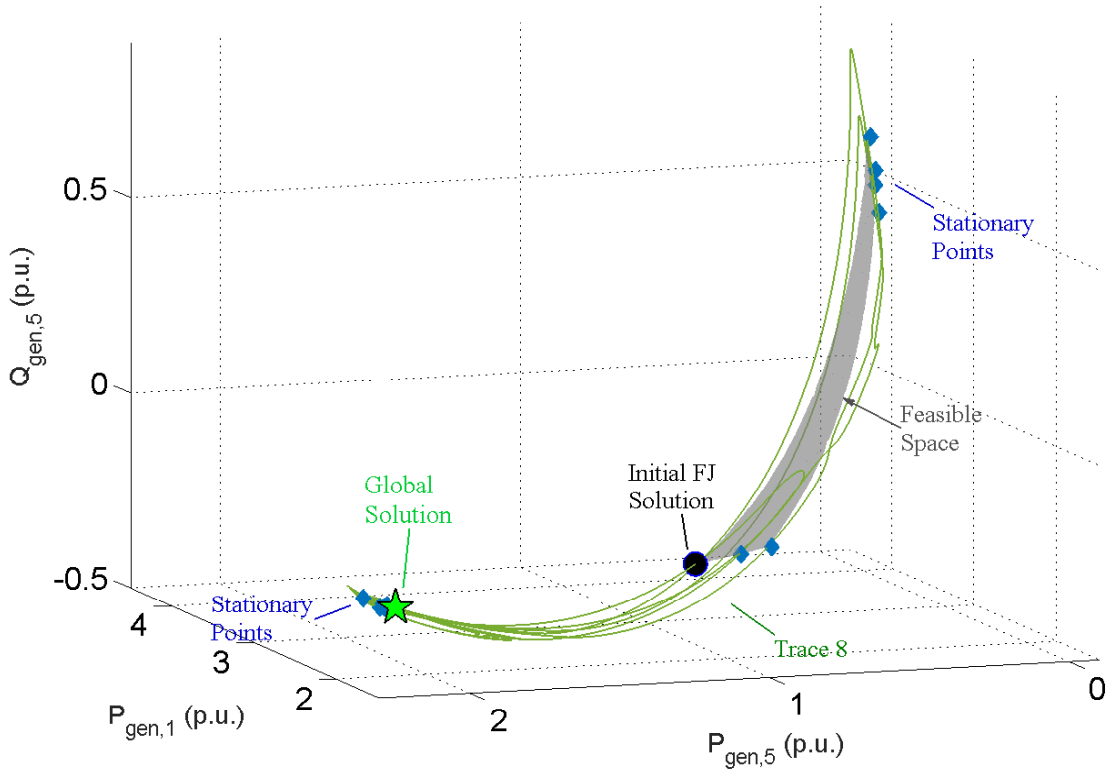


Figure 4.20: Selected Enumeration Search Trace for WB5

method finds at least seven local minima (including the known three). The newly found additional four local minima are listed in Appendix F.

To make an exact comparison to [26], we first enforced apparent power flow limits on transmission lines, which induces 924 many decision variables. Starting at the three known local minima, we computed a single trace of the paths associated with each constraint. This incomplete enumeration followed a total of 2772 paths to yield 10238 different eq-FJ points, including 150 KKT points for minimization, 204 KKT points for maximization, 7 local minima (the three known local minima and four new ones) and 6 local maxima. These KKT points and local extrema are shown separately in Figure 4.21, and shown integrally in Figure 4.22. All the eq-FJ solutions as well as KKT points are shown together in Figure 4.23. Note that starting the enumeration search from the second known local minimum yields 6 out of 7 minima (two known local minima and four

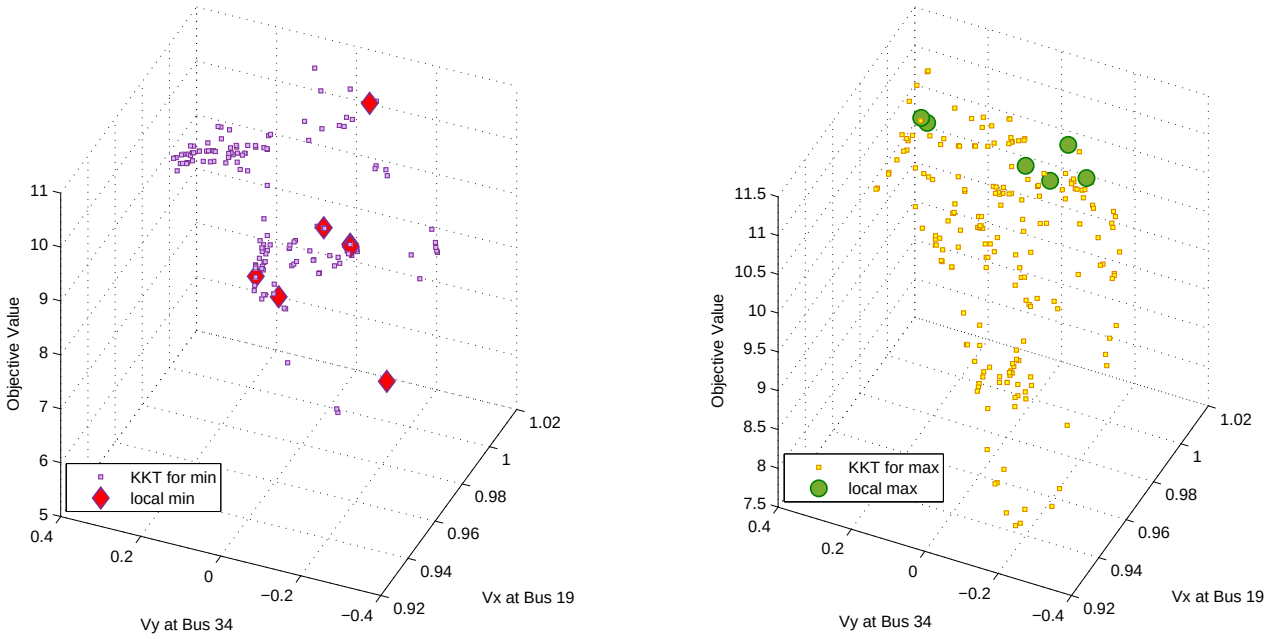


Figure 4.21: case39mod4 Line Apparent Power Model KKT Points for Local Minima (Left) and KKT Points for Local Maxima (Right)

new ones) using only 924 traces. Further investigation indicates that there exist over 3000 different concatenated paths[9] that connect every local minimum to the global minimum. The solutions found along some (but not all) of these concatenated paths have objective function values that are monotonically decreasing. Also note that despite the large number of eq-FJ points, a concatenated path starting from a known local solution to the global solution can be simple. For example, a concatenated monotone path starting at a local minimum with cost 1068.30/hr passes through an eq-FJ point with a lower cost of 866.80/hr, which then passes through another eq-FJ point with a lower cost 567.64/hr and eventually reaches the global minimum with the lowest cost 557.14/hr. This monotonically decreasing path only passed through two additional intermediate eq-FJ points to reach the global solution [111].

[9]We use the phrase "concatenated path" to refer to the sequence of several individual paths concatenated at common solution points.

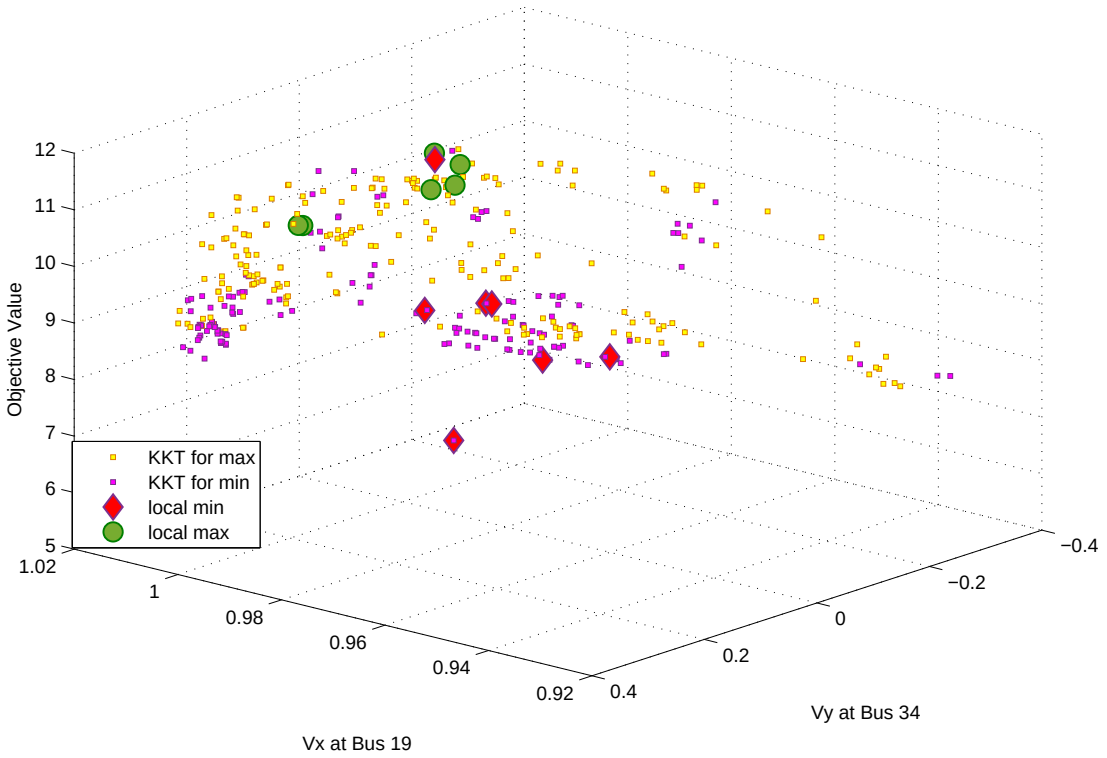


Figure 4.22: case39mod4 Line Apparent Power Model KKT Points and Local Extrema

Next, we consider "case39mod4" case with line current limits. As discussed in Chapter 2, line current model is superior in dimensionality to line apparent power model, which in this case reduces the number of decision variables to 556. An incomplete enumeration search yields 16677 different eq-FJ solutions, among which 85 of them are KKT points for minimization, 202 of them are KKT points for maximization, 6 of them are local minima, and 14 of them are local maxima. These KKT points are depicted separately in Figure 4.24, and presented integrally in Figure 4.25. We then applied monotone search strategy for this example to generate decreasing sequences of eq-FJ solutions. Figure 4.26 presents two different monotone search sequences based on two different ellipsoidal formulations. The green sequence approaches to the vicinity of the global solution with 31 intermediate eq-FJ points, while the blue sequence needs more intermediate eq-FJ points to approach the vicinity of the global solution.

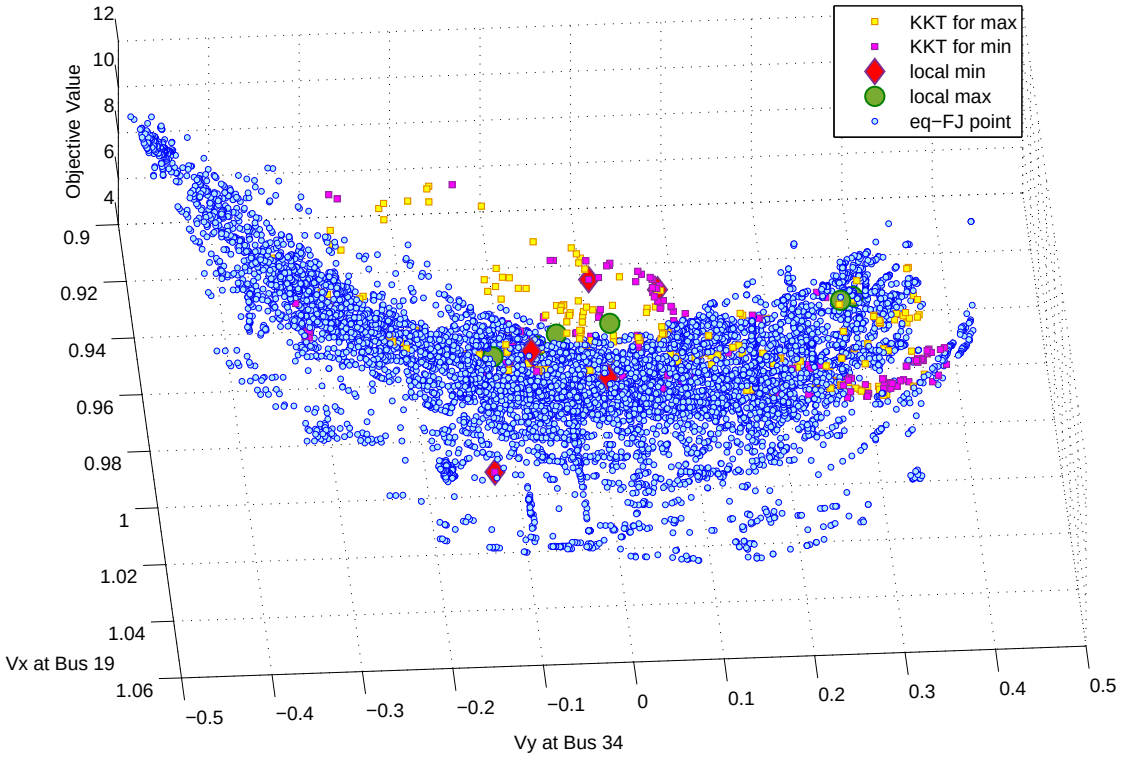


Figure 4.23: case39mod4 Line Apparent Power Model 10238 eq-FJ Solutions

4.4.2.4 WB5mod 5-Bus System

We present a modified version of "WB5" system called "WB5mod". In contrast to some commercial solvers, the proposed tracing algorithm finds the global optimum for this problem. The test case "WB5mod" has the following variations from "WB5" in Appendix E:

- Bus voltage magnitude limits are $[0.9, 1.1]$,
- The generator at bus 1 has active power limits of $[2, 15]$ per unit and reactive power limits of $[0.4, 18]$ per unit,
- The generator at bus 5 has active power limits of $[0, 20]$ per unit and reactive power limits of $[-0.5, 8]$ per unit,
- The cost function is $J = P_{gen,1}^2 + 2P_{gen,1} + 12P_{gen,5}^2 + 1200P_{gen,5}$.

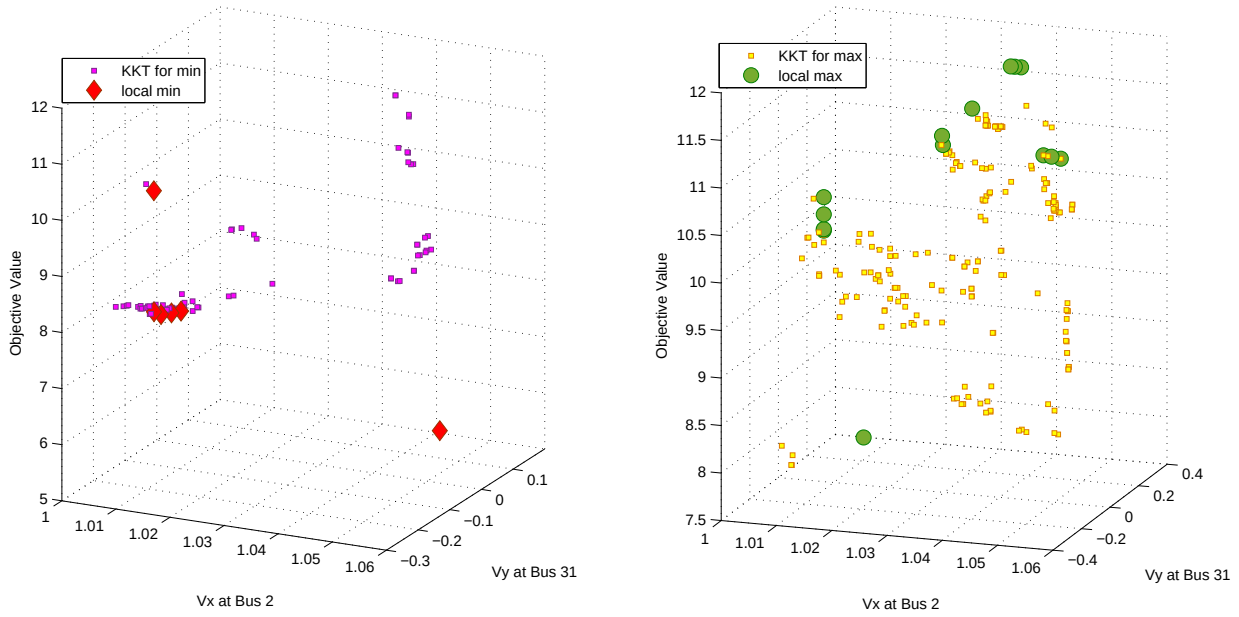


Figure 4.24: case39mod4 Line Current Model KKT Points for Local Minima (Left) and KKT Points for Local Maxima (Right)

This example can lead to failure of recognizing the global solution for some local solvers based on the interior point method. It also causes a trouble for the equality constrained ACOPF model (2.31) because the LICQ fails at the global solution in this model. Thus, the global solution doesn't satisfy the eq-KKT conditions under equality constrained ACOPF model (2.31). However, our tracing method is based on more general Fritz John conditions which do not require any constraint qualifications. Hence, the proposed method can locate the global solution for this example.

Fig. 4.27 shows four disconnected feasible regions for this example as well as eq-FJ points, local extrema, and selected paths that connect the starting point to the global solution. When initialized from a flat start, Matlab interior point solver "MIPS" converged to a local solution represented by a blue diamond with cost 161921.15 \$/hr. A trace initialized from this point connects to an intermediate eq-FJ point depicted by yellow square, which is not a local minimum. Another trace connects this intermediate eq-FJ point to the global solution at the green pentagram with a cost of 139875.00 \$/hr. (See Table G.1 in Appendix G.) Note that the angle differences

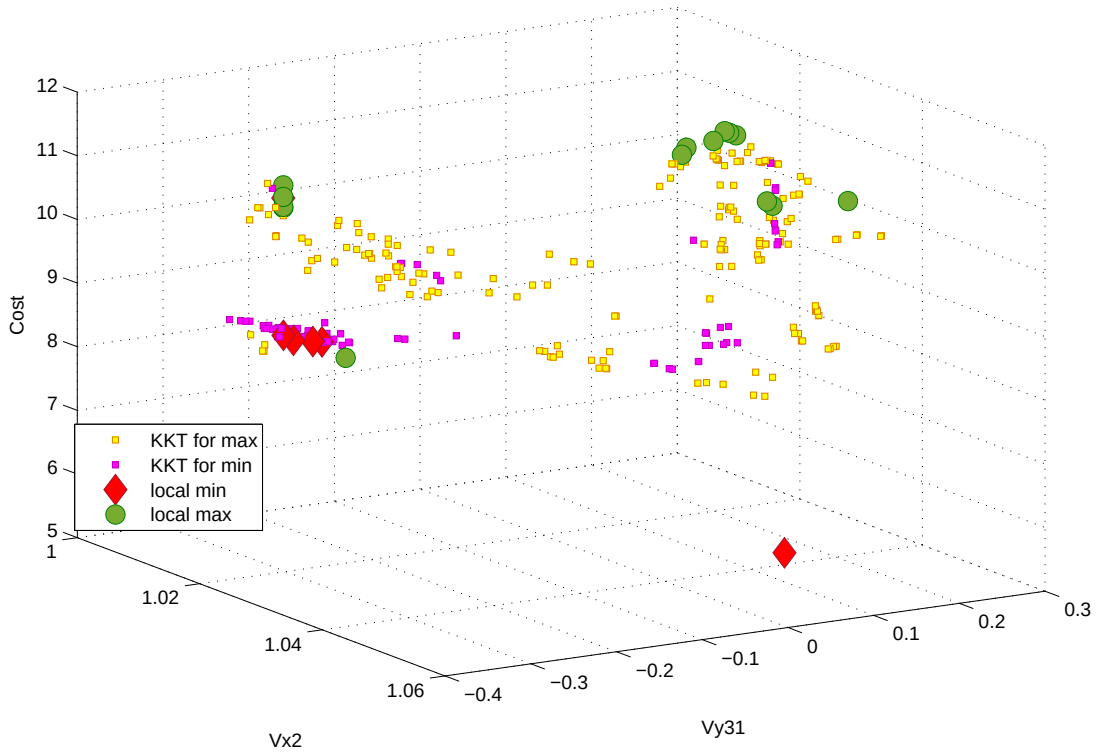


Figure 4.25: case39mod4 Line Current Model KKT Points and Extrema

between adjacent buses are less than 30° for both solutions, despite the fact that these solutions are in different disconnected regions of the feasible space.

To further verify our results, higher order moment-based SDP relaxations have been applied [112] to confirm the global solution. Several local solvers, i.e. "KNITRO", "IPOPTH", "CONOPT", "SNOPT", and a global solver "BARON" from "GAMS", are also implemented to this example. The results are provided in Table G.2 in Appendix G. The 2^{nd} order moment relaxation provided a solution close to our global solution with inexactness. The 3^{rd} order moment relaxation found an almost exact solution which coincides with the global solution we found. However, all other local solvers except "CONOPT" converged to a different local solution which has a much higher objective value initiated from a flat start. The solver "CONOPF" reported locally infeasibility.

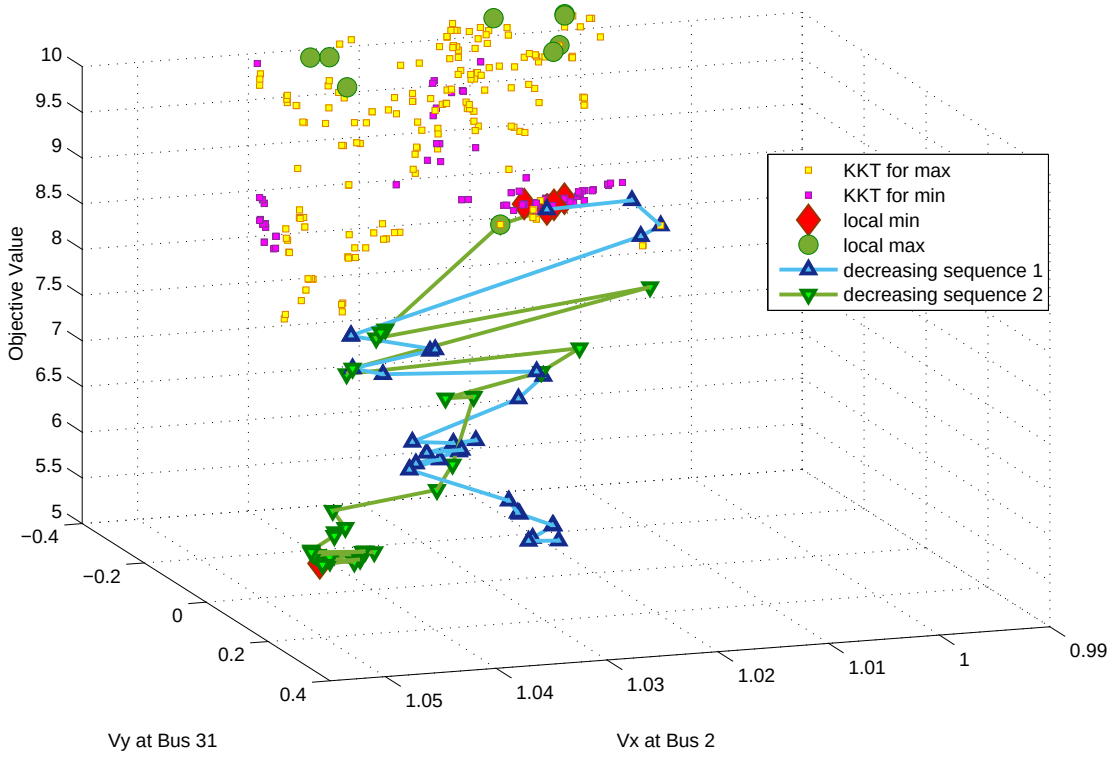


Figure 4.26: case39mod4 Line Current Model Monotone Search Sequence

4.4.3 Advanced Search Design

From previous simulation results we noticed that the proposed method has the following features:

- The enumeration search strategy can identify a lot of eq-FJ points, which is suitable to enumerate multiple local extrema. For example in "case39mod4" case, our incomplete enumeration identified over 10000 eq-FJ points, among which seven are local minima including four new local minima.

- The monotone search strategy is more efficient than the enumeration search strategy for finding solutions with lower objective values. For instance in Figure 4.26, the second monotone

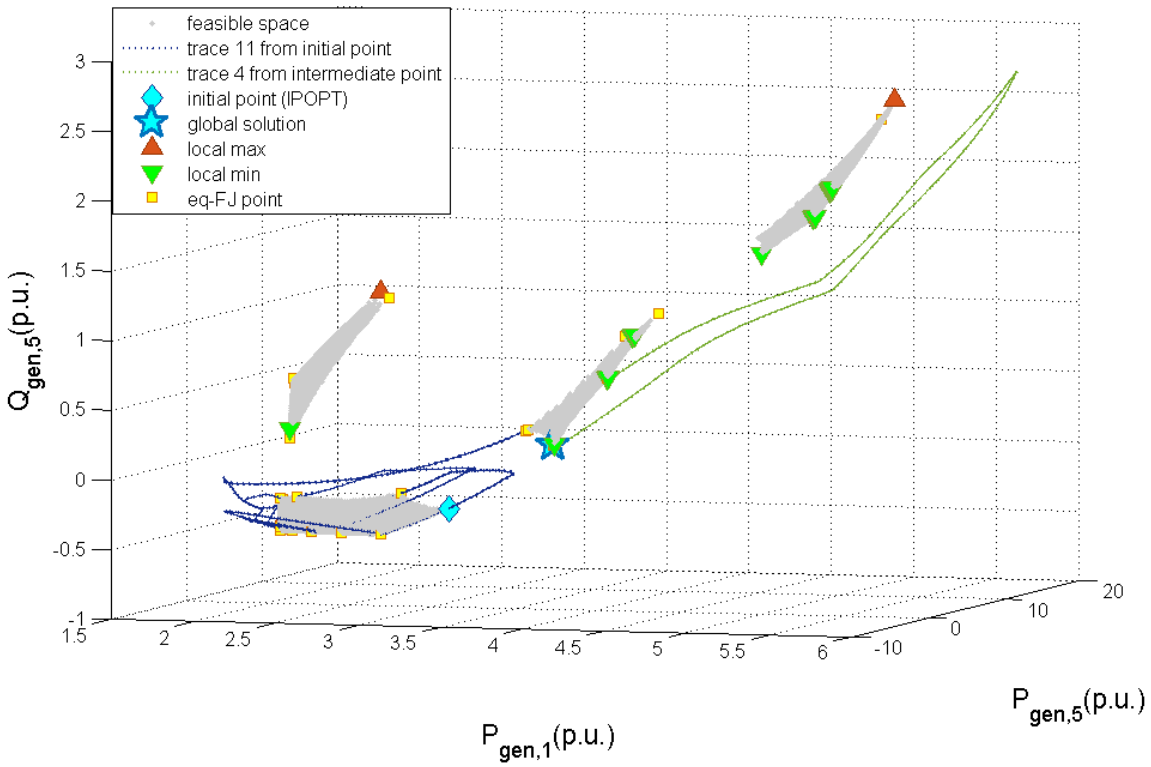


Figure 4.27: WB5mod 5-Bus Local Extrema, Global Optimum, and Linking Curves

sequence located only 31 intermediate eq-FJ solutions before reaching the vicinity of the global solution.

- The number of eq-FJ points can sometimes be much larger than the number of local extrema. For instance, “case39mod4” with either line apparent power model or line current model possesses over 10000 eq-FJ points but only dozens of local extrema.

- Despite a huge number of eq-FJ points for an ACOPF problem, the number of intermediate points required for linking the starting point to the global solution can be very few. For example, “case39mod4” only needs two intermediate eq-FJ points to connect any pair of the known local minima; “case9mod” only needs one; and “WB5” doesn’t need any intermediate point.

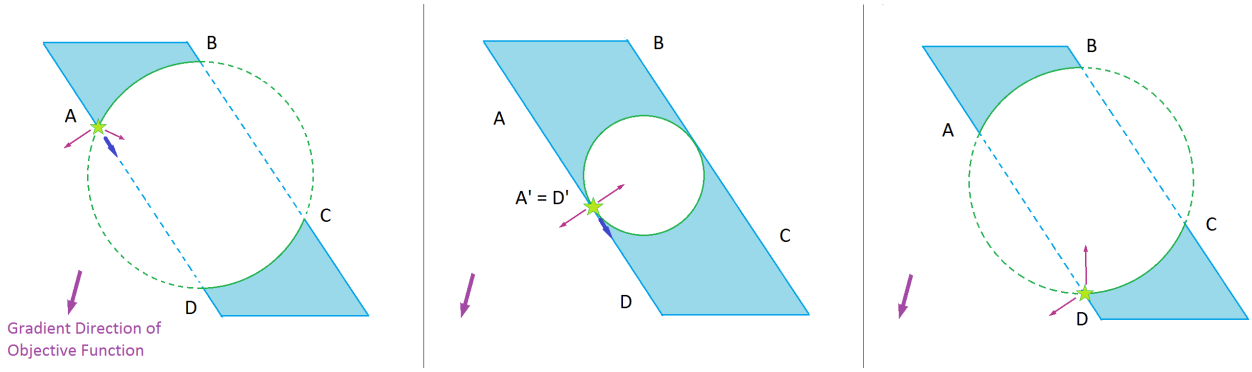


Figure 4.28: Bridge Two Disconnected Feasible Regions by Tracing A Circular Constraint: Start Tracing (Left), Pass Through Singularity (Middle), and Reach Another Feasible Region (Right)

According to these features, a more advanced search strategy may be designed to enhance the search efficiency for locating a better solution. An ideal situation would be choosing an appropriate curve to follow and pick an appropriate intermediate point as next starting point such that the number of curves to be followed before finding the global solution is minimal. For example, the minimal number of curves to be followed for "case9mod" is two, according to Figure 4.17. However, in practice, we followed at least nine curves to reach the global solution. In "WB5" case the ideal number of curves to be followed is just one, however, we followed at least eight curves to reach the global solution. The reason why the proposed methods are less efficient than the ideal case is that we didn't follow those curves in an informative order but treated them equally. To understand why the order of tracing matters, let's illustrate it through a toy example in Figure 4.28 and Figure 4.29.

Figure 4.28 shows how we can reach another feasible region by continuously following an appropriate constraint. The left plot of Figure 4.28 depicts an initial state of our tracing. The blue areas are two disconnected feasible regions which are comprised of an intersection of a polytope and a circular "hole". Our starting solution is at green pentagram labeled "A". When we continuously shrink the radius of the circular hole, our starting solution "A" will follow the secant line "AD" at the blue arrow direction. However, at a particular radius, the circular hole is tangent to "AD", which means constraint qualifications fail at this point. If we impose the KKT conditions to

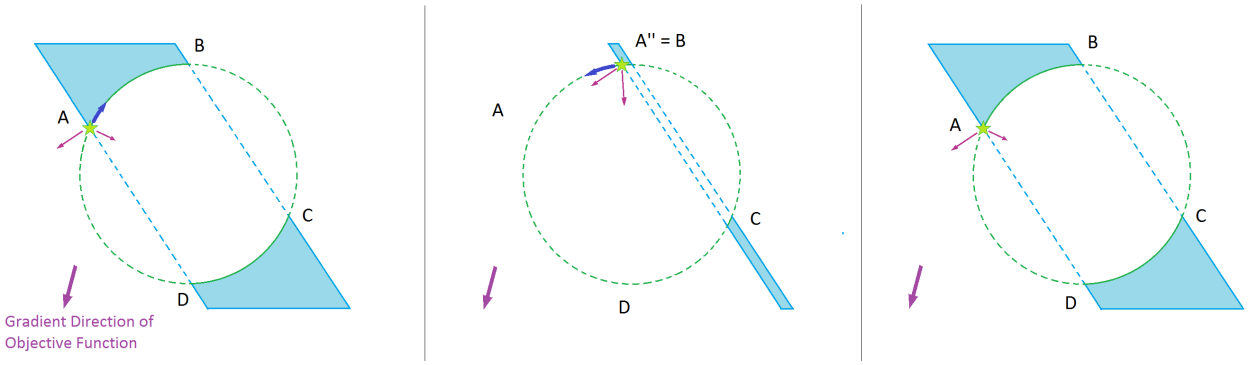


Figure 4.29: Fail to Bridge Two Disconnected Feasible Regions by Tracing An Affine Constraint: Start Tracing (Left), Reach Bound (Middle), and Return to Initial State (Right)

follow this curve, we will get stuck at this singular point. Fortunately, we instead apply the FJ conditions which can smoothly pass through this point by sending the multiplier associated with the objective function to zero. After passing through this singular point, the radius of the circular hole begins to increase continuously, which further drives our solution to reach point "D" in another feasible region on the right plot of Figure 4.28.

However, if we do not follow the path defined by continuously changing the radius of the circular hole but rather follow the path defined by continuously shifting the affine constraint "AD" to the right, we cannot reach another feasible region. In Figure 4.29, an initial state is depicted on the left. After a continuous shifting of constraint "AD", we reach an upper bound in the middle plot of Figure 4.29 such that any further shifting to the right will result in infeasibility. Thus, we shift "AD" back to return the initial state as shown on the right plot of Figure 4.29. Through this particular tracing, we never have a chance to reach another feasible region.

By this toy example we demonstrate that the order of tracing curves influences the search efficiency for disconnected feasible regions and better solutions. How to order those curves to improve search efficiency is still an ongoing research topic. We list a few heuristic rules that might be useful:

- If a starting point is an eq-FJ point but not an FJ point, one may choose to relax the active inequality constraint which has the largest positive multiplier (based on the models of (3.20) and (3.22)).
- If a starting point is an FJ point but not a local minimum, one may choose to relax a constraint whose gradient at the starting point has a non-zero direction which lies in the subspace spanned by the eigenvectors of negative eigenvalues of the projected Hessian at the starting point.
- If a starting point is a local minimum, one may choose to relax active constraints first.
- When finishing a trace, choose the solution associated with the lowest objective value, if lower than current objective value, as the next starting point.
- If no solutions have lower objective values than current objective value, choose the solution that has fewest positive multipliers associated with inequality active constraints (based on the models of (3.20) and (3.22)).

In practice, however, the situation is much more complicated because we convert the original problem into its equality constrained model, and map the eq-FJ conditions into an ellipsoidal formulation by an ellipsoidal mapping. We, then, trace the curves defined by the ellipsoidal formulation. Geometrically, this procedure firstly lifts the original feasible space to an intersection of higher dimensional hypersurfaces such that the original feasible space is a projection of the intersection of these hypersurfaces. Secondly, the ellipsoidal formulation maps each hypersurface into an ellipsoidal hypersurface but retaining their intersection. Finally, we follow the curves defined by these ellipsoidal hypersurface to reach multiple eq-FJ solutions. Hence, the behaviors of the curves we follow can be very sophisticated in higher dimensional space.

Finally, we would like to compare the proposed method to the simplex method for linear programming [108]. A geometric interpretation of simplex method is searching for a nearby vertex (called basic feasible point) which has a lower objective value than the objective value of current vertex, provided the problem is feasible and bounded. In a feasible linear programming problem,

any vertex of the feasible space satisfies the eq-KKT conditions which are the KKT conditions except the inequality conditions of multipliers associated with the inequality constraints. On the other hand, since the feasible space of a linear programming problem is a polytope, any point that satisfies the eq-KKT conditions is a vertex, provided the problem is non-degenerate. Therefore, the simplex method can be regarded as a strategy that seeks for non-increasing eq-KKT points by pivoting active constraints. An analogue interpretation for our proposed monotone search strategy is that the proposed tracing method seeks for non-increasing eq-FJ points by tracing curves. Although the simplex method and our proposed tracing methods are seemingly quite different, we will explain below that the simplex method for linear programming can be regarded as a special case in the tracing methods.

A major difference is that our proposed method is designed to follow curves to search for new solutions, while the simplex method seems never follow curves to reach next basic feasible point. However, the pivoting procedure intrinsically includes a path following process when selecting entering and leaving indices. Geometrically, it follows an edge of the feasible polytope to attain a new decreasing vertex [108]. The specialty for linear programming is that since everything is affine, the next decreasing vertex can be solved directly from linear algebra. Thus, the path following process is skipped in practice. For nonlinear programming problems, however, we no longer have a linear specialty to solve the next point algebraically, thus a numerical path following process is required.

The second difference is that the simplex method only searches among vertices, while our proposed methods can search other points besides vertices[10]. This is due to the special geometry in linear programming that any eq-KKT point is a vertex and vice versa, provided non-degenerate problems. However, this is not true in general for nonlinear case. Therefore, our proposed methods also search for other non-vertex eq-FJ points.

[10]The word "vertex" in nonlinear case is defined to be a point at where the tangent space is the whole space.

4.4.4 Summary

In this section we first showed that the proposed method found all the local minima for two numerical examples 'case9mod'" and "WB5" which had been studied in [25, 26, 109]. Then, we located four additional local minima to the three known local minima for "case39mod4" 39-bus system from [26]. Furthermore, we modified "WB5" 5-bus system to demonstrate that our proposed method can identify a global solution which is not a eq-KKT point under equality constrained ACOPF model (2.31). Finally, we discussed advanced search design ideas, and argued that the simplex method for linear programming is a special case in our tracing methods.

4.5 Conclusion

This chapter starts with a detailed explanation of how to implement the branch tracing algorithm. Specifically, it introduces a predictor-corrector algorithm, explores sparse structures of Jacobian matrices for solving Newton's steps, explains step length control, bad conditioning control, and solution identification criteria. Next, this chapter applies the branch tracing method to identify multiple real solutions to the power flow problems. It reports all the solved examples as well as a bunch of other benchmark systems that have never been completely solved before. It includes two examples to illustrate the number of power flow solutions can increase with the increment of load, and another example to show a non-trivial power flow problem can admit multiple high voltage solutions. Then, this chapter introduces enumeration search and monotone search strategies for solving ACOPF problems. The proposed methods successfully located multiple local extrema for several hard problems, and reports new local minima that have never been reported before. It also includes a special example to illustrate the capability of the proposed method for finding a non-KKT global solution. A final discussion of advanced search design with an analog of simplex method concludes this chapter.

One should note that the ellipsoidal formulation guarantees the boundedness of our 1-dimensional curves in this thesis. But at present we do not have a rigorous proof for the connectivity of these 1-dimensional curves defined by the ellipsoidal formulation. The simulation results in this chapter

showed that the ellipsoidal formulation can help us connect all the 1-dimensional curves for these specific examples including the 5-bus counter example for Ma-Thorp's design. It is promising to guarantee the completeness of the real solution set under the ellipsoidal formulation.

Chapter 5

Privacy Preserving Mappings of Multi-Party OPF Problems for Cloud Computing

5.1 Overview

In this chapter we consider a different application of our mappings for privacy preserving purposes in a multi-party OPF scenario in the shared computing platforms. Recall that in Chapter 2 we showed that the encryption mapping designed individually by each party is equivalent to a sub-block encryption mapping for all the parties, provided an intersection area compliance relation. This feature allows us to share a small amount of limited information between adjacent parties while solving the optimization problem simultaneously for all parties. The encrypted data in the shared computing platforms is processed directly without any decryption, which can prevent eavesdropping from a third party. This design provides two major data security advantages: enhancing security of sensitive information for the entire electric power grid, and maintaining privacy for each commercial participant of the power grid. The first advantage is achieved by representing the ACOPF problem in a structurally non-sensitive formulation with the help of each party's encryption mapping. The second advantage comes from the consequence that the solution returned to each party can only be deciphered privately by each party, hence it preserves privacy. However, this approach has drawbacks with the price of increasing computational complexity and data storage demand. Due to the loss of a sparse data structure, we trade off security and complexity. A rank-reduced data storage technique will be introduced for reducing the data storage demand, and it will be applied to our example for illustration.

5.2 Encryption Mappings for Multi-Party ACOPF Problems

In this section we revisit the encryption mapping from Chapter 2 to mask multi-party ACOPF problems. We relax the requirement of invertibility on the affine mappings of Equation (3.33) to enhance security and privacy. It is discussed in detail and simulated in our numerical example.

5.2.1 Design of Encryption Mapping

Suppose there are N_{area} regional parties jointly operating a power grid. These parties are commercial competitors seeking an optimized state for the entire power grid. To preserve privacy of each party and to enhance computational performance of this joint optimization problem, a shared computing platform can serve as a third-party operator to accomplish these purposes. Solving multi-party ACOPF problems in a shared computing platform has two major parts: an encryption part processed by each party, and an optimization part that is executed by the computing platform. Let's first consider the encryption part.

Consider the α-th party of the equality constrained multi-party ACOPF model in Equation (2.33). We normalize each constraint by its constant term if it is nonzero. Then the objective function, non-sharing constraints and compliance relation parts can be succinctly written in quadratic form as

$$\text{Objective:} \qquad \mathbf{X}_\alpha^T \mathbf{D}_\alpha \mathbf{X}_\alpha \tag{5.1a}$$

$$\text{Non-sharing:} \qquad \left[\mathbf{X}_\alpha^T \mathbf{M}_{\alpha,i} \mathbf{X}_\alpha - \Phi_{\alpha,i}\right]_{i\in[N_{c,\alpha}]} = \mathbf{0} \tag{5.1b}$$

$$\text{Compliance:} \qquad \mathbf{e}_\alpha^T \mathbf{X}_\alpha \tag{5.1c}$$

where $N_{c,\alpha}$ is the number of the non-sharing constraints for party-α; $\mathbf{X}_\alpha$ is the variable vector with its dimension $N_{x,\alpha}$; $\Phi_{\alpha,i}$ is either 1 or 0 according to the constant term of the i-constraint.

To obfuscate the sensitive information from those matrices $\mathbf{D}_\alpha$ and $\mathbf{M}_{\alpha,i}$, party-α first generates a random matrix $\mathbf{T}_\alpha$, which is called the encryption matrix, and a random vactor γ_α, which is called the encryption vector, such that

$$\mathbf{X}_\alpha = \mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha \tag{5.2}$$

where $\mathbf{Y}_\alpha$ is the new variable vector with its dimension $N_{y,\alpha}$; $\mathbf{T}_\alpha \in \mathbb{R}^{N_{x,\alpha} \times N_{y,\alpha}}$ and $\gamma_\alpha \in \mathbb{R}^{N_{x,\alpha}}$.

To ensure that every feasible point $\mathbf{X}_\alpha$ is accessible, $\mathbf{T}_\alpha$ must be full row rank, which further requires that $N_{y,\alpha} \geq N_{x,\alpha}$. If $N_{y,\alpha} = N_{x,\alpha}$, we have an invertible encryption mappings. However, it is possible that we can relax the invertibility requirement and still have solutions, provided $rank(\mathbf{T}_\alpha) = N_{x,\alpha}$. Hence in this context we let $N_{y,\alpha} > N_{x,\alpha}$ and substitute Equation (5.2) to Equation (5.1).

Party-α generates its own linear combination matrix $\mathbf{R}_\alpha$, which is required to be invertible, and apply it to the non-sharing constraints in Equation (5.1) as well. Equation (5.1) is re-written as

$$\text{Objective:} \qquad (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha)^T \mathbf{D}_\alpha (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha) \tag{5.3a}$$

$$\text{Non-sharing:} \qquad \mathbf{R}_\alpha \Big[(\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha)^T \mathbf{M}_{\alpha,i} (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha) - \Phi_{\alpha,i} \Big]_{i \in [N_{c,\alpha}]} = \mathbf{0} \tag{5.3b}$$

$$\text{Compliance:} \qquad \mathbf{e}_\alpha^T (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha) \tag{5.3c}$$

Once each party has encrypted its model, the shared computing platform gathers the data from all the parties and, conceptually, solves the following optimization problem.

$$\textbf{Minimize} \qquad \sum_{\alpha=1}^{N_{area}} (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha)^T \mathbf{D}_\alpha (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha)$$

Subject to:

$$\text{Non-sharing:} \qquad \mathbf{R}_\alpha \Big[(\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha)^T \mathbf{M}_{\alpha,i} (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha) - \Phi_{\alpha,i} \Big]_{i \in [N_{c,\alpha}]} = \mathbf{0} \tag{5.4a}$$

$$\text{Compliance:} \qquad \mathbf{e}_{\alpha,\beta}^T (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha) - \mathbf{e}_{\beta,\alpha}^T (\mathbf{T}_\beta \mathbf{Y}_\beta + \gamma_\beta) = \mathbf{0} \tag{5.4b}$$

$$\alpha \;=\; 1, \ldots, N_{area}$$

$$\beta \; (\neq \alpha) \;=\; 1, \ldots, N_{area}$$

This is done by solving the first order conditions of Equation (5.4) as

$$\mathbf{T}_\alpha^T \Big(2\mathbf{D}_\alpha (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha) - \sum_{k=1}^{N_{c,\alpha}} 2\lambda_{\alpha,k} [\mathbf{M}_{\alpha,i} (\mathbf{T}_\alpha \mathbf{Y}_\alpha + \gamma_\alpha)]_{i \in [N_{c,\alpha}]} \mathbf{R}_{\alpha,k}^T - \sum_{j=1}^{N_{area}} \mathbf{e}_{\alpha,j} \; \mu_{\alpha,j} \Big) = \mathbf{0} \tag{5.5}$$

where $\alpha \in [N_{area}]$; $\lambda_{\alpha,k}$'s are the Lagrangian multipliers associated with the non-sharing constraints, and the corresponding column vector is denoted as λ_α; $\mu_{\alpha,j}$ contains the Lagrangian multipliers associated with the compliance relations, and the corresponding column vector is denoted as μ_α; $\mathbf{R}_{\alpha,k}$ is the k-th row of the combination matrix $\mathbf{R}_\alpha$.

Note that the unknowns in Equation (5.5) for party-α are $\mathbf{Y}_\alpha$, λ_α and μ_α. Stacking these unknwons into a column vector and stacking over all the parties, define $\zeta := [\cdots, \mathbf{Y}_\alpha^T, \lambda_\alpha^T, \mu_\alpha^T, \cdots]^T$, and denote its dimension by N_ζ. We can stack all the equations from Equation (5.4a), (5.4b) and (5.5) in a column, denoted by $\mathbf{h}(\zeta)$, over all the areas. Denote the dimension of $\mathbf{h}(\zeta)$ as N_h. Since these equations are quadratic polynomials with respect to ζ, they can be expressed generally in the form of

$$\mathbf{h}(\zeta) = \left[\zeta^T \mathbf{H}_k \zeta + \mathbf{B}_k \zeta + r_k \right]_{k\in[N_h]} = \mathbf{0} \tag{5.6}$$

where $\mathbf{H}_k \in \mathbb{R}^{N_\zeta \times N_\zeta}$ represents a quadratic matrix, $\mathbf{B}_k \in \mathbb{R}^{N_\zeta}$ is a linear vector and $r_k \in \mathbb{R}$ is a constant scalar.

Generally, these $\mathbf{H}_k$, $\mathbf{B}_k$ and r_k are the information needed for shared computing and will be sent to the computing platform conceptually.

5.2.2 Least Square Formulation

Note that Equation (5.6) is over-determined due to the design that $N_{y,\alpha} > N_{x,\alpha}$. Thus, we sum up the squares of all the equations from Equation (5.6) and reformulate it as an unconstrained least square problem.

$$\textbf{Minimize} \quad R(\zeta) = \frac{1}{2}\mathbf{h}(\zeta)^T \mathbf{h}(\zeta) \tag{5.7}$$

The Jacobian matrix of $\mathbf{h}(\zeta)$ is

$$\mathbf{J}(\zeta) = \left[2\zeta^T \mathbf{H}_k + \mathbf{B}_k \right]_{k\in[N_\zeta]} \tag{5.8}$$

The gradient of $R(\zeta)$ is given by

$$\bigtriangledown R(\zeta) = \mathbf{J}(\zeta)^T \mathbf{h}(\zeta) \tag{5.9}$$

and the Hessian of $R(\zeta)$ after dropping the residue term [108] is

$$\bigtriangledown^2 R(\zeta) = \mathbf{J}(\zeta)^T \mathbf{J}(\zeta) \tag{5.10}$$

With the gradient and the Hessian of $R(\zeta)$, the least square problem can be solved by any Hessian dependent optimization algorithms. The overall procedure is demonstrated in Figure 5.1.

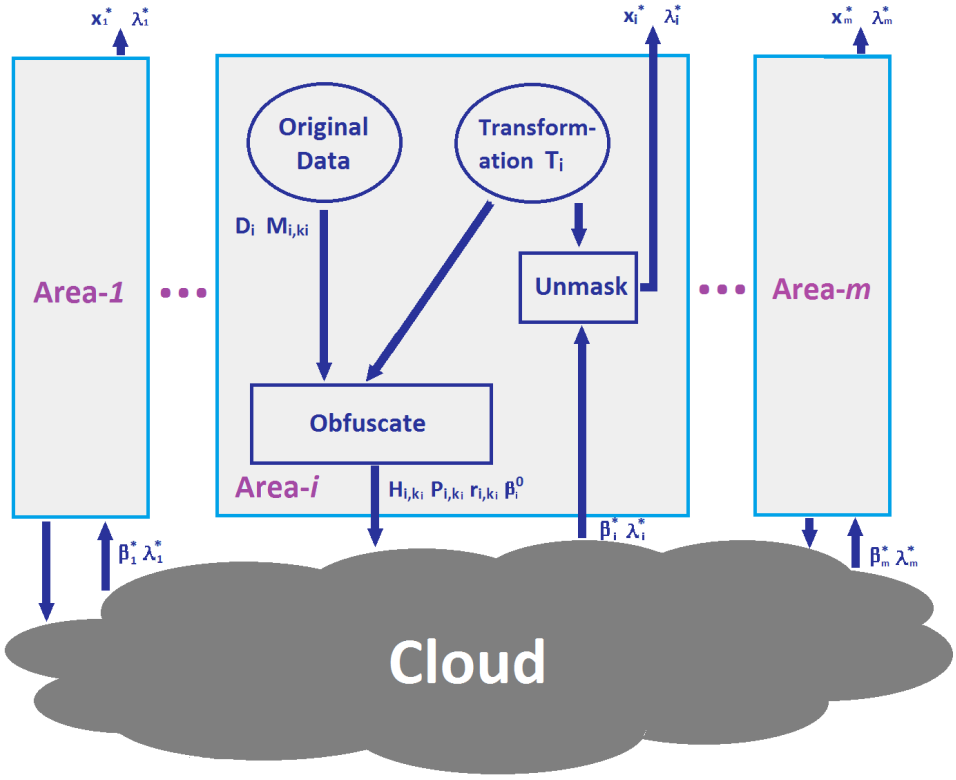


Figure 5.1: Multi-party ACOPF Encryption and Solving Procedure (Conceptual)

5.2.3 Improving Data Storage and Computation Efficiency

The formulation presented in Equation (5.4) masks an ACOPF problem by the encryption matrix $\mathbf{T}_\alpha$, encryption vector γ_α and the linear combination matrix $\mathbf{R}_\alpha$. However, it suffers from computational challenges because the sparse structure of the original matrices is likely lost during the encryption. Particularly for the least square formulation, each party must communicate $\mathbf{H}_k$, a large, dense matrix for every constraint. The amount of data describing the overall model is

immense, and needs to be simplified. To this end, we return to the original problem to exploit the low-rank properties of the matrices in the constraint equations.

Recall Proposition 3.4.2 in Chapter 3, the rank of power flow equations does not exceed four. Thus, in the equality constrained ACOPF model the rank of each quadratic matrix does not exceed six including the slack variables. This low rank structure is invariant to encryption. Although the linear combination matrix $\mathbf{R}_\alpha$ could potentially increase the rank of each matrix, it can be easily controlled by the sparsity of $\mathbf{R}_\alpha$ while keeping a certain security level. This will be seen in the next section.

Let's consider the original quadratic matrix $\mathbf{M}_{\alpha,i}$. It can be written as

$$\mathbf{M}_{\alpha,i} = \mathbf{V}_{R,\alpha,i}\mathbf{\Lambda}_{R,\alpha,i}\mathbf{V}_{R,\alpha,i}^T = \mathbf{V}_{+,\alpha,i}\mathbf{V}_{-,\alpha,i}^T \tag{5.11}$$

where the subscript 'R' indicated reduced, i.e., $\mathbf{V}_{R,\alpha,i}$ only contains eigenvectors associated with nonzero eigenvalues. On the right hand side of Equation (5.11) we incorporate the values of the eigenvectors via multiplying the eigenvectors by the square root of the associated eigenvalues. To accommodate negative eigenvalues, we negate one of the vectors. For those eigenvalues that are positive, $v_- = v_+ = \sqrt{\lambda}v$. For those eigenvalues that are negative, $v_- = -v_+ = -\sqrt{\lambda}v$. The matrices $\mathbf{V}_{+,\alpha,i}$ and $\mathbf{V}_{-,\alpha,i}$ may not be sparse, but they are very low rank, and one of their dimensions will be equal to the rank. We describe these matrices as "rank-dimensioned" matrices. This very low rank property is preserved by the encryption matrix $\mathbf{T}_\alpha$.

Before presenting the form of the relevant transformed equations, it is necessary to emphasize that we do not perform a computational eigenvalue decomposition for each constraint to obtain the eigenvectors and eigenvalues in Equation (5.4). Instead there is a simple closed-form description of the eigenvalues and eigenvectors for these matrices which has been discussed in Theorem 3.4.2 in Chapter 3.

Finally, we can reformulate Equation (5.4a), (5.4b) and (5.5) in terms of the rank-dimensioned matrices, dramatically reducing the size of the system representation.

The non-sharing equation in Equation (5.4a) are re-written as

$$\mathbf{Y}_\alpha^T\mathbf{U}_{+,\alpha,i}\mathbf{U}_{-,\alpha,i}^T\mathbf{Y}_\alpha + 2\mathbf{L}_{\alpha,i}^T\mathbf{U}_{+,\alpha,i}\mathbf{Y}_\alpha + r_{\alpha,i} = 0 \tag{5.12}$$

where $r_{\alpha,i}$ is a scalar, $\mathbf{L}_{\alpha,i}$ is a vector, and $\mathbf{U}_{+,\alpha,i}$ and $\mathbf{U}_{-,\alpha,i}$ are rank-dimensioned matrices that we will define shortly below.

The compliance relation in Equation (5.4b) is expressed as

$$\mathbf{t}_{\alpha,\beta}\mathbf{Y}_\alpha - \mathbf{t}_{\beta,\alpha}\mathbf{Y}_\beta + \mathbf{r}_{\alpha,\beta,s} - \mathbf{r}_{\beta,\alpha,s} = \mathbf{0} \tag{5.13}$$

where $\mathbf{t}_{\alpha,\beta}$, $\mathbf{t}_{\beta,\alpha}$, $\mathbf{r}_{\alpha,\beta,s}$ and $\mathbf{r}_{\beta,\alpha,s}$ are low-dimensional matrices and vectors depending on the number of the transmission lines in the intersection area between party-α and party-β, and are defined below.

The first-order conditions in Equation (5.5) can be written as

$$\mathbf{r}_{\alpha,g} + \mathbf{d}_\alpha^T\mathbf{d}_\alpha\mathbf{Y}_\alpha - \sum_{i=1}^{N_{c,\alpha}} \lambda_{\alpha,i}\mathbf{U}_{+,\alpha,i}^T\mathbf{L}_{\alpha,i} - \sum_{i=1}^{N_{c,\alpha}} \lambda_{\alpha,i}\mathbf{U}_{+,\alpha,i}^T\mathbf{U}_{-,\alpha,i}\mathbf{Y}_\alpha - \sum_{\beta=\alpha+1}^{N_{area}} \mathbf{t}_{\alpha,\beta}\mu_{\alpha,\beta} = \mathbf{0} \tag{5.14}$$

The rank-dimensioned matrices vectors, and scalars in Equations (5.12), (5.13), and (5.14) are listed below. Using

$$\begin{aligned} \mathbf{M}_{\alpha,i} &= \mathbf{V}_{-,\alpha,i}\mathbf{V}_{+,\alpha,i}^T \\ \mathbf{D}_\alpha &= \mathbf{d}_{\alpha,u}\mathbf{d}_{\alpha,u}^T \end{aligned}$$

we define

$$\begin{aligned} \mathbf{\Gamma}_{+,\alpha,i} &:= \left[\ \sqrt{\mathbf{R}_\alpha(i,j)}\mathbf{V}_{+,\alpha,j}^T\ \right]_{\forall j\in Z_i} \\ \mathbf{\Gamma}_{-,\alpha,i} &:= \left[\ \sqrt{\mathbf{R}_\alpha(i,j)}\mathbf{V}_{-,\alpha,j}^T\ \right]_{\forall j\in Z_i} \\ \mathbf{U}_{+,\alpha,i} &:= \mathbf{u}_{\alpha,i}\mathbf{\Gamma}_{+,\alpha,i}\,\mathbf{T}_\alpha \\ \mathbf{U}_{-,\alpha,i} &:= \mathbf{u}_{\alpha,i}\mathbf{\Gamma}_{-,\alpha,i}\,\mathbf{T}_\alpha \\ \mathbf{L}_{\alpha,i} &:= \mathbf{u}_{\alpha,i}\mathbf{\Gamma}_{-,\alpha,i}\,\gamma_\alpha \\ \mathbf{d}_\alpha &:= \mathbf{u}_{\alpha,d}\mathbf{d}_{\alpha,u}^T\,\mathbf{T}_\alpha \\ \mathbf{t}_{\alpha,\beta} &:= \mathbf{e}_{\alpha,\beta}^T\,\mathbf{T}_\alpha \\ r_{\alpha,i} &:= \sum_{j\in Z_i} \mathbf{R}_\alpha(i,j)\left(\gamma_\alpha^T\mathbf{M}_{\alpha,j}\gamma_\alpha - \Phi_{\alpha,j}\right) \\ \mathbf{r}_{\alpha,g} &:= \mathbf{T}_\alpha\mathbf{D}_\alpha\gamma_\alpha \\ \mathbf{r}_{\beta,\alpha,s} &:= \mathbf{e}_{\alpha,\beta}^T\gamma_\alpha, \end{aligned} \tag{5.15}$$

where $\mathbf{R}_\alpha(i,j)$ is the (i,j)-th entry of the linear combination matrix $\mathbf{R}_\alpha$; Z_i is the index set of the nonzero entries of the i-th row of $\mathbf{R}_\alpha$; $\mathbf{u}_{\alpha,i}$ and $\mathbf{u}_{\alpha,d}$ are arbitrary unitary matrices.

Therefore, the practical procedure of solving the rank-demensioned multi-party ACOPF in the shared platform is demonstrated in Figure 5.2.

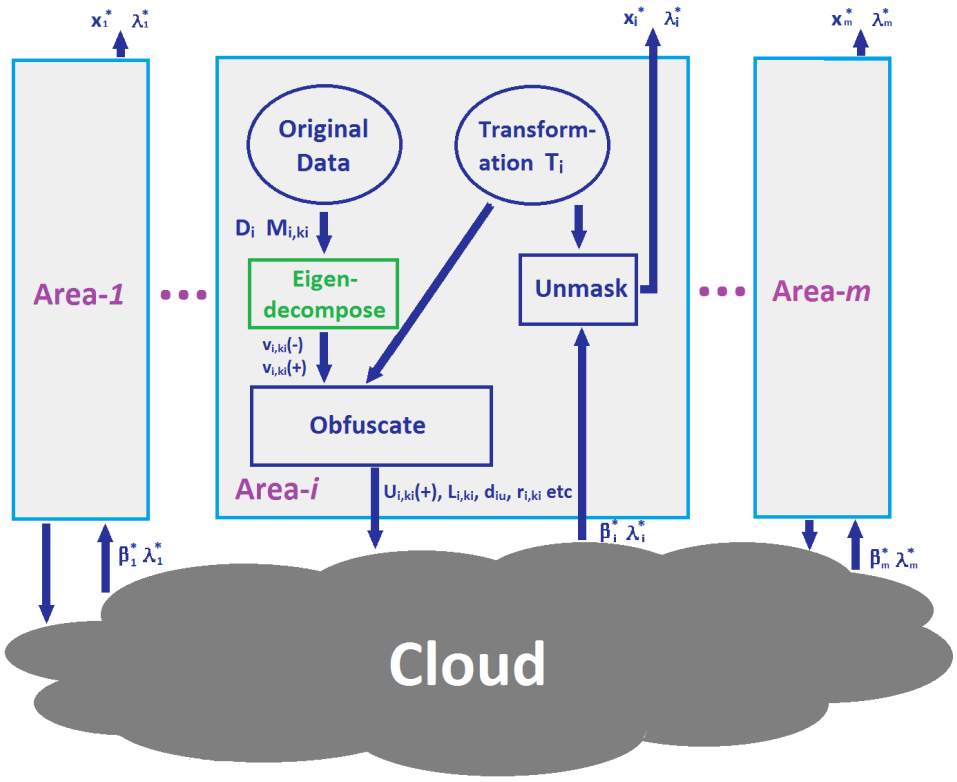


Figure 5.2: Multi-party ACOPF Encryption and Solving Procedure (Practical)

It's useful to compare how system models scale with the number of buses, generators, and lines, and we can estimate the memory required for the matrices of the dense system model in Equation (5.6) and the rank-dimensioned matrices in Equation (5.16). We can also estimate the memory required for the original model in Equation (2.33) for reference, however for that system the sparse representation will be best and nearly negligible in size.

Let $N_{r,\alpha}$ be the total rank of all the constraints, $\Omega_{masked,full}$ be the total number of the elements in the matrices used in Equation (5.6), and $\Omega_{masked,reduced}$ be the total number of the elements in the rank-dimensioned matrices presented in this section. $\Omega_{masked,full}$ appears to be $O(N^3)$ while

$\Omega_{masked,reduced}$ is comparably $O(N^2)$. For example, let $\mathbf{R}_\alpha$ be a monomial matrix, then

$$\Omega_{masked,full} \approx (N_{y,\alpha} + N_{c,\alpha})^3 \tag{5.16}$$

The total number of the elements in the rank-dimensioned matrices is much smaller and approximated by

$$\Omega_{masked,reduced} \approx 2(N_{y,\alpha} + N_{c,\alpha})N_{r,\alpha} \tag{5.17}$$

where $N_{r,\alpha} \approx 3N_{c,\alpha}$.

By the comparison it shows that the rank-dimensioned representation reduces the data storage demand drastically compared to the encrypted full model. This result will be further illustrated in the numerical example later in this chapter.

5.2.4 Privacy Analysis

This section will discuss security issues at three different levels. A preliminary question should be addressed to clarify what kind of information need to be obfuscated.

The first and foremost security concern comes from protecting power grids against malicious attacks. If the topology of a network and some actual values of operating units are revealed, an adversary may be capable of identifying the most vulnerable parts of the grid [113]. This can further enable the adversary to design malicious controls and attacks to the actual systems, causing power blackout and equipment damage. Thus, the network topology and its parameter values should be kept confidential. Another concern originates from competitive commercial activities. The power market participants are unwilling to share their commercial information to each other. Hence, the solution to each participant should be hidden from others. In summary, the network topology, the power grid parameter values and the solution for each participant should be obfuscated.

To illustrate security validity, we consider three different levels of information leakage in the cloud. At the beginning level we analyze particular information of encrypted data under an assumption that an adversary is educated with general ACOPF models but has no specific knowledge about the area he is attacking. We call this situation the ”blind attack”. Beyond “blind attack” we

assume that the attacker not only is educated with the general model but also applies the correct topology information of the area in his attack to identify the parameter values of the original system. We call this situation the "knowledgeable attack". Finally we assume that the attacker has acquired the topology and the parameter values of the grid, but little knowledge of linear combination matrices. We are interested in whether he can recover the ACOPF solution from the shared computing platform. It is called the "solution attack". Each level is more severe than the previous one as the attacker is assumed to be more knowledgeable about the grid. Our analysis shows that our encryption design makes it extremely hard for an adversary to retrieve any more information than what he already knows.

5.2.4.1 Security Analysis of Blind Attacks

Suppose an adversary has full authority to access all the uploaded data of party-α from the shared computing platform, i.e., $\mathbf{U}_{+,\alpha,i}$, $\mathbf{U}_{-,\alpha,i}$, $\mathbf{L}_{\alpha,i}$, $\mathbf{d}_{\alpha}$, $\mathbf{t}_{\alpha,\beta}$, $r_{\alpha,i}$, $\mathbf{r}_{\alpha,g}$ and $\mathbf{r}_{\alpha,\beta,s}$. And suppose he is well-educated with the general ACOPF models but has no specific knowledge about the grid of party-α, for example, no knowledge of the topology and the parameter values. The first concern is whether the rank information of the original constraints in Equation (5.4a) of party-α is deducible from the uploaded data. If this information is revealed, then through a combinatorial search the topology may be identified.

Recall in Equation (5.4a) that the linear combination matrix $\mathbf{R}_{\alpha}$ changes every single constraint in Equation (5.1b). The purpose of this refinement is to obscure the special rank information of each original constraint. The exact rank information can be revealed if and only if the original uncombined matrices in Equation (5.1b) are known, which requires the exact knowledge of $\mathbf{R}_{\alpha}$. Since $\mathbf{R}_{\alpha}$ is kept privately by party-α, the rank information is also kept private. The difficulty in recovering $\mathbf{R}_{\alpha}$ comes from the following observations:

1. The number of the choices of the nonzero structures of $\mathbf{R}_{\alpha}$ is huge. For example, suppose $\mathbf{R}_{\alpha}$ has a certain structure that each row only has three nonzero elements including one of them on the diagonal, then the total number of the possible choices of $\mathbf{R}_{\alpha}$ is $\binom{N_{c,\alpha}-1}{2}^{N_{c,\alpha}}$.

2. There is no other information to check if a guess of the structure of $\mathbf{R}_\alpha$ is correct in the blind attacks.

3. Even with the correct guess of the structure of $\mathbf{R}_\alpha$, the value of each nonzero element in $\mathbf{R}_\alpha$ also needs to be identified. This is shown to be hard in the discussion of solution attacks.

On the other hand, the adversary can try to break up $\mathbf{U}_{+,\alpha,i}$ and $\mathbf{U}_{-,\alpha,i}$ into its components, i.e., $\mathbf{u}_{\alpha,i}$, $\mathbf{\Gamma}_{+,\alpha,i}$, $\mathbf{\Gamma}_{-,\alpha,i}$ and $\mathbf{T}_\alpha$. The difficulty of recovering these matrices follows from the observation that there are uncountably many choices of $\mathbf{u}_{\alpha,i}$, $\mathbf{\Gamma}_{+,\alpha,i}$, $\mathbf{\Gamma}_{-,\alpha,i}$ and $\mathbf{T}_\alpha$ that satisfy

$$\begin{aligned} \mathbf{U}_{+,\alpha,i} &= \mathbf{u}_{\alpha,i}\mathbf{\Gamma}_{+,\alpha,i}\,\mathbf{T}_\alpha \\ \mathbf{U}_{-,\alpha,i} &= \mathbf{u}_{\alpha,i}\mathbf{\Gamma}_{-,\alpha,i}\,\mathbf{T}_\alpha \end{aligned} \tag{5.18}$$

To see this, let's suppose that $\mathbf{u}^*_{\alpha,i}$, $\mathbf{\Gamma}^*_{+,\alpha,i}$, $\mathbf{\Gamma}^*_{-,\alpha,i}$ and $\mathbf{T}^*_\alpha$ satisfy Equation (5.18). Choose any arbitrary unitary matrix $\mathbf{u}$ and any arbitrary invertible matrix $\mathbf{A}$ we have

$$\begin{aligned} \mathbf{U}_{+,\alpha,i} &= (\mathbf{u}^*_{\alpha,i}\mathbf{u}^T)(\mathbf{u}\mathbf{\Gamma}^*_{+,\alpha,i}\mathbf{A}^{-1})(\mathbf{A}\mathbf{T}^*_\alpha) \\ \mathbf{U}_{-,\alpha,i} &= (\mathbf{u}^*_{\alpha,i}\mathbf{u}^T)(\mathbf{u}\mathbf{\Gamma}^*_{-,\alpha,i}\mathbf{A}^{-1})(\mathbf{A}\mathbf{T}^*_\alpha) \end{aligned} \tag{5.19}$$

It indicates that one specific solution induces a family of uncountably many solutions which obscures the correct one. Therefore, in the blind attack scenario, even with the complete obfuscated data of an area, the essential information will be safe.

5.2.4.2 Security Analysis of Knowledgeable Attacks

Suppose an adversary has full authority to access all the uploaded data of party-α, i.e., $\mathbf{U}_{+,\alpha,i}$, $\mathbf{U}_{-,\alpha,i}$, $\mathbf{L}_{\alpha,i}$, $\mathbf{d}_\alpha$, $\mathbf{t}_{\alpha,\beta}$, $r_{\alpha,i}$, $\mathbf{r}_{\alpha,g}$ and $\mathbf{r}_{\alpha,\beta,s}$. And suppose he is well-educated with general ACOPF models. Furthermore, he knows the exact topology of the grid in party-α, and has successfully decoupled the linear combination matrix for each constraint, however he achieves it. Can he decrypt the parameter values of the original model?

First of all, to apply the topological information in the attack, the adversary has to recover the quadratic matrices $\mathbf{H}_{\alpha,i}$ for party-α

$$\mathbf{H}_{\alpha,i} = \mathbf{U}_{-,\alpha,i}^{T}\, \mathbf{U}_{+,\alpha,i} = \sum_{j \in Z_j} \mathbf{R}_{\alpha}(i,j) \mathbf{T}_{\alpha}^{T} \mathbf{M}_{\alpha,j} \mathbf{T}_{\alpha} \tag{5.20}$$

where $\mathbf{H}_{\alpha,i} \in \mathbb{R}^{N_{y,\alpha} \times N_{y,\alpha}}$; $i = 1, \cdots, N_{c,\alpha}$.

Since $\mathbf{U}_{+,\alpha,i}$ and $\mathbf{U}_{-,\alpha,i}$ are known, $\mathbf{H}_{\alpha,i}$ is known. Suppose all the $\mathbf{R}_{\alpha}(i,j)$'s are known, the correct grid topology information can reveal the structure of the nonzero elements of $\mathbf{M}_{\alpha,j}$, but the values of them are kept unknown. $\mathbf{T}_{\alpha}$ is unknown as well.

Solving the cubic matrix Equation (5.20) is equivalently to solve a set of cubic equations for every entry in $\mathbf{H}_{\alpha,i}$.

$$\mathbf{H}_{\alpha,i}(m,n) = \tau_{i,m,n}(\delta) \tag{5.21}$$

where $\mathbf{H}_{\alpha,i}(m,n)$ is the (m,n)-th entry of $\mathbf{H}_{\alpha,i}$; δ is the unknown entries of $\mathbf{T}_{\alpha}$ and $\mathbf{M}_{\alpha,j}$; and $\tau_{i,m,n}(\delta)$ is the corresponding cubic equation.

Note that Equation (5.21) is generally over-determined, which has to be formulated as a least square problem by summing up all the squares of the cubic equations into a sixth-order multivariable polynomial.

$$\textbf{Minimize } \mathbf{L}(\delta) = \sum_{i=1}^{N_{c,\alpha}} \sum_{m=1}^{N_{y,\alpha}} \sum_{n=m+1}^{N_{y,\alpha}} \left(\tau_{i,m,n}(\delta) - \mathbf{H}_{\alpha,i}(m,n)\right)^2 \tag{5.22}$$

We claim that under the assumption of correct topology guess, the actual values of the original power grid can be decrypted only if Equation (5.22) reaches its global optimum.

Note that plugging the actual values of the grid in Equation (5.22) will drive it to zero, which is the global optimum. Hence, the necessary condition holds. However, since Equation (5.22) is a sixth-order multivariable polynomial, it is generically non-convex. Hence solving Equation (5.22) is NP-hard [114, 115] even without the global optimality requirement. Furthermore, the dimension of Equation (5.22) is $N_{x,\alpha} \times N_{y,\alpha} + K \times N_{c,\alpha}$, where K is a scalar depending on the network topology. The dimension increases quadratically with the increase of the power grid scale, which

will quickly blow up for practical solvers. For example, in our 30-bus simulation case the first party consists of 18 buses and 26 transmission lines. The dimension of Equation (5.22) to this single party is over 44,000. When consider the 30-bus system as a whole, the dimension increases to over 156,000.

In the knowledgeable attack scenario, with the help of correct topological information, it is theoretically possible to decrypt the sensitive information from the uploaded data but computationally difficult, especially for a large system.

5.2.4.3 Security Analysis of Solution Attacks

Suppose an adversary has obtained the original data of party-α, i.e., the topology and the parameter values. We also provide him with the nonzero structure of the linear combination matrix $\mathbf{R}_\alpha$. Can he recover the solution to the original ACOPF problem?

Recall the affine map in Equation (5.2) we have

$$\mathbf{X}^*_\alpha = \mathbf{T}_\alpha \mathbf{Y}^*_\alpha + \gamma_\alpha \tag{5.23}$$

To obtain $\mathbf{X}^*_\alpha$ the adversary needs the exact information for both the vector $\mathbf{X}^*_{\alpha,0} := \mathbf{T}_\alpha \mathbf{Y}^*_\alpha$ and the encryption vector γ_α. Let's further suppose that vector $\mathbf{X}^*_{\alpha,0}$ is already known, then the only part remains to be decrypted is the encryption vector γ_α. With all the assumptions hold, we claim that the encryption vector γ_α can be solved only if the following equations are solved

$$\begin{aligned} \mathbf{L}^T_{\alpha,i}\mathbf{L}_{\alpha,i} &= \gamma^T_\alpha \mathbf{\Gamma}^T_{-,\alpha,i}\mathbf{\Gamma}_{-,\alpha,i}\,\gamma_\alpha \\ r_{\alpha,i} &= \sum_{j\in Z_j} \mathbf{R}_\alpha(i,j)\left(\gamma^T_\alpha \mathbf{M}_{\alpha,j}\gamma_\alpha - \Phi_{\alpha,j}\right) \end{aligned} \tag{5.24}$$

where $\mathbf{R}_\alpha(i,j)$ and γ_α are the unknowns.

Note that the actual $\mathbf{R}_\alpha(i,j)$ and γ_α are one particular solution to Equation (5.24), so the necessary condition holds. However, solving Equation (5.24) does not necessarily identify the actual value of γ_α because (5.24) is under-determined.

Suppose the linear combination matrix $\mathbf{R}_\alpha$ has a minimal number of nonzero elements that keeps it nonsingular, then the number of $\mathbf{R}_\alpha(i,j)$'s is $N_{c,\alpha}$. So the cardinality of the unknowns

is $N_{c,\alpha} + N_{x,\alpha}$, greater than the number of the equations which is $2N_{c,\alpha}$. It implies that if Equation (5.24) is solvable, it has infinitely many solutions which obscure the actual one. So without the exact knowledge of the encryption vector γ_α, the original solution $\mathbf{X}^*_\alpha$ remains private.

The analysis of the above three security levels is summarized below.

Table 5.1: A summary of three security levels

	Known Information	Difficulty of Decryption
Blind Attack	Encrypted data	Combinatorially many choices of $\mathbf{R}_\alpha$ structures, infinitely many values of parameters
Knowledgeable Attack	Encrypted data, topology of party-α, $\mathbf{R}_\alpha$	Requirement of global solution to NP-hard optimizations, scales quadratically with system size
Solution Attack	Original data, structure of $\mathbf{R}_\alpha$, $\mathbf{X}^*_{\alpha,0}$	Infinitely many solutions depending on encryption vectors

5.2.5 Numerical Example

Here we provide a 30-bus system with two parties in Figure 5.3. Its data can be found in Appendix H. The circles with G's inside represent the buses with generators; the triangles represent the buses with loads; and the heavy dots represent the buses with neither generators nor load. This system is operated by two parties, shown in different colors. Party-1 in purple contains sixteen buses from Bus-1 to Bus-16 and twenty-two transmission lines; Party-2 in green contains fourteen buses from Bus-21 to Bus-34 and eighteen transmission lines. These two areas are connected by two interconnection transmission lines in orange, one of which links Bus-3 to Bus-32, and another links Bus-16 to Bus-21. To further separate two areas, we add fictitious Bus-17, Bus-18, Bus-19

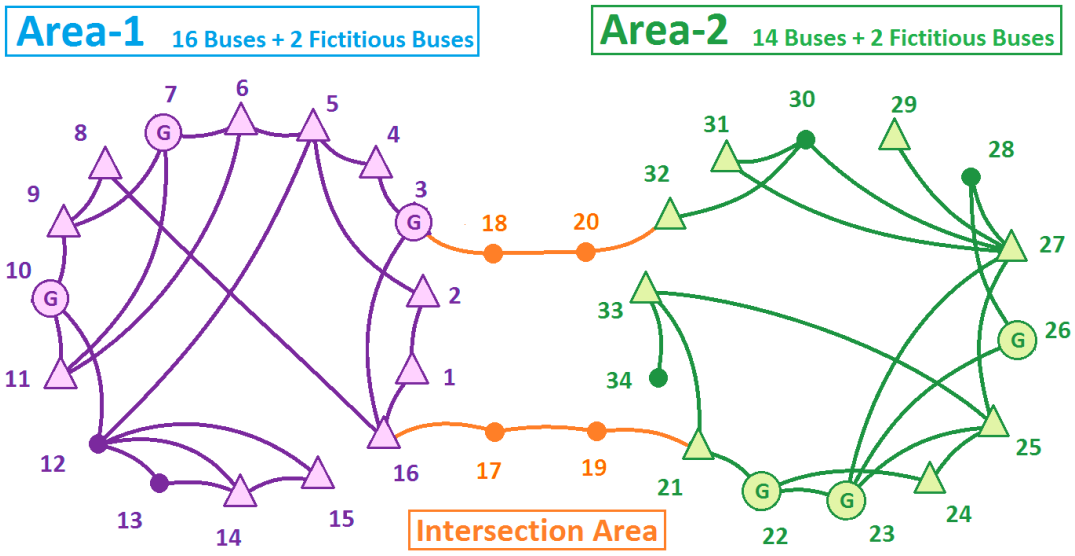


Figure 5.3: 30-Bus System with Two Parties

and Bus-20 on the interconnection transmission lines and divide each of them into three sub-lines. Note that the fictitious buses have neither generators nor loads.

Suppose Party-1 with fictitious Bus-17 and Bus-18 are operated by Party-1, while Party-2 with fictitious Bus-19 and Bus-20 are operated by Party-2. We follow our previous discussion to obtain the private encryption data, then convert the ACOPF model to the least square formulation, and finally solve all of them jointly. For easy demonstration, we choose the linear combination matrices $\mathbf{R}_1$ and $\mathbf{R}_2$ to be monomial matrices.

To mimic the loads changing, we increase the load demands on Bus-4, Bus-5, Bus-6, Bus-23 and Bus-29 simultaneously for seventy-five different cases. In each case, we choose the starting point to be the solution of the previous case and run the dog-leg algorithm to solve the least square problem. These results are verified using Matpower 5.0 [110] with the maximun error below 0.3%. The convergence performance is shown in Figure 5.4. We can see that the first few steps are profoundly efficient, converging from 10^4 to 10^{-6} within three iterations. However, as the load demands increase, it becomes difficult to reduce the least square function value to below 10^{-7}. This is attributed to a large condition number of $\mathbf{T}_1$ and $\mathbf{T}_2$ matrices. In our example, the

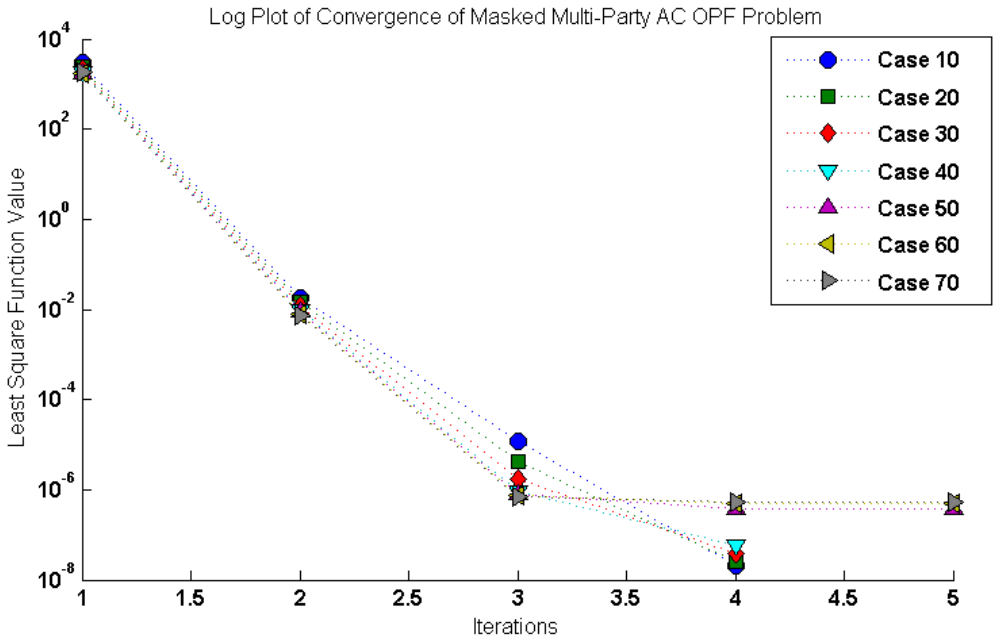


Figure 5.4: Convergence Rate for Seven Load Configurations.

condition numbers of $\mathbf{T}_1$ and $\mathbf{T}_2$ are around 100. There are several possible ways to reduce this condition number. Exploring these options is part of our ongoing research.

Next we compare the memory requirements for the example represented in the unmasked form of Equation (2.33), the masked system with full, dense matrices of Equation (5.6), and the rank-dimensioned matrices of Equation (5.16). The memory needs are estimated by counting the total number entries in the matrices and vectors and multiplying by 64 bits (i.e. 8 bytes) for double precision. In practice a certain amount of additional reduction may be possible with efficient compression and coding. The results are shown below in Table 5.2. The unmasked, sparse system

Table 5.2: A comparison of memory requirements for Multi-Party ACOPF Representation

	Sparse	Full	Rank-d
Unmasked (Equation (2.33))	23 kB	37 MB	0.9 MB
Masked (Equation (5.6))	–	1.5 GB	–
Rank-dimensioned (Equation (5.16))	–	–	4.6 MB

uses negligible memory for this example. We also calculate the size required if the matrices were full and dense, and using the rank-dimension representation for comparison. Sparsity is clearly lost in the transformation. The rank-dimensioned representation is roughly 40 times larger than the sparse representation for the unmasked model, and the masking increases the size by roughly a factor of 5. This rank-dimension representation is dramatically lower than the full, dense matrix representation. Further research on transformations that offer sparsity in the rank-dimensioned representation is warranted.

5.3 Advanced Encryption: Embedding Technique for Linear Programming

The encryption techniques discussed in this thesis are built on a different idea than common encryption techniques. A common encryption usually focuses on how to hide information from a plain text. For example, if we want to solve an OPF problem in a cloud, the data of our OPF problem is the plain text. To avoid eavesdropping along the data transmission channel, the OPF data should be ciphered before sending to the cloud. For common encryption, it is usually the data that has been masked, not the problem. Thus, to solve the problem, the cloud should know how to decipher the data, and recover the original OPF problem. This process can successfully prevent most of the data leakage during transmission, however, it may fail to keep data secure if the cloud is not trustworthy. There are plenty of techniques that can improve the security of a cloud, but on the customer side the most confident way is not revealing the original problem to the cloud. Then, it comes to a dilemma that the cloud cannot know our problem but attempt to solve it. This is where our encryption mappings make a difference.

We change the problem representation to obfuscate the original data and keep the cloud blind of our original problem. The reformulated problem, nevertheless, admits the solution set of the original problem which can be recovered easily and privately by us.

In section 5.2 we have applied an encryption mapping defined in Chapter 3 to nonlinear multi-party ACOPF problems. The encryption mapping didn't change redundancy of an ACOPF problem, but masked all the variables and constraints. To achieve better security, in this section we

will return to the linear programming scenario and discuss an advanced idea that can increase the problem redundancy for obfuscation.

5.3.1 Embedding Technique

The basic idea of embedding technique is to embed the original problem in a larger problem, and mask the larger problem by the techniques in [74, 75]. Two key steps need to be specifically investigated: how to design a larger problem that embeds the original problem while preserving the solution, and how to make the original problem obscure in the larger problem. These two questions are basically related with each other.

5.3.1.1 Problem Integration

Consider a linear programming problem LP($\mathbf{F}_1$)

$$\textbf{Minimize} \quad \mathbf{p}_1^T \mathbf{x}_1 \tag{5.25a}$$

$$\textbf{Subject to:} \quad \mathbf{A}_1 \mathbf{x}_1 \leq \mathbf{a}_1 \tag{5.25b}$$

$$\mathbf{B}_1 \mathbf{x}_1 = \mathbf{b}_1 \tag{5.25c}$$

where $\mathbf{x}_1 \in \mathbb{R}^{n_1}$ is the decision variable vector, $\mathbf{p}_1 \in \mathbb{R}^{n_1}$ is the cost function coefficient vector, $\mathbf{A}_1 \in \mathbb{R}^{m_{11} \times n_1}$ and $\mathbf{B}_1 \in \mathbb{R}^{m_{12} \times n_1}$ are the constant matrices for inequality and equality constraints, $\mathbf{a}_1 \in \mathbb{R}^{m_{11}}$ and $\mathbf{b}_1 \in \mathbb{R}^{m_{12}}$ are the corresponding constant vectors.

To construct a larger problem, let's first introduce a feasible synthetic problem LP($\mathbf{F}_2$)

$$\textbf{Minimize} \quad \mathbf{p}_2^T \mathbf{x}_2 \tag{5.26a}$$

$$\textbf{Subject to:} \quad \mathbf{A}_2 \mathbf{x}_2 \leq \mathbf{a}_2 \tag{5.26b}$$

$$\mathbf{B}_2 \mathbf{x}_2 = \mathbf{b}_2 \tag{5.26c}$$

where $\mathbf{x}_2 \in \mathbb{R}^{n_2}$ is the decision variable vector, $\mathbf{p}_2 \in \mathbb{R}^{n_2}$ is the cost function coefficient vector, $\mathbf{A}_2 \in \mathbb{R}^{m_{21} \times n_2}$ and $\mathbf{B}_2 \in \mathbb{R}^{m_{22} \times n_2}$ are the constant matrices for inequality and equality constraints, $\mathbf{a}_2 \in \mathbb{R}^{m_{21}}$ and $\mathbf{b}_2 \in \mathbb{R}^{m_{22}}$ are the corresponding constant vectors.

We require that (5.26) is feasible, and has a similar structure as (5.25). Problem (5.26) is artificial for the purpose of introducing redundant constraints and variables.

A larger problem LP($\mathbf{F}$) that embeds (5.25) and retains its solution can be simply constructed by direct integration of (5.25) and (5.26)

$$\textbf{Minimize} \quad \mathbf{p}^T\mathbf{x} \tag{5.27a}$$

$$\textbf{Subject to:} \quad \mathbf{A}\mathbf{x} \leq \mathbf{a} \tag{5.27b}$$

$$\mathbf{B}\mathbf{x} = \mathbf{b} \tag{5.27c}$$

where

$$\mathbf{x} := \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \in \mathbb{R}^{n_1+n_2} \tag{5.28a}$$

$$\mathbf{p} := \begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \end{bmatrix} \in \mathbb{R}^{n_1+n_2} \tag{5.28b}$$

$$\mathbf{A} := \begin{bmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_2 \end{bmatrix} \in \mathbb{R}^{(m_{11}+m_{21})\times(n_1+n_2)} \tag{5.28c}$$

$$\mathbf{B} := \begin{bmatrix} \mathbf{B}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_2 \end{bmatrix} \in \mathbb{R}^{(m_{12}+m_{22})\times(n_1+n_2)} \tag{5.28d}$$

$$\mathbf{a} := \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \end{bmatrix} \in \mathbb{R}^{m_{11}+m_{21}} \tag{5.28e}$$

$$\mathbf{p} := \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} \in \mathbb{R}^{m_{12}+m_{22}} \tag{5.28f}$$

Since (5.26) and (5.25) are decoupled in (5.27), their solutions comprise the solution of (5.27), and (5.27) preserves the solution of (5.25). Geometrically, the overall feasible space of (5.27) is a direct product of each individual feasible space of (5.25) and (5.26).

$$Direct Feasible\ Space\ (5.27) = Feasible\ Space\ (5.25) \times Feasible\ Space\ (5.26) \tag{5.29}$$

where "$\times$" represents the direct product operation.

5.3.1.2 Obscure Coupling

Although (5.27) simultaneously introduces redundant information and preserves solution of (5.25), it, nevertheless, provides a sub-diagonal structure in $\mathbf{A}$ and $\mathbf{B}$ which can be utilized by adversaries to break up the embedding. To overcome this drawback, one must obscure the structural information in $\mathbf{A}$ and $\mathbf{B}$.

Since $\mathbf{B}$ is associated with equality constraints, a linear mapping will destroy its special structure while preserving its solution set. Specifically, we apply an invertible matrix $\mathbf{T}_b$ on the left hand side of (5.27c)

$$[\mathbf{T}_b\mathbf{B}]\mathbf{x} = \mathbf{T}_b\mathbf{b} \tag{5.30}$$

However, this technique cannot be applied to inequality constraints (5.27b) because it changes the feasible set of inequalities. To eliminate the special structure in $\mathbf{A}$, one has to replace the off-diagonal zero sub-matrices by some nonzero entries. Geometrically, this process will couple two sub-problems together, and is called the coupled product of feasible spaces.

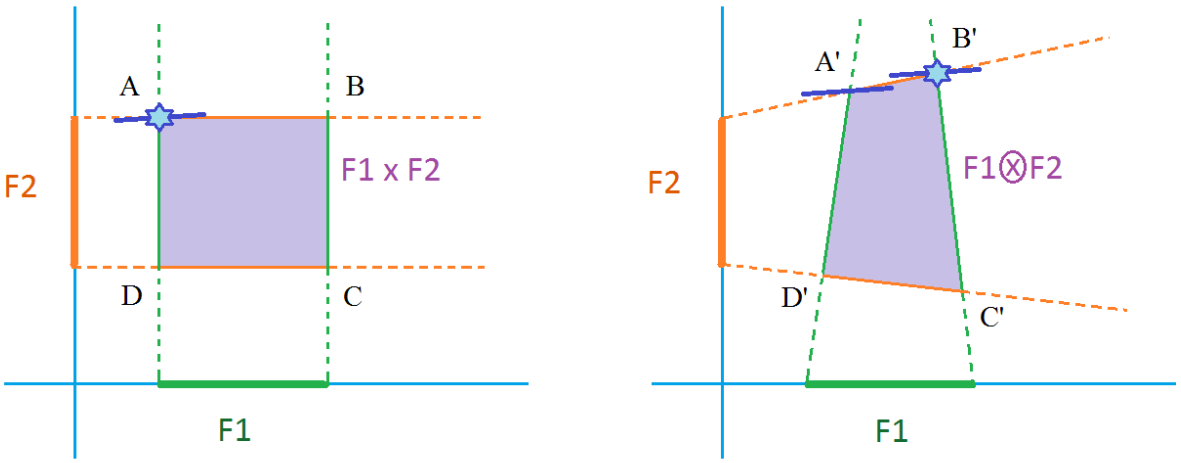
$$Coupled Feasible\ Space\ (5.27) = Feasible\ Space\ (5.25) \otimes Feasible\ Space\ (5.26) \tag{5.31}$$

where "$\otimes$" represents the coupled product operation.

Once the zero sub-matrices have been replaced, the originally decoupled sub-problems (5.25) and (5.26) become coupled together. This coupling process, if designed carelessly, will completely alter the problem and lose the track of the original solution. We demonstrate this behavior by a 2-dimensional example in Figure 5.5.

First consider the left plot in Figure 5.5. It illustrates a direct product "$\times$" of two feasible regions $F1$ and $F2$ in $\mathbb{R}^2$. The facets defined by $F1$ are perpendicular to the facets defined by $F2$ and vice versa, which is the geometric interpretation of the decoupled problem in (5.27). Suppose the vertex labeled "A" is the optimal solution on $F1 \times F2$, then it also includes the optimal solutions for both sub-problems $F1$ and $F2$.

If we replace the off-diagonal zero sub-matrices with nonzero entries, the overall feasible space is coupled in the sense that the facets defined by $F1$ are no longer perpendicular to the facets defined by $F2$. The right plot of Figure 5.5 depicts the coupled case. Note that in Figure 5.5



Direct Product of Two Regions

Coupled Product of Two Regions

Figure 5.5: Direct Product of Two Feasible Regions (Left) and Coupled Product of Two Feasible Regions (Right)

all the facets and vertices are altered, and the original optimal point "A", which alters to "A'", is no longer the optimal solution for the coupled problem. The optimal solution becomes a totally different point "B'" after coupling. To avoid the loss of the original optimal solution, we have to carefully design the coupling sub-matrices such that the active constraints preserve after coupling. To achieve this goal, a focus point $(\mathbf{x}_1^*, \mathbf{x}_2^*)$ is introduced for LP($\mathbf{F}_1$).

Let's consider the set defined by

$$\left\{ \mathbf{x}_1 \in \mathbb{R}^{n_1},\ \mathbf{x}_2 \in \mathbb{R}^{n_2} \middle| \begin{bmatrix} \mathbf{A}_1 & \mathbf{C}_1 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \\ \mathbf{x}_2 \end{bmatrix} \leq \mathbf{a}_1 \right\} \tag{5.32}$$

where the coupling sub-matrix $\mathbf{C}_1$ is determined (not necessarily unique) by

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{C}_1 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1^* \\ \\ \mathbf{x}_2^* \end{bmatrix} = \mathbf{a}_1 \tag{5.33}$$

Note that (5.32) determines a conic object in $\mathbb{R}^{n_1+n_2}$ for which the cross section with any fixed $\mathbf{x}_2$ has the same shape of the feasible space of LP($\mathbf{F}_1$). Similarly, one can pick another focus point for LP($\mathbf{F}_2$) and calculate the coupling sub-matrix $\mathbf{C}_2$.

The purposes of introducing focus points for sub-problems LP($\mathbf{F}_1$) and LP($\mathbf{F}_2$) are for maintaining the shape of feasible space for each individual sub-problem and coupling the big problem LP($\mathbf{F}$). However, keeping the shape of individual feasible space may not necessarily preserve the active constraints when solving the overall coupled problem. The positions of selected focus points play a crucial role in determining which constraints are active for the coupled problem. A simple rule can be applied to choose a good focus point.

Non-negativity rule for choosing a focus point for LP($\mathbf{F_1}$): Given a feasible point $(\mathbf{x}_{1,0}, \mathbf{x}_{2,0})$ for LP($\mathbf{F}$), choose a point $(\mathbf{x}_1^*, \mathbf{x}_2^*)$ such that

1. $(\mathbf{x}_1^*, \mathbf{x}_2^*)$ is not feasible for LP($\mathbf{F}$);
2. $\langle(\mathbf{x}_1^* - \mathbf{x}_{1,0}, \mathbf{x}_2^* - \mathbf{x}_{2,0}), \mathbf{p}\rangle \geq 0$.

If such a point $(\mathbf{x}_1^*, \mathbf{x}_2^*)$ exists, it is regarded as a good candidate for serving a focus point.

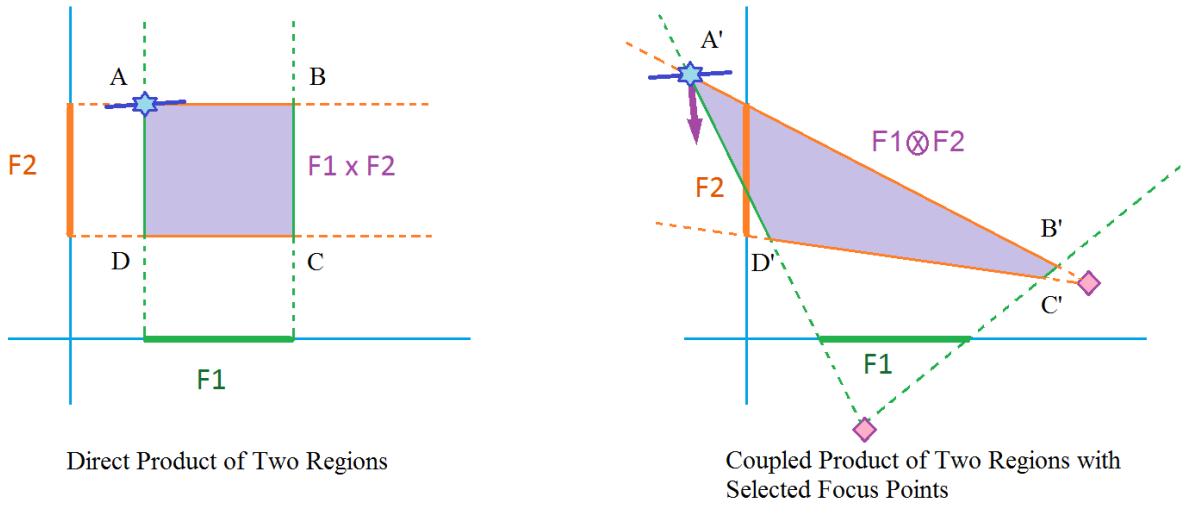


Figure 5.6: Direct Product of Two Feasible Regions (Left) and Coupled Product of Two Feasible Regions with Selected Focus Points (Right)

Geometrically, this rule attempts to select a point which, if serving as a focus point, forces the optimal point to be more "peaky" after coupling. The plot on the right of Figure 5.6 illustrates how the selected focus points force the optimal solution becoming more "peaky".

Once we have chosen a focus point for the original problem LP($\mathbf{F_1}$), we can obtain an obscure sub-matrix $\mathbf{C}_1$ by solving (5.33). Note that $\mathbf{C}_1$ is not uniquely determined. Then, we can further select a focus point for the synthetic problem LP($\mathbf{F_2}$).

Non-negativity rule for choosing a focus point for LP($\mathbf{F_2}$): Choose a point $(\mathbf{y}_1^*, \mathbf{y}_2^*)$ such that

1. $(\mathbf{y}_1^*, \mathbf{y}_2^*)$ is not in the cone of $\left\{\mathbf{x}_1 \in \mathbb{R}^{n_1},\ \mathbf{x}_2 \in \mathbb{R}^{n_2} \middle| \begin{bmatrix} \mathbf{A}_1 & \mathbf{C}_1 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}_1^* \\ \mathbf{x}_2 - \mathbf{x}_2^* \end{bmatrix} \leq \mathbf{0}\right\}$;
2. $(\mathbf{y}_1^*, \mathbf{y}_2^*)$ is not feasible for LP($\mathbf{F}$);
3. $\langle(\mathbf{y}_1^* - \mathbf{x}_{1,0}, \mathbf{y}_2^* - \mathbf{x}_{2,0}), \mathbf{p}\rangle \geq 0$.

Then, the obscure sub-matrix for LP($\mathbf{F}_2$) can be determined by solving $\begin{bmatrix} \mathbf{C}_2 & \mathbf{A}_2 \end{bmatrix} \begin{bmatrix} \mathbf{y}_1^* \\ \mathbf{y}_2^* \end{bmatrix} = \mathbf{0}$

Finally, we have to justify that $(\mathbf{x}_1^*, \mathbf{x}_2^*)$ is a good candidate.

Justification of the focus point for LP($\mathbf{F_1}$): $(\mathbf{x}_1^*, \mathbf{x}_2^*)$ is not in the cone of $\left\{\mathbf{x}_1 \in \mathbb{R}^{n_1},\ \mathbf{x}_2 \in \mathbb{R}^{n_2} \middle| \begin{bmatrix} \mathbf{C}_2 & \mathbf{A}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 - \mathbf{y}_1^* \\ \mathbf{x}_2 - \mathbf{y}_2^* \end{bmatrix} \leq \mathbf{0}\right\}$.

Although the non-negativity rule provides us a rough method to select potential focus points, it can still produce bad candidates. For example, the coupled problem under selected focus points becomes unbounded. Future investigation is needed to guarantee the preservation of feasibility and the active constraints at optimum.

5.3.2 An Illustrative Example

In this subsection we will demonstrate how to apply the embedding technique via an illustrative example.

Consider a linear programming problem LP($\mathbf{F}_1$) as follow.

$$\textbf{Minimize} \quad -x_1 + x_2 \tag{5.34a}$$

$$\textbf{Subject to:} \quad \begin{bmatrix} 0.5 & 0.5 \\ -2 & -1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 1 \\ -2 \\ 0 \end{bmatrix} \tag{5.34b}$$

To embed LP($\mathbf{F}_1$) in a larger problem, let's introduce a synthetic problem LP($\mathbf{F}_2$) for redundancy.

$$\textbf{Minimize} \quad y \tag{5.35a}$$

$$\textbf{Subject to:} \quad \begin{bmatrix} 1 \\ -1 \end{bmatrix} y \leq \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{5.35b}$$

Then, a larger problem LP($\mathbf{F}$) is constructed from a direct integration of LP($\mathbf{F}_1$) and LP($\mathbf{F}_2$).

$$\textbf{Minimize} \quad -x_1 + x_2 + y \tag{5.36a}$$

$$\textbf{Subject to:} \quad \begin{bmatrix} 0.5 & 0.5 & 0 \\ -2 & -1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ y \end{bmatrix} \leq \begin{bmatrix} 1 \\ -2 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{5.36b}$$

Now consider a coupled problem LP($\hat{\mathbf{F}}$)

$$\textbf{Minimize} \quad -x_1 + x_2 + y \tag{5.37a}$$

$$\textbf{Subject to:} \quad \begin{bmatrix} 0.5 & 0.5 & a \\ -2 & -1 & b \\ 0 & -1 & c \\ d & e & 1 \\ f & g & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ y \end{bmatrix} \leq \begin{bmatrix} 1 \\ -2 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{5.37b}$$

where a, b, c, d, e, f and g are to be determined by our coupling process.

Choose a feasible point $(1, 0, 0)$ for LP($\mathbf{F}$), and select $(-2, 3, 3)$ as a candidate of focus point for LP($\mathbf{F}_1$), we have

1. $(-2, 3, 3)$ is infeasible for LP($\mathbf{F}$);

2. $\langle(-2, 3, 3) - (1, 0, 0), (-1, 1, 1)\rangle > 0$.

Substituting this focus point in (5.37b) with inequalities being equalities, the non-unique parameters are calculated: $a = 0.167$, $b = -0.167$ and $c = 0.5$.

Select a new point $(-5, 5, 1)$ as a candidate of focus point for LP($\mathbf{F}_2$), we have

1. $(-5, 5, 1)$ is infeasible for LP($\mathbf{F}$);

2. $\langle(-5, 5, 1) - (1, 0, 0), (-1, 1, 1)\rangle > 0$.

Substituting this focus point in (5.37b) with inequalities being equalities, the non-unique parameters are calculated: $d = 0.2$, $e = 0.2$, $f = -0.1$ and $g = 0.1$.

Therefore, LP($\hat{\mathbf{F}}$) is determined as

$$\textbf{Minimize} \quad -x_1 + x_2 + y \tag{5.38a}$$

$$\textbf{Subject to:} \quad \begin{bmatrix} 0.5 & 0.5 & 0.167 \\ -2 & -1 & -0.167 \\ 0 & -1 & 0.5 \\ 0.2 & 0.2 & 1 \\ -0.1 & 0.1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ y \end{bmatrix} \leq \begin{bmatrix} 1 \\ -2 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{5.38b}$$

Solving the coupled problem LP($\hat{\mathbf{F}}$) and the decoupled problem LP($\mathbf{F}$) we observe that they share the same active constraints (constraint 1, 3, and 5) at optimal solutions.

One should note that this is an illustrative example that works with this approach. We currently do not have a set of rules that is ensured to work on every LP problem. Future research is needed to provide more rules for embedding techniques.

5.4 Conclusion

This chapter discusses how to obfuscate sensitive information for multi-party ACOPF problems. At first, an encryption mapping is introduced by each individual party. During the encryption, a rank-reduced data is generated instead of full-dense data. A cloud computes directly on the encrypted data for solving a least square formulation. The security and privacy are analyzed at three different levels, and a numerical example is presented. Then, this chapter describes an advanced encryption technique for obfuscating linear programming problems. This method integrates the original linear programming problem with a synthetic linear programming problem to form a larger problem. To further mask information, a coupling strategy is discussed, and a rule is introduced for selecting potential candidates of focus points. Finally, the feasibility of this method is demonstrated by an example.

Chapter 6

Conclusion and Future Work

This chapter summarizes important results and contributions of this thesis, and outlines open questions and research topics for future work in power system engineering.

6.1 Conclusion

This thesis is motivated by three different power engineering topics, the power flow problem, the optimal power flow problem, and the data security of cloud computing for multi-party optimal power flow problem. Each individual topic has been studied extensively over years but still leaves many open questions that are worth of in-depth investigation. In attempt to address some questions for these topics, a universal perspective appears and is the methodological core of this thesis. As introduced in Chapter 1, this perspective is called "problem representation (or reformulation)". Many non-trivial features arise with different problem representations which help us design novel tools to analyze these problems. There are many other problems beyond power system engineering that can also benefit from this methodology.

This thesis first introduces backgrounds of the three power system engineering topics in Chapter 2. It starts with a trigonometric description of the power flow problem in polar coordinates. The polar coordinates representation is commonly considered in both academia and industry for its physical meanings and a good performance in calculation. Chapter 2 briefly derives a DC approximation of the full AC power flow problem in polar coordinates. Then, it provides an algebraic representation of the power flow problem in rectangular coordinates. The rectangular coordinates representation has recently acquired more attention in academia because the semi-definite

programming favors this algebraic form rather than the trigonometric form. Next, Chapter 2 extends the algebraic representation to the optimal power flow (OPF) problem, and discusses how to convert inequality constraints into equalities. This equality-constrained OPF model is essential throughout this thesis because it enables some mappings to recast the problem while preserving the solution. Geometrically, the conversion from OPF to equality-constrained OPF lifts the feasible space of OPF to an intersection of some hypersurfaces in a higher dimensional space. To ensure the equality-constrained OPF model is quadratic, different treatments for linear and quadratic objective functions are discussed. Different models for transmission line current limits and apparent power limits are also provided. In the end, Chapter 2 details the multi-party ACOPF model which introduces fictitious buses in intersection areas to better separate adjacent parties. The information of intersection areas is required to be shared between adjacent parties for jointly optimizing operation cost, thus compliance between adjacent parties is articulated for maintaining information consistency.

Chapter 3 builds the theoretical foundations for the entire thesis. It starts with a background of real algebraic geometry with basic definitions and results. The most important concept in Chapter 3 is the "affine algebraic set preserving mapping (set mapping in short)" which maps one polynomial ring to another. It defines a class of mapping that changes the formulation of a problem but preserves an equivalent class of solution. The equivalence is specified by homeomorphisms among solutions of different problem representations. The space of set mapping is shown to be closed under map composition. This topological property becomes useful for the design of encryption mappings in multi-party ACOPF scenario.

The first set mapping detailed in Chapter 3 is called the "induced affine mapping (affine mapping in short)". It is induced by an invertible affine map from one rectangular coordinate system to another. Geometrically, an affine mapping alters nothing of an object, but rather rotates and rescales the coordinate system. This type of mapping proves useful for obfuscating the information of an object in the original coordinate system. The second set mapping articulated in Chapter 3 is called the "linear mapping" which is induced by a linear invertible map from an n-tuple of polynomial rings to another n-tuple of polynomial rings. This mapping preserves the algebraic set

of n-tuple polynomials, and can completely alter the algebraic set of each entry of the original n-tuple polynomials. It plays a crucial role in all three topics of this thesis by changing n-tuple polynomials to satisfy certain properties.

Chapter 3 continues with the construction of an ellipsoidal formulation of power flow equations. An ellipsoidal formulation is defined by a set of quadratic polynomials for which each polynomial represents a high-dimensional ellipsoid. The algebraic set of an ellipsoidal formulation is bounded and is almost everywhere smooth, given a positive dimensionality of the algebraic set. These properties benefit the branch tracing method in Chapter 4 for searching multiple real-valued solutions to power flow problems and OPF problems. Chapter 3 proves three different sufficient conditions to ensure the existence and construction of an ellipsoidal formulation. The first condition claims that if the network has no shunt elastance, its power flow problem admits an ellipsoidal formulation (not unique). A more detailed transmission model can have shunt elastance, thus another sufficient condition is proposed for an ellipsoidal formulation (not unique). The second condition only requires a particular principal sub-matrix of the bus susceptance matrix to be positive definite, which is usually true in practice. Finally, if the transmission model includes both shunt conductance and shunt elastance, a third sufficient condition, similar to the second condition, guarantees an ellipsoidal formulation of the power flow problem. An interesting proposition from the third condition indicates that the power flow problem of a lossy network always admits an ellipsoidal formulation, which further induces a byproduct of the boundedness of QV curves.

Another important result presented in Chapter 3 is the ellipsoidal formulation of the first order conditions for the ACOPF problem. While the Karush-Kuhn-Tucker (KKT) conditions are the most frequently used first order conditions for nonlinear optimization, the KKT conditions for ACOPF cannot be successfully represented in an ellipsoidal formulation. The Lagrangian multipliers in the KKT conditions only appear in cross-product terms, no univariate quadratic terms for constructing ellipsoids. To address this difficulty, Chapter 3 involves the Fritz John (FJ) conditions to introduce one extra multiplier for the objective function. The original FJ conditions requires that

all the multipliers shouldn't be zero simultaneously, leaving uncountably many choices of multipliers. To collapse the uncountable multiplier set to a finite set, a sum of squares for all the multipliers replaces the nonzero condition in the FJ conditions. The modified FJ conditions are called the sphere confined FJ conditions in this thesis. This modification enables an ellipsoidal formulation for ACOPF problems because the square terms of multipliers can be scaled to enforce a positive definite matrix. Another benefit of the sphere confined FJ conditions comes from independence of constraint qualifications. In the KKT conditions, a constraint qualification is the prerequisite for the validity of the KKT conditions. However, it is known that in nonlinear optimization problems a local solution can fail constraint qualifications, thus can fail the KKT conditions. Since the sphere confined FJ conditions do not require any constraint qualifications, they will not miss out such special solutions. The superiority of the sphere confined FJ conditions is presented in Chapter 4 by a numerical example. Chapter 3 further investigates the possibility of constructing ellipsoidal formulations for a class of quadratic constrained quadratic programming (QCQP) problem. An analysis shows constructively that any bounded QCQP problem can admit an ellipsoidal formulation for its sphere confined FJ conditions. The constructions of ellipsoidal formulations for both the power flow problem and the ACOPF problem are proved to be linear mappings, and are specifically referred as "ellipsoidal mappings". These ellipsoidal mappings send each polynomial in these problems to a high-dimensional ellipsoid but preserve their intersections. A further investigation explores the topological properties of ellipsoidal mappings and suggests that the space of ellipsoidal mapping at an ellipsoidal base is convex and closed under addition.

Finally, Chapter 3 proposes an encryption mapping for multi-party ACOPF problems. This kind of mapping is a composition of an affine mapping and an linear mapping. The affine mapping is described to mask the information of the ACOPF problem in the original coordinate system, while the linear mapping is designed to obfuscate the network topology information. A proof of equivalence ensures that the solution to the original ACOPF problem can be recovered from the solution of the obfuscated problem.

This thesis next explores innovative search strategies for identifying multiple real-valued solutions for both the power flow problem and the ACOPF problem in Chapter 4. It starts with a

detailed description of the branch tracing method which is the basic routine for the designs of efficient search strategies. Multiple real roots to the target polynomial system are then collected through the path following process.

Chapter 4 spends great efforts to improve the computational efficiency and reliability of the branch tracing method. Computationally, the most intensive part for the branch tracing method comes from the corrector step which requires solving a system of linear equations. A general dense LU factorization for solving these equations has a complexity of $O(n^3)$, which is very expensive in practice for large applications. To enhance the efficiency, Chapter 4 explores sparse structures of Jacobian matrices for ACOPF problems, which gives a nonzero diagonal sparse structure for Jacobian matrices. Such structure is favored by LU factorization because the nonzero diagonal elements can be used to eliminate off-diagonal elements, henceforth, largely accelerates the solving speed. Another issue that has been addressed for the branch tracing method is the bad conditioning of some 1-dimensional curves. Numerical error inflates rapidly through bad conditioning arcs, resulting in the failure of corrector step, and makes it hard to continue following the curve. Chapter 4 introduces a "phase II" corrector which is triggered by numerical instability. It then rescales decision variables to bring bad conditioning entries back to tolerable ranges. Once the tracing process leaves the bad conditioning arc, phase II returns to normal. Such bad conditioning control helps the branch tracing method successfully follow certain arcs with curve gradient entries around 10^9, and makes the branch tracing method much more reliable.

Next, Chapter 4 applies the branch tracing method to identify multiple real solutions of power flow problems. While previous work [16] applied the branch tracing method to the power flow problem and claimed that this approach could reliably identify all the real solutions, recent work [18] disproved it by a counter example. To enhance the chance of finding more solutions, Chapter 4 instead applies the branch tracing method to ellipsoidal formulations of power flow equations rather than directly to power flow equations. This modification helps the branch tracing method follow every curve and empirically locate the entire real solution set for every known power flow test case (including the counter example in [18]) for which the entire real solution set has been known. Chapter 4 further provides several more real solution sets solved by the proposed method

for much larger test cases which cannot be completely solved via any other existing methods within reasonable time with current computing capabilities. Chapter 4 also provides two numerical examples to show that the number of real solutions for a power flow problem can be non-monotonic with respect to the increase of load demand. The first example illustrates that by increasing the active power demand at a certain bus the number of real solutions to the power flow problem increases temporarily. The second example demonstrates that the number of real solutions can also increase with the increase of reactive power demand at some bus. Another interesting example provided in Chapter 4 admits twelve real solutions including two "high-voltage solutions". Although it is well known that a lossless mesh power grid with all PV buses and zero active power demand can admit multiple high-voltage solutions [11], the example presented in Chapter 4 has resistance on transmission lines, and non-zero active power demands at PQ buses.

Chapter 4 continues by designing a deterministic method to identify multiple local solutions to ACOPF problems. Although it has been known that ACOPF problems can be nonconvex and can admit multiple local solutions, no existing literatures except this thesis have proposed a deterministic method to reliably locate multiple local solutions. Chapter 4 introduces two deterministic search strategies to achieve this goal. The first search strategy, which is called the "enumeration search", applies the same algorithm as in the procedure of searching for multiple real solutions to the power flow problem. Specifically, it implements the branch tracing method to follow curves defined by the ellipsoidal formulation of the sphere confined FJ conditions of ACOPF problems. The ellipsoidal formulation guarantees that every curve is bounded and forms a closed loop. Thus, the enumeration search strategy can reliably return many first order solutions to the ACOPF problem, among which local minima and local maxima are identified by the second order sufficient condition. The enumeration search strategy is capable of searching for many interesting points, nevertheless, it is not an efficient strategy if a better local solution (or the global solution) is of primary interest. Thus, a more efficient search strategy, which is called the "monotone search", is introduced in Chapter 4 as well. The monotone search strategy adds one extra constraint to the original ACOPF problem, forcing the objective function values on curves to be below the objective function value at the starting solution. If a first order solution is encountered with a lower objective

function value, it abandons tracing the rest and set the new solution as a starting point. Both the enumeration search and the monotone search are testified on a few hard numerical examples which have multiple disconnected feasible regions and are inexact for their first order SDP relaxations. The enumeration search successfully identifies all the local minima for a 5-bus system and a 9-bus system. More interestingly, it locates seven local minima for a 39-bus system, among which three are known and the rest four have never been reported before. The monotone search works on the same examples and successfully identifies all the global solutions to these examples. Then, Chapter 4 presents a special example, for which the global solution does not satisfy the LICQ and the eq-KKT conditions under equality constrained ACOPF model (2.31). The global solution is verified by the third order SDP relaxation. To make a comparison, some local solvers from "GAMS" are testified on this example, none of them can identify the global solution initiated from a flat start. However, both the enumeration search and the monotone search successfully locate the global solution for this particular example.

Finally, Chapter 4 explains how the branch tracing method smoothly crosses singular points with the help of the FJ conditions, and how it bridges disconnected feasible regions to search for better solutions. Chapter 4 further shows that the tracing order of curves can significantly influence the search efficiency. Therefore, some rules for advanced search strategies are discussed. Chapter 4 concludes with a discussion of the relation between the simplex method for linear programming and the proposed methods. It claims that the simplex method is a special case of the proposed method.

Investigations regarding jointly solving multi-party ACOPF problems in the cloud are then addressed in Chapter 5. This chapter first recalls the design of encryption mapping from Chapter 3 to obfuscate the ACOPF model for each party. The cloud should gather the masked information from all the parties to generate a least square problem and solve it, conceptually. However, since the obfuscated model is very likely to lose its sparse structure during the encryption, the storage demand for masked full matrices is much larger at $O(n^3)$ than the original data demand at $O(n)$. To reduce huge data storage demand, Chapter 5 explores the eigen-structure of ACOPF problems and utilizes the low rank property (rank ≤ 6, independent of system size n) of matrices to improve

the data storage demand. Instead of storing the masked full matrices, each party only needs to store the masked eigenvectors of these matrices, which decrease the data storage from $O(n^3)$ to $O(n^2)$. Although $O(n^2)$ is still larger than $O(n)$, it is the price to pay for privacy and security. Therefore, each party first generates the eigen information of their ACOPF data (the eigen-decomposition for ACOPF matrices has closed form, thus is computationally neglectable), then obfuscate the eigen-information by an encryption mapping as discussed before. The cloud gathers the masked eigen-information from all the parties, solves a least square problem based on the encrypted information, and sends the solution to each party. During the whole process, the cloud never decryptes data, but only manipulates data for solving problems. Thus, competitive power market participants can take advantage of cloud resources without eavesdropping concerns to jointly solve the ACOPF problem for the entire area. An analysis of three different levels of security concerns is presented in Chapter 5. The first level requires no specific information leakage from a party, and claims that an adversary will find it near impossible to break down the sensitive information. The second level allows an adversary to be informed with the network topology of a party and its combination matrix. It has been shown that the original data of this party can be revealed theoretically, but computationally it could be very expensive in practice. The third level provides an adversary with the original data of a party, the structure of combination matrix, and the solution obtained from cloud. Even with so much information, the original solution of the attacked party can still remain private.

Chapter 5 then discusses an advanced encryption technique by embedding the original linear programming problem into a larger problem to further enhance security and increase the difficulty in recognizing structures in the original problem. The premise of this technique is to ensure that the solution of the original problem is easily recoverable. To construct such a larger problem, Chapter 5 first generates a synthetic problem which has similar data structures as the original problem. This synthetic problem serves as an "camouflage" for the original problem only if all the encryptions have been deciphered by an adversary. Then, the original problem and the synthetic problem are integrated directly into a larger problem which completely preserves the solutions for both problems. However, the direct integration can be easily separated since these two problems

are not entangled with each other. To couple these two problems, Chapter 5 discusses a method to preserve the binding constraints of the original problem at the optimal solution. It introduces a focus point for each problem, computes coupling matrices to force every constraint going through the focus point, and lifts the feasible space of a problem into a conic object with its cross section being the same shape of the original feasible space. Chapter 5 also introduces a rule that can help choose a good candidate of a focus point. This method couples the original problem with the synthetic problem, making them hard to be distinguished, thus enhances security. Finally, Chapter 5 demonstrate the feasibility of the embedding technique with a small illustrative example.

6.2 Publications

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

Work from Section 3.4 and Section 4.3 is published as

[89] Bernard Lesieutre, and **Dan Wu**, "An efficient method to locate all the load flow solutions-revisited," in *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2015.

Work from Section 3.5 and Section 4.4 is submitted for publication as

[111] **Dan Wu**, Daniel Molzhan, Bernard Lesieutre, and Krishnamurthy Dvijotham, "A Deterministic Method to Identify Multiple Local Extrema of the AC Optimal Power Flow Problem," to appear in *IEEE Transactions on Power Systems*.

Work from Section 3.7 and Section 5.2 are published as

[78] **Dan Wu**, Bernard C. Lesieutre, and Parmesh Ramanathan, "Feasibility of power system structure preserving linear transformations for the AC optimal power flow problem," in *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2014.

[77] Dan Wu, Bernard Lesieutre, Parmesh Ramanathan, and Bhuvana Kakunoori, "Preserving Privacy of AC Optimal Power Flow Models in Multi-Party Electric Grids," *IEEE Transactions on Smart Grid*, vol. 7, no. 4, pp. 2050-2060, June 2016.

6.3 Future Work

This work has made certain progress for three problems in power system engineering: locating multiple real roots for the power flow problem, identifying multiple local extrema for the nonlinear ACOPF problem, and preserving privacy for the multi-party ACOPF problem in cloud computing. This work also provides several promising avenues and insights for further research.

The power flow problem plays a central role in power system engineering. Although having been studied for several decades, we still lack a complete understanding of it. For example, how many real roots does a specific power flow problem have? How many of these roots are stable? How can we efficiently identify all the real roots? What is the relation between the load demand and the roots? How does the network topology influence the number of real roots? Answering these questions requires both new tools from real algebraic geometry and a deeper understanding of the structure of the power flow problem itself. This work firstly reveals a hidden geometric structure, which has never been reported before, for most power flow problems: an ellipsoidal formulation. Section 3.4 provides a few sufficient conditions to guarantee an ellipsoidal formulation of the power flow problem. These conditions are very mild and easy to be satisfied for most realistic power system cases, however, a power flow problem can still have an ellipsoidal formulation if all these conditions fail. One research topic for better understanding the geometry of power flow equations in algebraic ways is to derive a sufficient and necessary condition for an ellipsoidal formulation of the power flow problem. The ellipsoidal formulation of the power flow problem promises to provide new geometric insights for analyzing power flow real roots because high-dimensional ellipsoids are simpler than general high-dimensional surfaces determined by the original power flow quadratic equations.

Another important open question is whether the ellipsoidal formulation of the power flow problem can ensure the branch tracing method is able to locate all the real roots. An example in [24]

shows that the original power flow equations can fail the branch tracing method for enumerating all the real roots, but the proposed method with ellipsoidal formulation in Section 4.3 has successfully identified the entire real solution sets for all the existing test cases for which the real solution sets are known, including the example in [24]. Although the ellipsoidal formulation works well empirically, a theoretical guarantee is still elusive. Considering the search efficiency of the proposed method, a statement of completeness under ellipsoidal formulation would set it apart from all other existing methods for solving all the real roots of the power flow problem.

This work also provides novel tools and insights to better analyze and understand the nonlinear ACOPF problem. The ACOPF problem is studied for power system economic operation and designs, thus it is very important for modern power system engineering. Any efforts to reduce the operational cost of a wide area power grid will save a huge amount of capital. Recently, semi-definite programming (SDP) techniques have been successfully implemented in some ACOPF problems to achieve global optimality. While promising, a lot of power system models can induce non-zero duality gap, which makes the SDP relaxation inexact and results in an infeasible solution. To overcome this inexactness, [112] introduces the moment-based relaxation which can sequentially reduce the duality gap by higher order relaxations. The problem dimension of this method expands dramatically and research continues.

Section 4.4 provides another way to search for global optimality: enumerating local solutions. While existing work has attempted to enumerate multiple local solutions to look for global optimality, it is not a principled deterministic method but rather a random search with many random initial guesses. However, the proposed method in Section 4.4 is a deterministic method[1] that can search orderly for multiple local solutions, or a sequence of monotonic first order feasible solutions. To ensure the full capability of the proposed method, one important open question should be addressed: under what condition the proposed method can locate all the local extrema for the ACOPF problem. A future research direction is to provide a sufficient condition for the connectedness of all the 1-dimensional curves defined by an ellipsoidal formulation of the eq-FJ conditions

[1] It is the first deterministic method to our best knowledge for solving multiple local solutions of ACOPF problem.

of ACOPF problem. Such condition will induce the completeness of the local extremum set by the proposed method in Section 4.4, hence, can ensure global optimality.

On the other hand, identifying multiple first order points and local solutions has its own research interest. As suggested in [116], the difficulty of an optimization problem can be caused by a huge number of saddle points instead of multiple local extrema. Understanding the clustering patterns of these saddle points or first order solutions may reveal important geometric properties of the feasible space of a class of problem, and may help us design more efficient solvers. Specifically in power system engineering, the global solution of an ACOPF problem may not be the best choice for a reliable operation point. Recall the feasible space of "WB5" system in Figure 4.20, the global solution to this example lies in a tiny piece of feasible region. Such global solution should be avoided when choosing an economic and reliable operation point because the ACOPF model cannot exactly reflect the realistic situation and the parameters are varying with the change of load and other environmental factors. Thus, that tiny feasible region could disappear when the physical model varies, leaving the system operating at an infeasible state. No existing work addresses this potential jeopardy at present, and few works deal with estimating the area of a feasible region. However, by analyzing the clustering patterns of first order solutions and local extrema, this issue may be addressed. For example, given a continuous objection function and a bounded feasible region, there must exist at least one local minimum and one local maximum on it by Weierstrass. A small feasible region will enforce all the first order solutions and local extrema on it to be clustered closely. Therefore, if several first order solutions as well as at least one local minimum and one local maximum cluster closely with each other, it may provide information about the existence of a small feasible region. If the global solution happens to be among these first order solutions, one should pay particular attention to such a situation. A future research topic should address clustering patterns of the first order solutions provided by the proposed method and design informative index for testing the reliability of ACOPF solutions.

Another insight of this work comes from the observation that the proposed method is not exclusively suitable for solving ACOPF problems. As discussed in Section 4.4, any bounded quadratic constrained quadratic programming (BQCQP) problem always admit an ellipsoidal formulation.

This result leads a natural extension of the proposed method to arbitrary BQCQP problems, for example, the maximum loadability problem. Typical methods for solving the maximum loadability problem are either by continuation power flow methods at a chosen direction or by formulating it in QCQP form and applying local or global solvers. These continuation power flow methods only determine a maximal load changing in a certain loading direction, suffering from limited information about the critical load changing direction at current operation state. Other work has been done to solve the global solution of QCQP representation of the maximum loadability problem by convex relaxation techniques. It faces the difficulty that many non-radial networks are inexact and provide no physically meaningful solution. Assuming the QCQP representation of the maximum loadability problem is bounded[2], its sphere confined eq-FJ conditions can be reformulated in an ellipsoidal form, which favors the proposed method. A specific application for future investigation is to apply the proposed method for solving multiple local solutions of the maximum loadability problem. It will provide power grid operators with detailed information about different maximum loading profiles at current operation state. Another specific BQCQP model in power system engineering is the unit commitment problem which includes binary variables for choosing different generator configurations. These binary variables are bounded and can be cast in quadratic form as $b(b-1)=0$. So the unit commitment problem also has an ellipsoidal formulation for its sphere confined eq-FJ conditions. Therefore, another promising application is to solve the unit commitment problem by the proposed method.

To improve the search efficiency of the proposed method of Section 4.4, an advanced search design will also be a future research topic. As discussed in Section 4.4.3, the number of intermediate eq-FJ solutions bridging a local solution to the global solution can be very few if selected appropriately. Thus, choosing the right curve to follow and selecting the right eq-FJ point as next starting point will dramatically improve the search efficiency of the proposed method. Some ideas have been suggested in Section 4.4.3, but more principles are required to deal with different situations. A comparison between the proposed method and the simplex method for linear programming is

[2]It is bounded because a realistic power system is always constrained by all kinds of physical and engineering limits.

also presented, which further indicates that the simplex method is a very special case for the proposed method. Therefore, developing the proposed method as a generalized "simplex method" for smooth nonlinear programming can be a total new optimization tool to study hard nonlinear optimization problems.

Another potential research direction to better enhance security of cloud computing for multi-party power system optimization problems is development of an embedding technique. In Chapter 5 we successfully design an encryption mapping which transforms the original problem to another problem while preserving the solution. Although such an encryption transformation obfuscates information of the original problem, it is not theoretically impossible to recover it. To increase recovery difficulty, Section 5.3 discusses a new idea that embeds the original problem into a larger problem for deep camouflage. It has been shown feasible for linear programming problems. Future research will investigate a systematic way to generate such embedding without loss of binding constraint information. An extension of this technique to nonlinear programming problems is also anticipated.

Appendix A: Case7Salam Mod1 Data and Solutions

The parameters in per unit are listed below. The base power is 100MVA.

Table A.1: Case7Salam Mod1 Data

Bus	P Load	Q Load	From	To	r	x	b	Gen No.	P gen	Q gen	V gen
1	-0.9	-0.3	1	2	0.082	0.192	0				
2	0.478	0.039	2	3	0.067	0.171	0				
3	0.942	0.19	2	5	0.058	0.176	0	3	0.4	0	1
4	0.135	0.058	2	6	0.013	0.042	0				
5	0.183	0.127	3	4	0.024	0.100	0				
6	0.076	0.016	4	5	0.024	0.100	0				
7	0	0	5	6	0.057	0.174	0	7*	0.8	0	1
			5	7	0.019	0.059	0				
			6	7	0.054	0.223	0				

* indicates the slack bus.

Table A.2: Case7Salam Mod1 Power Flow Solutions

	Solu 1	Solu 2
V_1	$1.118 \angle 5.904°$	$0.236 \angle 109.143°$
V_2	$1.009 \angle -1.647°$	$0.704 \angle -7.863°$
V_3	$1.0000 \angle -5.204°$	$1.0000 \angle -16.432°$
V_4	$0.992 \angle -3.799°$	$0.938 \angle -11.505°$
V_5	$0.994 \angle -1.681°$	$0.896 \angle -5.171°$
V_6	$1.004 \angle -1.552°$	$0.774 \angle -6.299°$
V_7	$1.0000 \angle 0°$	$1.0000 \angle 0°$

Appendix B: Case7Salam Mod2 and Mod3 Solutions

Table B.1: Case7Salam Mod2 Power Flow Solutions

	Solu 1	Solu 2	Solu 3	Solu4
V_1	$1.116 \angle 3.642°$	$0.521 \angle -5.533°$	$0.435 \angle 17.282°$	$0.238 \angle 105.791°$
V_2	$1.007 \angle -3.933°$	$0.391 \angle -52.262°$	$0.366 \angle -51.339°$	$0.697 \angle -10.964°$
V_3	$1.000 \angle -10.166°$	$1.000 \angle -129.855°$	$1.000 \angle -125.572°$	$1.000 \angle -23.204°$
V_4	$0.989 \angle -7.020°$	$0.480 \angle -100.163°$	$0.500 \angle -96.059°$	$0.929 \angle -15.994°$
V_5	$0.991 \angle -3.136°$	$0.532 \angle -22.116°$	$0.538 \angle -22.015°$	$0.887 \angle -7.011°$
V_6	$1.003 \angle -3.396°$	$0.459 \angle -33.998°$	$0.459 \angle -32.768°$	$0.767 \angle -8.617°$
V_7	$1.000 \angle 0°$	$1.000 \angle 0°$	$1.000 \angle 0°$	$1.000 \angle 0°$

Table B.2: Case7Salam Mod3 Power Flow Solutions

	Solu 1	Solu 2	Solu 3	Solu4
V_1	$1.112 \angle 6.030°$	$0.514 \angle -1.566°$	$0.450 \angle 14.425°$	$0.238 \angle 108.815°$
V_2	$1.003 \angle -1.606°$	$0.386 \angle -50.282°$	$0.365 \angle -49.840°$	$0.698 \angle -7.988°$
V_3	$1.000 \angle -5.368°$	$1.000 \angle -128.817°$	$1.000 \angle -126.091°$	$1.000 \angle -17.021°$
V_4	$0.954 \angle -3.388°$	$0.394 \angle -96.835°$	$0.410 \angle -94.414°$	$0.898 \angle -11.321°$
V_5	$0.982 \angle -1.513°$	$0.536 \angle -19.165°$	$0.537 \angle -19.195°$	$0.884 \angle -5.032°$
V_6	$0.998 \angle -1.499°$	$0.458 \angle -32.170°$	$0.446 \angle -31.391°$	$0.768 \angle -6.354°$
V_7	$1.000 \angle 0°$	$1.000 \angle 0°$	$1.000 \angle 0°$	$1.000 \angle 0°$

Appendix C: Case5Salam Mod Solutions

Table C.1: Case5Salam Mod Power Flow Solutions

	V_1	V_2	V_3	V_4	V_5
Sol 1	$1\angle -2.824°$	$1.268\angle -10.008°$	$1.207\angle -9.395°$	$1.044\angle -7.629°$	$1.06\angle 0°$
Sol 2	$1\angle -12.714°$	$1.105\angle -19.395°$	$0.983\angle -19.294°$	$0.053\angle -74.276°$	$1.06\angle 0°$
Sol 3	$1\angle -130.629°$	$0.805\angle -132.134°$	$0.749\angle -133.139°$	$0.58\angle 169.780°$	$1.06\angle 0°$
Sol 4	$1\angle -100.672°$	$0.403\angle 148.642°$	$0.283\angle 167.578°$	$0.082\angle -164.680°$	$1.06\angle 0°$
Sol 5	$1\angle -36.496°$	$0.341\angle -166.659°$	$0.183\angle -149.805°$	$0.085\angle -96.871°$	$1.06\angle 0°$
Sol 6	$1\angle -32.135°$	$0.322\angle -140.781°$	$0.044\angle -164.968°$	$0.084\angle -88.709°$	$1.06\angle 0°$
Sol 7	$1\angle -24.395°$	$0.473\angle -48.947°$	$0.020\angle -105.494°$	$0.082\angle -80.843°$	$1.06\angle 0°$
Sol 8	$1\angle -18.423°$	$0.476\angle -44.550°$	$0.018\angle -99.168°$	$0.625\angle -25.173°$	$1.06\angle 0°$
Sol 9	$1\angle -25.332°$	$0.317\angle -140.656°$	$0.045\angle -150.415°$	$0.615\angle -32.818°$	$1.06\angle 0°$
Sol 10	$1\angle -28.491°$	$0.321\angle -163.983°$	$0.145\angle -134.645°$	$0.611\angle -39.069°$	$1.06\angle 0°$
Sol 11	$1\angle -107.936°$	$0.386\angle 133.766°$	$0.263\angle 164.826°$	$0.635\angle -121.428°$	$1.06\angle 0°$
Sol 12	$1\angle -140.282°$	$0.902\angle -142.445°$	$0.914\angle -142.897°$	$0.945\angle -143.936°$	$1.06\angle 0°$

Appendix D: ACOPF Data and Solutions of Case9mod 9-Bus System

The paramethers in per unit are listed below. The base power is 100MVA.

Table D.1: Case9mod 9-Bus System Bus Data and Line Data

Bus No.	P Load	Q Load	Vmax	Vmin	From	To	r	x	b	Imax
1	0	0	1.1	0.9	1	4	0	0.0576	0	2.5
2	0	0	1.1	0.9	4	5	0.017	0.092	0.158	2.5
3	0	0	1.1	0.9	5	6	0.039	0.17	0.358	1.5
4	0	0	1.1	0.9	3	6	0	0.0586	0	3
5	0.54	0.18	1.1	0.9	6	7	0.0119	0.1008	0.209	1.5
6	0	0	1.1	0.9	7	8	0.0085	0.072	0.149	2.5
7	0.6	0.21	1.1	0.9	8	2	0	0.0625	0	2.5
8	0	0	1.1	0.9	8	9	0.032	0.161	0.306	2.5
9	0.75	0.3	1.1	0.9	9	4	0.01	0.085	0.176	2.5

Table D.2: Case9mod 9-Bus System Generator Data

Gen.	Pmax	Pmin	Qmax	Qmin	d	c	e
1^*	2.5	0.1	3	-0.05	0.11	0.05	0.015
2	3	0.1	3	-0.05	0.085	0.012	0.06
3	2.7	0.1	3	-0.05	0.1225	0.01	0.0335

* indicates the slack bus. d is the coefficient for the quadratic term in the objective function. c is the coefficient for the linear term. e is the constant term.

Table D.3: Case9mod 9-Bus System Local Minima Obained by Enumeration Strategy

	Solu 1	Solu 2	Solu 3	Solu 4
V_1	0.9020	0.9027	0.9064	0.9095
V_2	0.9175	0.9169	0.9255	0.9218
V_3	0.9247	0.9272	0.9326	0.9388
V_4	0.9098	0.9104	0.9096	0.9127
V_5	0.9104	0.9120	0.9109	0.9160
V_6	0.9279	0.9307	0.9387	0.9426
V_7	0.9177	0.9182	0.9271	0.9284
V_8	0.9213	0.9204	0.9299	0.9291
V_9	0.9000	0.9000	0.9000	0.9000
α_1	$0°$	$0°$	$0°$	$0°$
α_2	$-9.304°$	$-11.555°$	$7.245°$	$12.367°$
α_3	$-11.150°$	$-8.619°$	$12.115°$	$7.006°$
α_4	$-5.770°$	$-5.722°$	$-0.400°$	$-0.398°$
α_5	$-10.044°$	$-9.507°$	$0.219°$	$-0.732°$
α_6	$-11.542°$	$-10.128°$	$7.592°$	$4.842°$
α_7	$-12.855°$	$-12.884°$	$4.173°$	$4.516°$
α_8	$-10.906°$	$-11.980°$	$4.548°$	$7.118°$
α_9	$-10.383°$	$-10.722°$	$-1.511°$	$-0.631°$
S_1	1.432 -j0.05	1.422 -j0.05	0.1 -j0.05	0.1 -j0.05
S_2	0.378 -j0.05	0.100 -j0.05	0.648 -j0.05	1.254 -j0.05
S_3	0.1 -j0.05	0.388 -j0.05	1.178 -j0.05	0.570 -j0.05
Cost	0.424655	0.426504	0.339797	0.308789

V_i is the voltage magnitude on the i-th bus. α_i is the voltage angle on the i-th bus. S_i is complex power generated by the i-th generator. The base power is 100MVA.

Table D.4: FJ Solutions for case9mod from the Monotone Search

	FJ Sol. 1	FJ Sol. 2	FJ Sol. 3
$\|V_1\|$	0.9020	0.9075	0.9095
$\|V_2\|$	0.9175	0.9178	0.9218
$\|V_3\|$	0.9247	0.9251	0.9388
$\|V_4\|$	0.9098	0.9107	0.9127
$\|V_5\|$	0.9104	0.9101	0.9160
$\|V_6\|$	0.9279	0.9304	0.9426
$\|V_7\|$	0.9177	0.9174	0.9284
$\|V_8\|$	0.9213	0.9211	0.9291
$\|V_9\|$	0.9000	0.9000	0.9000
θ_1	0°	0°	0°
θ_2	-9.304°	-1.982°	12.367°
θ_3	-11.150°	8.511°	7.006°
θ_4	-5.770°	-2.151°	-0.398°
θ_5	-10.044°	-2.383°	-0.732°
θ_6	-11.542°	-3.506°	4.842°
θ_7	-12.855°	-1.614°	4.516°
θ_8	-10.906°	-2.409°	7.118°
θ_9	-10.383°	-5.080°	-0.631°
S_1	$1.432-j0.05$	$0.538-j0.05$	$0.1-j0.05$
S_2	$0.378-j0.05$	$0.100-j0.05$	$1.254-j0.05$
S_3	$0.1-j0.05$	$1.281-j0.05$	$0.570-j0.05$
Cost	4246.55	3829.84	3087.89

$|V_i| \angle\theta_i$ and S_i denote the voltage and power injection at bus i in per unit. Costs are given in \$/hr.

Appendix E: ACOPF Data and Solutions of WB5 5-Bus System

The paramethers in per unit are listed below. The base power is 100MVA.

Table E.1: WB5 5-Bus System Bus Data and Line Data

Bus No.	P Load	Q Load	Vmax	Vmin	From	To	r	x	b	Imax
1	0	0	1.05	0.95	1	2	0.04	0.09	0	25
2	1.3	0.2	1.05	0.95	1	3	0.05	0.1	0	25
3	1.3	0.2	1.05	0.95	2	4	0.55	0.9	0.45	25
4	0.65	0.1	1.05	0.95	3	5	0.55	0.9	0.45	25
5	0	0	1.05	0.95	4	5	0.06	0.1	0	25
					2	3	0.07	0.09	0	25

Table E.2: WB5 5-Bus System Generator Data

Gen.	Pmax	Pmin	Qmax	Qmin	d	c	e
1^*	50	0	18	-0.3	0	0.04	0
5	50	0	18	-0.3	0	0.01	0

* indicates the slack bus. d is the coefficient for the quadratic term in the objective function. c is the coefficient for the linear term. e is the constant term.

Table E.3: WB5 5-Bus System Local Minima Obained by Enumeration (Left) and Monotone Search (Right) Strategy

	Solu 1	Solu 2	Solu 1	Solu 2
V_1	1.0137	1.0467	1.0137	1.0467
V_2	0.9540	0.9567	0.9540	0.9567
V_3	0.9500	0.9500	0.9500	0.9500
V_4	1.0143	0.9840	1.0143	0.9840
V_5	1.0500	1.0500	1.0500	1.0500
α_1	$0°$	$0°$	$0°$	$0°$
α_2	$-6.5245°$	$-3.4619°$	$-6.5245°$	$-3.4619°$
α_3	$-6.5734°$	$-3.2143°$	$-6.5734°$	$-3.2143°$
α_4	$-1.3415°$	$37.6505°$	$-1.3415°$	$37.6505°$
α_5	$3.1073°$	$45.4797°$	$3.1073°$	$45.4797°$
S_1	2.4605 +j0.2927	1.8143 +j1.2409	2.4605 +j0.2927	1.8143 +j1.2409
S_5	0.9815 -j0.3	2.2087 -j0.3	0.9815 -j0.3	2.2087 -j0.3
Cost	0.108235	0.094659	0.108235	0.094659

V_i is the voltage magnitude on the i-th bus. α_i is the voltage angle on the i-th bus. S_i is complex power generated by the i-th generator. The base power is 100MVA.

Appendix F: Case39mod4 39-Bus Additional Local Minima

Table F.1: New Local Minima to case39mod4: Bus Voltage Part I

	Sol. # 1		Sol. # 2		Sol. # 3		Sol. # 4	
Bus #	$\|V_1\|$	$\delta°$	$\|V_1\|$	$\delta°$	$\|V_1\|$	$\delta°$	$\|V_1\|$	$\delta°$
1	1.0381	-7.7180	1.0407	-8.8822	1.0381	-8.3789	1.0414	-9.7286
2	10500	-13.6302	1.0500	-13.9056	1.0500	-14.5736	1.0500	-14.8025
3	1.0311	-14.3376	1.0301	-14.2221	1.0318	-14.7228	1.0308	-14.5328
4	1.0164	-12.9683	1.0158	-12.8135	1.0176	-12.9721	1.0172	-12.7766
5	1.0171	-10.7174	1.0171	-10.7211	1.0182	-10.7163	1.0184	-10.7173
6	1.0178	-10.2212	1.0178	-10.2211	1.0189	-10.2100	1.0191	-10.2076
7	1.0141	-10.8564	1.0145	-10.9631	1.0152	-10.8776	1.0159	-11.0022
8	1.0141	-10.8809	1.0146	-11.0416	1.0152	-10.9191	1.0160	-11.1080
9	1.0338	-6.4855	1.0364	-7.5748	1.0348	-6.7911	1.0379	-8.0887
10	1.0269	-11.3978	1.0264	-11.2401	1.0283	-11.2899	1.0281	-11.1044
11	1.0234	-11.0195	1.0231	-10.9123	1.0247	-10.9427	1.0247	-10.8160
12	1.0203	-11.4129	1.0198	-11.2552	1.0217	-11.3049	1.0215	-11.1195
13	1.0239	-11.7426	1.0233	-11.5344	1.0255	-11.6035	1.0251	-11.3594
14	1.0182	-12.6373	1.0171	-12.2983	1.0200	-12.4170	1.0192	-12.0209
15	1.0027	-14.0066	0.9996	-13.0677	1.0058	-13.2255	1.0032	-12.1547
16	1.0020	-13.7728	0.9981	-12.5641	1.0056	-12.7498	1.0024	-11.3771
17	1.0148	-14.4987	1.0117	-13.8149	1.0157	-14.4508	1.0127	-13.6434
18	1.0208	-14.6582	1.0185	-14.2112	1.0217	-14.7768	1.0193	-14.2087
19	1.0080	-17.6118	1.0068	-16.4359	1.0108	-10.9482	1.0099	-9.5788
20	0.9500	-20.5983	0.9500	-19.4292	0.9500	-9.4907	0.9500	-8.1227

Table F.2: New Local Minima to case39mod4: Bus Voltage Part II

	Sol. # 1		Sol. # 2		Sol. # 3		Sol. # 4	
Bus #	$\|V_1\|$	δ°	$\|V_1\|$	δ°	$\|V_1\|$	δ°	$\|V_1\|$	δ°
21	0.9836	-10.1557	0.9781	-8.0968	0.9894	-10.5188	0.9853	-8.2832
22	0.9760	-5.1747	0.9713	-2.1926	0.9807	-7.0300	0.9776	-3.8677
23	0.9720	-4.7886	0.9664	-2.1269	0.9777	-6.1090	0.9738	-3.2773
24	0.9989	-12.9764	0.9942	-11.5725	1.0036	-12.2741	0.9998	-10.7015
25	1.0383	-14.3653	1.0375	-14.6704	1.0332	-15.6994	1.0325	-15.9161
26	1.0312	-15.2774	1.0294	-15.7300	1.0263	-18.0348	1.0238	-18.2826
27	1.0221	-15.5262	1.0196	-15.4763	1.0196	-17.0044	1.0166	-16.7739
28	1.0147	-14.8419	1.0131	-16.2208	1.0079	-20.7534	1.0060	-21.8038
29	1.0052	-13.7943	1.0036	-15.4733	0.9980	-20.7266	0.9966	-22.0356
30	1.0486	-13.6302	1.0486	-13.9056	1.0486	-14.5736	1.0486	-14.8025
31	0.9500	0	0.9500	0	0.9500	0	0.9500	0
32	0.9900	-11.3978	0.9896	-11.2401	0.9913	-11.2899	0.9911	-11.1044
33	0.9500	-17.6357	0.9500	-16.4630	0.9500	-10.9642	0.9500	-9.5974
34	0.9552	-20.6398	0.9566	-19.4751	0.9552	-3.6968	0.9563	-2.3393
35	0.9500	-0.3855	0.9500	4.0725	0.9500	-4.6879	0.9500	-0.0062
36	0.9611	4.9342	0.9552	7.7125	0.9670	3.4967	0.9630	6.4074
37	1.0130	-14.2653	1.0122	-14.6704	1.0080	-15.6994	1.0074	-15.9161
38	0.9575	-10.8805	0.9558	-13.2279	0.9500	-20.0854	0.9500	-21.9683
39	1.0213	-3.3572	1.02442	-5.0848	1.0217	-3.8438	1.0253	-5.9033

Table F.3: New Local Minima to case39mod4: Power Generation

	Sol. # 1		Sol. # 2		Sol. # 3		Sol. # 4	
Gen. #	P	Q	P	Q	P	Q	P	Q
1	0	1.4000	0	1.4000	0	1.4000	0	1.4000
2	6.46	0.5511	6.46	0.5507	6.4600	0.5113	6.4600	0.5027
3	0	1.5000	0	1.5000	0	1.5000	0	1.5000
4	0	0.5339	0	0.6054	0	0.3568	0	0.4153
5	0	0.7243	0	0.8012	5.0800	0.7255	5.0800	0.7868
6	5.2816	0.0734	6.8700	0.5351	2.5976	-0.3985	4.2670	-0.1056
7	5.8000	0	5.800	0	5.8000	0	5.8000	0
8	0	0	0	0	0	0	0	0
9	2.9827	-1.5000	2.2734	-1.5000	0.5882	-1.4680	0	-1.3570
10	11.0000	-1.0000	10.1394	-1.0000	11.0000	-1.0000	9.9438	-1.0000
Cost $/hr	841.74066		848.92950		842.81718		847.54988	

Bus power injections are in per unit with a base power of 100 MVA.

Appendix G: Solutions of WB5mod 5-Bus System

Table G.1: MIPS Solution, Intermediate Stationary Point and Global Solution for WB5mod

	MIPS Sol.	Intermediate Sol.	Global Sol.
$\|V_1\|$	1.1000	0.9881	1.0892
$\|V_2\|$	1.03202	0.9053	1.0098
$\|V_3\|$	1.0255	0.9000	1.0045
$\|V_4\|$	0.9000	0.9000	0.9000
$\|V_5\|$	0.9232	0.9461	0.9358
θ_1	$0°$	$0°$	$0°$
θ_2	$-6.8867°$	$-10.1765°$	$-8.6595°$
θ_3	$-7.0106°$	$-10.5328°$	$-8.9542°$
θ_4	$-15.3728°$	$-38.0141°$	$-36.5139°$
θ_5	$-11.5972°$	$-36.1330°$	$-34.9495°$
S_1	$3.074+j0.4$	$3.5709+j0.4047$	$3.73+j0.4$
S_5	$0.3982-j0.299$	$0.1651+j0.2840$	$0+j0.1785$
Cost	161921.15	151310.41	139875.00

$|V_i|\angle\theta_i$ and S_i denote the voltage and power injection at bus i in per unit. Costs are given in \$/hr.

Table G.2: Summary of Solutions Obtained from KNITRO, IPOPTH, BARON, 2nd Order SDP, 3rd Order SDP, and Proposed Tracing Method for WB5mod

	KNITRO	IPOPTH	SNOPT	BARON	2^{nd} SDP	3^{rd} SDP	Tracing
P_1 (p.u.)	3.074	3.074	3.074	3.73	3.7239	3.7297	3.73
P_5 (p.u.)	0.398	0.398	0.398	0	4e-4	6.3e-5	0
Cost (\$/hr)	161925	161925	161925				139875
Lower Bd				139873	139467	139870	

Appendix H: 30-bus Multi-party ACOPF System Data

Table H.1: Party-2 Bus and Transmission Line Data

Bus No.	P Load	Q Load	Vmax	Vmin	From	To	r	x	b	Imax
21	0.112	0.043	1.05	0.95	21	19	0.0315	0.0662	0	1.05
22	0	0	1.05	0.95	32	20	0.03	0.06	0	1.05
23	0.192	0.052	1.05	0.95	22	21	0.0662	0.1304	0	1.05
24	0.024	0.012	1.03	0.95	25	33	0.0192	0.0575	0.0528	1.05
25	0.076	0.016	1.05	0.95	26	31	0.0452	0.1652	0.0408	1.05
26	0	0	1.05	0.95	23	25	0.057	0.1737	0.0368	1.05
27	0.392	0.13	1.03	0.95	24	25	0.0132	0.0379	0.0084	1.05
28	0	0	1.03	0.95	23	26	0.0472	0.1983	0.0418	1.05
29	0.225	0.105	1.03	0.95	23	27	0.0581	0.1763	0.0374	1.05
30	0	0	1.03	0.95	25	27	0.0119	0.0414	0.009	1.05
31	0.058	0.02	1.03	0.95	26	28	0.046	0.116	0.0204	1.05
32	0.328	0.0645	1.03	0.95	27	30	0.267	0.082	0.017	1.05
33	0.112	0.075	1.05	0.95	27	29	0.012	0.042	0.009	1.05
34	0	0	1.03	0.95	27	28	0	0.208	0	1.05
20	0	0	1.05	0.95	22	24	0	0.556	0	1.05
19	0	0	1.05	0.95	30	32	0	0.208	0	1.05
18	0	0	1.05	0.95	30	31	0	0.11	0	1.05
17	0	0	1.05	0.95	22	23	0	0.256	0	1.05
					33	34	0	0.14	0	1.05
					33	21	0.1231	0.2559	0	1.05
					18	20	0.03	0.06	0	1.05
					17	19	0.0315	0.0662	0	1.05

Table H.2: Party-1 Bus and Transmission Line Data

Bus No.	P Load	Q Load	Vmax	Vmin	From	To	r	x	b	Imax
1	0.085	0.018	1.05	0.95	1	16	0.091	0.1997	0.03	1.05
2	0.09	0.058	1.03	0.95	1	2	0.0524	0.1923	0.04	1.05
3	0.082	0.039	1.05	0.95	16	3	0.073	0.2185	0.05	1.05
4	0.495	0.084	1.03	0.95	3	4	0.0639	0.1292	0.01	1.05
5	0.122	0.047	1.03	0.95	4	5	0.034	0.068	0	1.05
6	0.175	0.112	1.05	0.95	6	5	0.0936	0.209	0.04	1.05
7	0	0	1.05	0.95	5	2	0.0324	0.0845	0	1.05
8	0.082	0.036	1.05	0.95	14	15	0.0348	0.0749	0	1.05
9	0.137	0.082	1.03	0.95	11	10	0.0727	0.1499	0.03	1.05
10	0	0	1.05	0.95	6	7	0.0116	0.0236	0	1.05
11	0.335	0.0765	1.03	0.95	14	13	0.1	0.202	0	1.05
12	0	0	1.03	0.95	7	9	0.115	0.179	0	1.05
13	0	0	1.03	0.95	12	14	0.132	0.27	0	1.05
14	0.034	0.009	1.03	0.95	9	10	0.1885	0.3292	0	1.05
15	0.076	0.016	1.03	0.95	7	11	0.2544	0.38	0	1.05
16	0.052	0.035	1.05	0.95	10	12	0.1093	0.2087	0	1.05
17	0	0	1.05	0.95	13	12	0	0.396	0	1.05
18	0	0	1.05	0.95	12	5	0.2198	0.4153	0	1.05
19	0	0	1.05	0.95	16	8	0.202	0.6027	0	1.05
20	0	0	1.05	0.95	6	11	0.2399	0.4533	0	1.05
					15	12	0.0636	0.2	0	1.05
					8	9	0.0169	0.0599	0	1.05
					16	17	0.0315	0.0662	0	1.05
					3	18	0.03	0.06	0	1.05
					17	19	0.0315	0.0662	0	1.05
					18	20	0.03	0.06	0	1.05

Table H.3: Party-2 Generator Data

Gen.	Pmax	Pmin	Qmax	Qmin	d	c	e
22	1.3	0.2	0.4	0.01	0	0.2512	0
23	1.4	0.4	0.35	-0.4	0	0.2372	0
26	0.9	0.1	0.4	-0.4	0	0.2452	0

d is the coefficient for the quadratic term in the objective function. c is the coefficient for the linear term. e is the constant term.

Table H.4: Party-1 Generator Data

Gen.	Pmax	Pmin	Qmax	Qmin	d	c	e
3	1.11	0.1	0.4	-0.1	0	0.2522	0
7	2.6	0.11	0.24	-0.06	0	0.2392	0
10	1.35	0.15	0.24	-0.06	0	0.2352	0

d is the coefficient for the quadratic term in the objective function. c is the coefficient for the linear term. e is the constant term.

Table H.5: Intersection Area Compliance

	Party-1	Party-2
Bus Match	17	18
Bus Match	18	17
Bus Match	19	16
Bus Match	20	15

LIST OF REFERENCES

[1] A. Klos and A. Kerner, "The non-uniqueness of load flow solution," in *Proc. PSCC*, vol. 5, no. 3.118, 1975.

[2] B. Johnson, "Extraneous and false load flow solutions," *Power Apparatus and Systems, IEEE Transactions on*, vol. 96, no. 2, pp. 524–534, 1977.

[3] Y. Tamura, K. Iba, and S. Iwamoto, "A method for finding multiple load-flow solutions for general power-systems," in *IEEE TRANSACTIONS ON POWER APPARATUS AND SYSTEMS*, vol. 99, no. 4. IEEE-INST ELECTRICAL ELECTRONICS ENGINEERS INC 345 E 47TH ST, NEW YORK, NY 10017-2394, 1980, pp. 1322–1322.

[4] Y. Tamura, Y. Nakanishi, and S. Iwamoto, "On the multiple solution structure, singular point and existence condition of the multiple load-flow solutions." in *IEEE New York*, 1980.

[5] A. Pai, *Energy function analysis for power system stability*. Springer Science & Business Media, 2012.

[6] P. Varaiya, F. F. Wu, and R.-L. Chen, "Direct methods for transient stability analysis of power systems: Recent results," *Proceedings of the IEEE*, vol. 73, no. 12, pp. 1703–1715, 1985.

[7] H.-D. Chiang, *Direct methods for stability analysis of electric power systems: theoretical foundation, BCU methodologies, and applications*. John Wiley & Sons, 2011.

[8] C. L. DeMarco and T. J. Overbye, "An energy based security measure for assessing vulnerability to voltage collapse," *Power Systems, IEEE Transactions on*, vol. 5, no. 2, pp. 419–427, 1990.

[9] T. J. Overbye and C. L. DeMarco, "Improved techniques for power system voltage stability assessment using energy methods," *Power Systems, IEEE Transactions on*, vol. 6, no. 4, pp. 1446–1452, 1991.

[10] H. Chen, "Cascaded stalling of induction motors in fault-induced delayed voltage recovery (fidvr)," Ph.D. dissertation, Masters Thesis, University of Wisconsin-Madison, Department of Electrical and Computer Engineering, 2011.

[11] C. J. Tavora and O. J. Smith, "Equilibrium analysis of power systems," *Power Apparatus and Systems, IEEE Transactions on*, no. 3, pp. 1131–1137, 1972.

[12] J. Baillieul and C. I. Byrnes, "Geometric critical point analysis of lossless power system models," *Circuits and Systems, IEEE Transactions on*, vol. 29, no. 11, pp. 724–737, 1982.

[13] F. Salam, L. Ni, X. Sun, and S. Guo, "Parallel processing for the steady state solutions of large-scale non-linear models of power systems," in *Circuits and Systems, 1989., IEEE International Symposium on*. IEEE, 1989, pp. 1851–1854.

[14] F. Salam, L. Ni, S. Guo, and X. Sun, "Parallel processing for the load flow of power systems: the approach and applications," in *Decision and Control, 1989., Proceedings of the 28th IEEE Conference on*. IEEE, 1989, pp. 2173–2178.

[15] I. A. Hiskens, "Energy functions, transient stability and voltage behavior in power systems," Ph.D. dissertation, Electrical Engineering and Computer Science, The University of Newcastle, Australia, 1990.

[16] W. Ma and J. S. Thorp, "An efficient algorithm to locate all the load flow solutions," *Power Systems, IEEE Transactions on*, vol. 8, no. 3, pp. 1077–1083, 1993.

[17] R. Seydel, *Practical bifurcation and stability analysis*. Springer Science & Business Media, 2009, vol. 5.

[18] D. K. Molzahn, B. C. Lesieutre, and H. Chen, "Counterexample to a continuation-based algorithm for finding all power flow solutions," *Power Systems, IEEE Transactions on*, vol. 28, no. 1, pp. 564–565, 2013.

[19] D. Mehta, H. Nguyen, and K. Turitsyn, "Numerical polynomial homotopy continuation method to locate all the power flow solutions," *arXiv preprint arXiv:1408.2732*, 2014.

[20] T. Chen and D. Mehta, "On the network topology dependent solution count of the algebraic load flow equations," *arXiv preprint arXiv:1512.04987*, 2015.

[21] D. K. Molzahn, M. Niemerg, D. Mehta, and J. D. Hauenstein, "Investigating the maximum number of real solutions to the power flow equations: Analysis of lossless four-bus systems," *arXiv preprint arXiv:1603.05908*, 2016.

[22] J. Carpentier, "Contribution to the economic dispatch problem," *Bulletin de la Societe Francoise des Electriciens*, vol. 3, no. 8, pp. 431–447, 1962.

[23] I. A. Hiskens and R. J. Davy, "Exploring the power flow solution space boundary," *Power Systems, IEEE Transactions on*, vol. 16, no. 3, pp. 389–395, 2001.

[24] B. Lesieutre and I. A. Hiskens, "Convexity of the set of feasible injections and revenue adequacy in FTR markets," *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1790–1798, Nov. 2005.

[25] W. A. Bukhsh, A. Grothey, K. I. McKinnon, and P. A. Trodden, "Local solutions of the optimal power flow problem," *Power Systems, IEEE Transactions on*, vol. 28, no. 4, pp. 4780–4788, 2013.

[26] K. M. W A Bukhsh, A Grothey and P. A. Trodden, "Test case archive of optimal power flow (opf) problems with local optima," [Online]. Available, http://www.maths.ed.ac.uk/optenergy/LocalOpt/.

[27] J. A. Momoh, M. El-Hawary, and R. Adapa, "A review of selected optimal power flow literature to 1993. part i: Nonlinear and quadratic programming approaches," *IEEE transactions on power systems*, vol. 14, no. 1, pp. 96–104, 1999.

[28] ——, "A review of selected optimal power flow literature to 1993. part ii: Newton, linear programming and interior point methods," *IEEE Transactions on Power Systems*, vol. 14, no. 1, pp. 105–111, 1999.

[29] Z. Qiu, G. Deconinck, and R. Belmans, "A literature survey of optimal power flow problems in the electricity market context," in *Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES*. IEEE, 2009, pp. 1–6.

[30] Y. Xia and K. W. Chan, "Dynamic constrained optimal power flow using semi-infinite programming," *IEEE transactions on power systems*, vol. 3, no. 21, pp. 1455–1457, 2006.

[31] X. Bai, H. Wei, K. Fujisawa, and Y. Wang, "Semidefinite programming for optimal power flow problems," *International Journal of Electrical Power & Energy Systems*, vol. 30, no. 6, pp. 383–392, 2008.

[32] R. A. Jabr, "Radial distribution load flow using conic programming," *Power Systems, IEEE Transactions on*, vol. 21, no. 3, pp. 1458–1459, 2006.

[33] B. C. Lesieutre, D. K. Molzahn, A. R. Borden, and C. L. DeMarco, "Examining the limits of the application of semidefinite programming to power flow problems," in *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*. IEEE, 2011, pp. 1492–1499.

[34] D. K. Molzahn, S. S. Baghsorkhi, and I. A. Hiskens, "Semidefinite relaxations of equivalent optimal power flow problems: An illustrative example," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2015, pp. 1887–1890.

[35] D. K. Molzahn and I. A. Hiskens, "Convex relaxations of optimal power flow problems: An illustrative example," *arXiv preprint arXiv:1510.04330*, 2016.

[36] M. Farivar and S. H. Low, "Branch flow model: Relaxations and convexification," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*. IEEE, 2012, pp. 3672–3679.

[37] S. Bose, D. F. Gayme, S. Low, and K. M. Chandy, "Optimal power flow over tree networks," in *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*. IEEE, 2011, pp. 1342–1348.

[38] S. Bose, D. F. Gayme, K. M. Chandy, and S. H. Low, "Quadratically constrained quadratic programs on acyclic graphs with application to power flow," *arXiv preprint arXiv:1203.5599*, 2012.

[39] B. Zhang and D. Tse, "Geometry of feasible injection region of power networks," in *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*. IEEE, 2011, pp. 1508–1515.

[40] B. Zhang, "David tse. geometry of the injection region of power networks," *IEEE Trans. Power Systems*, vol. 28, no. 2, pp. 788–797, 2013.

[41] L. Gan and S. H. Low, "Optimal power flow in direct current networks," *Power Systems, IEEE Transactions on*, vol. 29, no. 6, pp. 2892–2904, 2014.

[42] L. Gan, N. Li, U. Topcu, and S. H. Low, "Exact convex relaxation of optimal power flow in radial networks," *Automatic Control, IEEE Transactions on*, vol. 60, no. 1, pp. 72–87, 2015.

[43] J. Lavaei, A. Rantzer, and S. Low, "Power flow optimization using positive quadratic programming," *Proc. 18th IFAC World Congr*, pp. 10 481–10 486, 2011.

[44] J. Lavaei and S. H. Low, "Zero duality gap in optimal pwoer flow problem," *Power Systems, IEEE Transactions on*, vol. 27, no. 1, pp. 92–107, Feb. 2012.

[45] S. Low, "Convex relaxation of optimal power flow, i: formulations and relaxations," *IEEE Trans. on Control of Network Systems*, vol. 1, no. 1, pp. 15–27, 2014.

[46] S. H. Low, "Convex relaxation of optimal power flow-part ii: Exactness," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 2, pp. 177–189, 2014.

[47] D. K. Molzahn and I. A. Hiskens, "Sparsity-exploiting moment-based relaxations of the optimal power flow problem," *Power Systems, IEEE Transactions on*, vol. 30, no. 6, pp. 3168–3180, 2015.

[48] J. B. Lasserre, "Global optimization with polynomials and the problem of moments," *SIAM Journal on Optimization*, vol. 11, no. 3, pp. 796–817, 2001.

[49] A. H. Land and A. G. Doig, "An automatic method for solving discrete programming problems," in *50 Years of Integer Programming 1958-2008*. Springer, 2010, pp. 105–132.

[50] D. R. Morrison, S. H. Jacobson, J. J. Sauppe, and E. C. Sewell, "Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning," *Discrete Optimization*, vol. 19, pp. 79–102, 2016.

[51] M. Tawarmalani and N. V. Sahinidis, *Convexification and global optimization in continuous and mixed-integer nonlinear programming: theory, algorithms, software, and applications*. Springer Science & Business Media, 2002, vol. 65.

[52] D. T. Phan, "Lagrangian duality and branch-and-bound algorithms for optimal power flow," *Operations Research*, vol. 60, no. 2, pp. 275–285, 2012.

[53] A. Gopalakrishnan, A. U. Raghunathan, D. Nikovski, and L. T. Biegler, "Global optimization of optimal power flow using a branch & bound algorithm," in *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*. IEEE, 2012, pp. 609–616.

[54] C. Chen, A. Atamtürk, and S. S. Oren, "Bound tightening for the alternating current optimal power flow problem," 2015.

[55] R. G. Jeroslow, "Trivial integer programs unsolvable by branch-and-bound," *Mathematical Programming*, vol. 6, no. 1, pp. 105–109, 1974.

[56] X. Liu and Z. Li, "Local load redistribution attacks in power systems with incomplete network information," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1665–1676, May 2014.

[57] O. Kosut, L. Jia, R. J. Thomas, and L. Tong, "Malicious data attacks on the smart grid," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 645–658, Dec. 2011.

[58] S. Gorman, "Electricity grid in the U.S. penetratedby spies," Wall Street Journal, http://online.wsj.com/artcile/SB123914805204099085.html, 2009.

[59] A. Hahn and M. Govindarasu, "Cyber attack exposure evaluation framework for the smart grid," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 835–843, Dec. 2011.

[60] A. Giani, E. Bitar, M. Garcia, M. McQueen, P. Khargonekar, and K. Poolla, "Smart gird data integrity attacks," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1244–1253, Sep. 2013.

[61] L. Xie, Y. Mo, and B. Sinopoli, "Integrity data attacks in power market operations," *IEEE Transactions on Smart Grid*, vol. 2, no. 4, pp. 659–666, Dec. 2011.

[62] O. Goldreich, *Foundations of cryptography: Basic applications*. Cambridge University Press, 2004, vol. 2.

[63] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology: EUROCRYPT 1999*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 1999, vol. 1592, pp. 223–238.

[64] C. Gentry, "A fully homomorphic encryption scheme," Ph.D. Thesis, Stanford University, 2009.

[65] A. López-Alt, E. Tromer, and V. Vaikuntanathan, "On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption," in *Proceedings of the Symposium on Theory of Computing (STOC)*, 2012, pp. 1219–1234.

[66] J. Dreier and F. Kerschbaum, "Practical privacy-preserving multiparty linear programming based on problem transformation," in *Proceedings of Privacy, Security, Risk and Trust (PASSAT)*, Oct. 2011, pp. 916–924.

[67] S. G. Choi, K.-W. Hwang, J. Katz, T. Malkin, and D. Rubenstein, "Secure multi-party computation of boolean circuits with applications to privacy in on-line marketplaces," in *Proceedings of the Conference on Topics in Cryptology*. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 416–432.

[68] P. Bogetoft *et al.*, "Secure multiparty computation goes live," in *Financial Cryptography and Data Security*, R. Dingledine and P. Golle, Eds. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 325–343.

[69] A. Ben-David, N. Nisan, and B. Pinkas, "FairplayMP: A system for secure multi-party computation," in *Proceedings of the ACM Conference on Computer and Communications Security*. New York, NY, USA: ACM, 2008, pp. 257–266.

[70] T. Toft, "Primitives and applications of secure multi-party computation," Ph.D. Thesis, University of Aarhus, Denmark, 2007.

[71] A. C.-C. Yao, "How to generate and exchange secrets," in *Proceedings of Foundations of Computer Science (FOCS)*, Oct. 1986, pp. 162–167.

[72] C. Dwork, "Differential privacy," in *Theory and applications of models of computation*. Springer, 2008, pp. 1–19.

[73] J. L. Nu and G. J. Pappas, "Differentially private filtering," *IEEE Transactions on Automatic Control*, vol. 59, no. 2, pp. 341–354, Feb. 2014.

[74] A. R. Borden, D. K. Molzahn, B. C. Lesieutre, and P. Ramanathan, "Power system structure and confidentiality preserving transformation of optimal power flow problem," in *Proceedings of Allerton Conference on Communication, Control, and Computing*, Sep. 2013.

[75] A. R. Borden, D. K. Molzahn, P. Ramanathan, and B. C. Lesieutre, "Confidentiality-preserving optimal power flow for cloud computing," in *Proceedings of Allerton Conference on Communication, Control, and Computing*, Sep. 2012.

[76] Y. Jadeja and K. Modi, "Cloud computing – Concepts architecture and challenges," in *Proceedings of International Conference on Computing, Electronics and Electrical Technologies*, Mar. 2012, pp. 877–880.

[77] D. Wu, B. Lesieutre, P. Ramanathan, and B. Kakunoori, "Preserving privacy of ac optimal power flow models in multi-party electric grids."

[78] D. Wu, B. C. Lesieutre, and P. Ramanathan, "Feasibility of power system structure preserving linear transformations for the AC optimal power flow problem," in *Proceedings of Allerton Conference on Communication, Control, and Computing*, Oct. 2014.

[79] E. Fink, *Changes of problem representation: Theory and experiments*. Physica, 2013, vol. 110.

[80] A. Newell and H. Simon, "Limitations of the current stock of ideas about problem solving," *Electronic information handling*, pp. 195–208, 1965.

[81] J. Van Baalen, "Toward a theory of representation design," DTIC Document, Tech. Rep., 1989.

[82] D. Peterson, *Re-representation and emergent information in three cases of problem solving*. Springer, 1994.

[83] S. Amarel, "On representations of problems of reasoning about actions," *Machine intelligence*, vol. 3, no. 3, pp. 131–171, 1968.

[84] A. Newell, H. A. Simon *et al.*, *Human problem solving*. Prentice-Hall Englewood Cliffs, NJ, 1972, vol. 104, no. 9.

[85] R. E. Korf, "Toward a model of representation changes," *Artificial intelligence*, vol. 14, no. 1, pp. 41–78, 1980.

[86] J. H. Larkin and H. A. Simon, "Why a diagram is (sometimes) worth ten thousand words," *Cognitive science*, vol. 11, no. 1, pp. 65–100, 1987.

[87] R. C. Holte, "An analytical framework for learning systems," Ph.D. dissertation, University of Texas at Austin, Austin TX, USA, 1988.

[88] J. E. Tate and T. J. Overbye, "A comparison of the optimal multiplier in polar and rectangular coordinates," *Power Systems, IEEE Transactions on*, vol. 20, no. 4, pp. 1667–1674, 2005.

[89] B. Lesieutre and D. Wu, "An efficient method to locate all the load flow solutions-revisited," in *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2015, pp. 381–388.

[90] J. D. Glover, M. Sarma, and T. Overbye, *Power System Analysis & Design, SI Version*. Cengage Learning, 2011.

[91] R. Hartshorne, *Algebraic geometry*. Springer Science & Business Media, 1977, vol. 52.

[92] J. Bochnak, M. Coste, and M.-F. Roy, *Real algebraic geometry*. Springer Science & Business Media, 2013, vol. 36.

[93] J. D. Foster, "Mixed-integer quadratically-constrained programming, piecewise-linear approximation and error analysis with applications in power flow," Ph.D. dissertation, School of Mathematical and Physical Sciences, The University of Newcastle, Australia, 2013.

[94] K. U. Rao, *Computer Techniques and Models in Power Systems*. IK International Pvt Ltd, 2008.

[95] M. S. Bazaraa, H. D. Sherali, and C. Shetty, *Nonlinear Programming: Theory and Algorithms*, 3rd ed. Wiley, 2006.

[96] T. A. Davis, "Suitesparse," [Online]. Available, http://faculty.cse.tamu.edu/davis/suitesparse.html.

[97] "IEEE 14-bus system." [Online]. Available: https://www2.ee.washington.edu/research/pstca/pf14/pg_tca14bus.htm

[98] J. J. Grainger and W. D. Stevenson, *Power system analysis*. McGraw-Hill, 1994.

[99] A. Wood and B. Wollenberg, "Power generation operation and control2nd edition," in *Fuel and Energy Abstracts*, vol. 3, no. 37, 1996, p. 195.

[100] "IEEE 30-bus system." [Online]. Available: http://www2.ee.washington.edu/research/pstca/pf30/pg_tca30bus.htm

[101] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Transactions on Power delivery*, vol. 4, no. 2, pp. 1401–1407, 1989.

[102] "IEEE 39-bus system." [Online]. Available: http://icseg.iti.illinois.edu/ieee-39-bus-system/

[103] "IEEE 118-bus system." [Online]. Available: http://www2.ee.washington.edu/research/pstca/pf118/pg_tca118bus.htm

[104] J. Verschelde, "Algorithm 795: Phcpack: A general-purpose solver for polynomial systems by homotopy continuation," *ACM Transactions on Mathematical Software (TOMS)*, vol. 25, no. 2, pp. 251–276, 1999.

[105] D. J. Bates, J. D. Hauenstein, A. J. Sommese, and C. W. Wampler, "Bertini: Software for numerical algebraic geometry," Available at bertini.nd.edu with permanent doi: dx.doi.org/10.7274/R0H41PB5.

[106] A. J. Korsak, "On the question of uniqueness of stable load-flow solutions," *IEEE Transactions on Power Apparatus and Systems*, no. 3, pp. 1093–1100, 1972.

[107] H. D. Nguyen and K. S. Turitsyn, "Appearance of multiple stable load flow solutions under power flow reversal conditions," in *PES General Meeting— Conference & Exposition, 2014 IEEE*. IEEE, 2014, pp. 1–5.

[108] J. Nocedal and S. Wright, "Least squares problems," in *Numerical optimization*, 2nd ed. Springer, 2000.

[109] D. K. Molzahn, "Computing the feasible spaces of optimal power flow problems," *IEEE Transactions on Power Systems*, 2017.

[110] C. E. M.-S. R. D. Zimmerman and R. J. Thomas, "Matpower: Steady-state operations, planning and analysis tools for power systems research and education," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, Feb. 2011.

[111] D. Wu, D. Molzahn, B. Lesieutre, and K. Dvijotham, "A Deterministic Method to Identify Multiple Local Extrema of the AC Optimal Power Flow Problem," to appear in *IEEE Transactions on Power Systems*.

[112] D. Molzahn and I. Hiskens, "Moment-Based Relaxation of the Optimal Power Flow Problem," *18th Power Syst. Comput. Conf. (PSCC)*, Aug. 2014.

[113] C. L. Demarco, J. V. Sarlashkar, and F. Alvarado, "The potential for mailicious control in a competitive power systems environment," in *Proceedings of International Conference on Control Applications*, Sep. 1996, pp. 462–467.

[114] K. G. Murty and S. N. Kabadi, "Some NP-complete problems in quadratic and nonlinear programming," *Mathematical programming*, vol. 39, no. 2, pp. 117–129, 1987.

[115] P. Manyem and J. Ugon, "Computational complexity, NP completeness and optimization duality: A survey." in *Electronic Colloquium on Computational Complexity (ECCC)*, vol. 19, 2012, p. 9.

[116] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, "Identifying and Attacking the Saddle Point Problem in High-Dimensional Non-Convex Optimization," in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 2933–2941.