

**Statistical and Intertemporal Methods Using Embeddings for Non-linear AC Power System  
State Estimation**

Submitted in Partial Fulfillment

of the Requirements for

the Degree of

Doctor of Philosophy

in

Electrical and Computer Engineering

Yang Weng

M. S., Electrical and Computer Engineering, Carnegie Mellon University

M. S., Statistics, University of Illinois at Chicago

B. E., Automation, Huazhong University of Science and Technology

Carnegie Institute of Technology

Carnegie Mellon University

Pittsburgh, PA

August, 2014

Approximation Methods for Nonlinear AC Electric Power System State  
Estimation

© Copyright 2014

by

Yang Weng

All Rights Reserved

This dissertation is dedicated to my parents and my wife.

## Abstract

This thesis aims to improve the robustness of state estimation (SE) methods currently used by the electric power industry. The main objective of today's SE is to estimate system state using redundant measurements in the presence of bad data and constantly varying network topology. The hoped-for of this thesis contributions concerns the use of embedding techniques to overcome problems related to non-linearities and uncertainties.

First, in order to improve static SE when reliable historical data is unavailable, we reformulate the least-squares non-convex static SE currently used as a convex optimization problem by using an embedding and convex relaxation techniques. We demonstrate a significant improvement in accuracy, particularly in reactive power/voltage estimates. We propose that the added computational complexity caused by this embedding can be managed by implementing the method as a distributed algorithm developed on the basis of an underlying power system graph.

On the other hand, when reliable historical data is available, this thesis proposes utilizing them by using both static and dynamic data-driven state estimation methods. The static estimator uses the an embedding of the current measurements and learns the state estimator using similar measurement-state pairs in the historical records. Several speedup techniques from machine learning are proposed for overcoming the initial computational complexity of the proposed method and making it potentially useful for online applications. A dynamic data-driven state estimator requires a much faster sampling of the historical data to capture dynamics. An expectation maximization algorithm for SE is proposed for learning in embedded state space.

Finally, the uncertainties in SE caused by a massive number of distributed energy resources motivate a fully distributed probabilistic state estimation method. This thesis provides a Bayesian Network solution to the SE problem in that setting. The proposed method employs a probabilistic graphical model and embeds it in a ceratin probability space, which allow the use of a variational belief propagation method that is both scalable and exact for tree networks in distribution systems.

To assess the improved performance achieved by applying the proposed methods in this thesis, we compare them with currently used methods in various simulation scenarios. Using the results, a flow chart is presented to determine which methods to apply in different scenarios.



## Acknowledgement

I would like to dedicate this acknowledgement to my advisors, Professor Rohit Negi and Professor Marija D. Ilić, for their continuous support and inspiration over my Ph.D. study period at Department of Electrical and Computer Engineering in Carnegie Mellon University (CMU). Professor Negi trained me well with precise attitude and strong logic for research and analyses. His critical thinking and reasoning is deeply rooted in theoretical foundation. Often times, critical thinking provides the soil for developing my own new ideas and innovations when conducting researches. Professor Ilić has an extraordinary talent of linking scientific theories with practical applications. She helped me abridge the gap between the idea I have developed in academic researches and its application in power systems. Her earnest guidance, resourcefulness, creativity and flexibility led me into the core of power systems analyses. Her passionate commitment to power grid monitoring analyses illustrated the great practical value that may be achieved by our research, which enlightens me to develop an insightful vision in the research field. I am deeply honored to have the opportunity of working with them and other eminent professors and faculties in CMU.

Secondly, I would like to present my gratitude to my M.S. advisor, Professor Christos Faloutsos at Department of Machine Learning in School of Computer Science, CMU. His expertise on data-mining analysis and his innovative ideas from computer science's perspective benefited me by paving the way for cross-discipline researches and advantages. My sincere thanks to Professor Tom Mitchell, Professor Larry Wasserman, Professor Geoffrey Gordon, and Mr. Evangelos (Vagelis) Papalexakis, who provided friendly support for my study on Computer Science.

I am grateful to my Ph.D dissertation committee members, Professor Vijayakumar Bhagavatula and Dr. Bruce Fardanesh, for their valuable participation and thoughtful inquiries from the initial stage of my thesis preparation. I would also like to pay reverence and appreciation to other faculty members in the ECE Department, including Professor Byron Yu, Professor Ozan K. Tonguz, Professor Jovan Ilić, Professor José Moura, Professor Bruno Sinopoli, Professor Gabriela Hug-Glanzmann, Professor Franz Franchetti, Professor Soumya Kar, Professor Ole Mengshoel, Professor Pulkit Grover, and Professor Le Xie from Texas A&M University. Thanks to their insightful suggestions and comments, my research skills gradually improved and after four and a half years full-time commitment in this academic field, I finally become a qualified Ph.D. candidate and a professional. The informative and constructive discussions we had have helped me refine my blueprint for my career in the future.

Special appreciation is devoted to Ms. Samantha Goldstein and Ms. Elaine Lawrence in the ECE Department, and Diane Stidle in the Machine Learning Department. They are always there to help and their invaluable help has made the academic path towards my Ph.D degree more efficient and comfortable.

I would further dedicate my gratefulness to the EESG research groups and fellow students from Porter Hall B-Level. Many thanks to Ms. Claire Bauerle for her administrative support. My thankfulness and respect to my colleagues there: Dr. Qixing Liu, Dr. Sergio Pequito, Dr. Jhi-Young Joo, Dr. Tao Cui, Dr. Lu Zheng, Dr. Dinghuan Zhu, Xiaoqi Yin, Rui Yang, Andrew Hsu, Dr. Milos Cvetković, Jonathan Donadee, Kevin Bachovchin, Nipun Popli, Dr. Sanja Civić, Chin Yen Tee, Dr. Yang Zhang, Brian Swenson, Stefanos Baros, Siripha (Pui) Junlakarn, José Prada, Dr. Noha Abdel-Karim, Dr. Masoud Nazari, Dr. Yilin Mo, Javad Mohammadi, Xia Miao, Xiao Zhang, June Zhang, Yaoqing Yang, Xiaohui Wang, Liangyan Gui. They were important sources of support and joy when I was facing academic challenges or difficulties in life. They were also my fellow travellers in the testing journey towards academic accomplishments.

I am also in debt to Ms. Marilyn L. Patete. She made the Hamerschlag Hall B200 Wing a warm and pleasant environment for me to work in. Salutations to my friends in the Wing: Dr. Qiao Li, Dr. Can Ye, Dr. Qi Wu, Zhiding Yu, Vishnu Naresh Boddeti, Yibin Ng, Andres Rodriguez, Euiseok Hwang, Jiun-Ren Lin, Junyao Guo, Dr. Wantanee Viriyasitavat, Hsiao-Yu Fan, Hugo Conceicao, Jonathan Becker, Hyunggi Cho, and Zhiding Yu.

Finally, my utmost gratitude to my parents, Mr. Junqiu Weng and Ms. Ruihua Chen, who brought me up with unconditional love and a life-time commitment. My love and gratitude also to my wife, Xiaoyi Liang, the person who has and will share the ups and downs in life with me.

I gratefully appreciate Dr. Ernst Scholtz and Dr. Xiaoming Feng's help from ABB. The research work presented in this dissertation was supported in part by ABB Fellowship and US National Science Foundation awards 0931978, 0831973 and 0347455.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Power System State Estimation: some Background . . . . .	1
1.2	Challenges in AC Electric Power System State Estimation . . . . .	4
1.2.1	Non-convexity . . . . .	4
1.2.2	Unused Historical Data . . . . .	5
1.2.3	Scalable Monitoring in a Smart Grids . . . . .	6
1.3	Our Contributions . . . . .	7
1.3.1	Convex Relaxation-Based Approach . . . . .	7
1.3.2	Data-Driven Approach . . . . .	8
1.3.3	Graphical Model Approach . . . . .	9
1.4	Notion of Embedding . . . . .	10
1.5	Thesis Outline . . . . .	11
<b>2</b>	<b>Problem Formulation</b>	<b>13</b>
2.1	Static AC State Estimation . . . . .	14
2.1.1	Classical Estimation Model . . . . .	14
2.1.2	Bayesian Estimation Model . . . . .	16
2.1.3	Discriminative Estimation Model . . . . .	17



2.2	Dynamic AC State Estimation . . . . .	17
2.2.1	State Space Model . . . . .	17
2.3	Chapter Summary . . . . .	18
<b>3</b>	<b>Semidefinite Programming for Convexifying Static State Estimation</b>	<b>19</b>
3.1	Introduction to Semidefinite Programming . . . . .	20
3.2	Current State-of-the-Art of Semidefinite Programming (SDP) for Power System Analysis . .	22
3.3	Semidefinite Programming-Based State Estimation . . . . .	23
3.3.1	Embedding of State . . . . .	24
3.3.2	Alternative SDP-Based SE Formulations . . . . .	25
3.3.3	Convex Relaxation of the SDP-Based SE . . . . .	28
3.3.4	Computational Cost and Extensions . . . . .	29
3.3.5	New Metric for Optimality Evaluation . . . . .	29
3.3.6	Simulation Results . . . . .	30
3.4	Distributed Implementation for Semidefinite Programming-Based State Estimation . . . . .	36
3.4.1	Alternative Form for a Distributed Algorithm . . . . .	38
3.4.2	Problem Decomposition . . . . .	39
3.4.3	Distributed Algorithm for Parallel Estimation of State . . . . .	44
3.4.4	New Metric for Local Optimality Evaluation. . . . .	46
3.4.5	Simulation Results . . . . .	46
3.5	A Special Case for Achieving a Global Optimum . . . . .	50
3.5.1	The New SDP-Based Approach . . . . .	51
3.5.2	Simulation Results . . . . .	55
3.6	Chapter Summary . . . . .	56
<b>4</b>	<b>Data-Driven State Estimation</b>	<b>58</b>

4.1	Current State-of-the-Art Power System Analysis with Historical Data . . . . .	59
4.2	Static Data-Driven Approach to State Estimation . . . . .	61
4.2.1	The Nearest Neighbors Approach . . . . .	62
4.2.2	Bayesian Inference: Kernel Ridge Regression . . . . .	63
4.2.3	Model Selection . . . . .	66
4.2.4	Robust Data-Driven State Estimation . . . . .	66
4.3	Speed Up for Static Data-Driven State Estimation . . . . .	70
4.3.1	Dimension Reduction . . . . .	70
4.3.2	$K$ -dimensional ( $k$ -d) Tree for Indexing . . . . .	71
4.3.3	Simulation Results . . . . .	73
4.4	Data-Driven Approach for Topology Identification . . . . .	79
4.4.1	Current State-of-the-Art Power System Topology Error Identification . . . . .	80
4.4.2	Problem Definition . . . . .	81
4.4.3	The Nearest Neighbors Approach . . . . .	82
4.4.4	Bayesian Inference: Logistic Kernel Regression . . . . .	82
4.4.5	Simulation Results . . . . .	86
4.5	Dynamic Approach . . . . .	89
4.5.1	Current State-of-the-Art Kalman Filter Approaches for State Estimation . . . . .	89
4.5.2	Kalman Filter Approach . . . . .	91
4.5.3	Stage 1: Historical Data-Driven Parameter Estimation . . . . .	92
4.5.4	Stage 2: On-line Kalman Filter State Estimation . . . . .	97
4.5.5	Simulation Results . . . . .	98
4.6	Chapter Summary . . . . .	99

## 5 Graphical Model-Based State Estimation

101

5.1	State-of-the-Art Bayesian Method and Probabilistic Analysis for Power Systems . . . . .	104
5.2	Probabilistic Graphical Modeling for State Estimation . . . . .	105
5.2.1	Belief Propagation for Tree Networks . . . . .	107
5.2.2	Variational Belief Propagation for Mesh Networks . . . . .	108
5.2.3	Sequential Variational Belief Propagation Algorithm . . . . .	109
5.2.4	Variational Belief Propagation for AC Power System State Estimation . . . . .	110
5.2.5	Illustration on a Small Example . . . . .	113
5.3	Simulation Results . . . . .	115
5.4	Chapter Summary . . . . .	119
<b>6</b>	<b>Comparison of the Various Proposed Methods using IEEE 30-bus systems</b>	<b>121</b>
6.1	Simulation Scenario I: A SCADA Center without Historical Data. . . . .	122
6.2	Simulation Scenario II: A New SCADA Center without Historical Data where the Noise is Very Small. . . . .	124
6.3	Simulation Scenario III: A SCADA Center with Some Historical Data about the Estimated States. . . . .	125
6.4	Simulation Scenario IV: A SCADA Center with Long Labeled Historical Data. However, Undetected Bad Data and Malicious Attacks May Exist in the Data Set. . . . .	128
6.5	Simulation Scenario V: A SCADA Center with Labeled Historical Data. The Data Rate is Very Fast. . . . .	128
6.6	Simulation Scenario VI: A SCADA Center with Poor Historical Data. . . . .	130
6.7	Chapter Summary . . . . .	133
<b>7</b>	<b>Conclusions</b>	<b>134</b>
	<b>Appendices</b>	<b>137</b>
.1	Other Measurement Type for Distributed Semidefinite Programming-Based State Estimation	137



# List of Tables

3.1	The expression of $A_i$ in (3.8)	25
3.2	Stage 1: Measurement normalized residuals	35
3.3	Stage 2: Estimated status of circuit breakers	35
3.4	Stage 1: Measurement normalized residuals for distributed SDP	49
3.5	Stage 2: Estimated status of circuit breakers for distributed SDP	50
3.6	IEEE 300 bus simulation results	50
4.1	Standard deviation of measurement noise	74
5.1	Convergence	119

# List of Figures

1.1	State Estimation in power system operation. . . . .	2
1.2	Benefits from State Estimation. . . . .	3
1.3	Logic among the three contributions. . . . .	8
1.4	Embedding process. . . . .	10
2.1	IEEE 14 bus system. . . . .	14
2.2	Physical network and cyber network (14 bus system). . . . .	16
3.1	Flow chart of the proposed algorithm. . . . .	30
3.2	A new metric and its application. . . . .	31
3.3	SCADA. . . . .	31
3.4	IEEE 14 bus (Gaussian noise). . . . .	32
3.5	IEEE 14 bus (Gaussian noise plus bad data). . . . .	34
3.6	IEEE 30 bus. . . . .	36
3.7	Computational time . . . . .	36
3.8	Local confidence in the SCADA room. . . . .	46
3.9	Extended chordal graph decomposition of IEEE 14 bus system. . . . .	47
3.10	Computational time comparison. . . . .	47
3.11	Accuracy comparison. . . . .	48

3.12 Four Scenarios of SE problems. . . . .	51
3.13 IEEE 14-bus test-bed. . . . .	54
4.1 Power grid. . . . .	60
4.2 Simulation results for speed up . . . . .	61
4.3 Flow chart . . . . .	62
4.4 Diagram for robust nearest neighbors search . . . . .	68
4.5 Visualization of voltage magnitude pairs in a two bus system . . . . .	69
4.6 Dimension reduction . . . . .	70
4.7 Binary tree and $k$ -d tree . . . . .	72
4.8 $K$ -d tree for clustered data set . . . . .	72
4.9 Simulation results with only the first block of Fig.4.4. . . . .	76
4.10 Simulation results with the first and the second blocks of Fig.4.4 . . . . .	76
4.11 Simulation results with all three blocks of Fig.4.4 . . . . .	77
4.12 Simulation results for the speedup . . . . .	78
4.13 Problem definition. . . . .	81
4.14 Bus branch model . . . . .	88
4.15 Averaged topology identification success probability . . . . .	88
4.16 Block diagram of the proposed historical data-driven approach. . . . .	92
4.17 Linear feature space model. . . . .	93
4.18 Normalized Euclidean state distance. . . . .	99
4.19 IEEE 300 bus. . . . .	100
5.1 Physical network and cyber network (14 bus system). . . . .	106
5.2 A graph with four nodes. . . . .	107
5.3 Flow chart for the proposed approach. . . . .	114

5.4	Generation of spanning trees to obtain the value of $\rho_{ij}$ . . . . .	115
5.5	Normalized sum square errors (14 bus system). . . . .	116
5.6	Results obtained from the IEEE 30 bus. . . . .	116
5.7	Results obtained from the IEEE 118 bus. . . . .	117
5.8	CPU time comparison. . . . .	118
5.9	Centralized and decentralized BP . . . . .	120
6.1	Simulation scenario II: accuracy comparison (measurement domain). . . . .	123
6.2	Simulation scenario II: computational time comparison. . . . .	124
6.3	Simulation scenario II: accuracy comparison (measurement domain). . . . .	125
6.4	Simulation scenario III: accuracy comparison (state domain). . . . .	126
6.5	Simulation scenario III: accuracy comparison (state domain). . . . .	127
6.6	Simulation scenario III: computational time comparison. . . . .	127
6.7	Simulation scenario IV: accuracy comparison (state domain). . . . .	129
6.8	Simulation scenario V: accuracy comparison (state domain). . . . .	130
6.9	Simulation scenario VI: accuracy comparison (state domain). . . . .	131
6.10	Decision tree for the various new SE approaches. . . . .	133



# Chapter 1

## Introduction

### 1.1 Power System State Estimation: some Background

A modern society without an electricity supply is unimaginable. Electric power grids have undergone continuous developments since late 1881 when the world's first public electricity supply was provided in the United Kingdom. As one of the most significant infrastructures in human society, the electric power grid not only provides electricity as a form of flexible, convenient energy for industrial and individual uses, but also provides it in a clean and relatively easy way to transmit. The current U.S. electric power grid consists of generation and distribution systems, and is designed with interconnections to power lines of different voltage levels to improve its transmission efficiency.

Despite 130-plus years of development and engineering, electric power systems are still under intense pressure to achieve stability against outages and blackouts. A blackout can raise serious issues in the operation of power systems regardless of its scale of the system; these issues include physical hardware damage, unexpected network interruption, and subsequent economic loss. After the 1965 blackout, the electric power industry started to realize that an in-depth understanding of power grid behavior would be necessary in order to better control grid operations. Accordingly, various funding is provided by government agencies to support grid-related research. First-generation computers were introduced to the monitoring system; these enabled power grid operators to supervise and manage operations by simply watching the telemetered data showing on the computer screens and deciding whether the systems were functioning normally. With the assistance of computers, power system monitoring and control evolved into what is now called a Superviso-

ry Control and Data Acquisition (SCADA) system. Such a computer-aided monitoring centralized system is useful in many respects since it replaces manual measurements with automatic measurements that run online simultaneously and sound the alarm when the data gathered exceeds setting limits. However, operators have found that computer-aided monitoring is unable to track the status of the power system as a whole and has difficulty predicting a system's stability. In addition, SCADA could not deal with suspicious, delayed, or missing data arisen from measurements.

Schweppe, a pioneer in power system security analysis, first pointed out to the power system society the possibility of using computer assistance to estimate the “states” of the power system. His comments led to a revolutionary concept - State Estimation (SE). He described in his introductory papers [1] two parts of the SE process, namely, power system modeling and subsequent state analysis. In the modeling, all generators, transformers, and loads are represented by network buses that are interconnected via branches. Branches represent electric transmission lines. In the analysis, SE is designed to extract hidden state information from redundant measurements and use such state information to calculate real time load flow, etc. Fig.1.1 displays the major role played by SE in power system operation.

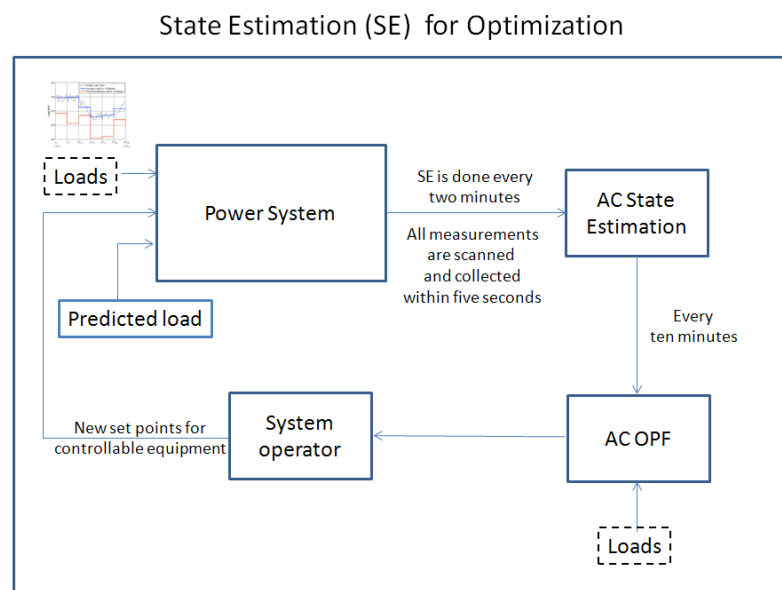


Figure 1.1: State Estimation in power system operation.

The introduction of SE into the power grid was a breakthrough that led to significant benefits. Turning to the multifold characteristics of SE, as displayed in Fig.1.2, we find the first benefit comes from allowing operators to visualize the operating point of a power grid so that proper control can be applied accordingly. The second benefit comes from SE's complete system representation, which operators can input into various security and economic analysis programs. Finally, the most beneficial characteristic is SE's ability to provide estimated measurements for real time power flow analysis when actual measurements are not available or bad. Specifically, SE can recover by estimation the specialized data necessary to analyze real time flow when such data cannot be obtained through actual measurements, such as real power and reactive power (PQ) values from all load buses, or real power and voltage magnitude (PV) values from all generator buses. Without the recovery of needed data, the power system will not be able to function in an economic, efficient manner as desired by both users and operators. SE's characteristics of completeness attract a large number of electric system engineers, who believe SCADA and SE can actually complement each other, i.e., SCADA is very fast in displaying various measured data without mathematical computation, while SE computes many unmeasured data based on available measurements. Together they form the basis for the Energy Management System (EMS), which is basically a database with various security analysis programs.

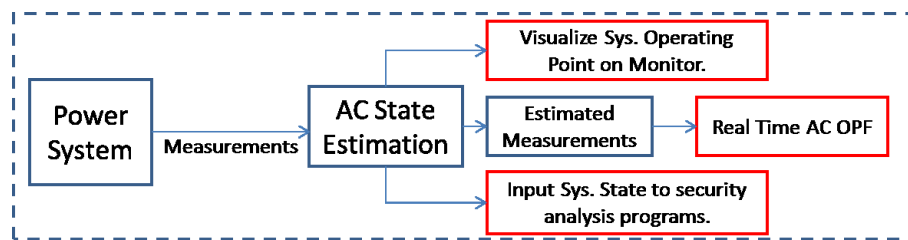


Figure 1.2: Benefits from State Estimation.

Inspired and encouraged by SE's potential benefits, power system engineers have been delving into SE application for the past few decades, with the following systematic steps achieved: (1) Measurements are read into sensors and communicated to the SCADA Data Center; (2) A data verification process is run to filter out obvious inconsistent data; (3) Bus breaker status data are loaded to form a bus-branch level topology; (4) Based on the topology, the grid is divided into a core grid and islands with a bus number assigned to each of them; (5) Observability analysis is performed with observable and unobservable areas marked by different colors on the screen; (6) In the observable areas, SE is conducted; (7) The SE result is fed into a bad data program to check for possible gross errors. If there are gross errors, they are filtered

out, and the program goes back to step (5); (8) Topology error is checked based on the SE results and analog power flow measurements in the substation. If an incorrect topology is detected and identified, the program goes back to step (4); (9) The SE results and measurements are marked on the monitors. Critical measurements are drawn with colors; (10) The SE result is outputted to the other security analysis programs; and (11) The SE result is used for AC Optimal Power Flow calculation.

SE is currently referred to as the static SE described above, which is conducted at a two-minute rate. However, when a topology change is detected or upon the operator's request, a new SE process will start based on the most current information. In the case of unexpected data loss, pseudo measurements performed prior to the loss will be used to create an estimate that is better than the operator's own guess and certainly better than the absence of necessary data. However, pseudo measurements also lead to inaccuracy when solving the missing data problem since it becomes more difficult to retrieve the lost data when time passes and possible status change can occur within the unobservable areas. In general, though, the systematic SE steps described above are useful in achieving stability and an economic operation of the power system despite the potential problems they raise.

## **1.2 Challenges in AC Electric Power System State Estimation**

### **1.2.1 Non-convexity**

A major challenge in power system control is obtaining accurate and computationally efficient state estimation (SE). Many widely used algorithms solve the SE problem by using a Weighted Least Square (WLS) [2–6] estimation form with Newton's method (i.e. [7]). However, these approaches have local optimum issues due to their problem formulation. For instance, although the WLS approach is known to give closed form solution and a global optimum result for linear Least Square problems with Gaussian noise, the approach loses both properties when dealing with an AC power system SE problem. This is caused by the inherent nonlinear power system equations, which turns the SE problem into a nonlinear Least Square problem. Mathematically, different than the unconstrained convex quadratic programming (QP) problem generated by the linear least square problem, the SE problem is formulated with a non-convex objective with polynomials of order four. For a non-convex optimization, Newton's method cannot guarantee a global optimum, even if it is useful in obtaining a local optimum by successively finding a better approximation.

The key observation from above is that: trying to deal with noise and nonlinearity in one-shot is difficult for state estimation. If Newton’s method hits the global optimum by luck, the two problems are solved perfectly and simultaneously. However, there is no guarantees over it. If Newton’s method hits a local optimum, neither the nonlinearity nor the noise problem is solved. Beside this troublesome property, there is also no way to check whether the solution is a global optimum or a local optimum, making the sub-optimal result undetectable. Therefore, there are two big challenges for the static state estimation approach. One is how to deal with noise and nonlinearity separately from the search for optimal result. The other lies in how to evaluate the optimality of the SE results.

**Remark** One method to solve the nonlinearity is to rely on Phasor Measurement Unit (PMU) measurements [8] alone. For instance, the ongoing industry pilot experiments plan to have PMU measurements installed at all major 230 KV substations and all large generators. However, because the deploying process is gradual, there will be limited PMU measurements [9] for a long time. Further, current experience shows that PMU measurements are imperfect and their accuracy depends on the manufacturing qualities and various sources of uncertainty (GPS synchronization, instrument transformers, A/D converters, etc.). Even if all buses are equipped with PMUs, we still need redundancy from other measurement types for robustness [10]. Therefore, we need a general SE method that deals with combined SCADA and PMU data.

### 1.2.2 Unused Historical Data

On the other side, the static SE above and Power Flow computation both aim at providing better system operation (BSO) with optimality and feasibility. However, a proper definition of BSO is needed. Furthermore, questions like “how much accuracy is needed?” and “what resources are available for a better BSO?” remain to be answered. For example, in one of our preliminary works on static SE, we showed how to solve the WLS SE problem exactly under no noise assumption. We concluded that when the noise is small in the measurement, the true state is also close to the state estimate of the new approach under no noise assumption. This is true under the smoothness condition of power equations. However, what if the noise is large—i.e. due to bad data and topology errors, or a compromised data set? Although there are have been heuristic methods to deal with them statically in the past, recent advances in new devices and corresponding investments such as advanced meter infrastructures (AMIs) and synchrophasors have led to unprecedented measurements and data in the electric power industry and opened the door for smart learning to deal with

large uncertainties. High-performance computers that are becoming available in SCADA centers can further assist such a data-driven approach to create a more efficient modeling process.

### **1.2.3 Scalable Monitoring in a Smart Grids**

In addition to historical data and high performance computers at the SCADA center, decentralized computational resources are also becoming available in the smart grid. How to use these small but smart devices in a distributed manner to conduct robust monitoring raises another big question about reliable power grid operation. This is particularly important now that distributed renewable generators are built in the distribution system. A large-scale implementation of this type of generator can lead to strong power fluctuations in the power grid. The voltage, for example, will no longer be imposed by the large, centralized power stations, and voltage stability of the system becomes an issue. Traditionally, the distribution network is a passive network that depends totally on the transmission network for energy delivery, frequency control, and voltage regulation. In a future horizontally-operated power system, the power is not only consumed by, but also generated in, the distribution network. Therefore, both the transmission and distribution networks need to change into active, distributed, and intelligent networks.

In addition to a distributed monitoring for robustness, scalability will also be important for future large-scale smart grids. The industrial method for SE is non-scalable and computationally complex [11], and is typically used for Extra High Voltage (EHV) and High Voltage (HV), and is only occasionally used for Medium Voltage (MV) representation of complex multi-voltage level power grids. Such a non-scalable method makes it difficult to estimate the status and states of new diverse resources and users connected to Low Voltage (LV) level distribution systems. Generally, the power system operators of traditional power grids have faced inherent difficulties in managing the effects of small-scale generations and loads, including but not limited to renewable energy generators, responsive small electricity users, and electricity users capable of storage, such as electric cars. While these new components are more environmentally friendly, and capable of increasing fuel diversity and bringing in economic benefits, they also raise serious concerns regarding the secure and reliable operation of the backbone EHV/HV power grids. Therefore, their state needs to be estimated in order to account for their effects on the state of the backbone power grid. This need to estimate the on-line state in the entire electric power grid makes it even more difficult than before to manage all data in a centralized way.

For future electric energy systems, a multi-layered, distributed implementation of SE will be the preferred approach. This application requires a systematic design of distributed algorithms whose performance does not fall short of the centralized methods.

## 1.3 Our Contributions

Our major contribution lies in re-modeling state estimation by utilizing new optimization tools, historical data, high performance computers, and distributed intelligence aided by relevant mathematical embeddings of the problem. These new modeling approaches can be extended to bad data detection and the topology identification process.

### 1.3.1 Convex Relaxation-Based Approach

To deal with noise and nonlinearity separately for static SE, and to evaluate the optimality of the obtained SE results, our first contribution is to propose a Semidefinite Programming (SDP)-based approach first. Specifically, to deal with the noise, such an approach first relaxes the nonlinear non-convex optimization problem into a convex optimization problem [12, 13] by embedding the state space into a higher dimensional space, which can be solved exactly. Second, eigenvalue decomposition is used to deal with the nonlinearity by choosing the most likely state, which helps eliminate local optimum issue in current SE. Third, an universal lower bound unavailable in the past is derived, which can be used to evaluate the optimality of arbitrary state estimation result.

Using simulations on several IEEE Test Systems of up to 118 buses, the centralized convex relaxation approach results in significantly higher accuracy than the currently used WLS SE. These accuracy improvements are evidently shown in the reactive power residuals. Such fact offers the major promise for accurate voltage-reactive power estimation. Unfortunately, the simulation results also show that the SDP-based approach has an undesirable computational burden which grows as a polynomial in the number of buses in the grid.

Therefore, a graph-based decomposition is subsequently conducted to decompose the positive semidefinite constraint, and the Lagrangian dual method is applied to decompose the objective to design a distributed SE algorithm, so that the computational complexity raised by centralized positive semidefinite programming

can be reduced. Further simulation results show that such a distributed SDP approach can further advance the centralized relaxation approach by obtaining the same accuracy with significantly less computation time.

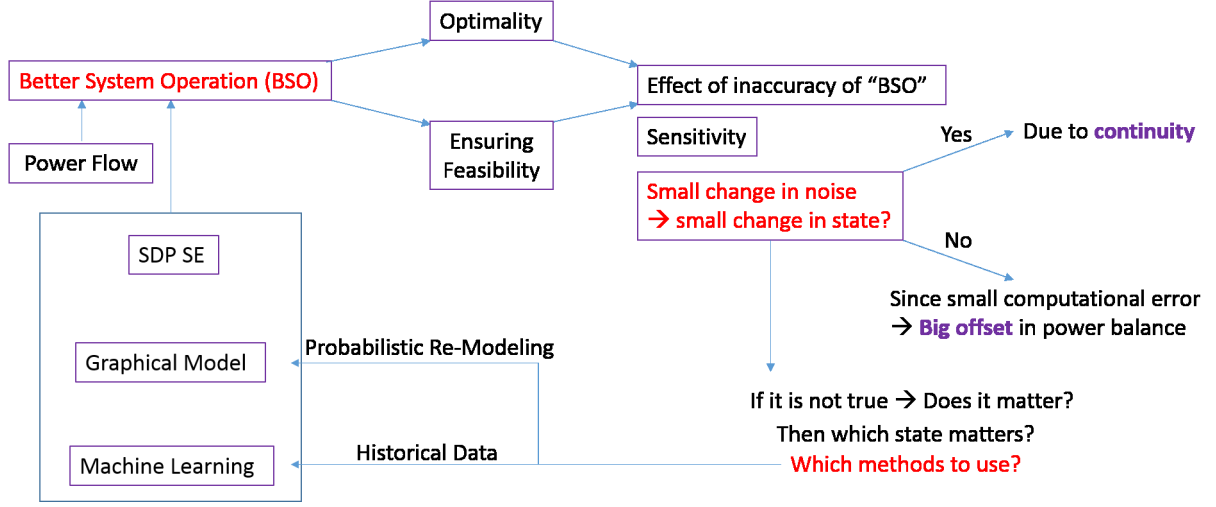


Figure 1.3: Logic among the three contributions.

Further, although the SDP-based approach above is an approximation approach, we can show that a similar approach guarantees the global optimum when there is no noise. Although this is a special case, we can show an approximate global optimum SE result via the statement of “Small change in the noise causes small change in the state” under the continuity assumption.

Finally, numerical results shows that our proposed SDP approach can improve upon the industrial approach by 30% on average in accuracy.

### 1.3.2 Data-Driven Approach

**The second contribution** of this thesis is demonstrating the potential to enhance current static SE with data mining over previously unused historical data.

For data-driven static SE, this thesis proposes a Bayesian approach based on a discriminative model. The proposed method is based on the idea that two similar system measurement sets usually indicate two similar operation conditions (system states). In particular, it first computes the distance between the current measurement set and the historical measurement set to identify a measurement set similar to the current one. Then, the identified set is fed into the discriminative model, obtained by embedding the measurement



space into a higher dimensional feature space. Then, parameters of the discriminative model were obtained via kernel ridge regression [14] in a supervised learning. Under the same framework, we further extend the kernel ridge regression from continuous variables to the kernel logistic regression for discrete variables. Under the same framework of Nearest Neighbors search, a historical data-driven topology identification approach is introduced.

For data-driven dynamic SE, this thesis aims at learning the parameters of a linear dynamic model based on historical data. Different from other works [15,16], the parameters in the proposed model are obtained via supervised learning over historical data with an Expectation-Maximization (EM) algorithm. This approach outperforms traditional static SE methods and dynamic SE with a linearized state space model [15,16]. The advantage is provided by its purely data-constructed modeling, where data can be flexibly organized for learning in accordance with the cyclic tendency of power system operation. To solve the inherent system nonlinearity, a polynomial kernel function inspired by the proposed SDP approach is employed for feature space embedding.

Our simulations show that the proposed data-driven approach is particularly useful when reliable historical data is available. The improvement could be as high as 90% in the state domain.

### 1.3.3 Graphical Model Approach

Besides historical data, the increasing computational resource in various part of the power network gives the grid many powerful local computational abilities. **Our third contribution** lies in the graphical modeling of uncertainties in the smart grid, which utilizes these local computational abilities and historical data for a probabilistic and distributed state estimation. Such a graph representation is motivated by an abstract model of ‘large-scale physical networks’ that are represented as a graph. The vertices of the graph represent state variables, which have a concrete physical interpretation as voltage. Other auxiliary variables of interest represent physical quantities, quantities that obey known physical laws (such as Kirchoff’s laws) that determine the interaction of these state variables. In the graph, these interactions are represented by the edges of the graph. Based on the graphical model, this thesis conducts a computationally-tractable distributed inference called Variational Belief Propagation (VBP) [17], whose basic form of “Belief Propagation” has recently had significant success in application to areas such as expert reasoning, information theory, communications, and image processing. VBP embeds the probability model into a larger probability space, so that the

estimation problem can be solved approximately.

Our numerical results show that when distributed intelligences are available and the historical data reflect good priori on the state, the accuracy improvement is 80% on average. The improvement in computation time is even higher; only  $\frac{1}{10}$  time of the current industrial approach is needed.

## 1.4 Notion of Embedding

An embedding process is described by a mathematical structure, where one structure is included in another one. Such a structure can be a topological object, manifold, graph, field, etc. Mathematically,  $f : A \hookrightarrow B$  is used to indicate that  $A$  is embedded in  $B$  in such a way that its connectivity or algebraic properties are preserved [18].

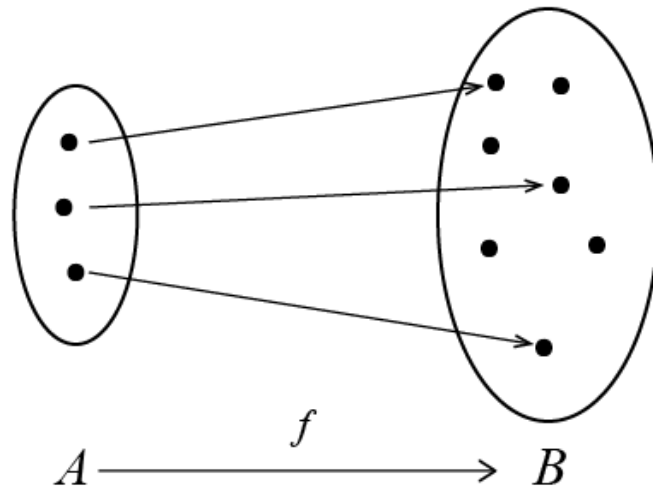


Figure 1.4: Embedding process.

Embedding techniques are used in different methods to handel non-linearity and design approximate SE algorithms.

- In Chapter 3, we embed system state in the space of positive semidefinite matrices to allow the use of a convex optimization relaxation;

- In Chapter 4, we embed the measurement space into a high-dimensional “feature space”, to allow the use of discriminative regression of the state.
- In Chapter 5, we embed the probabilistic graphical model related to the power grid in a larger probability space, so that Bayes estimation can be approximated using tree-based estimation for which exact solution exists.

## 1.5 Thesis Outline

The remainder of this thesis is organized as follows.

Chapter 2 presents various models for AC state estimation that will be employed in the thesis. For static SE, we review the classical estimation model and propose a Bayesian estimation model and supervised learning model. For dynamic SE, we employ dynamic state space representation.

Chapter 3 introduces a convex relaxation-based semidefinite programming approach to deal with non-linearity and noise in static state estimation model. To evaluate its performance, a new metric called “confidence” is derived. To reduce the computational complexity, a distributed semidefinite programming approach is introduced. Finally, we obtain an exact solution for a special case, where no noise exists.

To utilize unused historical data, Chapter 4 designs a supervised learning framework to improve state estimation. Kernel ridge regression is proposed in a Bayesian framework based on robust Nearest Neighbors search. To enable online data-driven SE, techniques such as dimension reduction and  $k$ -dimensional tree indexing are employed with a 1000 time speedup in the simulations. As an extension, topology identification is demonstrated with the Nearest Neighbors approach but with kernel logistic regression.

Chapter 5 proposes a probabilistic graphical model to account for new uncertainties in the power grids. Subsequently, a single (joint) most likely state(s) is efficiently located via local sum-product (or max-product) computation in a distributed belief propagation manner achieved by the increasingly available cyber (i.e., computational and communication) intelligence. To improve the algorithm, a sequential tree re-weighted message passing algorithm is proposed to guarantee optimality, deal with convergence issues, and reduce required computer memory.

Chapter 6 carries out numerical simulations of the proposed methods under different scenarios using the IEEE standard test-bed systems. The simulation results illustrate the tradeoffs among various proposed

approaches, which are compactly represented in a flow chart.

Chapter 7 provides a summary of this thesis and points out future research directions.

## Chapter 2

# Problem Formulation

While somewhat under-appreciated, modeling that captures the phenomena of interest is the basis for analysis and control approaches that address the temporal and spatial complexities of large-scale electric power systems. In recent years, new sensing and communication technologies have been deployed to enable the identifications of parameters in the various models. At the same time, since the electrical power industry is being deregulated, open access has been provided to competitive independent entities. In this chapter, we introduce models needed for analysis and control design in future electric power grids. Various static and dynamic modelings of the state estimation (SE) problem are presented. The goal is to capture the impacts of new entities on system-level response, so that the local behavior of these entities and the interactions among them can be derived.

In Section 2.1, we introduce several static SE models. In particular, in subsection 2.1.1, we introduce the widely used WLS model with AC system measurement equations. In subsection 2.1.2, a Bayesian SE model is introduced by additional compact priori information. In sub-section 2.1.3, a supervised learning model is introduced by using historical information. All of the above can be regarded as static approaches, and we define a dynamic state space model in Section 2.2. In Section 2.3, the inherent relationship among these models is summarized.

## 2.1 Static AC State Estimation

### 2.1.1 Classical Estimation Model

For static AC power system state estimation, the measurement model is typically expressed as

$$z_i = h_i(\mathbf{v}) + u_i, \quad (2.1)$$

where the vector  $\mathbf{v} = (|v_1|e^{j\delta_1}, |v_2|e^{j\delta_2}, \dots, |v_n|e^{j\delta_n})^T$  represents the power system states (the complex voltages of all the buses) in Fig.2.1.  $u_i$  is the  $i^{th}$  additive measurement noise, assumed to be an independent Gaussian random variable with zero mean, i.e.,  $\mathbf{u} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ , where  $\Sigma$  is a diagonal matrix, with the  $i^{th}$  diagonal element  $\sigma_i^2$ .  $z_i$  is the  $i^{th}$  telemetered measurement, such as power flow and voltage magnitude.  $h_i(\cdot)$  is the nonlinear function associated with the  $i^{th}$  measurement.

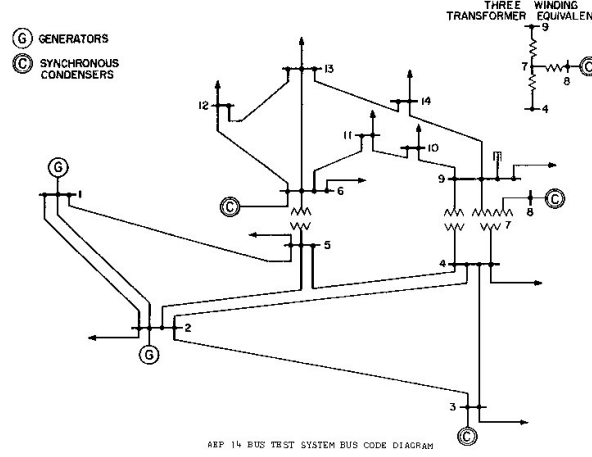


Figure 2.1: IEEE 14 bus system.

The power system state estimator aims to find an estimate ( $\hat{\mathbf{v}}$ ) of the true state ( $\mathbf{v}$ ) that best fits the measurement set  $\mathbf{z}$  according to the measurement model in (2.1), which is usually achieved by minimizing the following criterion

$$\min_{\mathbf{v}} J_p(\mathbf{v}) = \sum_{i=1}^m \left| \frac{z_i - h_i(\mathbf{v})}{\sigma_i} \right|^p, \quad (2.2)$$

where the parameter  $p$  ( $p \geq 0$ ) is chosen to achieve the desired performance. For example, for  $p = 1$ , the above problem reduces to the WLAV estimation [7] known to be robust against bad data. For  $p = 2$ ,

the above problem corresponds to the conventional WLS estimation. The WLS or WLAV estimations are to solve (2.2), subjecting to (2.1). However, the optimization problem in (2.2) is highly nonconvex and difficult to solve optimally. Therefore, in practice, such a nonconvex estimation problem is conventionally solved by applying Newton's method, a local search algorithm highly sensitive to the initial guess.

**Extension** One can also extend the formulation to include substation topology identification by adding  $M_i \mathbf{f}$  into the measurement model (2.1), where the vector  $\mathbf{f}$  is used to model the vector of power flows through the circuit breakers of suspect substations [7], and  $M_i$  represents the circuit breaker incidence matrix. Therefore, the extended power system state estimators aim to find an estimate  $(\hat{\mathbf{v}}, \hat{\mathbf{f}})$  of the true state  $(\mathbf{v}, \mathbf{f})$  that best fits the measurement set  $\mathbf{z}$  according to the measurement model in (2.1).

$$\min_{\mathbf{v}, \mathbf{f}} J_p(\mathbf{v}, \mathbf{f}) = \sum_{i=1}^m \left| \frac{z_i - h_i(\mathbf{v}) - M_i \mathbf{f}}{\sigma_i} \right|^p, \quad (2.3)$$

Topology identification can be achieved with the formula above using a two-step method in [7]. In step one, we let  $p = 2$ , and  $\mathbf{f} = \mathbf{0}$ .

$$\min_{\mathbf{v}} J_2(\mathbf{v}) = \sum_{i=1}^m \left( \frac{z_i - h_i(\mathbf{v})}{\sigma_i} \right)^2. \quad (2.4)$$

Then, the estimated  $\hat{\mathbf{v}}$  is used to calculate the estimated measurement residual:  $\left( \frac{z_i - h_i(\hat{\mathbf{v}})}{\sigma_i} \right)^2$ . Afterwards, the bus with the largest residual is extended into the sub-station model. In the second step, (2.3) with  $p = 2$  and  $\mathbf{f} \neq \mathbf{0}$  is used to identify the circuit breaker status based on estimated sub-station flow information  $\mathbf{f}$ .

$$\min_{\mathbf{v}, \mathbf{f}} J_2(\mathbf{v}, \mathbf{f}) = \sum_{i=1}^m \left( \frac{z_i - h_i(\mathbf{v}) - M_i \mathbf{f}}{\sigma_i} \right)^2. \quad (2.5)$$

Once the analog  $\mathbf{f}$  is obtained, the analog estimate is used to predict the discrete status of the bus breakers in the substation for topology identification.

### 2.1.2 Bayesian Estimation Model

When prior information or a Bayesian inference is preferred, the following model can be used. Later, we can show that as a generalization of the classic estimation model, Bayesian model is a natural extension of WLS approach for scalability with probabilistic estimates, which are desirable in the future smart grid.

The probabilistic measurement model of static AC power system state estimation is expressed the same as (2.1), but with  $\mathbf{v}$  representing the probabilistic power system states, instead of the conventionally used deterministic states. In this new method, a power grid is defined as a physical graph  $G(V, E)$  with vertices  $V$  representing the buses (slack, generator, load buses) and edges  $E$  representing their interconnection (transmission lines, etc.). It can be visualized as the physical layer in Fig.2.2. The probabilistic power system state

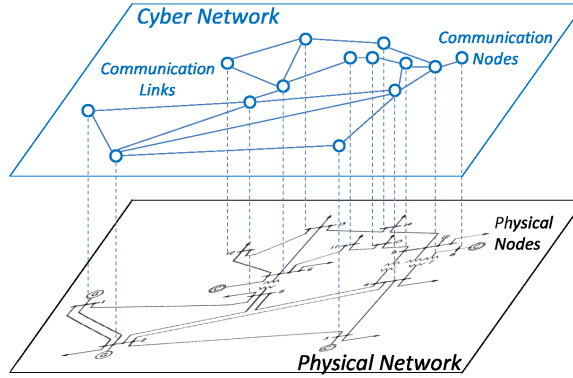


Figure 2.2: Physical network and cyber network (14 bus system).

estimator aims to find an estimate ( $\hat{\mathbf{v}}$ ) of the true state ( $\mathbf{v}$ ) that achieves the maximum a posteriori probability (MAP), given the measurement set  $\mathbf{z}$  and the priori information on state  $\mathbf{v}$  according to the measurement model in (2.1). It is mathematically expressed as

$$\max_{\mathbf{v}} p(\mathbf{v}|\mathbf{z}) = \frac{p(\mathbf{v})p(\mathbf{z}|\mathbf{v})}{p(\mathbf{z})}, \quad (2.6)$$

where  $p(\cdot)$  represents the probability density function. Such a process is achieved via the cyber network layer as in Fig.2.2. In this thesis, the cyber network topology is the same as the physical network topology.



### 2.1.3 Discriminative Estimation Model

Discriminative models are a class of models that models the dependence between unobserved variables and observed variables. Different from Generative model, which relies joint distribution between unobserved variables and observed variables, Discriminative model does not require such information. It is usually designed to directly learn the posterior probabilities under a supervised learning framework [19].

Mathematically, we assume that the conditional distribution of state set  $\mathbf{x}$  given the measurement set  $\mathbf{z}$  follows a probability distribution.

$$\mathbf{x}|\mathbf{z} \sim p(\mathbf{x}|\mathbf{z}) \quad (2.7)$$

The goal is to use historical data below to learn the parameters that describe the conditional probability.

- a sequence of historical state estimates column vectors:  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k, \dots, \mathbf{x}_Q$ ;
- a sequence of historical measurement function sets:  $\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_k, \dots, \mathbf{h}_Q$ .

## 2.2 Dynamic AC State Estimation

### 2.2.1 State Space Model

If fast measurements, i.e. PMU, are available, one can utilize the physical dynamics in addition to the historical data.

To introduce historical data-driven SE into power systems, we consider a sequence of historical SCADA measurements obtained every 2 seconds or less. These measurements are represented as a discrete time sequence  $\{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k, \dots, \mathbf{z}_Q\}$ . The corresponding states are  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k, \dots, \mathbf{x}_Q\}$ . The measurement equations (i.e. power flow equations) relate the measurements and states as

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{w}_k, \quad (2.8a)$$

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{u}_k, \quad (2.8b)$$

where the state vector  $\mathbf{x}_k = (|v_{k,1}|e^{j\delta_{k,1}}, |v_{k,2}|e^{j\delta_{k,2}}, \dots, |v_{k,n}|e^{j\delta_{k,n}})^T$  represents the power system states ( $n$ -bus system) at time slot  $k \geq 0$ .  $f : R^n \rightarrow R^n$  represents the state transition function that maps the

state from time slot  $k - 1$  to  $k$ , and  $h : R^n \rightarrow R^m$  represents the measurement function taken at time slot  $k$ .  $\mathbf{w}_k$  and  $\mathbf{u}_k$  are additive processing noise and measurement noise vectors, both of which are assumed to be independent Gaussian random variables with zero mean and a diagonal covariance matrix, i.e.,  $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \Gamma)$ ,  $\mathbf{u}_k \sim \mathcal{N}(\mathbf{0}, \Sigma)$ .  $\mathbf{z}_k$  is a telemetered measurement column vector at time slot  $k$ , such as power flow and voltage magnitude. Notice that  $u_i$  and  $\mathbf{u}_k$  represent two different but related terms.  $u_i$  is used in a static fashion, where the subscript  $i$  represents the measurement noise index in the measurement set in the current time slot. Elsewhere, the  $\mathbf{u}_k$  represents a noise vector in the measurement set with an assigned historical time slot index  $k$ .

Power system dynamic state estimation aims to find a sequence of estimates  $\{\hat{\mathbf{x}}_k\}$  of the true state sequence  $\{\mathbf{x}_k\}$  that fits the measurement set sequence  $\{\mathbf{z}_k\}$  best according to the dynamic model in (2.8) with a minimized mean-square error (MMSE). Due to the difficulty in dealing with nonlinearity seen in (2.8), the following linear dynamic system is adopted in practice,

$$\mathbf{x}_k = A\mathbf{x}_{k-1} + \mathbf{w}_k, \quad (2.9a)$$

$$\mathbf{z}_k = C\mathbf{x}_k + \mathbf{u}_k, \quad (2.9b)$$

where matrices  $A$  and  $C$  are the state transition matrix and the measurement matrix respectively.

## 2.3 Chapter Summary

While facing many challenges, the fast evolving power grid also incorporates various information components, such as high performance computers, large set of historical data, and distributed computation. How to model these new components and try one's best to eliminate problems, such as local optimum and the inability to detect a large number of topology errors, is extremely meaningful work for both the academic and engineering fields. Furthermore, since the Electric Power Network is an example of case for more general concepts such as cyber-physical systems, such a modeling process with its strong logic will help the modeling processes for other CPS, such as watering systems.

In the next three chapters, we will utilize the models introduced in this chapter in different scenarios. Then several state estimators are proposed. Performance guarantees are obtained in some cases.

## Chapter 3

# Semidefinite Programming for Convexifying Static State Estimation

While a decoupled-real-power-voltage state estimator that uses a linear (DC) power flow formulation is a convex weighted least square optimization problem, a fully coupled power-voltage AC state estimation problem is generally a very nonlinear, nonconvex problem [1, 20–26]. For example, the currently used static AC State Estimation (SE) has both a nonconvex performance objective (2.2) and nonlinear measurement model (2.1). This causes numerical problems in state estimators (2.2)–(2.5), which rely on Newton’s method. Newton’s method is known as a local search algorithm that is highly sensitive to the initial guess. Due to its non-convexity, current SE has a local optimum issue that leads to inaccurate estimation results. A further issue is the lack of an analytical tool to assess the performance of the current static state estimator. Therefore, conventional approaches in today’s SE are prone to sub-optimal solutions, which creates an unacceptable gap between the true and estimated voltages.

The proposed approach in this chapter is necessitated by open questions concerning the ability to compute the SE global optimum. Specifically, we propose a convex relaxation approach that convexifies the SE problem for a global optimum estimate by looking into the structure of static SE formulation. To lay the foundation for our analysis, we introduce briefly Semidefinite Programming (SDP) in Section 3.1. Then we review the current state-of-the-art SDP method for power system analysis in Section 3.2. We introduce in Section 3.3 a centralized SDP-based SE approach to solve the non-convexity problem for the current sub-optimal SE problem by using the embedding techniques introduced in Section 1.4. We then explain

the specific structure of the problem before reaching an efficient algorithm for finding the global optimum, namely, the most accurate estimate of the state. To evaluate the performance of our proposed SE, we derive a lower bound, which can also be used for arbitrary static state estimator in the measurement domain. We show that our estimate is close to the global optimum even when measurement noise is present, while the current SE only finds a local optimum. Unfortunately, the proposed centralized method requires a large computational overhead. To solve this problem, we introduce in Section 3.4 a distributed version to facilitate the practical applications of the proposed SDP-based SE. Finally, in Section 3.5, we derive another SDP-based approach capable of solving the static SE precisely in a no-noise setting. Numerical results are illustrated in each section, and Section 3.6 summarizes the entire chapter.

We briefly introduce SDP formulation in below. Interested readers may refer to the book Convex Optimization [27].

### 3.1 Introduction to Semidefinite Programming

Generally speaking, Semidefinite programming (SDP) is an optimization problem that aims to minimize (maximize) a linear objective function over a region described by a positive semidefinite cone. Although a recent theory, SDP has become increasingly popular, and has drawn increased research interest over the past few decades. Researchers may have different opinions, but one direct cause is the positive-semidefiniteness constraint's natural appearance for many practical problems in operations research, combinatorial optimization, and control theory. Examining its relationship to other optimization classes, we observe that many existing and well-known convex optimization problems, such as linear programming and (convex) quadratically constrained quadratic programming, can be reformulated in an SDP context, offering a unified way to study the properties and derive algorithms for a wide variety of convex optimization problems. Most importantly, SDP can be solved very efficiently both in theory and in practice in a polynomial time frame due to its convex characteristics.

Mathematically, SDP optimization [28] is defined as a problem to minimize a linear function  $J$  of a variable  $\mathbf{x} \in R^n$ , subjecting to a linear matrix inequality.

$$\begin{aligned} \min_{\mathbf{x}} J(\mathbf{x}) &= \mathbf{c}^T \mathbf{x} \\ \text{subject to } F(\mathbf{x}) &\preceq 0, \end{aligned} \tag{3.1}$$

where the coefficient vector  $\mathbf{c} \in R^n$  and  $F(\mathbf{x}) \triangleq F_0 + \sum_{i=1}^n x_i F_i$ , with matrices  $F_0, \dots, F_n \in R^{n \times n}$  symmetric. The inequality sign in  $F(\mathbf{x}) \preceq 0$  means that  $F(\mathbf{x})$  is positive-semidefinite, i.e.,  $\mathbf{c}^T F(\mathbf{x}) \mathbf{c} \geq 0$  for all  $\mathbf{c} \in R^n$ .

In semidefinite programming, the convexity property of positive semidefinite constraints plays a key role. For example, let  $F(\mathbf{x})$  be a matrix based on variable  $\mathbf{x}$ , and use  $F(\mathbf{x}) \succeq 0$  to indicate that the matrix  $F(\mathbf{x})$  is positive semidefinite. Then if  $F(\mathbf{x}) \succeq 0$  and  $F(\mathbf{y}) \succeq 0$ , we can show that the convex combination of the  $\mathbf{x}$  and  $\mathbf{y}$  inputs to the function  $F$  equals the convex combination of  $F(\mathbf{x})$  and  $F(\mathbf{y})$ .

$$F(\lambda \mathbf{x} + (1 - \lambda) \mathbf{y}) = \lambda F(\mathbf{x}) + (1 - \lambda) F(\mathbf{y}) \succeq 0 \quad (3.2)$$

where  $\lambda$ ,  $0 \leq \lambda \leq 1$ .

We will now use an example to explain how semidefinite programming works and we will form a semidefinite programming for power system analysis later in the chapter [28]. Consider the minimization of  $\frac{(\mathbf{c}^T \mathbf{x})^2}{\mathbf{d}^T \mathbf{x}}$ , where  $\mathbf{x}$  is the optimization variable, and  $\mathbf{c}$ ,  $\mathbf{d}$  are known vectors,

$$\underset{\mathbf{x}}{\text{minimize}} \quad \frac{(\mathbf{c}^T \mathbf{x})^2}{\mathbf{d}^T \mathbf{x}}. \quad (3.3)$$

To convert (3.3) into a convex form, we rewrite its equivalent formulation as follows:

$$\begin{aligned} \min_t \quad & t \\ \text{s.t.} \quad & \begin{pmatrix} t & \mathbf{c}^T \mathbf{x} \\ \mathbf{c}^T \mathbf{x} & \mathbf{d}^T \mathbf{x} \end{pmatrix} \preceq 0. \end{aligned} \quad (3.4)$$

The optimization above has a linear objective, which is convex. It also features a positive semidefinite constraint, which is convex as shown in (3.2). (3.4) is a standard form for positive semidefinite programming. Notably, the quadratic form in (3.3) is similar to the square in the sum square error of WLS. This is one of the keys as to why Semidefinite Programming is suitable for SE.

### 3.2 Current State-of-the-Art of Semidefinite Programming (SDP) for Power System Analysis

Similar to State Estimation (SE), Optimal Power Flow (OPF) has played an important role in power system analysis ever since its introduction in 1962 [29]. Due to the non-convexity of OPF formulation, conventional iterative algorithms can only guarantee local optimal solutions. An interesting feature of the OPF problem is that power flow equations can be formulated as a quadratic form of complex voltages in a Semidefinite matrix. By writing the power flow equations with this semidefinite matrix, the original nonconvex OPF problem can be re-written into a convex Semidefinite Programming form. Therefore, the formulation can convexify partial OPF optimization. Once a convex relaxation techniques are applied, the convexified OPF problem can be solved with a global optimum result within polynomial time.

Following this observation, [30] conducted initial analysis in convexifying Optimal Power Flow (OPF) problems. Recently, [12, 13] studied AC OPF problems and showed that the SDP-based approach is a promising method for overcoming the nonlinearity problem by working in the more structured and embedded space induced by SDP formulation. Their SDP-based approach applies convex relaxation techniques to transform the nonlinear, non-convex optimization problem into a convex optimization. But a rank-one constraint involved may hinder the reformulation process, because it can not be easily removed. The study reached its conclusion by assuming that such a non-convexity constraint does not affect the final optimization results. However, practical evaluation shows the opposite. While practical OPF problems where the SDP method generates zero duality gaps [31, 32] do exist, there are also many cases where the duality gap is strictly nonzero.

For example, some current ongoing research is uncovering the conditions that could bring about for zero duality gap. These conditions include highly limiting requirements on power injection and voltage magnitude limits. They also impose either constraints on the radial networks or an unrealistically extensive placement of controllable phase shifting transformers [12]. [32] shows that non-convexities in association with small subsections of the network are responsible for non-zero duality gap solutions in larger networks.

While providing many promising results, the SDP approach was also observed to be computationally expensive [33], because the optimization variable expands from a vector variable to a more complex matrix variable. To remove such limit, [34] utilizes graph theory results to conduct a parallel computation of OPF that leads to a faster computation time for the SDP approach.

Motivated by the ongoing success in solving OPF problems with an SDP-based approach, [33,35] applied similar ideas to AC state estimation, which also has a major challenge of non-convexity. The application focuses on improving SE accuracy by convexifying the SE problem. In particular, our prior work [33] successfully achieved complex voltage re-construction by utilizing the dependency among the intermediate states with the more structured and embedded space induced by SDP. Here, we refer to this method as the SDP-based convex relaxation method for state estimation.

A recent work [36] also tries to embed state space in order to solve the nonlinear SE problem non-iteratively. By treating a pair-wise, voltage-product as a new state, a linear set of equations is obtained. Subsequently, the state of a non-linear AC power system can be solved directly in a non-iterative manner by applying the “re-linearization” technique promoted by the study. This solution is direct because it utilizes neither the traditional Least-Squares objective function nor its derivatives.

### 3.3 Semidefinite Programming-Based State Estimation

It has been observed that solving nonlinearity and noise together at one time is difficult. One can try to use a DC approximation to resolve the nonlinearity problem first, and focus on the noise in the optimization. Although such a method creates large estimation error, it eliminates the local optimum problem, and has a close form solution with global optimum properties. Similarly, to solve them separately, we seek to explore the special structure of AC SE in this section and propose an SDP-based approach.

Such an SDP-based approach is motivated by the very recent research about convexifying AC Optimal Power Flow (AC OPF) using an SDP approach [12] as described in Section 3.2. This thesis attempts to convexify the AC SE problem given in [7] by posing the SE problem formulation as an SDP problem. This idea is based on the recognition that AC OPF and AC SE have similar structures. Notably, AC SE problem formulation (2.1) - (2.2) is distinctive from AC OPF problem formulation in that it does not impose explicit inequality constraints on system input (generation, demand), system output (line flows, reactive power of generators), nor on system states  $v$ . As such, it does not have the related feasibility problems of AC OPF [13].

Specifically, we perform an algebraic transformation of the direct SE problem formulation by embedding the state space in a feature space of positive semidefinite matrices. As a result, the original non-convex problem is relaxed into an equivalent but convex semidefinite programming (SDP) problem. Notably, the

nonlinearity problem temporarily disappears in the relaxed problem, and a global optimum can be reached. Then, we recover an estimate of the work in the original state space by using an eigenvalue decomposition. In the following, we will illustrate the know-how to convexify the static SE problem with generalization to bad data detection and topology error identification. We will also provide a new metric called SE confidence to measure the performance of various estimators.

### 3.3.1 Embedding of State

Much the same as in the recent SDP formulation of AC OPF, one can reformulate a conventional SE problem as an SDP problem. This is done with the intent to arrive at a good convex approximation to the non-convex SE problem. To start with, instead of using complex voltage  $\mathbf{v}$  as the optimizing variable, we use  $W \triangleq \mathbf{x}\mathbf{x}^T$ <sup>1</sup> in state estimation where  $\mathbf{x} \triangleq (Re(\mathbf{v})^T, Im(\mathbf{v})^T)^T$  has been embedded as a rank-one matrix  $W$ ; the coupled AC power flow equations can be expressed in a quadratic form in  $\mathbf{x}$  but linear in  $W$ . For instance, if the  $i^{th}$  measurement  $z_i$  is on the  $k^{th}$  active power injection, then we enforce  $\mathbf{f} = \mathbf{0}$  in (2.3) and let  $h_i(\mathbf{v}) = P_k$ , which can be reformulated as:

$$P_k = Re\{\mathbf{v}_k^H \mathbf{i}_k\} = Re\{\mathbf{v}^H \mathbf{e}_k \mathbf{e}_k^T Y \mathbf{v}\} \triangleq Re\{\mathbf{v}^H Y_k \mathbf{v}\} \quad (3.5)$$

$$= \mathbf{x}^T \tilde{Y}_k \mathbf{x} = tr(\tilde{Y}_k \mathbf{x} \mathbf{x}^T) = tr(\tilde{Y}_k W), \quad (3.6)$$

where matrix  $Y_k = \mathbf{e}_k \mathbf{e}_k^T Y$ ,  $\mathbf{e}_k$  is the  $k^{th}$  standard basis vector.  $Y$  is the bus admittance matrix defining  $\mathbf{i} = Y \mathbf{v}$ . (3.5) comes from the definition of active power and (3.6) comes from the definition of  $\tilde{Y}_k$  below

$$\tilde{Y}_k = -\frac{1}{2} \begin{pmatrix} Im(Y_k + Y_k^T) & Re(Y_k - Y_k^T) \\ Re(Y_k^T - Y_k) & Im(Y_k^T + Y_k) \end{pmatrix}. \quad (3.7)$$

As a result, the nonlinear measurement model (2.1) expressed in terms of voltages becomes a linear model in terms of the matrix  $W$  in lifted state space of positive semidefinite matrices as  $z_i = tr(\tilde{Y}_k W) + u_i$ . This simpler mathematical formulation requires the  $W$  matrix to be both positive semidefinite and rank-one; this is necessary to map uniquely a solution for  $W$  to the voltage state space. Unfortunately, rank-one condition is a non-convex constraint. To preserve the convex SDP-based SE formulation it becomes

---

<sup>1</sup>A recent study [36] also tries to extend state space in order to solve the nonlinear problem non-iteratively. By treating a pair-wise, voltage-product as a new state, a linear set of equations is obtained. Subsequently, the state of a non-linear AC power system can be solved directly in a non-iterative manner by applying the “re-linearization” technique promoted by the study. This solution is direct because it utilizes neither the traditional Least-Squares objective function nor its derivatives. Interestingly, this method does not need an initial guess.



necessary to relax this rank-one condition on matrix  $W$ ; this relaxed solution is expected to be a good approximation of the full SDP-based nonlinear problem. In this thesis, we further extend the SDP-based SE to include topology identification explicitly where we no longer assume  $\mathbf{f} = \mathbf{0}$ ; instead,  $z_i = \text{tr}(\tilde{Y}_k) + M_i \mathbf{f} + u_i$ . In such a case, although the states are extended, convexity holds because of the linear combination with no additional constraints.  $L_1$  norm is also added for robustness. This is derived next in some detail.

### 3.3.2 Alternative SDP-Based SE Formulations

Based on the discussion above, the nonconvex optimization that is equivalent to (2.3) is

$$\begin{aligned} \min_{W, \mathbf{f}} J_p(W, \mathbf{f}) &= \sum_{i=1}^m \left| \frac{z_i - \text{tr}(A_i W) - M_i \mathbf{f}}{\sigma_i} \right|^p \\ \text{subject to} \quad &W \succeq 0, \quad \text{rank}(W) = 1, \end{aligned} \quad (3.8)$$

where  $A_i$  is similar to  $\tilde{Y}_k$  in (3.6) depending on the measurement type. Its specific form can be found in Table 3.1 with the formulation below the table.

Table 3.1: The expression of  $A_i$  in (3.8)

Measurement type for $z_i$	Corresponding $A_i$
The $k^{th}$ active (reactive) power injection	$\tilde{Y}_k$ ( $\hat{Y}_k$ )
The $k^{th}$ active (reactive) power flow ‘from’	$\tilde{Y}_{fk}$ ( $\hat{Y}_{fk}$ )
The $k^{th}$ active (reactive) power flow ‘to’	$\tilde{Y}_{tk}$ ( $\hat{Y}_{tk}$ )
The $k^{th}$ voltage magnitude (phase angle)	$V_k$ ( $\Delta$ )

- Active power flow at the from (to) side  $P_{fk}$  ( $P_{tk}$ ) of the  $k^{th}$  branch can be computed similar to  $P_k$ . Instead of using  $Y_k$ , use  $Y_{fk} = \mathbf{e}_{g(k)} \mathbf{e}_k^T Y_f$ , where  $g(k)$  is a function that maps the branch index  $k$  into its ‘from’ side bus index  $g(k)$ .  $Y_f$  is the branch admittance matrix relating the bus voltages to the branch current vector  $\mathbf{i}_f$ , at the ‘from’ end of all branches, respectively.  $Y_{tk}$  is defined similarly with the replacement of  $f$  with  $t$ .

- Reactive power injection formulation:  $Q_k = \text{tr}(\hat{Y}_k W)$ ,

$$\hat{Y}_k = -\frac{1}{2} \begin{pmatrix} \text{Im}(Y_k + Y_k^T) & \text{Re}(Y_k - Y_k^T) \\ \text{Re}(Y_k^T - Y_k) & \text{Im}(Y_k^T + Y_k) \end{pmatrix}, \quad (3.9)$$

where  $Y_k = e_k e_k^T Y$ .

- Reactive power flow at the from (to) side  $Q_{fk}(Q_{tk})$  of the  $k^{th}$  branch can be computed similar to  $Q_k$ . Instead of using  $Y_k$ , use  $Y_{fk}$  and  $Y_{tk}$ .
- Voltage magnitude formulation: Since the direct use of voltage magnitude  $|v_k|$  with the SDP form is hard, an alternative form  $|v_k|^2$  is provided as  $|v_k|^2 = \text{tr}(S_k W)$ , where

$$V_k = \begin{pmatrix} e_k e_k^T & 0 \\ 0 & e_k e_k^T \end{pmatrix}. \quad (3.10)$$

**Remark** Here, it is supposed that  $|v_k|' = |v_k| + u_{v_k}$ , where  $u_{v_k}$  is the  $k^{th}$  voltage magnitude measurement noise, defined as a Gaussian random variable with zero mean and variance  $\sigma_{v_k}^2$ . By approximation,  $|v_k|'^2 = (|v_k| + u_{v_k})^2 = |v_k|^2 + 2|v_k|u_{v_k} + u_{v_k}^2 \approx |v_k|^2 + 2|v_k|e_k$ , can be regarded as a random variable with mean  $|v_k|^2$  and variance  $4|v_k|^2\sigma_{v_k}^2 \approx 4\sigma_{v_k}^2$ , because the voltage magnitude  $|v_k|$  is close to 1. Note that,  $|v_k|^2$  does not have a zero mean and causes biases in this approximation. However, this bias issue does not occur when the voltage measurements are not very noisy. This is especially true when a PMU is used to generate the voltage magnitude Measurements. The bias issue is a question open for future investigation.

- Voltage phase angle formulation: Again, since the direct use of voltage phase angle  $|\delta_k|$  with the SDP form is hard, an alternative form of  $\sin^2 \delta_k$  is used. Since the bus phase angles are usually small with respect to the reference bus, we employ the common approximation as follows:  $\sin^2(\delta_k') = \sin^2(\delta_k + u_{\delta_k}) \approx (\delta_k + u_{\delta_k})^2 \approx \delta_k^2 + 2\delta_k u_{\delta_k} + u_{\delta_k}^2$ . Therefore we can model it as a random variable with mean  $\delta_k^2$  and variance  $2\delta_k u_{\delta_k} + u_{\delta_k}^2$ . Finally, in SDP form

$$\Delta_k = \begin{pmatrix} e_k e_k^T & 0 \\ 0 & 0 \end{pmatrix}. \quad (3.11)$$

**Remark** Notice that one of the differences between the OPF problem and the SE problem lies in the voltage magnitude measurements and phase angle measurements of SE, when we conduct an approximation using SDP-Based SE. They do not appear in the OPF framework.

**Lemma 3.3.1.** *Problems (2.2) and (3.8) are equivalent.*

*Proof.* We have already shown that  $h_i(\mathbf{v}) = \text{tr}(A_i W)$  for the bus branch level in (3.5) - (3.6) where  $M_i \mathbf{f}$  is omitted.  $M_i \mathbf{f}$  can be added to the measurement model owing to its linear nature. Therefore we just need to illustrate the one-to-one mapping between the state vector  $\mathbf{v}$  and the state matrix  $W$  as follows. First, by the definition of  $\mathbf{x}$  and  $W$ , there is only one constructed  $W$  for every  $\mathbf{v}$ . Reversely, since  $W$  is positive semidefinite, symmetric and rank-one,  $W$  has one unique pair of eigenvalue  $\lambda_1$  and eigenvector  $\mathbf{g}_1$  owing to the spectrum decomposition theorem. Therefore, there is only one  $\mathbf{x} = \sqrt{\lambda_1} \mathbf{g}_1$  for every  $W$  such that  $\mathbf{x}\mathbf{x}^T = W$ .  $\square$

Problem (3.8) can be written in an equivalent SDP form with various norms. In particular, if  $p = 1$ , we obtain

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^m \alpha_i \\ \text{s.t.} \quad & \begin{pmatrix} z_i - t_i - \alpha_i \sigma_i & 0 \\ 0 & -z_i + t_i - \alpha_i \sigma_i \end{pmatrix} \preceq 0, \\ & t_i = \text{tr}(A_i W) + M_i \mathbf{f}, \quad W \succeq 0, \quad \text{rank}(W) = 1. \end{aligned} \quad (3.12)$$

where  $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_m)^T$ . If  $p = 2$ , we obtain

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^m \alpha_i \\ \text{s.t.} \quad & \begin{pmatrix} z_i^2 - 2t_i z_i - \alpha_i \sigma_i^2 & t_i \\ t_i & -1 \end{pmatrix} \preceq 0, \\ & t_i = \text{tr}(A_i W) + M_i \mathbf{f}, \quad W \succeq 0, \quad \text{rank}(W) = 1. \end{aligned} \quad (3.13)$$

**Lemma 3.3.2.** *Problems (3.8) and (3.12) are equivalent for  $p = 1$ ; Problems (3.8) and (3.13) are equivalent for  $p = 2$ .*

*Proof.* When  $p = 2$ , following the definition of negative semidefinite constraint in (3.13), we have

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix}^T \begin{pmatrix} z_i^2 - 2t_i z_i - \alpha_i \sigma_i^2 & t_i \\ t_i & -1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \leq 0 \quad (3.14)$$

for all  $y_1$  and  $y_2$  with  $t_i \triangleq \text{tr}(A_i W) + M_i \mathbf{f}$ . In rewritten form,

$$y_2^2 - (2y_1 t_i) y_2 - (z_i^2 - 2t_i - \alpha_i \sigma_i^2) y_1^2 \geq 0 \quad (3.15)$$

By choosing  $y_1 = 1$  and  $y_2 = t_i$ , we see that condition  $(z_i - t_i)^2 \leq \alpha_i \sigma_i^2$  must hold for (3.8). Proof of “(3.8) and (3.12) are equivalent when  $p = 1$ .” is similar and thus omitted.  $\square$

### 3.3.3 Convex Relaxation of the SDP-Based SE

Again, due to the nonconvex rank-one constraint in (3.12), the problem is hard to solve. Since most of the optimization component is convexified, we consider its relaxed form by dropping the rank-one constraint, which results in the new approximate problem stated below. (When  $p = 2$ , the formulation is the same as (3.13), but with the rank-one constraint over  $W$  dropped.)

$$\begin{aligned} \min_{\alpha} \quad & \hat{J} = \sum_{i=1}^m \alpha_i \\ \text{s.t.} \quad & \begin{pmatrix} z_i - t_i - \alpha_i \sigma_i & 0 \\ 0 & -z_i + t_i - \alpha_i \sigma_i \end{pmatrix} \preceq 0, \\ & t_i = \text{tr}(A_i W) + M_i \mathbf{f}, \quad W \succeq 0. \end{aligned} \tag{3.16}$$

Now, since the optimization in (3.16) is with a linear objective function over the intersection of a cone of positive semidefinite matrices in an affine space, we successfully constitute a convex semidefinite programming problem according to [27]. As a subfield of convex optimization, such a problem can be efficiently solved with software. Later, we will illustrate that the solution here is close to the global optimal solution of (3.8), leading to an efficient estimate.

### Reconstructing Estimated States

Although there is a one-to-one mapping between the state matrix  $W$  and the state vector  $\mathbf{v}$  by Lemma 3.3.1, we lose such a unique mapping due to the removal of the rank-one constraint in (3.16) for convexification purposes. Since the  $\widehat{W}$  from (3.16) is now only approximately rank-one, we will use the largest eigenvalue to recover first the intermediate vector variable  $\mathbf{x}$ , and then the state vector  $\mathbf{v}$ . Specifically, with a matrix  $W$  having rank  $r \geq 1$ , we can use the eigenvalue decomposition for state recovery:  $\widehat{W} = \sum_{i=1}^r \lambda_i \mathbf{g}_i \mathbf{g}_i^T$ , where  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_r \geq 0$  are the eigenvalues and  $\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_r$  are the corresponding eigenvectors. Since  $\lambda_1 \mathbf{g}_1 \mathbf{g}_1^T$  is the optimal rank-one approximation for  $W$ , we define  $\hat{\mathbf{x}} \triangleq \sqrt{\lambda_1} \mathbf{g}_1 \in \mathcal{R}^{2n \times 1}$  as the estimate for  $\mathbf{x}$ , which can be uniquely mapped into  $\hat{\mathbf{v}} = \hat{\mathbf{x}}_{1:n} + j \cdot \hat{\mathbf{x}}_{n+1:2n}$ , with  $j$  as the imaginary unit. Notably, further enhancement can be readily conducted by feeding  $\hat{\mathbf{v}}$  into the initial guess for the classical Newton’s method.

Another advantage of such an initial guess is that it is based on the current measurements rather than the commonly used previous state estimation, making the SDP method suitable for a system emergency restart.

### 3.3.4 Computational Cost and Extensions

A major disadvantage of the proposed SDP method arises when one looks at the increasing optimization variables number, which increases from  $\mathcal{O}(n)$  in the pure WLS method to  $\mathcal{O}(n^2)$ . As a result, the SDP approach is much more computationally expensive than the WLS approach, which suggests the need for a high end computer for a large system. Later in this chapter, we will propose a distributed algorithm approach to avoid the need for such large computing power.

### 3.3.5 New Metric for Optimality Evaluation

The drawback of WLS that uses Newton's method as a solver is not only its inability to find the global optimum, but also its inability to evaluate the goodness of a local optimum. While the first problem was solved with an SDP-based approximation, the second part will be solved in this subsection by giving a tight lower bound. Specifically, after obtaining the state matrix  $\widehat{W}$  (may not be rank-one), one can directly compute the cost function in (3.16). Problem (3.16) has a global minimum smaller than or equal to the global minimum in (3.8); this provides a lower bound to the original problem given in (2.2), which is illustrated in the flow chart of the new approach in Fig.3.1.

The lower bound helps us understand how close we are to the best possible solution to the problem. The gap tells us the maximum possible savings we could hope to achieve by improving our algorithm. Notably, such a lower bound holds under broad conditions for different state estimations, bad data detections, and topology identification with the same types of measurements. For instance, in state estimation, the system operator can simultaneously run several estimation methods (i.e., WLS with Newton's method or the SDP-based method) as in Fig.3.2a. In each time slot of Fig.3.2b, different estimators may give different results. For example, Method 1 is the best at the first interval near time slot zero, and method 2 is the best in the second time interval. Then one can compute the lower bound derived from our SDP method, and calculate the distance between the two sum square errors as in Fig.3.2b. This gap (the distance in between) is called "confidence" in this thesis by using the current best estimator. Therefore, the increasing computational ability in the power grid not only enables the system operator to have the current best estimate, but also

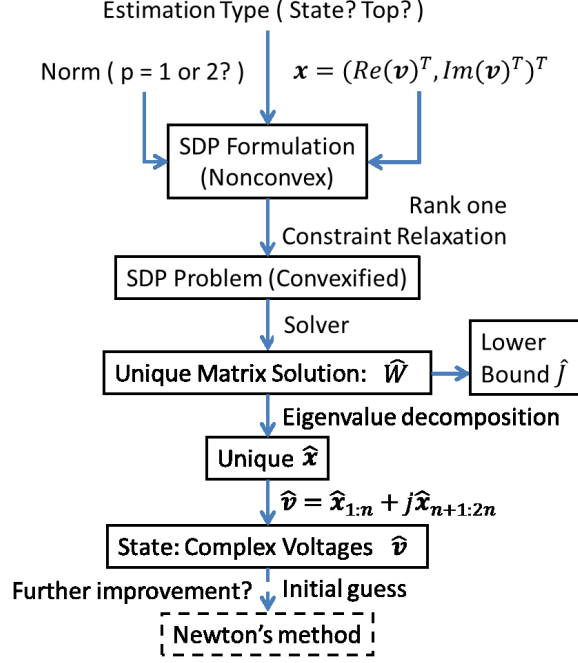


Figure 3.1: Flow chart of the proposed algorithm.

to obtain a new metric to visualize the confidence of the current best estimate. Similarly, system operator can use such a metric to obtain a confidence for different estimators in bad data detection and topology identification as shown in Fig.3.3.

### 3.3.6 Simulation Results

In this section, we simulate and verify the significantly improved performance of the convex relaxation method for the various estimators (State Estimator, Bad Data Detector, and Topology Estimator) and the usage of the new metric of confidence. No historical data is used.

#### Simulation Set-Up

The simulations are implemented on the IEEE standard test systems for IEEE 14, 30, 39, 57, 118 buses. Similar performance improvements were observed. The 14 bus and 30 bus simulation results are presented. The data has been preprocessed by the MATLAB Power System Simulation Package (MATPOWER) [37, 38]. To obtain the measurements, a power flow is first run to generate the true states of the power system,

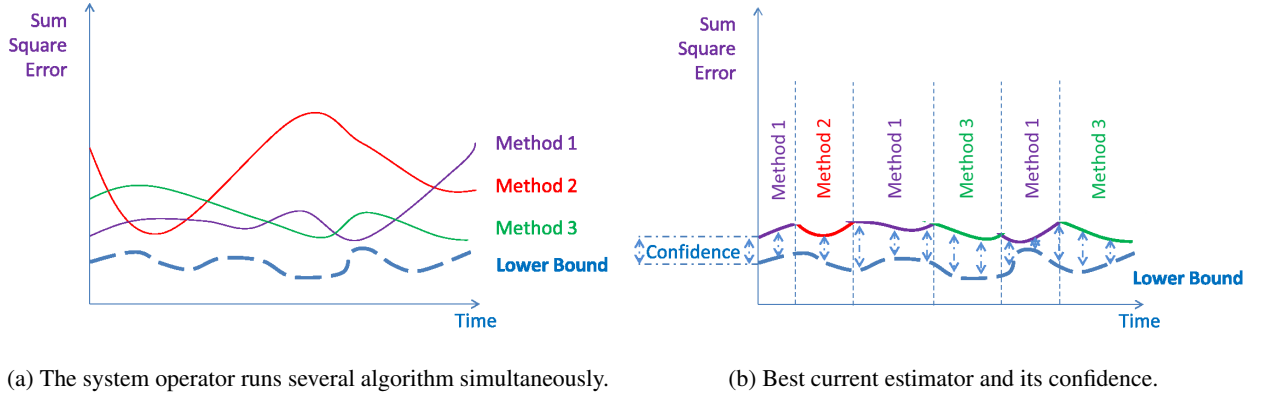


Figure 3.2: A new metric and its application.

after which Gaussian noise is added to the corresponding measurements. The measurements include the following: (1) the power injection on each bus; (2) the transmission line power flow ‘from’ or ‘to’ each bus that it connects; (3) the direct voltage magnitude of each bus and (4) the voltage phase angle of each bus. The measurements are randomly chosen assuming guaranteed system observability. The measurement number is usually chosen to be around three times the bus number.

## Numerical Results

**State Estimation** First, the performance of the SDP approach is demonstrated in the conventional state estimation scenario, where measurements are corrupted only by Gaussian noise. Besides, the tolerance

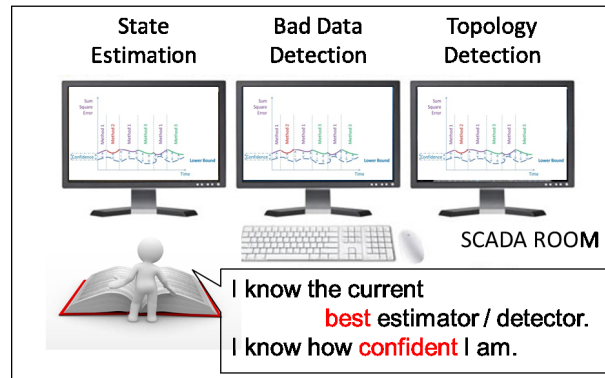
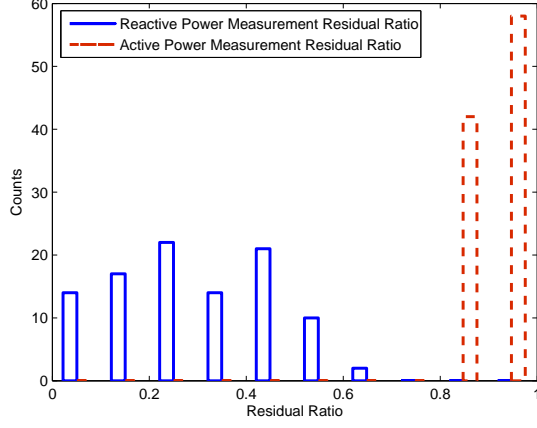
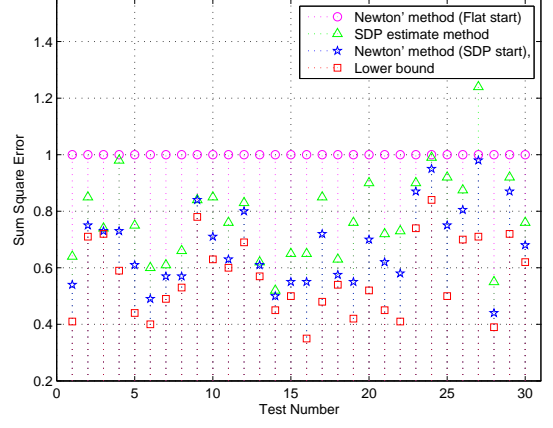


Figure 3.3: SCADA.



(a) Histogram



(b) WRSS

Figure 3.4: IEEE 14 bus (Gaussian noise).

(stopping criteria) is the same as what is defined in Matpower ( $10^{-5}$ ), no matter whether Newton's method is initialized with a flat start or an SDP start. The simulation results in this section are checked to satisfy the first-order necessary condition ( $H^T \Sigma^{-1} [z - h(x)] = \mathbf{0}$ , where  $H$  is the measurement Jacobian matrix). The noise is generated according to standard deviations: (1) 1.5% for power injection measurements, (2) 2% for power flow measurements, and (3) 1% for the other measurements. Second, the SDP is solved by the "SEDUMI" package [39]. After obtaining the state matrix  $\widehat{W}$  in (3.16), low rank matrix recovery from subsection 3.3.3 is used to obtain the estimate  $\hat{v}$ . For comparison purposes, this estimate is used as an initial guess for Newton's method to obtain a locally optimal estimate.

When using the Matpower SE simulation package in the IEEE 14 bus case, we observed a large residual in the reactive power measurements. For instance, one can run the "test\_se\_14bus.m" file in the Matpower path "matpower4.0\extras\se" to see the large residuals for the WLS with a flat start. In Fig.3.4a we plot the histogram of the averaged active (reactive) power measurement residual ratio of the SDP method over Newton's method for a flat start in the 14 bus system over the course of 100 simulations. The red bars are for the active power measurement residual ratio and the blue bars are for the reactive power measurement residual ratio. No numerical problem appears in the simulation. It can be observed that the two methods do not differ much (the ratios are close to 1.) in active power measurement performance but show significant differences (the ratios are much smaller than 1.) in reactive power measurement performance. The proposed SDP method successfully reduces the reactive power residuals, which has been a well known problem in



traditional state estimation. The ability of our approach to deal with large reactive power residuals can be attributed to the convexification process. By solving the convexification problem, we greatly reduce the large residual due to more accurate state estimation.

To display the near optimum estimation and the new metric of confidence, Fig.3.4b displays Weighted Residual Sum of Squares (WRSS) error for all measurement types, which is defined as

$$\text{WRSS} = \sum_{i=1}^m \left( (z_i - \text{tr}(A_i W)) / \sigma_i \right)^2. \quad (3.17)$$

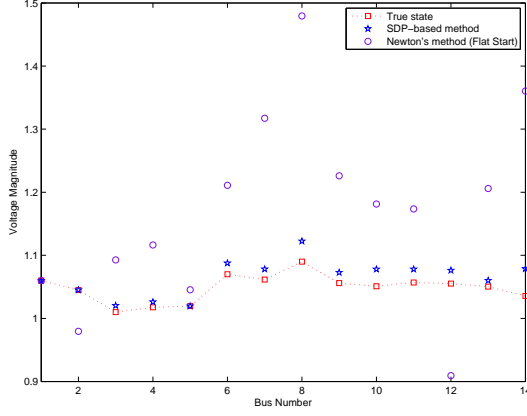
The  $x$  coordinate is the test number for the simulation. The  $y$  coordinate is for the metric WRSS. Through observation, we see that Newton's method with an SDP initial guess can significantly reduce the error of WLS (by at least 20%) compared to Newton's method with a flat start. At some points, such as simulation number 6, 14 and 28, more than 50% improvement is achieved. In addition, the small distance (the new confidence metric) existing between the SDP initial guess method and the lower bound informs that the proposed SDP initial guess method is close to the global optimum. Such facts lead to a natural interpretation of the proposed SE procedure: the possibility for Newton's method to hit the global optimum is greatly increased since the SDP method finds a closer initial state than the flat start method does.

**Bad Data** In this subsection, we consider both Gaussian noise and bad data and a measurement set is randomly generated by choosing an observable set. To make the estimation robust against bad data, the cost function (2.3) with  $p = 1$  (WLAV approach) in this simulation is chosen so that the WLAV problem is solved with a bus-branch model. Similar results can be obtained to those in the state estimation in Fig.3.5.

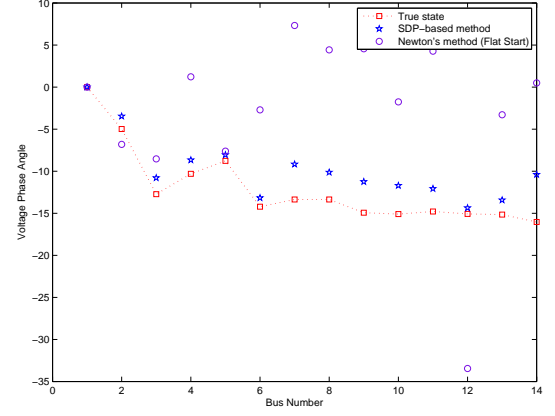
Fig.3.5a and Fig.3.5b show the improvements in the state domain in one simulation. From the figure, we can see that the blue start (SDP approach) is close to the red rectangle (true state). But the purple circle (Flat start) is much further away from the red rectangle.

**Topological Estimation** Topological estimation deals with the occasionally unreported removal of customer nodes, sensor failures, and manually updated breakers. Without accurate topological estimation, SCADA systems will provide the wrong topology to the subsequent state estimation, resulting in erroneous consumer power cut-offs or even a blackout throughout the grid.

The well-known method [7, 40, 41] for topological estimation has two steps: the first step is to use the bus branch model with WLS and the Chi-square test to locate the suspected substation (abnormal mea-



(a) Voltage magnitudes



(b) Voltage phase angles

Figure 3.5: IEEE 14 bus (Gaussian noise plus bad data).

surements); in the second step, the buses associated with the suspected measurements are extended with a substation model to include the circuit breaker power flows as supplemented states for further topology analysis. Since the breaker status can be regarded as bad data, the WLAV method is usually employed [42] to enhance robustness.

In this subsection, simulations are conducted on the IEEE 30-Bus test system (Fig.3.6a) with one randomly chosen topology error. Similar performance improvements are observed. The closed branch 12 – 15 is recorded to be open in this simulation.

In stage 1, the WLS problem is solved by the SDP approach in (3.13) on a bus-branch level in Fig.3.6a to search suspicious buses. Demonstrated in Table 3.4, with the new method, the residual increases significantly in measurements related to the wrong topology, and vice versa. For example, the residual for the wrong measurement  $PF_{12-16}$  (active power flow on branch 12 – 16 at the ‘from’ end) increases from 0.0044 to 0.012, with increasing suspicions about bus 12 as a consequence.

In stage 2, the bus-branch model is extended to include the detailed substation model [40,41] in Fig.3.6b. The SDP estimate with (3.12) is used. Finally, the estimated power flow states are used to determine the correct topology. In Table 3.5, the estimated power flow (SDP method) decreases significantly for the actually open circuit breaker, compared to Newton’s method with a flat start.

Table 3.2: Stage 1: Measurement normalized residuals

Measurement	Relative to the wrong topology?	$ r_N _{SDP}$	$ r_N _{Flat}$
PF 12-16	Yes	0.0120	0.0044
QT 15-23	Yes	0.0271	0.0068
PF 2-6	No	0.0064	0.0312

Table 3.3: Stage 2: Estimated status of circuit breakers

Circuit breaker	Power flow (SDP)	Power flow	Actual state
PF 12-32	0.0389	0.0380	Closed
QT 31-36	0.0042	0.0166	Open
PF 37-38	0.0106	0.0317	Open

### Computational Cost

Fig.3.7 shows a comparison of the computational time used by the regular WLS method and the SDP method. The computational time of the SDP method exceeds that of the regular WLS method, in both the length of time used and its growing rate. Notably, the IEEE 300 bus runs out of time due to the computational burden for centralized SDP-Based SE. From the linear relationship of  $\log(time)$  in the  $y$  coordinate with respect to the bus number in the  $x$  coordinate, a prediction of  $10^8$  seconds equal to 3.17 year can be obtained for 300 buses. As a result, one can observe a tradeoff between accuracy and complexity. By obtaining better accuracy, the price to be paid for robustness is computational complexity, which can be deduced from the sophisticated mathematical approach proposed in this thesis.

To deal with the computational complexity, we will address in the next section how to employ a distributed algorithm to implement parallel computing for such a computational problem.

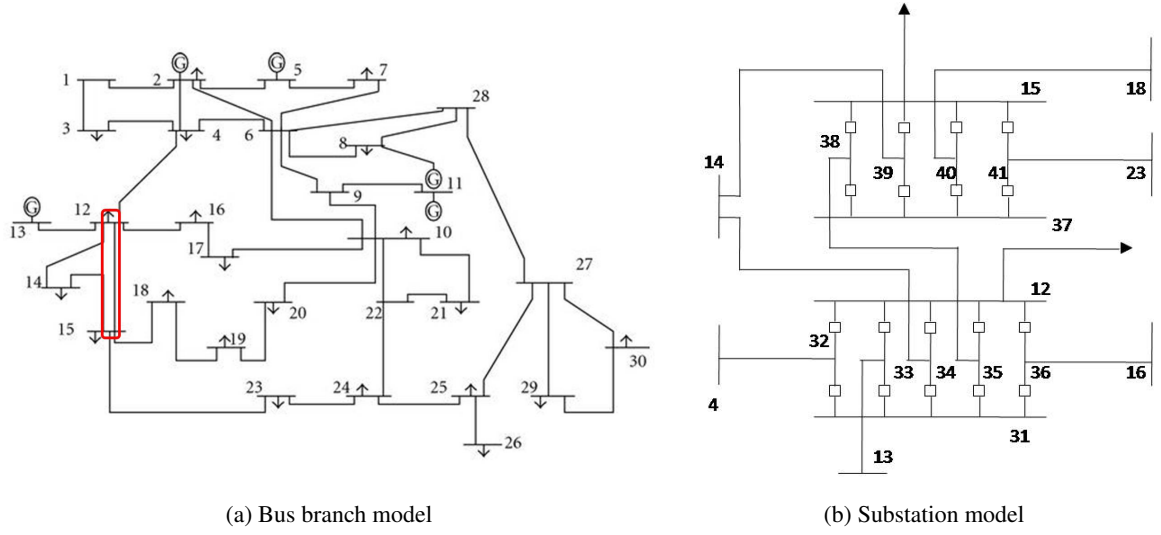


Figure 3.6: IEEE 30 bus.

### 3.4 Distributed Implementation for Semidefinite Programming-Based State Estimation

The SDP-based generalized State Estimator approach in the last section is shown to significantly reduce large residuals seen in conventional SE [7, 11] for large-scale electric power systems by convexifying the inherently nonlinear AC power flow problem embedded in the SE formulation; it is, therefore, near-globally

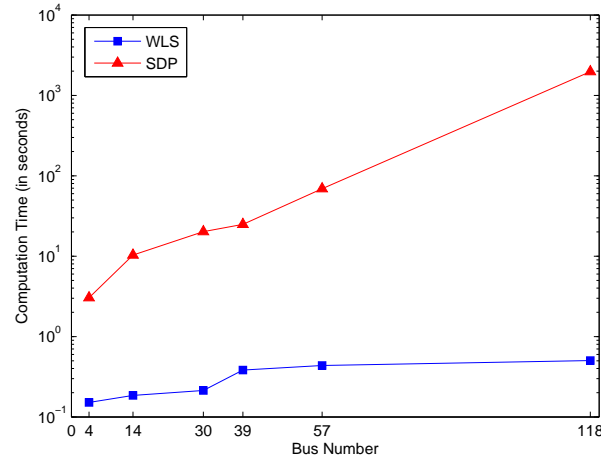


Figure 3.7: Computational time

optimal.

However, previous analysis also shows that a centralized implementation of the SDP-based SE is computationally prohibitive because it is fundamentally based on lifting the original state variables (complex-valued voltages) to a much higher-order state space defined by the matrix  $W$ . We have illustrated that even the IEEE test system comprising 300 network nodes is no longer computationally feasible using the SDP-based SE centralized method, implying that this method cannot be used for the typical electric power grids representing EHV/HV/MV utilities, which have nodes that number in the thousands; using this centralized algorithm for SE in the interconnected EHV/HV/MV/LV electric power grids is completely out of the question. Moreover, the SDP-based centralized SE approach also requires full information gathering and is generally non-robust with respect to partial network communication failures. This section attempts to overcome such disadvantages by proposing a distributed SDP-based SE algorithm that reformulates the centralized SDP computation into distributed SDP problems with no loss of accuracy.

Such algorithms are likely to form the basis for “smart” grids by enabling even many small system users to participate in enhancing system operation in predictable ways. For example, the state (power consumed, voltage) of smart meters would not have to be estimated by the operator of the backbone system based on the measurements provided by a particular location. Instead, the state of a small user gets estimated in a distributed, message-passing manner with neighboring system users which have smart meters. The aggregated information is then communicated in a bottom-up way to the backbone system operator.

For this multi-layered distributed data processing and SE vision to be implemented in a rigorous way, many questions must be answered. They concern the number of measurements necessary within each group of system users passing information multi-laterally, the type of information exchange necessary within the smaller groups of system users, and the information exchange necessary between the group of users and the rest of the system. Notably, the observability requirements and the ability to manage bad data and topological changes may become qualitatively different for distributed approaches than they are for today’s centralized approaches. In what follows we describe how a combined use of the Lagrangian dual decomposition method and a graph-theoretic approach to creating subnetworks (“cliques”) on chordal graphs supports a distributed implementation of an SDP-based SE method. We describe how such an algorithm could begin to answer the difficult questions above.

This reformulation is achieved by using the dual Lagrangian decomposition [43] method to repose this

problem first in a form that can be decomposed. Next, the decomposition of the system-level positive-semidefinite constraint on the  $W$  matrix is done by using the notion of induced chordal graphs and their decomposition, similar to the distributed implementation of the SDP-based AC OPF method [34, 44, 45]. However, we do not set limits on state, input variables, or output variables. This is because the performance objectives for state estimators are qualitatively different from the performance objective used for an AC OPF; instead of setting limits on operating ranges, as proposed in [34], approximate formula for some measurements (voltages and phase angles, in particular) are proposed. Different from [35], no topology assumption in control areas is required here. This section also extends the objective to deal with topology error. In addition, we add the  $L_1$  norm over the extension to deal with bad data. A proof of equivalence between the centralized algorithm and the distributed algorithm is added. To enable fully distributed estimation, coordinate descent approaches are proposed as an alternative to parallel computing for robustness [46]. To characterize the performance of local estimators, lower bounds are further derived to generate a metric called local confidence for the system operators.

As a result, by using message exchange on coupling nodes between neighboring local networks, the original centralized SDP problem can be characterized in a distributed manner by performing local SDP SE computation. Obviously, a direct consequence of this – if parallel processors are used – is computational time reduction. Notably, the proposed algorithm can also be implemented in a fully distributed way via coordinate decent. Each local network computes its local SDP-based estimator and passes the message to the neighbor nodes, which is robust to local network failure. A significant reduction in computational time and a fast rate of convergence, both critical for power system analysis, are illustrated through simulations.

### 3.4.1 Alternative Form for a Distributed Algorithm

The optimization problem in (2.2) is highly non-convex and difficult to solve optimally. In practice, the state estimation problem is conventionally solved by applying Newton’s method, a local search algorithm that is extremely sensitive to the initial guess. To convexify this, in Section 3.3, we proposed to state the SE problem as a convex relaxation-based SDP problem [33, 35]. Necessary approximations are described that help transform the centralized SE problem into a convex relaxation-based SDP problem. This formulation is summarized here for completeness, although we use a slightly different formulation that lends itself better to introducing the distributed implementation in the next section. Specifically, if we define a new complex matrix  $W = vv^H$ , where  $v^H$  represents the Hermitian of the vector  $v$ , the complex-valued power flow

constraints without considering the substation topology can be expressed as follows:

$$\mathbf{s}_{\text{inj}} = \text{diag}\{\mathbf{v}\}\mathbf{i}^H = \text{diag}\{\mathbf{v}\mathbf{v}^H\mathbf{Y}^H\} = \text{diag}\{\mathbf{W}\mathbf{Y}^H\}. \quad (3.18)$$

Further, by defining  $T^{\text{act}} = \frac{1}{2}(\mathbf{Y}^H + \mathbf{Y})$  and  $T^{\text{rea}} = \frac{1}{2}(\mathbf{Y}^H - \mathbf{Y})$ , we obtain real and reactive power balance equations

$$\begin{aligned} \mathbf{p}_{\text{inj}} &= \text{diag}\{\mathbf{W}(\mathbf{Y}^H + \mathbf{Y})\} = \text{diag}\{T^{\text{act}}\mathbf{W}\} \\ \mathbf{q}_{\text{inj}} &= \text{diag}\{\mathbf{W}(\mathbf{Y}^H - \mathbf{Y})\} = \text{diag}\{T^{\text{rea}}\mathbf{W}\} \end{aligned}$$

with the  $i^{\text{th}}$  real power injection  $p_{\text{inj},i} = \sum_{k=1}^n M_{ik}^{\text{act}}W_{ki}$ , and the  $i^{\text{th}}$  reactive power injection  $q_{\text{inj},i} = \sum_{k=1}^n M_{ik}^{\text{rea}}W_{ki}$ . As an example, if one were to use the real and/or reactive power measurements to estimate system state, the performance objective would become

$$\begin{aligned} \underset{\mathbf{W}, \mathbf{f}}{\text{minimize}} \quad & \sum_{i=1}^n \left[ \left( \sum_{k=1}^n T_{ik}^{\text{act}}W_{ki} - z_i^{\text{act}} - M_i\mathbf{f} \right)^2 \right. \\ & \left. + \left( \sum_{k=1}^n T_{ik}^{\text{rea}}W_{ki} - z_i^{\text{rea}} - M_i\mathbf{f} \right)^2 \right] \\ \text{subject to} \quad & \mathbf{W} \succeq 0. \end{aligned} \quad (3.19)$$

Examples of using other measurements are described in the Appendix. For illustrative purposes, we formulate next the distributed SE algorithm when power injection measurements are given.

### 3.4.2 Problem Decomposition

#### Objective Decomposition

In this section, we describe how one could use the Lagrangian dual decomposition [43] method to estimate states using real and reactive power injection measurements.

To decompose the objective in a quadratic form, we use the Lagrangian dual method. To conduct such a

technique, auxiliary vector variables  $\mathbf{y}^{\text{act(rea)}}$  are defined to make the optimization in (3.19) equivalent to

$$\begin{aligned}
& \underset{W, \mathbf{f}}{\text{minimize}} && \sum_{i=1}^n (y_i^{\text{act}})^2 + \sum_{i=1}^n (y_i^{\text{rea}})^2 \\
& \text{subject to} && W \succeq 0, \\
& && y_i^{\text{act}} = \sum_{k=1}^n T_{ik}^{\text{act}} W_{ki} - z_i^{\text{act}} - M_i^{\text{act}} \mathbf{f}, \\
& && y_i^{\text{rea}} = \sum_{k=1}^n T_{ik}^{\text{rea}} W_{ki} - z_i^{\text{rea}} - M_i^{\text{rea}} \mathbf{f}.
\end{aligned}$$

The associated Lagrangian is

$$\begin{aligned}
& L(W, \mathbf{f}, \boldsymbol{\lambda}^{\text{act}}, \boldsymbol{\lambda}^{\text{rea}}, \mathbf{y}^{\text{act}}, \mathbf{y}^{\text{rea}}) \\
&= \sum_{i=1}^n \left[ (y_i^{\text{act}})^2 + \lambda_i^{\text{act}} (y_i^{\text{act}} - \sum_{k=1}^n T_{ik}^{\text{act}} W_{ki} + z_i^{\text{act}} + M_i^{\text{act}} \mathbf{f}) \right] \\
&+ \sum_{i=1}^n \left[ (y_i^{\text{rea}})^2 + \lambda_i^{\text{rea}} (y_i^{\text{rea}} - \sum_{k=1}^n T_{ik}^{\text{rea}} W_{ki} + z_i^{\text{rea}} + M_i^{\text{rea}} \mathbf{f}) \right].
\end{aligned} \tag{3.20}$$

Its dual problem is formed by

$$\begin{aligned}
& \max_{\boldsymbol{\lambda}} \min_{W, \mathbf{f}, \mathbf{y}} L(W, \mathbf{f}, \boldsymbol{\lambda}^{\text{act}}, \boldsymbol{\lambda}^{\text{rea}}, \mathbf{y}^{\text{act}}, \mathbf{y}^{\text{rea}}) \\
& \text{subject to } W \succeq 0,
\end{aligned}$$

which can be reorganized into

$$\begin{aligned}
& \max_{\boldsymbol{\lambda}} \min_{W, \mathbf{f}, \mathbf{y}} \sum_{i=1}^n \left[ (y_i^{\text{act}})^2 + (y_i^{\text{rea}})^2 + \lambda_i^{\text{act}} (y_i^{\text{act}} + z_i^{\text{act}} + \right. \\
& \left. M_i^{\text{act}} \mathbf{f}) + \lambda_i^{\text{rea}} (y_i^{\text{rea}} + z_i^{\text{rea}} + M_i^{\text{rea}} \mathbf{f}) \right] - \sum_{i=1}^n \sum_{k=1}^n N_{ik} W_{ki} \\
& \text{subject to } W \succeq 0,
\end{aligned} \tag{3.21}$$

with  $N_{ik} \triangleq \lambda_i^{\text{act}} T_{ik}^{\text{act}} + \lambda_i^{\text{rea}} T_{ik}^{\text{rea}}$ .

Since the objective has a piecewise quadratic form over  $y_i^{\text{act}}$  and  $y_i^{\text{rea}}$  respectively, the optimal  $y_i^{\text{act(rea)}} =$



$-\frac{\lambda_i^{\text{act(rea)}}}{2}$ , which simplifies (3.21) into

$$\begin{aligned} \max_{\lambda} \min_{W, \mathbf{f}} \quad & \sum_{i=1}^n \left[ \lambda_i^{\text{act}} \left( z_i^{\text{act}} - \frac{\lambda_i^{\text{act}}}{4} \right) + M_i^{\text{act}} \mathbf{f} \right. \\ & \left. + \lambda_i^{\text{rea}} \left( z_i^{\text{rea}} - \frac{\lambda_i^{\text{rea}}}{4} \right) + M_i^{\text{rea}} \mathbf{f} \right] - \sum_{i=1}^n \sum_{k=1}^n N_{ik} W_{ki} \\ \text{subject to} \quad & W \succeq 0. \end{aligned} \quad (3.22)$$

As a result, unlike the initial quadratic objective, we successfully convert the optimization into a linear objective with respect to the element of matrix  $W$  and  $\mathbf{f}$ . This gain is obtained by performing an extra optimization step over the dual decoupled variables  $\lambda_i$ s.

### $L_1$ norm

If robustness is desired,  $L_1$  norm with the following reformulation can be used.

$$\begin{aligned} \text{minimize}_{W, \mathbf{f}} \quad & \sum_{i=1}^n |y_i^{\text{act}}| + \sum_{i=1}^n |y_i^{\text{rea}}| \\ \text{subject to} \quad & W \succeq 0, \\ & y_i^{\text{act}} = \sum_{k=1}^n T_{ik}^{\text{act}} W_{ki} - z_i^{\text{act}} - M_i^{\text{act}} \mathbf{f}, \\ & y_i^{\text{rea}} = \sum_{k=1}^n T_{ik}^{\text{rea}} W_{ki} - z_i^{\text{rea}} - M_i^{\text{rea}} \mathbf{f}, \end{aligned}$$

which can be reformed into

$$\begin{aligned} \max_{\lambda} \min_{W, \mathbf{f}, \mathbf{y}} \quad & \sum_{i=1}^n \left[ |y_i^{\text{act}}| + |y_i^{\text{rea}}| + \lambda_i^{\text{act}} (y_i^{\text{act}} + z_i^{\text{act}} + M_i^{\text{act}} \mathbf{f}) \right. \\ & \left. + \lambda_i^{\text{rea}} (y_i^{\text{rea}} + z_i^{\text{rea}} + M_i^{\text{rea}} \mathbf{f}) \right] - \sum_{i=1}^n \sum_{k=1}^n N_{ik} W_{ki} \\ \text{subject to} \quad & W \succeq 0. \end{aligned}$$

The only difference between the expression above and (3.22) is that we need to calculate  $y$  numerically, instead of obtaining closed-form solutions as we did in (3.22). In the next section, we will introduce the positive semidefinite (PSD) constraint decomposition method for (3.19). Naturally, elements of  $W$  in the objective can be grouped in the same way for distributed computation.

**Remark** The problem of an incomplete measurement set arises when measurements are not available on all buses, due to initial power grid planning, missing data, communication errors, etc. To deal with these problems, we can assume any unavailable  $z_i$  to be the same as the estimated measurement in each iteration for an iterative algorithm. As a result, the associated term in our problem formulation disappears. One may also employ pseudo-measurements if they are available.

## Constraint Decomposition

In this section, we decompose the PSD constraint over  $W$  of (3.19) into equivalent submatrices PSD constraints, according to [34, 47–49]. We also introduce some graph-theoretic concepts necessary for subsequent discussion. Particular emphasis is on using chordal graphs.

## Graph Basics

An undirected graph is denoted by  $G(V, E)$  with  $V$  indicating the set of its vertex, and  $E \subseteq V \times V$  the set of edges. In this paper, we assume no self-loops within a graph, or  $(v, v) \notin E$  for any  $v \in V$ . Two vertices  $u, v \in V$  are defined to be adjacent if  $(u, v) \in E$ . A graph is considered a complete graph if every pair of vertices is adjacent. For a subset  $V' \subseteq V$ , an induced subgraph  $G(V', E')$  has edge set  $E' = E \cap (V' \times V')$ . A clique of a graph is an induced subgraph that is complete. Formally, a clique is a set of pairwise adjacent vertices and a clique is maximal if its vertices do not constitute a proper subset of another clique. In our succeeding discussions, we call  $C \subseteq V$  a clique of  $G(V, E)$  whenever it induces a clique of  $G(V, E)$ . Further, a chord in a cycle is an edge connecting two non-consecutive vertices of the cycle. Therefore, a graph  $G(V, E)$  is said to be chordal, or equivalently triangulated, if every cycle of length exceeding four has a chord.

## PSD decomposition

In order to understand Proposition 3.4.1 below from another research area, we introduce here a graph-based mathematical problem called matrix completion, which is defined as the process of adding entries to a matrix with some unknown or missing values. If there are no underlying assumptions about the matrix, the unknown elements can be arbitrarily chosen, making matrix completion theoretically impossible. However, given a few assumptions about the matrix characteristics, such as positive semidefinite constraints, some algorithms

allow a complete reconstruction of the matrix. Mathematically, define a partially specified symmetric matrix  $W$  based on the graph  $G(V, F)$  for some edge set  $F \subseteq E$ . Its element  $w_{ij}$  is defined if and only if edge  $\{i, j\} \in F$ , so only parts of the matrix entries are specified. The completion process for such a partial symmetric matrix  $W$  aims to find a fully specified positive semidefinite matrix based on  $W$ . Denote by  $\{C_r \subseteq V : r = 1, 2, \dots, l\}$  the family of all maximal cliques of  $G(V, F)$ . An obvious necessary condition for  $W$  to have a PSD matrix completion is that each  $W_{C_r, C_r}$  is PSD ( $r = 1, 2, \dots, l$ ), with all submatrices  $W_{C_r, C_r}$  being completely specified. We refer to the above results as the clique-PSD condition.

**Proposition 3.4.1.** (*Grone et al. [47]*) *Any partial symmetric matrix  $W$  with specified entries  $(i, j) \in F$  representing a chordal graph, has a positive semidefinite matrix completion, if and only if all submatrices  $W_{C_r, C_r}$  ( $r = 1, 2, \dots, l$ ) are PSD.*

### Generate Chordal Graph

Unfortunately, the underlying graphs of typical power grids are non-chordal, so a power system engineer may ask the feasibility of such decomposition for power system state estimation problems. Luckily, some research, such as [48], illustrate how to apply chordal graph results to a non-chordal graph. The key idea is about fill-in; i.e. the triangulation of a graph  $G$  by adding virtual edges. In other words, one would like to fill in the unknown matrix terms with respect to the graph so that cliques are generated. So, how do we determine a chordal extension which has the smallest number possible since this number directly affects the algorithmic performance? Admittedly, the problem of finding a fill-in ordering that minimizes the total fill-in number is  $\mathcal{NP}$  complete [48] without a polynomial time algorithm. Hence, it seems reasonable at least in practice to employ some existing heuristic methods to obtain an ordering that might possibly produce lesser fill-in [48].

One method is to employ Cholesky decomposition. Because  $W$  is positive semidefinite, Cholesky factorization can be performed over it, creating  $W = LL^H$  (under the genericized assumption that no numerical cancellations occur in the elimination process). If we further define a graph  $G(V, F)$  based on  $L$ , we will have  $F \supseteq E$ , since the sparsity pattern of  $L$  is determined by that of the matrix  $W$ . The added edges, belonging to  $F \setminus E$ , correspond to the fill-ins, making graph  $G(V, F)$  a chordal extension of the underlying power grid graph  $G(V, E)$ .

Upon generating the chordal graph for a power grid, one needs to search all maximal cliques within it

via the efficient tool of perfect elimination ordering [50] in linear time with respect to the vertex and edge numbers in the graph. Here an elimination ordering is simply a numbering of the vertices with integers from 1 to  $n$ . So, based on the maximal cliques, we finally reach decomposition of the positive semidefinite constraint of  $W$  into many smaller semidefinite constraints of its submatrices.

**Remark** So far, the discussion of constraint decomposition is solely based on the structure of  $W$ , which is unknown in our problem formulation. Nevertheless, since  $W$  lies in the objective with an inner product by the system structure matrix  $N$ , one can extract the structure of  $W$  via  $N$  [48]. More details are given in the toy example.

### 3.4.3 Distributed Algorithm for Parallel Estimation of State

Next, we summarize the distributed SDP SE computing steps with the following theorem.

**Proposition 3.4.2.** *Optimization of (3.19) is the same as the following distributed algorithm:*

- *Step 1: Fix  $\lambda^{\text{act(rea)}}$  and the coupling variables in  $W$ . For each clique, compute its local SDP SE for the independent variables in  $W$  (and  $\mathbf{f}$ , if a substation model is used in topology identification).*
- *Step 2: Fix  $\lambda^{\text{act(rea)}}$  and the independent variables in  $W$ . Compute the SDP SE for the coupling variables in  $W$ .*
- *Step 3: Fix  $W$ . Distributively update  $\lambda$  with respect to the cliques.*
- *Step 4: If the algorithm does not converge, go to Step 1.*

*Proof.* In the ‘dual Lagrangian decomposition method’, each iteration aims at maximizing the primal variable (matrix  $W$ ) while fixing the Lagrangian multiplier  $\lambda$ . In our proposed parallel approach, this step is replaced by two maximization steps alternating between the independent variables and the coupling variables in  $W$ . This is because we can regard the optimization over  $W$  to be another primal decomposition problem according to [43]. The  $i$ th subproblem is:

$$\min_{W_i} f_i(W_i, W_c), \quad (3.23)$$

where  $W_i$  represents the local variables in the  $i^{th}$  area, and  $W_c$  is the set of coupling (boundary) variables in  $W$ . With a fixed  $W_c$ , the  $i$ th subproblem is assumed to have optimal value  $\phi_i(W_c)$ . Then the original problem is equivalent to the master problem:

$$\min_{W_c} \sum_i \phi_i(W_c) \quad (3.24)$$

with variable set  $W_c$ . This problem is called primal decomposition since the master problem manipulates the primal (complicating) variables. Since the original problem is convex, so is the master problem. Therefore, the two-step optimization is equivalent to a one-step optimization over  $W$ , thus leading to a legitimate decomposition.  $\square$

### Fully Distributed Algorithm for Robustness

The parallel algorithm in Theorem 3.4.2 needs coupling variables to be computed together, which limits the ability to deploy the algorithm in a fully distributed way. To enable the distributed computing of coupling variables, we introduce the coordinate descent method, which is based on the idea that the minimization of a convex multivariate function  $F(W)$  can be realized by minimizing it along one direction (one area) at a time [46]. Since the objective is convex, one can achieve the global optimum by fixing the descent direction at the outset, instead of varying the descent direction according to gradient. The key to this result is the separability of the objective function. For instance, one chooses some basis for the search directions:  $W_1, W_2, \dots, W_n$ , where  $W_i$  includes the independent and coupling variables in the  $i^{th}$  clique. One cyclically iterates through each direction, one at a time, minimizing the objective function with respect to that coordinate direction. It follows that, if  $x^k$  is given, the  $i^{th}$  coordinate of  $x^{k+1}$  is given by

$$\hat{W}_i^{k+1} = \arg \min_{W_i^{k+1}} f(\hat{W}_1^{k+1}, \dots, \hat{W}_{i-1}^{k+1}, W_i^{k+1}, W_{i+1}^k, \dots, W_n^k), \quad i \in \{1, 2, \dots, n\}. \quad (3.25)$$

Therefore, a fully distributed SDP SE can be achieved by locally updating the clique matrix  $\{\overline{W}_i\}$ , the dual variable  $\lambda$  associated with the clique, and then optimization of (3.19) is the same as the following distributed algorithm:

- Step 1: Each clique receives the  $W$  matrix from a neighbor clique.
- Step 2: Update  $\lambda$  in the clique according to  $W$ .

- Step 3: Fix  $\lambda$  in the clique, and update the local and coupling variables in the clique.
- Step 4: Pass the updated  $W$  matrix to the neighbor cliques.
- Step 5: If the algorithm does not converge, go to Step 1.

### 3.4.4 New Metric for Local Optimality Evaluation.

Note that, after obtaining the local state matrix  $\widehat{W}_c$  (may not be rank-one), one can directly compute its local cost function, which has a global minimum smaller than or equal to the global minimum in using  $x_c$ ; this provides a lower bound for local state estimation, bad data detection, and topology identification problems. Such an idea can be seen in Fig.3.8. Therefore, the system operator can use this lower bound for arbitrary estimators to evaluate their optimality, leading to a new metric of local estimation confidence for each area.

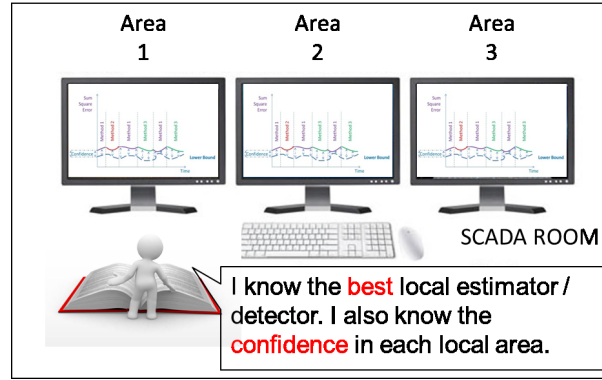


Figure 3.8: Local confidence in the SCADA room.

### 3.4.5 Simulation Results

In this section, simulations are implemented on the IEEE 4-Bus, 14-Bus, 30-Bus, 39-Bus, 57-Bus and 118-Bus test systems. Data generation is similar to the simulations for the centralized approach in Section 3.3.6.

#### Centralized Parallel Computing

**Computational time speed up** As an illustration, the extended chordal graph decomposition of the 14-bus case is shown in Fig. 3.9. Fig. 3.10 shows the a comparison of the computational time for centralized



time than the distributed one. This is because, for a small system, dividing the network into multiple sub-networks cannot substantially lower the optimization variable number. Instead, the computation of the dual variables may lay heavy burdens on the algorithm. Therefore, Fig. 3.10 suggests distributed computation for large systems and centralized computation for small systems.

**Same accuracy** To evaluate performance, we compare the weighted residual sum of squares error (WRSS) among the centralized SDP, decentralized SDP and traditional Newton's method, where  $WRSS = \sum_{i=1}^m ((z_i - \text{tr}(T_i W) - M_i f) / \sigma_i)^2$ ,  $m$  is the total measurement number. Here, we conduct the comparison multiple times with the IEEE 14-bus system. Fig.3.11 illustrates the simulation results after 30 times, where the  $x$  coordinate represents the test index number and the  $y$  coordinate represents the WRSS metric defined above. Evidently our proposed distributed algorithm essentially provides the same performance improvement as does the centralized SDP method in [33]. The slightly bigger error for the distributed algorithm may result from the convergence conditions.

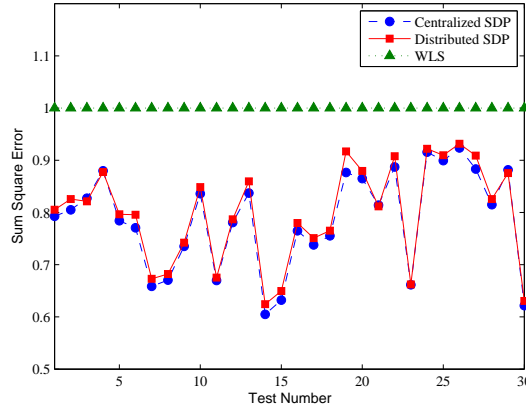


Figure 3.11: Accuracy comparison.

**Topological estimation improvement** One of the major contributions of this chapter is its generalization of SDP with respect to topology estimation, which deals with occasionally unreported removal of customer nodes, sensor failures, and manually updated breakers. Without accurate topological estimation, SCADA systems will provide the wrong topology for subsequent state estimation, resulting in erroneous consumer power cut-offs or even a blackout throughout the grid.



In this subsection, in order to compare our distributed algorithm with centralized algorithms in terms of accuracy and time, multiple simulations are conducted on the IEEE 30-Bus test system (Fig.3.6a) with one randomly chosen topology error. Similar performance improvements are observed.

In stage 1 [7, 40, 41], the WLS problem is solved by the SDP approach ( $\mathbf{f} = \mathbf{0}$ ) and Newton's method on a bus-branch level in Fig.3.6a. The goal is to search suspicious nodes. Table 3.4 refers to one simulation result obtained with the assumption that the actually closed branch 12 – 15 is open. We see in Table 3.4 that, similarly to centralized SDP SE, the residual of the distributed SDP SE increases significantly in measurements related to the wrong topology, and vice versa. For example, the residual for the wrong measurement  $PF_{12-16}$  (the active power flow on branch 12 – 16 at the 'from' end) increases from 0.0044 to 0.0102, with increasing suspicions on bus 12 as a consequence.

In stage 2 [7], the bus-branch model is extended to include the detailed substation model (Fig.3.6b) ( $\mathbf{f}$  is no longer  $\mathbf{0}$ ) of buses with the maximum number of suspect measurements from which an SDP estimate is obtained. An SDP estimate is obtained by efficiently solving the convex relaxation form of the centralized SDP SE optimization problem. Finally, the estimated power flow states are used to determine the correct topology. In table 3.5, we see that the estimated power flow (SDP method) decreases significantly for the actually open circuit breaker, compared to Newton's method with a flat start. For instance, in an actually open state, the calculated flow estimate of  $QT_{31-36}$  (the reactive power flow on the circuit breaker '31 – 36' at the 'to' end.) for an SDP initial guess (0.0067) is much smaller than the calculated flow estimate for a flat start one (0.0166), which prevents a mis-correction of the topology error. Therefore, the distributed SDP method also increases significantly robustness against topology errors. We use the 300-bus case to illustrate

Table 3.4: Stage 1: Measurement normalized residuals for distributed SDP

Measurement	Relative to the wrong topology?	$ r_N _{SDP}$	$ r_N _{Flat}$
PF 12-16	Yes	0.0102	0.0044
QT 15-23	Yes	0.0253	0.0068
PF 2-6	No	0.0066	0.0312

larger systems. In this case, we randomly disconnect one branch, but without telling the system about the topology change. Then the residual near the topology changes are computed based on (1) the WLS estimate with a flat start, and (2) the distributed SDP-based method. The following table shows the simulation results.

Table 3.5: Stage 2: Estimated status of circuit breakers for distributed SDP

Circuit Breaker	Power Flow (SDP)	Power Flow (Flat)	Actual State
PF 12-32	0.0387	0.0380	Closed
QT 31-36	0.0082	0.0166	Open
PF 37-38	0.0144	0.0317	Open

Table 3.6: IEEE 300 bus simulation results

Simulation Times	Residual Improvement	Identification Improvement
200	49.7%	17%

### 3.5 A Special Case for Achieving a Global Optimum

Since solving the problems of nonlinearity and noise together is hard, we proposed an SDP-based approach in the previous sections to deal with them separately. Although we are able to achieve a global optimum, the SDP-based approach is an approximation method. In this section, we analyze a special case where there is no noise. Although not realistic, it does give us an insight into why SDP-based approach can handle nonlinearity. In Fig.3.12 we illustrate the four scenarios when considering the noise problem and the nonlinear problem. The current SE result lets us have closed-form solution in linear cases regardless of whether there is noise. And this closed-form solution is globally optimal. When there is a nonlinearity problem, we can still try to separate it into two cases, where noise appears and where noise does not appear. It is well known from the previous sections that trying to resolve nonlinearity and noise together is hard. However, there is a special cases where the noise is extremely small or does not exist. By solving this special case, one can obtain insight into how to deal with nonlinearity, which may eventually help us understand how to solve nonlinearity and noise together in a perfect and less difficult way.

In this subsection, we seek to explore the special structure of the SE performance metrics more deeply. Specifically, we perform an algebraic transformation of the direct SE problem formulation. As a result, the original non-convex problem is converted into an equivalent, but convex semidefinite programming (SDP) problem formulation; this formulation is different from other SDP-based methods in Sections 3.3 and 3.4.

The advantages of the proposed formulation are: 1) the convex formulation is free from local minima

	With Noise	Noiseless
Nonlinear (AC model)	No closed-form Solution <b>Local Optimum</b>	No closed-form Solution <b>Global Optimum</b>
Linear (DC model)	Closed-form Solution Global Optimum	Closed-form Solution True Value

Figure 3.12: Four Scenarios of SE problems.

and there exist mature theories/technologies for algorithm initialization and step size choice. This makes the proposed convex optimization approach to SE robust; and 2) highly efficient algorithms are available for its solution. As a result, the simulation results in this paper indicate that our convex approach performs better than the standard approaches.

### 3.5.1 The New SDP-Based Approach

The SDP approach in Sections 3.3 and 3.4 optimizes over the extended space in matrix  $W$ , instead of  $\mathbf{v}$ , to offer more linearity. The drawback of this approach is the rank-one condition, which has neither a nice interpretation nor helps to obtain the global optimum. Recent publications show that the SDP-based OPF is generally hard because of its constraints, such as lower bounds on neighboring buses [13]. Interestingly, the proposed SE formulation does not have these constraints. The SE problem, when constructed as an unconstrained optimization problem, should be easier to solve.

In this subsection, we revisit the WLS formulation, look into the structure of the objective, and represent it in the real space

$$\min_{\mathbf{x}} f(\mathbf{x}) = \sum_{i=1}^m \left| \frac{z_i - \mathbf{x}^T \tilde{\mathbf{Y}}_i \mathbf{x}}{\sigma_i} \right|^2. \quad (3.26)$$

By writing it in this form without the rank-one constraint, the objective becomes a quartic polynomial. We recognize that this performance objective can be regarded as a composition of quadratic forms. The following lemma states the state-of-the-art [51] in the convexity of polynomials.

**Lemma 3.5.1.** *The question of deciding convexity is trivial for odd degree polynomials. Indeed, it is easy to check that linear polynomials ( $d = 1$ ) are always convex and that polynomials of odd degree  $d \geq 3$  can never be convex. Deciding convexity of degree for four polynomials is strongly NP-hard. This is true even when the polynomials are restricted so as to be homogeneous.*

Since the convexity is hard to check, we look into its equivalent problems. The above problem in (3.26) can be recast as

$$\begin{aligned} \max_{\alpha} \quad & \alpha \\ \text{subject to} \quad & f(\mathbf{x}) - \alpha \geq 0 \quad \forall \mathbf{x}, \end{aligned} \tag{3.27}$$

where  $f(\mathbf{x}) = \mathbf{x}^T \tilde{Y}_i \mathbf{x}$ .

In this conversion, dummy variable  $\alpha$  represents the horizontal hyperplane that lies beneath  $f(\mathbf{x})$  for every value of  $\mathbf{x}$ . By maximizing its value with the upper bound  $f(\mathbf{x})$ , it achieves the global minimum of  $f(\mathbf{x})$ . After this transformation, both the objective and the constraints are linear with respect to variable  $\alpha$ , leading to a convex optimization problem (3.27). Unfortunately, the constraint number is infinite because for each  $\mathbf{x} \in \mathcal{R}^{2n}$  there is an inequality constraint.

In order to explore the fourth-order polynomial structures of the objectives in (3.26), we define the convex cone  $\mathcal{A}$  of real-valued fourth-order polynomials of  $\mathbf{x}$  [52]

$$\mathcal{A} = \left\{ q \mid q(\mathbf{x}) \text{ is a real-valued fourth-order polynomial of } \mathbf{x} \text{ and } q(\mathbf{x}) \geq 0, \forall \mathbf{x} \right\},$$

which is the convex cone of nonnegative real-valued polynomials of degree 4.

**Lemma 3.5.2.** *Optimization in (3.26) is equivalent to the following optimization with respect to  $\mathcal{A}$*

$$\begin{aligned} \max_{\alpha} \quad & \alpha \\ \text{subject to} \quad & f(\mathbf{x}) - \alpha \in \mathcal{A}. \end{aligned} \tag{3.28}$$

Further, to explore the composition of quadratic forms of the objectives in (3.26), we define another convex cone of real-valued polynomials of  $\mathbf{x}$ :

$$\mathcal{B} = \left\{ q \mid q(\mathbf{x}) = \sum_i q_i(\mathbf{x})^2, \text{ with each } q_i(\mathbf{x}) \text{ being a real-valued second-order polynomial of } \mathbf{x} \right\},$$

which is the convex cone of all fourth-order polynomials that can be represented as the sum of squares of some real-valued quadratic polynomials.

Notice that  $\mathcal{B}$  is a subset of  $\mathcal{A}$ . We further define the following optimization problem to compare with (3.28)

$$\begin{aligned} \max_{\alpha} \quad & \alpha \\ \text{subject to} \quad & f(\mathbf{x}) - \alpha \in \mathcal{B}. \end{aligned} \tag{3.29}$$

We show later that (3.29) is a convex semidefinite-programming problem. Here, although (3.29) is not equivalent to (3.28) in general, it is equivalent to (3.28) for the important case of when there is no noise in the system. In such a case, since  $\alpha_{opt} \geq 0$  (due to the non-negativity of the objective in (3.27)) and  $\alpha_{opt} \leq 0$  (due to no noise),  $\alpha_{opt}^A = 0$ . Interestingly,  $\alpha_{opt}^B = 0$  as well because  $\alpha_{opt}^B \leq \alpha_{opt}^A$  (due to (3.29)'s restricted set size when compared to (3.28)) and  $\alpha_{opt}^B \geq 0$  (due to the non-negativity of the objective in (3.27)). Subsequently,  $\alpha_{opt}^B = \alpha_{opt}^A$ . Thus, we can use (3.29) to solve exactly the non-convex problem in (3.26).

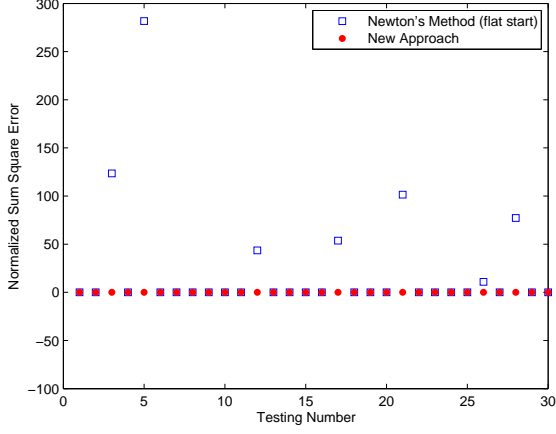
**Proposition 3.5.3.** *If the globally optimal solution of (3.28) is  $\alpha_{opt}^A = 0$ ,  $\alpha_{opt}^B = 0$  is also the global optimal solution from (3.29) [52].*

Proposition 3.5.3 shows that, in the noise-free case when (3.28) achieves the global optimum, (3.29) can be used instead and achieve the same result. Since (3.29) is convex and with a smaller feasible set than (3.28), its global optimum solution, or the state estimate is the same as (3.28), which is equivalent to the WLS problem in (3.26). When the noise-free assumption is violated, (3.29) is still preferable, since it is convex and can be solved with highly efficient software solvers, i.e., the interior point method, and it features convergence properties to the true state as the signal to noise ratio (SNR) grows.

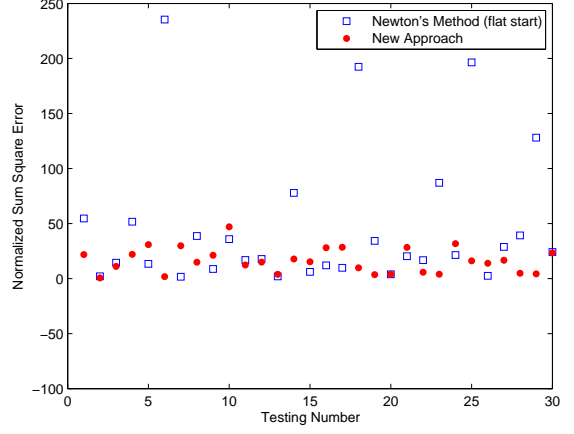
To show that (3.29) can be converted into an equivalent convex semidefinite programming problem, the following lemma [53] is used.

**Lemma 3.5.4.** *Given any fourth-order polynomial  $q(\mathbf{x})$ , the following relation holds:  $q(\mathbf{x}) \in \mathcal{D} \Leftrightarrow q(\mathbf{x}) = \bar{\mathbf{x}}^T G \bar{\mathbf{x}}$  for some Hermitian matrix  $G \succeq \mathbf{0}$ , where  $\bar{\mathbf{x}} = (\mathbf{w}_{(2)}^T, \mathbf{w}_{(0)}^T)^T$  with  $\mathbf{w}_{(2)}$  being a vector whose components are the products  $\{x_i x_j, 1 \leq i \leq j \leq 2n\}$  and  $\mathbf{w}_{(0)} = 1$ .*

For example, if  $\mathbf{x} = [x_1, x_2, x_3]^T$ ,  $\bar{\mathbf{x}} = [x_1^2, x_1 x_2, x_1 x_3, x_2^2, x_2 x_3, x_3^2, 1]^T$ . Further, to obtain the linear constraint for  $G$ , one needs to compare the coefficient in  $q(\mathbf{x})$  and the coefficient in  $\bar{\mathbf{x}}^H G \bar{\mathbf{x}}$ . For instance, if



(a) Noise Free Test Cases



(b) Test Cases with Gaussian Noise

Figure 3.13: IEEE 14-bus test-bed.

we consider the following partial objective  $[\mathbf{x}^T \tilde{\mathbf{Y}}_1 \mathbf{x} - z_1]^2$  without the scaling factor  $\sigma_1$

$$(\mathbf{x}^T \tilde{\mathbf{Y}}_1 \mathbf{x} - z_1)^2 \quad (3.30)$$

$$= \left( \sum_i \sum_j \tilde{Y}_{1,ij} x_i x_j - z_1 \right)^2 \quad (3.31)$$

$$= \sum_i \sum_j \tilde{Y}_{1,ij}^2 x_i^2 x_j^2 + 2 \sum_i \sum_j \sum_k \sum_l \tilde{Y}_{1,ij} \tilde{Y}_{1,kl} x_i x_j x_k x_l + z_1^2 - 2z_1 \sum_i \sum_j \tilde{Y}_{1,ij} x_i x_j, \quad (3.32)$$

then the coefficient  $\tilde{Y}_{1,ij}^2$  should be associated with  $x_i^2 x_j^2$  for matrix  $G$ . Detailed algorithms can be found in [52] and are denoted by linear constraints as  $G(\tilde{\mathbf{Y}})$ , where  $\tilde{\mathbf{Y}}$  represents the set of matrices  $\{\tilde{\mathbf{Y}}_i\}, i \in [1, m]$ .

Subsequently, we obtain the convex semidefinite programming formulation, which has a linear objective function, linear equality constraints over variables  $\alpha$  and  $G$ , and a linear matrix inequality constraint  $G \succeq 0$ .

$$\begin{aligned} & \max_{\alpha} \quad \alpha \\ & \text{subject to } G \text{ satisfying linear equations in } G(\tilde{\mathbf{Y}}), \\ & \quad G \succeq 0. \end{aligned} \quad (3.33)$$

## State Recovery

Note that the solution above is for  $G_{opt}$  and  $\alpha_{opt}$ . In order to obtain physical meaningful states, or the complex voltages, we need to compute  $\bar{\mathbf{x}}_{opt}$ ,  $\mathbf{x}_{opt}$  and ultimately  $\mathbf{v}_{opt}$ . Following Proposition 3.5.3, global

optimum is achieved in the noiseless case with

$$q(\bar{\mathbf{x}}_{opt}) = \bar{\mathbf{x}}_{opt}^H G_{opt} \bar{\mathbf{x}}_{opt} = 0. \quad (3.34)$$

Therefore,  $\bar{\mathbf{x}}_{opt}$  lies in the null space of the positive semidefinite matrix  $G_{opt}$ . If the null space of  $G_{opt}$  from (3.33) has dimension 1,  $\bar{\mathbf{x}}_{opt}$  can be uniquely calculated. In the presence of sensor noise, which may leads to an  $\mathcal{N}(G_{opt})$  with dimension greater than 1, the eigenspace  $\mathcal{N}(G_{opt})$  can be used for the almost zero eigenvalues of  $G_{opt}$ . This is because when noiseless measurement is perturbed with a small noise, a small perturbation in the optimal solution can also been seen.

### 3.5.2 Simulation Results

The simulations are implemented on the IEEE standard 14, 30, and 39 bus test systems. Similar performance improvements are observed. Only the 14-bus simulation results are presented here. The simulation set-up is similar to that in subsection 3.3.6 but with small noise added to the measurements.

#### Numerical Results

The SDP is solved by the “SEDUMI” package [39]. After obtaining the state matrix  $G_{opt}$  in (3.33),  $\bar{\mathbf{x}}$  is obtained from the null space of  $G_{opt}$ , from which voltage estimate  $\hat{\mathbf{v}}$  in the complex domain is extracted.

To display the optimal estimation, Fig.3.13a shows us the noise-free case. The Weighted Residual Sum of Squares (WRSS) error for all measurement types is defined as

$$\text{WRSS} = \sum_{i=1}^m \left( \frac{z_i - \text{tr}(\mathbf{A}_i \mathbf{W})}{\sigma_i} \right)^2. \quad (3.35)$$

The  $x$  axis is the test number of each simulation. The  $y$  axis is for the metric WRSS. The proposed method in red dot achieves global optimum, and constantly reduces the estimated error to zero during all test cases. However, Newton’s method in the blue rectangle with a flat start occasionally converges to the local optimum. This can be seen in test case 3, 5, 12, 17, 21, 26, and 28. One can see the sensitivity of Newton’s method in all these cases, where significant improvement can be realized via the new approach. These facts lead to the natural conclusion that since the new approach converts the WLS into equivalent convex semidefinite programming in the noiseless case, it achieves the global optimum for the WLS problem.

In the presence of Gaussian noise, the significant improvement achieved by the proposed approach is also observed by the WRSS (30 simulations) in Fig.3.13b. Expectedly, since the measurement vector is perturbed, a perturbation of the optimal solution is seen in the red dot line with non-zero WRSS. However, the line is close to zero. For Newton's method with a flat start (the blue rectangle), the WRSS objective occasionally jumps far away from the zero line, indicating a local optimal solution. This can best be seen in test case 7, 17, 25, and 29, which illustrates the converging property of our proposed method in contrast to the local optimal solution.

### 3.6 Chapter Summary

This chapter shows the possibility of achieving better SE accuracy with the WLS formulation. In particular, we introduce a general framework for solving various AC power system estimation problems that have to do with nonconvexity in EMS/SCADA. We first formulate a Semidefinite Programming problem that relaxes the WLS or WLAV estimation problems into rectangular state variable forms. Such a form is motivated by a technique called embedding. Since the estimation problem becomes convex after being relaxed, a global optimal state estimate can be efficiently located with low rank matrix recovery. We demonstrate by simulation that the proposed estimation process can significantly reduce the error therein and achieve a much more accurate estimate than when one applies conventional methods initialized with a flat start. These improvements take place despite the chance of bad data occurrence and regardless of whether the problem is cast as a state estimation problem or a topology estimation problem. Finally, we provide a new confidence metric to highlight the approximate global optimum achieved by our proposed method.

Due to the computational complexity of the proposed methods, we utilize achievements in graph theory and Lagrangian dual method to compute the proposed method in a distributed way. As a result, with message exchanges on coupling nodes between neighboring local networks, the original centralized SDP approach can be characterized in a distributed manner via local SDP SE computation. Obviously, using parallel processors directly reduces the computational time needed for the SE process. We support this conclusion with numerical results that also ensure a fast rate of convergence and robustness to local bad data, both of which are critical for power system analysis.

Finally, we propose for the first time a global optimum solver for the electric power grid state estimation problem in the noise-free case. In this approach, we start by formulating an equivalent convex linear



programming problem, followed by restricted convex semidefinite programming (SDP). A global optimum can be efficiently located in an SDP problem. It is shown that the convex SDP formulation is equivalent to the non-convex WLS formulation in the noise-free case. When noise appears, the proposed method can still be used to obtain an estimate, because a small disturbance in the measurement usually causes a small disturbance in the state. The simulation results show that the proposed approach is capable of preventing local optimum, making it suitable for robust state estimation. In the subsequent chapter, we will discuss various methods for conducting historical data-driven state estimation.

## Chapter 4

# Data-Driven State Estimation

In the last chapter, we proposed convex relaxation-based methods to improve static state estimation (SE) results. Although they are able to achieve global or near-globally optimal results in the measurement domain, further improvement can be accomplished by using historical data to enhance estimation accuracy in the state domain. This historical data-based approach has previously been used for power system analysis. For example, field engineers have used historical data in the past when conducting topology identification by noticing that some groups of circuit breakers were more suspicious than others. As such, when they performed analysis, historical data were taken into account by conducting priority checks on these buses after a topology error was detected. Another example lies in the static state estimation itself. To be more concrete, a previous state estimate, which is a piece of recent historical information, is usually used as a heuristic initial guess for SE in the WLS approach with Newton's method with the assumption that no significant change appears in a short time.

However, a more systematic data-driven approach is needed for smart power grid. This is because, in a smart grid, intermittent generations (i.e. wind and solar farms), consumptions (i.e. plug-in hybrid-electric vehicles), and frequent topological changes can lead to significant state shifts in power system operations. In such cases, a previous state estimate computed around 2 minutes before [54] may not truly reflect the operating point of the current power system and, accordingly that previous estimate generates suboptimal results. Although not suitable for smart grids, utilizing previous state estimate as prior knowledge for current SE does reflect an important idea in power systems analysis: using historical data in a smart way can enhance real time analysis against uncertainties in operation.

Therefore, this chapter aims to utilize valuable historical data resources to improve SE accuracy against the ever-changing and hard-to-predict uncertainties that will exist in future smart power grids. Instead of using a single data point (the last state estimate), the key idea presented in this paper is to use more historical data (i.e. state, topology, and measurement) for the robustness of SE with the embedding techniques introduced in Section 1.4. In Section 4.1, we review the state-of-the-art data-driven approaches currently used in power system analysis. In Section 4.2, we propose static robust data-driven SE approaches. To cure the time-consumption issue of the proposed data-mining approach, we introduce methods in Section 4.3 to speedup the data-driven approach. In Section 4.4, we introduce a static data-driven approach for topology identification to extend the application of the method proposed in Section 4.2; this extension shows an improvement in results when compared with the traditional chi-square test-based topology detection and identification method. Finally, inspired by machine learning, a dynamic data-driven approach, in which the sensors have a very fast sampling rate to capture dynamics, is proposed in Section 4.5. Section 4.6 summarizes the entire chapter.

## **4.1 Current State-of-the-Art Power System Analysis with Historical Data**

Initiated by the U.S. government, the rapidly expanding smart grid aims to evolve into an efficient, reliable and sustainable modern grid by adopting, integrating, and advancing the communication and computing technologies that already exist. To achieve such an ambitious goal, namely the “smartness” of the power grid, a large amount of investment is devoted to sensor purchase and data storage devices. This has resulted in recent advances in communications, sensing, computing and control, as well as in targeted investments toward the deployment of advanced meter infrastructures (AMIs) and synchrophasors, which have become drivers and sources of data previously unavailable in the electric power industry. Such a database is expected to exhibit exponential growth. With vast amounts of data being generated in the power grids, researchers and engineers need to address questions, such as what patterns and trends to extract and how to use them to improve power system reliability, security, sustainability, efficiency and flexibility.

As such, data-driven power system analysis has become popular in recent years. In the following we summarize its various applications.

- **Fault Detection:** [55] presents data-driven fault detection in nuclear power plants to distinguish the effects of sensor degradation from those of actual system faults. [56] and [57] propose data-driven

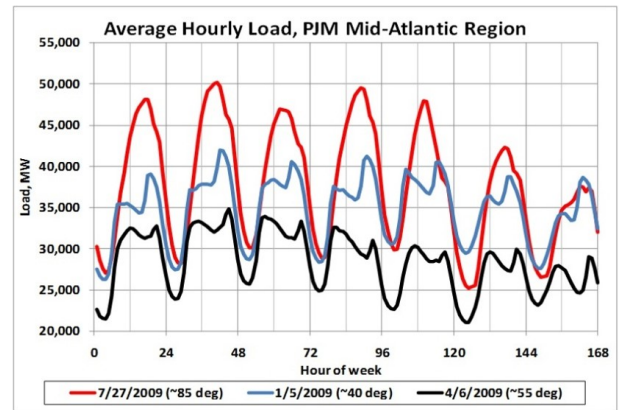
approaches for fault diagnosis in wind energy conversion systems to deal with nonlinearity, unknown disturbances, and significant measurement noise in wind turbines.

- **Model Identification:** [58] presents a practical approach for identifying a global model of a wind turbine from operational data, while the turbine operates in a turbulent wind field with a varying mean wind speed and under closed-loop control.
- **Estimation:** [59] proposes a data-driven multi-scale extended Kalman filtering based on parameter and state estimation approach to the batteries in electric vehicles. [60] presents a domain driven data mining approach for estimating the risk of systems' unavailability based on their component equipments historical data, within one of the biggest Brazilian electric sector companies.
- **Prediction:** [61] discusses short-horizon prediction of wind speed and power using wind turbine data collected at 10s intervals. A time-series model approach to examine wind behavior is studied. Both exponential smoothing and data-driven models are developed for wind prediction. In 2011, a data-driven model is presented in [62] to predict the ice load on transmission lines so that icing disasters can be anticipated. In 2012, [63] proposes a data-driven predictive functional control of power kites for high altitude wind energy generation.
- **Visualization:** [64] proposes a data-driven approach to interactive visualization of power systems.

In the rest of this chapter, we will propose both static and dynamic data-driven approaches to state estimation.



(a) Data center.



(b) Periodic pattern.

Figure 4.1: Power grid.

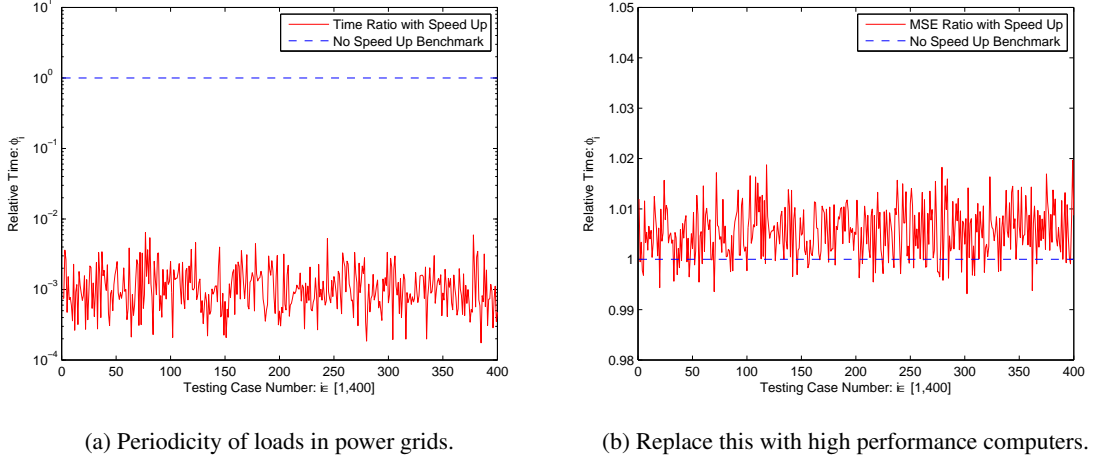


Figure 4.2: Simulation results for speed up

## 4.2 Static Data-Driven Approach to State Estimation

In the static data-driven approach, we aim to find a static state estimate based on the current measurement set and the historical data. This is to be compared to the dynamic data driven approach introduced in Section 4.5. One possible advantage of such an approach for outperform traditional static state estimation method may come from the cyclic tendency of power system operation, with strong daily, weekly and seasonal cycles (e.g., very high loads on weekday afternoons during the summer, and much lower loads on weekend nights in the spring and fall). Hence, similar to an experienced system operator who may readily detect patterns that seem abnormal, a data-based state estimator may be enhanced by looking back to similar patterns in the past.

### Problem Definition

Recall the formal definition in Section 2.1.3 from Chapter 2:

- Problem: Obtain a data-driven state estimator
- Given:
  - a sequence of historical measurement column vectors:  $z_1, z_2, \dots, z_k, \dots, z_Q$ , where  $k$  is the time index, and  $Q$  is the total number of data points in the database;

- a sequence of historical state estimate column vectors:  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k, \dots, \mathbf{x}_Q$ ;
  - a sequence of historical measurement function sets:  $\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_k, \dots, \mathbf{h}_Q$ ;
  - the current measurement column vector:  $\mathbf{z}_{current}$ ;
  - the current measurement function set:  $\mathbf{h}_{current}$ .
- Find:  $\hat{\mathbf{x}}_{current} = g(\mathbf{z})$ , where estimator  $g(\cdot)$  is estimate using historical data. This avoid solving (2.2) directly.

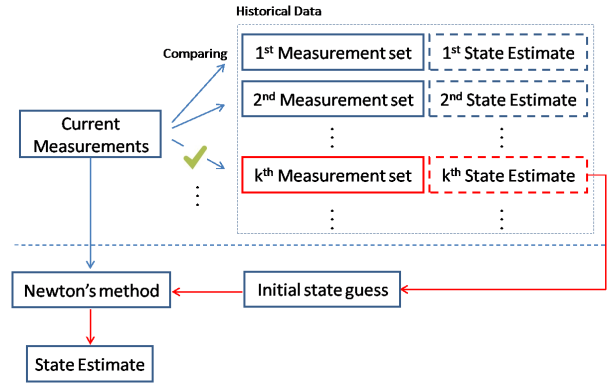


Figure 4.3: Flow chart

Mathematically, our proposed Robust Data-Driven SE algorithm can be decomposed into two parts:

- A minimization problem to obtain a group of likely historical data;
- A kernel ridge regression problem to obtain an “optimal” state estimate from the group.

We will detail the idea behind such a method in the next two subsections.

#### 4.2.1 The Nearest Neighbors Approach

Intuitively, with the same topology, close-by states usually produce similar measurements. Therefore, a smaller distance between the current measurement set  $\mathbf{z}_{current}$  and a historical measurement set  $\mathbf{z}_k$  at time  $k$  implies a strong probability that the associated historical state vector  $\mathbf{x}_k$  stays closer to the current true state vector  $\mathbf{x}_{true}$ . Knowing this, we proposed a historical data search method. To make the estimation unbiased

toward any single data point, a group of nearest neighbors is obtained instead of a single historical data point.

Such a method, as shown in Fig.4.3, is called the K-Nearest Neighbors (K-NNs)<sup>1</sup> approach in Statistics, which is a nonparametric method requiring no model to fit. Specifically, given a query point  $z_{current}$ , we find  $K$  training points ( $z_k$ ) closest in distance to the query point. Despite mild structural assumptions and algorithmic simplicity, K-NNs' predictions are often accurate, leading to its successes in a large number of classification problems, including handwritten digits, satellite image scenes, and EKG patterns.

## 4.2.2 Bayesian Inference: Kernel Ridge Regression

Since the major goal is to conduct inference for the hidden parameters (states) based on the labeled data (historical measurements-state pairs), one employs Discriminative model rather than Generative model, which requires the joint distribution between unobserved variables and observed variables. Such a Discriminative model is only for inference tasks.

### Kernel Ridge Regression

- Ridge regression

We first consider a Normal model below, which is a popular discriminative model with unknown hyperparameters  $\mathbf{q}$  and  $\Sigma_d$ :

$$\mathbf{x}|\mathbf{z} : N(\mathbf{q}^T \mathbf{z}, \Sigma_d). \quad (4.1)$$

To identify such a discriminative model for our inference, a regularized (ridge regression) estimator is commonly used:

$$\hat{\mathbf{q}} = \arg \min_{\mathbf{q}} \sum_{i=1}^K (\mathbf{x}_i - \mathbf{q}^T \mathbf{z}_i)^2 + 2\gamma \|\mathbf{q}\|^2, \quad (4.2)$$

where  $\sum_{i=1}^K (\mathbf{x}_i - \mathbf{q}^T \mathbf{z}_i)^2$  is used to minimize the sum square error, and the regularization term  $2\gamma \|\mathbf{q}\|^2$  is used to improve the estimator performance against ill-conditioned problems.

---

<sup>1</sup>The tuning parameter  $K$  can be chosen by cross-validation [14].

For the Normal model, with the historical data stored in  $\mathbf{Z}_{\text{mat}}$  and  $\mathbf{X}_{\text{mat}}$  as follows,

$$\begin{aligned}\mathbf{Z}_{\text{mat}} &= (\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_K), \\ \mathbf{X}_{\text{mat}}^T &= (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_K),\end{aligned}\tag{4.3}$$

we can obtain a closed-form solution:

$$\hat{\mathbf{q}} = (\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T + 2\gamma \mathbf{I})^{-1} \mathbf{Z}_{\text{mat}} \mathbf{X}_{\text{mat}},\tag{4.4}$$

where the unknown hyper-parameter  $\Sigma_d$  has been absorbed into the penalty constant  $\gamma$ . Notice that due to the ridge regularization (since  $\gamma > 0$ ),  $\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T + 2\gamma \mathbf{I} \succeq 2\gamma \mathbf{I} \succ 0$ . Therefore, the matrix  $\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T + 2\gamma \mathbf{I}$  is always invertible. Thus, the regularized estimator in (4.4) always exists. In other words, with a choice of quadratic penalty, the ridge regression solution is again a linear function of the labels (states) in  $\mathbf{X}_{\text{mat}}$ . The solution adds a positive constant to the diagonal of  $\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T$  before the inversion. This makes the problem nonsingular, even if  $\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T$  is not of full rank, and was the main motivation for ridge regression when it was first introduced in statistics.

Once the hyper-parameter  $\hat{\mathbf{q}}$  is estimated in (4.4), it can be used for Bayesian inference to generate the current state estimate  $\hat{\mathbf{x}}^B$ , as follows:

$$\hat{\mathbf{x}}_{\text{current}}^B = \hat{\mathbf{q}}^T \mathbf{z}_{\text{current}}\tag{4.5}$$

$$= \mathbf{X}_{\text{mat}}^T \mathbf{Z}_{\text{mat}}^T (\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T + 2\gamma \mathbf{I})^{-1} \mathbf{z}_{\text{current}}\tag{4.6}$$

$$= \mathbf{X}_{\text{mat}}^T \mathbf{T}\tag{4.7}$$

Notice that such a form is based on  $\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T$ . One can also conduct the following derivation to obtain an alternative form based on  $\mathbf{Z}_{\text{mat}}^T \mathbf{Z}_{\text{mat}}$ . By employing the Matrix Inversion Lemma  $(A + BDC)^{-1} = A^{-1} - A^{-1}B(D^{-1} + CA^{-1}B)CA^{-1}$  to expand the inversion above, an alternative form of  $\mathbf{T}$  can be obtained that may further simplify the computational need:

$$\mathbf{T} = \mathbf{Z}_{\text{mat}}^T (\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T + 2\gamma \mathbf{I})^{-1} \mathbf{z}_{\text{current}}\tag{4.8}$$

$$= \frac{1}{2\gamma} (\mathbf{Z}_{\text{mat}}^T \mathbf{z}_{\text{current}} - \mathbf{Z}_{\text{mat}}^T \mathbf{Z}_{\text{mat}} (\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T + 2\gamma \mathbf{I})^{-1} \mathbf{Z}_{\text{mat}}^T \mathbf{z}_{\text{current}})\tag{4.9}$$

$$= \frac{1}{2\gamma} (\mathbf{Z}_{\text{mat}} \mathbf{z}_{\text{current}}^T - (\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T + 2\gamma \mathbf{I} - 2\gamma \mathbf{I}) \cdot (\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T + 2\gamma \mathbf{I})^{-1} \mathbf{Z}_{\text{mat}}^T \mathbf{z}_{\text{current}})\tag{4.10}$$

$$= (\mathbf{Z}_{\text{mat}}^T \mathbf{Z}_{\text{mat}} + 2\gamma \mathbf{I})^{-1} \mathbf{Z}_{\text{mat}}^T \mathbf{z}_{\text{current}},\tag{4.11}$$



where

$$\mathbf{Z}_{\text{mat}}^T \mathbf{Z}_{\text{mat}} = (\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n)^T (\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n) = \begin{pmatrix} \mathbf{z}_1^T \mathbf{z}_1 & \dots & \mathbf{z}_1^T \mathbf{z}_n \\ \vdots & \ddots & \vdots \\ \mathbf{z}_n^T \mathbf{z}_1 & \dots & \mathbf{z}_n^T \mathbf{z}_n \end{pmatrix}, \quad (4.12)$$

$$\mathbf{Z}_{\text{mat}}^T \mathbf{z}_{\text{current}} = (\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n)^T \mathbf{z}_{\text{current}} = \begin{pmatrix} \mathbf{z}_1^T \mathbf{z}_{\text{current}} \\ \vdots \\ \mathbf{z}_n^T \mathbf{z}_{\text{current}} \end{pmatrix}. \quad (4.13)$$

Because the matrix  $\mathbf{Z}_{\text{mat}}^T \mathbf{Z}_{\text{mat}}$  appears in the calculation (4.11), as opposed to the original calculation (4.8) involving  $\mathbf{Z}_{\text{mat}} \mathbf{Z}_{\text{mat}}^T$ , the pairwise inner product in  $\mathbf{Z}_{\text{mat}}^T \mathbf{Z}_{\text{mat}}$  creates the potential to improve estimation performance in the nonlinear kernel space. This is because the power system's measurement functions are usually nonlinear. Direct use of (4.1) and (4.5) implicitly assumes a linear model, leading to a relatively poor state estimate.

- The Kernel trick for the Normal Discriminative model

Kernels are important building blocks in high-dimensional learning techniques. There is a trick called kernelization for improving a computationally simple classifier/regressor [14]. The idea is to embed the covariate  $\mathbf{z}_j$  into a higher dimensional space and apply the regression in the embedding space. This can yield a more flexible estimator while retaining computational simplicity. The point is that to get a richer set of regression models we do not need to give up the convenience of linear regression model. We simply map the covariates to a higher-dimensional space. This is akin to making linear regression more flexible by using polynomials.

Now for every so-called kernel  $K(\cdot, \cdot)$  [65], there exists a high-dimensional mapping  $\mathbf{u}_i = w(\mathbf{z}_i)$ , from which the inner product  $\mathbf{u}_i^T \mathbf{u}_j = (w(\mathbf{z}_i))^T w(\mathbf{z}_j)$  can be calculated by the kernel  $K(\cdot, \cdot)$ , as below,

$$\mathbf{u}_i^T \mathbf{u}_j = K(\mathbf{z}_i, \mathbf{z}_j). \quad (4.14)$$

Therefore, the kernel calculation uses only (low-dimensional)  $\mathbf{z}$ 's, rather than the high-dimensional  $\mathbf{u}$ 's. Therefore, the computational complexity of calculating the inner products in (4.12) and (4.13) is low, even

though  $\dim(\mathbf{u})$  itself may be very large. This idea of using a cost-effective kernel calculation to implement a high-dimensional Normal model is called ‘the kernel trick’. In this section, we employ the following kernel forms as candidates. This process is called kernel model assessment and selection.

- Homogeneous polynomial:  $K(\mathbf{u}_i, \mathbf{u}_j) = (\mathbf{u}_i^T \mathbf{u}_j)^d$ .
- Inhomogeneous polynomial:  $K(\mathbf{u}_i, \mathbf{u}_j) = (1 + \mathbf{u}_i^T \mathbf{u}_j)^d$ .
- Gaussian (Radial Basis function):  $K(\mathbf{u}_i, \mathbf{u}_j) = \exp(-\mu \|\mathbf{u}_i^T \mathbf{u}_j\|^2)$ ,  $\mu > 0$ .

### 4.2.3 Model Selection

In order to choose the best model, we need to assess the performance of various models based on different  $\gamma$ s in (4.2) and the kernels above.

If we are in a data-rich situation, the best approach for both problems is to randomly divide the data-set into three parts: a training set, a validation set, and a test set. The training set is used to fit the models; the validation set is used to estimate the prediction error in model selection; and the test set is used to assess the generalization error of the final chosen model. Ideally, the test set should be kept in a “vault”, and be brought out only at the end of the data analysis. In Section 4.3.3, we use the inconsecutive data between validation and testing phases for this purpose.

Therefore, we divide the data into three phases.

- In the training phase, one applies part of the historical data to different kernel functions and  $\gamma$  pairs to calculate different  $T$ s.
- In the validating phase, another part of the historical data is used to choose the best kernel function and  $\gamma$ .
- Finally, the chosen  $\hat{\mathbf{x}}^B$ , computed from the validated  $T$ , is used in (5.2) for the testing phase of state estimation.

### 4.2.4 Robust Data-Driven State Estimation

Although showing promising results in smart grids, the proposed data-driven SE [66] implicitly places strong assumptions on the historical data it uses. For instance, it assumes that the historical data are without: 1)

topology changes [7], 2) bad data [10, 67], and 3) malicious attack [68]. Unfortunately, such assumptions are frequently violated in practice [7, 10].

Specifically, the naive NNs comparison in the measurement space did not include scenarios concerning topology changes, bad data, and malicious attack:

1. The method above is non-robust to topology changes. If topology change occurs, a small distance between measurements may result in an undesired large distance between states.
2. Data points with relatively large errors in the historical measurement may diminish the quality of the collected data points. Different levels of bad data may harm the overall quality of the estimation result.
3. If unobserved malicious bad data injection appears in static SE, it should not be collected in the robust K-NNs due to its false information.

To make the algorithm robust, we generalize the system learning process. Instead of using only one step, we propose three steps to systematically locate good and robust nearest neighbor points. We first utilize historical state and system topology information to deal with topology changes. Second, historical measurements are used to refine the data set against bad data. Third, we conduct a maximum agreement algorithm for the collected states in order to identify malicious attack. Then the resulting information is used in a supervised learning process with Kernel Ridge Regression of Section 4.2.2, leading to a robust data-driven state estimate.

## Topology Changes

To capture the topology changes in the smart grids, historical pseudo-measurements, instead of historical measurements, are used for comparison.

$$\hat{z}_k = \mathbf{h}_{current}(\mathbf{x}_k), \quad (4.15)$$

where  $\mathbf{h}_{current}$  is the measurement function associated with the current topology. Mathematically, the K-NNs results that are robust to the topology changes can be expressed as the following optimization.

$$\hat{\mathbf{s}} = \arg \min_{|\mathbf{s}|=K} d(\mathbf{s}) = \sum_{k \in \mathbf{s}} \|\mathbf{z}_{current} - \hat{z}_k\|_2^2, \quad k \leq Q, \quad (4.16)$$

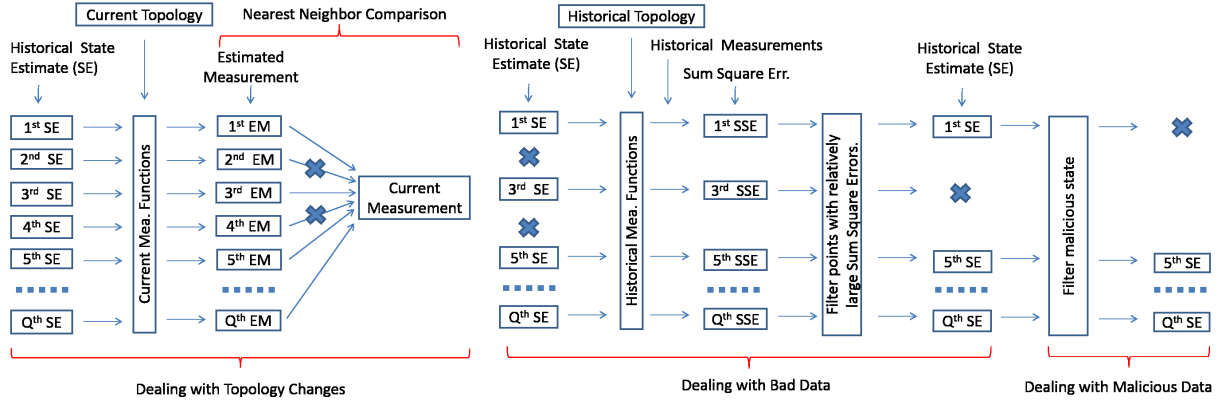


Figure 4.4: Diagram for robust nearest neighbors search

i.e., minimizing the sum distance function  $d(s)$ . Here  $Q$  is the number of total data points in the database.  $k$  represents a particular index for a data point. As a result,  $z_k$  indicates the measurement set within the  $k^{th}$  time slot. Finally,  $K$  indicates the cardinality of the set  $s$ . Essentially, during the searching step, the algorithm looks for an index set  $s$  with  $K$  elements that represents a set of pseudo-measurement vectors that have nearest distance to the current measurement  $z_{current}$ . Such a process is illustrated on the left side of Fig.4.4. Notice that, such a search process is time consuming, due to the high dimensionality of the measurement vectors and a large volume of data points over a long time. Section 4.3 is devoted to this problem.

### Dealing with unfiltered bad data

Bad data occurs due to equipment failure, finite accuracy, infrequent instrument calibration and measurement scaling procedure at the control center. Telecommunication errors and incorrect topology information may also cause bad data. Traditional static bad data detection and filtering are based on the Chi-square test [7]. After choosing a threshold over the probability of error, e.g. 5%, the weighted sum square error is evaluated in Chi-square distribution under the rules of the Chi-square test. Unfortunately, the threshold is usually subjective and can not guarantee the absence of relatively large amounts of bad data. Luckily, when many data points are used in a data-driven framework, one obtains the opportunity to improve the SE result even more by selecting relative better data points. In this subsection, instead of using the hard decision process, we propose comparing the sum square errors in different time slots by utilizing the extra degree of freedom

over time. In this paper, as illustrated in the center of Fig. 4.4, we proposed to remove data points with relatively large sum square errors, i.e. data points with 10% or more will be deleted.

### Dealing with malicious bad data

[68] presents a new class of bad data, namely the false data injection, in the process of state estimation. By exploiting the measurement function (the topology information) of a power system, an attacker can successfully introduce arbitrary errors into certain state variables while bypassing existing techniques for bad measurement detection.

Such an attack, due to its unobservability, is hard for static SE to detect. However, with long historical data, one can compare different data points across the time domain. For this reason, we can compare all the collected states and filter out the outliers. In other words, instead of looking for false data injection among different measurements in a single time slot, we propose to examine the state vectors in multiple time slots to filter out data points that are inconsistent with others according to some metric, such as the definition of relative outliers [69–71]. Fig.4.5 illustrates the idea over a two bus system via synthetic data. The  $x$  coordinate represents the voltage magnitude of the first bus. The  $y$  coordinate represents the voltage magnitude of the second bus. Since power system data are highly clustered, as shown in Section 4.3, points  $o_1$  and  $o_2$  become suspected data points to be filtered out.

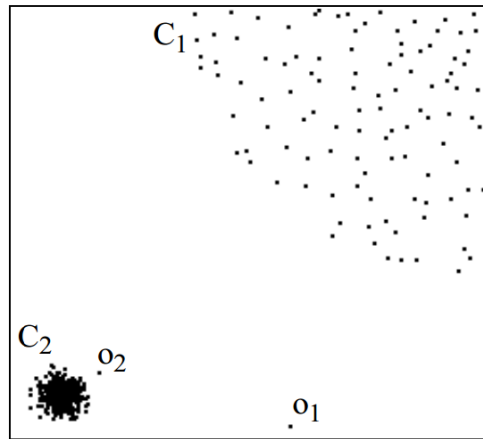
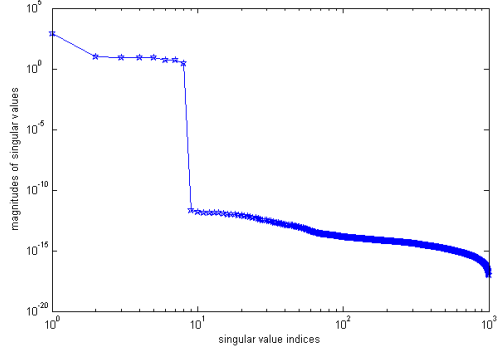
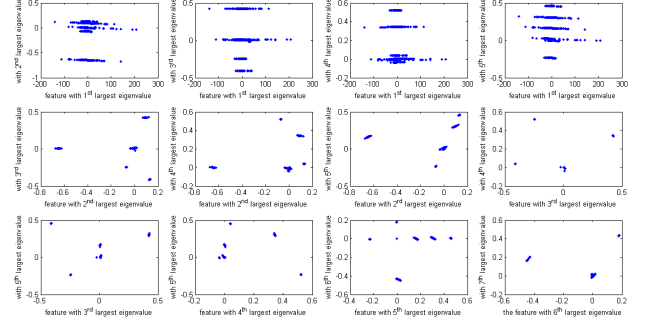


Figure 4.5: Visualization of voltage magnitude pairs in a two bus system

Overall, the robust K-NNs method discussed in this subsection (Fig.4.4) aims to filter out unnecessary



(a) Singular values



(b) Measurement data clustering

Figure 4.6: Dimension reduction

data information in order to shrink the amount of data. Afterwards, we will “average” this historically similar information by conducting the kernel regression problem in subsection 4.2.2. Together, such a two-stage process is called Robust K-Nearest Neighbors Regression [72].

### 4.3 Speed Up for Static Data-Driven State Estimation

In the proposed data-driven approach described in the last section, the nearest neighbors search requires an exhaustive exploration of all data points and is slow in large network with huge amount of historical data. We analyze below the historical data structure of a power system and propose three steps to speed up the data search process but preserve the accuracy.

#### 4.3.1 Dimension Reduction

In order to reduce the NN search time, we start by exploring the data structure. Since electrical power systems exhibit periodicity, one would expect the measurement data to be highly clustered, creating the possibility for a great dimension reduction in measurement data. As an illustration, singular value decomposition (SVD) is conducted over the historical data of 300-bus systems [73]. 1073 measurements per time slot over one year are used to form historical measurement matrix  $Z = [z_1, z_2, \dots, z_Q]$ . Mathematically, the SVD decomposition is represented as

$$Z = U \times S \times V', \quad (4.17)$$

where the diagonal entries of  $S$  are known as the singular values.  $U$  and  $V$  are unitary matrices [73]. By plotting the magnitude of singular values in  $Z$  with a  $\log - \log$  scale, only 8 significant singular values show up in Fig.4.6a. Besides, in Fig.4.6b, we show the result of mapping the historical data onto some two-dimensional features (two left-singular vectors of  $U$ ) associated with significant singular values. Since the data in the figure are highly clustered and far away from each other, dimension reduction is possible with historical electric power system data, and we can remove redundancy from the NN search.

In this section, we propose using random projection for this highly clustered historical measurement data set. This is because other dimension reduction techniques such as SVD tend to be very time consuming, making them unsuited for the online state estimation process [74]. Besides, the dimension reduction of other techniques is typically a one-time operation, which means that the entire process has to be done from scratch every time new power system data come up. This makes them non-adaptive to new incoming data, which is important in real-time power system analysis.

On the contrary, random projection is fast and adaptive. Mathematically, the original  $m$ -dimensional measurement data is simply projected onto a  $m'$ -dimensional ( $m' \ll m$ ) subspace using a random  $m' \times m$  matrix  $R$ ,

$$\mathbf{y}_{m' \times 1} = R_{m' \times m} \mathbf{z}_{m \times 1}. \quad (4.18)$$

The elements  $r_{ij}$  of the random matrix  $R$  are often chosen to be normally independent and identically distributed with zero means, and the columns of  $R$  are normalized with unit lengths.

The key idea of such a random mapping arises from the Johnson-Lindenstrauss Lemma [75]: if points in a vector space are projected onto a randomly selected subspace of suitably high dimensions, then the distance between the points is approximately preserved.

### 4.3.2 $K$ -dimensional ( $k$ -d) Tree for Indexing

Since the historical data is highly clustered, we propose to use a tree to index the data after dimension reduction. The basic idea of the proposed  $k$ -d tree approach comes from the binary-tree as illustrated on the left of Fig.4.7. In the binary-tree, all nodes after the left pointer have smaller values than the current root value, and all nodes after the right pointer have bigger values. If one wants to search for a number in this seven-node tree, the maximum searching time changes from 7 (by exhaustive search) to 3 by using the tree structure.

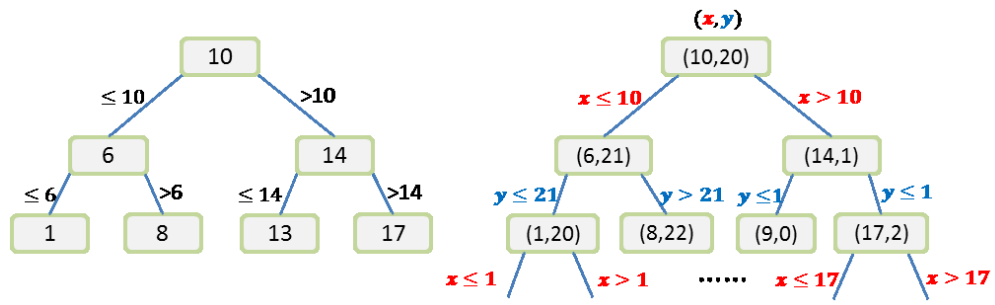


Figure 4.7: Binary tree and  $k$ -d tree

To extend the one dimensional data to high dimensional data such as the power system measurement vectors, we substitute for the binary-tree with a  $k$ -d tree as illustrated on the right of Fig.4.7. The  $k$ -d tree alternates over different measurements as a discriminator on every level of the tree. Because of the efficient ‘pruning’ of the search space, the  $k$ -d tree has an average nearest neighbor search time of  $O(\log(Q))$ , where  $Q$  is the total number of historical data points.

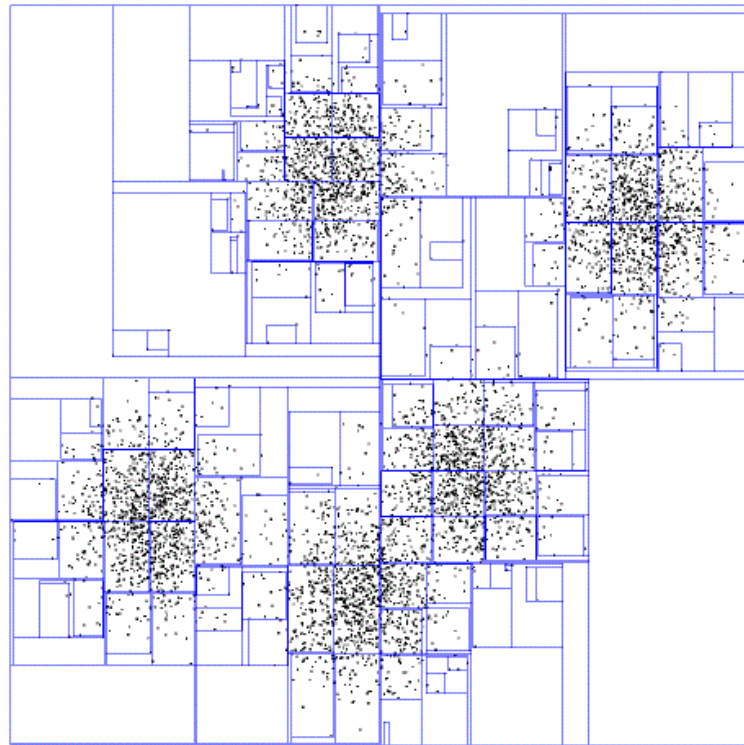


Figure 4.8:  $K$ -d tree for clustered data set



As an illustration, Fig.4.8 shows a case where the  $k$ -d tree is conducted over a two-dimensional data set. By properly using the clustering properties of the spatial data points, the  $k$ -d tree achieves good performance since the search within it can omit a large portion of the clustered points in the space.

### **Adaptation to the Robust K-NNs Approach**

The speed-up method introduced in this section is based on historical measurements. However, our robust K-NNs approach in Section 4.2.4 is based on the pseudo-measurements in (4.15). To adapt the algorithm, we first choose  $10 \times K$  Nearest Neighbors based on the historical measurements. Then we reduce the neighbor numbers via the robust K-NNs approach discussed in Section 4.3.

### **4.3.3 Simulation Results**

In this section, we simulate and verify the performance of the proposed robust K-NNs regression approach, and compare it in a standard IEEE 300 test case to the industrially used Newton's method.

**Data Preparation** The simulations are completed in MATLAB environment in accordance with the MATLAB Power System Simulation Package (MATPOWER) [37, 38]. In addition, to simulate power system behavior in a more practical pattern, the online load profile from New York ISO [76] is adopted in the subsequent simulation. Specifically, it has online load profiles from the New York ISO area recorded every five minutes. The load data used is between February 2005 and December 2013 with a consistent data format. Therefore, we use the load data between February 2005 and May 2013 in training and validation sessions. The load profiles between July 2013 and December 2013 are used in the testing session.

To generate data for SE, we first fit the normalized load data into the case file. Subsequently, an AC power flow is run to generate the true states of the power system, followed by creating true measurement sets with Gaussian noises (standard deviations in Table 4.1). Hereby, we assume that the measurement set includes 1) power injection; 2) line power flows; 3) voltage magnitudes; and 4) some phase angle measurements.

**Data Adjustment** Since we propose a data-driven SE that we claim to be robust to topology changes, bad data, and malicious attack, we adjust the generated data above for simulation with respect to each of them.

Table 4.1: Standard deviation of measurement noise

Measurement type	Standard deviation
Active (Reactive) power injection	0.015
Active (Reactive) power flow (from)(to)	0.02
Voltage magnitude	0.01
Phase angle	0.002

- **Topology Changes:** In this case, before running the power flow to generate the measurements, several randomly chosen topology connections are changed with a 20% probability of imitating certain features of the smart grid.
- **Bad Data:** In addition to the topology changes, we randomly generate bad data and insert them into the measurements.
- **Malicious Attack:** In addition to topology changes and bad data, we intentionally inject several malicious data into the pre-May 2013 historical database, which is unobservable by the Chi-square test [68].

### Training, Validation, and Testing

- **Training Phase:** By randomly selecting one measurement between July 2013 and December 2013 as a test case, a group of robust nearest neighbor measurements in (4.16) between Feb. 2005 and Dec. 2012 is selected via Fig.4.4 in Sec.4.2.1. Then matrix  $T$  in (4.11) is computed for different choices of  $\gamma$  and kernel function.
- **Validation Phase:** Matrix  $T$  is validated for the data between January 2013 and May 2013 to obtain the best choices for  $\gamma$  and the kernel pair.
- **Testing Phase:** We use the matrix  $T$  chosen in the validating phase to calculate the Bayesian state estimate  $x^B$  via (4.7) for the data between July 2013 and December 2013. For comparison purposes, the industrial approach of Newton's method initialized via the last state estimate is also applied to the same testing data.

**Improved Accuracy** In the testing phase, filtering out bad data and malicious data reduces the measurement number. Therefore, instead of Sum Square Error, we will employ Mean Squared Error (MSE) defined as

$$\text{MSE} = \frac{1}{m} \sum_{i=1}^m \left( \frac{z_i - h_i(\hat{x}_{\text{current}})}{\sigma_i} \right)^2. \quad (4.19)$$

### Simulation: Robustness to Topology Changes

In this part, we assume there is no bad data and malicious attack in the historical data to emphasize the robustness feature of the proposed method. We employ the first building block only (the left block) of (Fig.4.4) to conduct the R-KNNs search. Then, the gathered data is outputted to the kernel ridge regression method discussed in Section 4.2.2 for an estimate. The Training, Validation and Testing procedures, discussed in Section 4.4.4, are subsequently used in this estimation process. For fairness, the testing is conducted 400 times. Each time, we obtain an MSE, voltage magnitude estimates, and voltage phase angle estimates. For comparison purposes, we also compute the corresponding results with Newton's method initialized by a previous state estimate.

To show the improved accuracy over the 400 testing cases, we define the relative error in the  $i^{\text{th}}$  testing case as

$$\gamma_i = \frac{\text{MSE}_{\text{RobustK-NearestNeighborsMethod}}}{\text{MSE}_{\text{Newton'sMethodwithPre.Est.Start}}}. \quad (4.20)$$

Subsequently, Fig.4.9a shows a histogram of the 400 simulation results. By looking at the  $x$  coordinate, we observe that the Robust K-Nearest Neighbors (R-KNNs) approach has greatly reduced estimation errors by an average ratio of  $10^{-2.5}$ . From this fact, we can reasonably conclude that the proposed R-KNNs method is able to handle topology changes.

As for the comparison in the MSE domain, Fig.4.9b and Fig.4.9c provide state domain plots. The  $x$  coordinate represents the bus number. The  $y$  coordinate represents the voltage magnitude ratio ( $\frac{|V_{R-KNNs}|}{|V_{True}|}$  and  $\frac{|V_{Pre.Est.}|}{|V_{True}|}$ ) in Fig.4.9b and the voltage phase angle ratio ( $\frac{\angle V_{R-KNNs}}{\angle V_{True}}$  and  $\frac{\angle V_{Pre.Est.}}{\angle V_{True}}$ ) in Fig.4.9c, respectively. It can be observed that the R-KNNs method has a voltage ratio in red of close to 1, and its lesser variance (in red) indicates its ability to track the true system states while Newton's method with a previous estimate start has a ratio (in blue) far away from 1 with a large variance. Such a poor result is caused by the local-search behavior of Newton's method, which is suboptimal with an inferior initial guess.

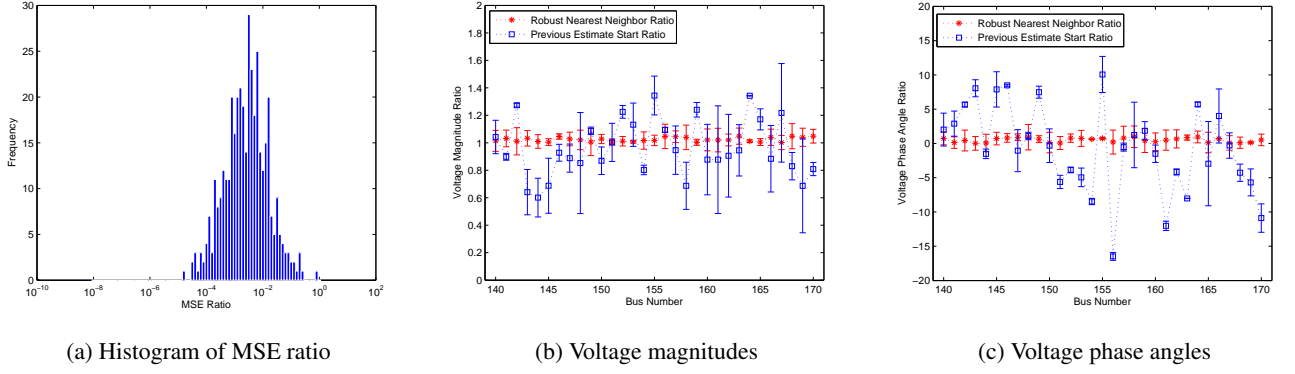


Figure 4.9: Simulation results with only the first block of Fig.4.4.

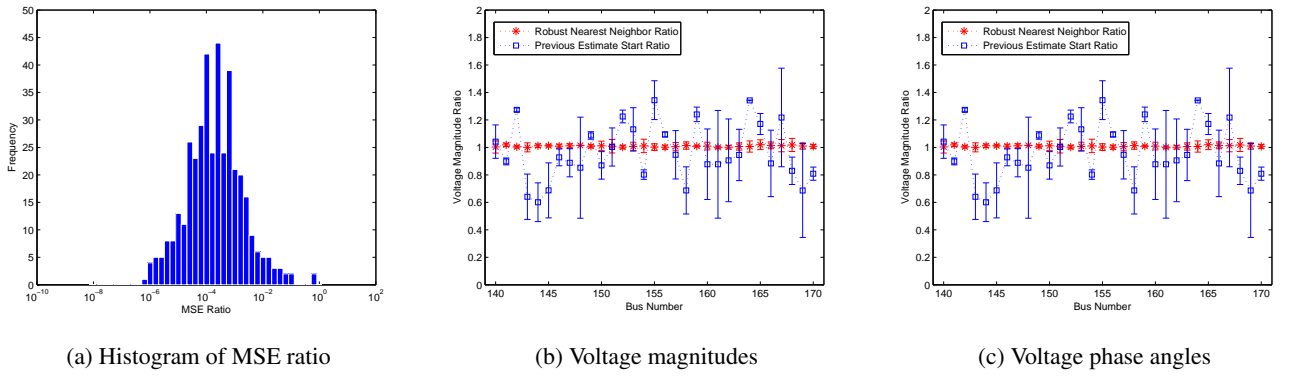


Figure 4.10: Simulation results with the first and the second blocks of Fig.4.4

### Simulation: Robustness to Bad Data

In this section, we conduct similar simulations to those in the last subsection. However, besides using the first building block (the left block of Fig.4.4, we will use the second (middle) building block as well. This block targets at improving the result, by filtering out data points with relatively large residuals and by creating a soft decision rule as discussed in Section 4.2.4.

For comparison purposes, we simulate this part in parallel to simulations in the last subsection, namely the robustness to topology changes. Therefore, we observe the same estimation mean and variance in blue for Newton's method initialized by the previous state estimate. As one can observe, the only difference between Fig.4.10b and Fig.4.9b are that the mean of R-KNNs ratio in Fig.4.10b is flatter and its variance is smaller than the ratio in Fig.4.9b. This is caused by the filtering out of measurements with relatively large residuals, thanks to richer data comparison across the horizontal time line, instead of the single data point

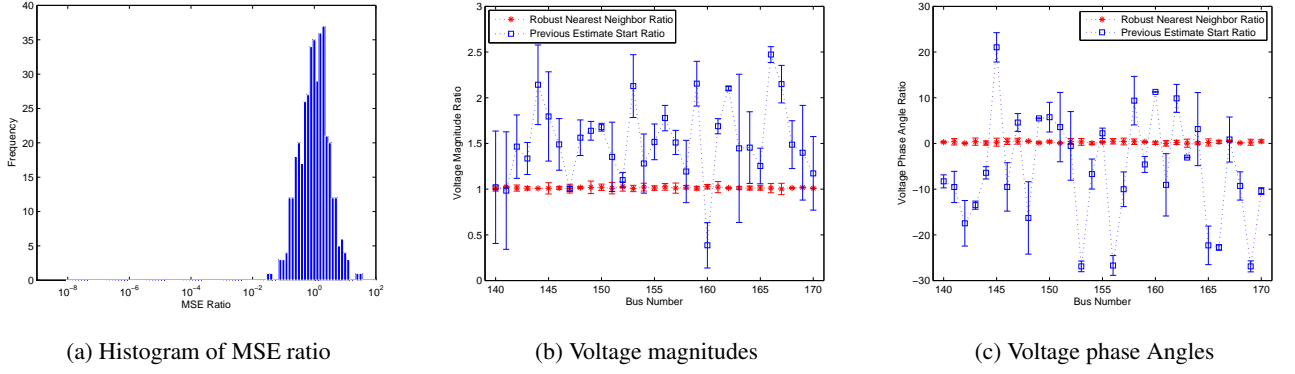


Figure 4.11: Simulation results with all three blocks of Fig.4.4

used in traditional static SE. Another piece of evidence supporting to this observation is that Fig.4.10a shows a smaller MSE ratio than does Fig.4.9a.

### Simulation: Robustness to Malicious Attack

The data used in this subsection are slightly different from the two simulations above. To mimic the possibility that at some point in time human beings might try to inject bad data that could pass the Chi-square test, we injected malicious data intentionally into the measurement set only in the testing sets. From Fig.4.11a, we can see that the Mean Square Error before and after the malicious-data-filtering are not so different. However, we can also observe large errors in the state space in Fig.4.11b and Fig.4.11c. For example, the rectangular plot in blue shows a long distance to 1 in Fig.4.11b. We can see a similar thing in Fig.4.11c as well. This shows that the malicious data can change the state estimate dramatically without triggering the bad data alarm.

In contrast, the red star-like curve is close to one in the voltage magnitude plotting and close to zero in the voltage phase angle plotting. The variance in red is also much smaller than the variance in blue, illustrating the robustness of the proposed data drive approach against malicious data.

### Speed Up

In this section the speedup method proposed in Section 4.3 is used. The comparison process is conducted 400 times with and without speedup.

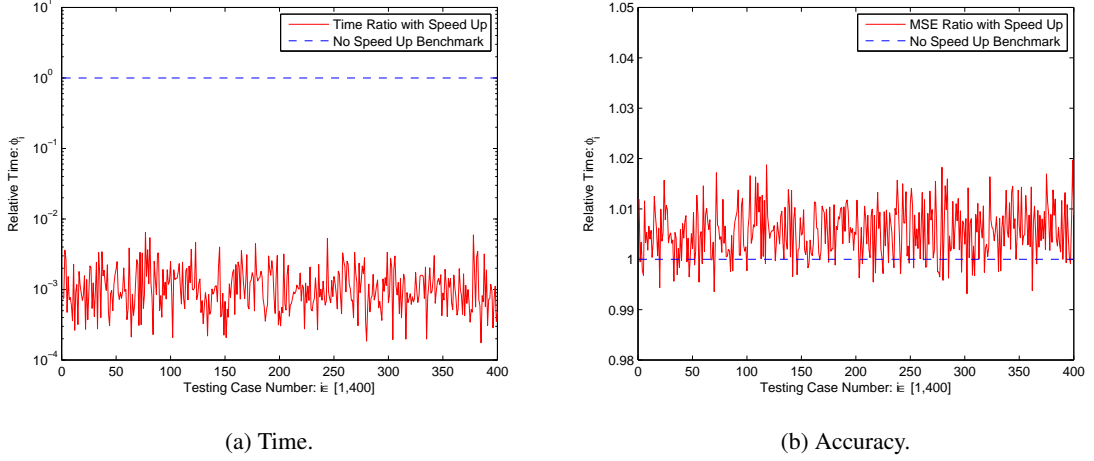


Figure 4.12: Simulation results for the speedup

**Time Reduction** To illustrate the speedup, we define the relative execution time  $\phi$  for the  $i^{th}$  testing data during the R-KNNs search:

$$\phi_i = \frac{t_{wp,i}}{t_{ex,i}}, \quad (4.21)$$

where  $t_{wp}$  represents the R-KNNs search time with preprocessing (dimension reduction and  $k$ -d tree indexing).  $t_{ex}$  represents the exhausted R-KNNs search time without preprocessing.

The plot of  $\phi_i$  is drawn in Fig.4.12a. It shows that the proposed preprocessing approach has greatly reduced the historical NN search time. For example, the average ratio ( $\phi$ ) for all testing cases is around  $10^{-3}$ , creating a 99.9% reduction in search time. Such a result leads to a rational interpretation for the proposed procedure: since the historical data is organized in a compact way, the nearest neighbors search can be conducted much more efficiently.

**Accuracy** Similar to the ratio metric defined in the last subsection, we define the  $i^{th}$  relative error  $\pi_i$  for the testing data as

$$\pi_i = \frac{MSE_{wp,i}}{MSE_{ex,i}}. \quad (4.22)$$

The plot of  $\pi_i$  shows that the relative error is only slightly larger than 1 (less than 0.01 on average), so the proposed two-step approach returns a group of historical data highly similar to the groups in the exhaustive search. The slightly larger error ( $\approx 1\%$ ) may be caused by dimension reduction. But it is negligible when

compared to the 1000 time speedup. Therefore the proposed approach achieves approximately the same accuracy as the slow but highly accurate data-driven SE as shown in Fig.4.9, Fig.4.10, and Fig.4.11.

## Conclusions

In this section, we discussed how to systematically obtain a robust data-driven state estimation for AC power systems. Based on the intuition that similar measurements and topology reflect similar power system states, we formulated the finding of the initial SE as a minimum distance search problem. Further, a Bayesian estimate was obtained via kernel ridge regression. We further proposed how to systematically reduce the computational costs of the proposed method. Such a computational reduction was based on the observation that power system consumption displays periodic pattern. In particular, dimension reduction and efficient indexing over trees were proposed. The numerical results showed that the proposed method can achieve a highly accurate robust data-driven state estimate in a short time with a 1000 time speedup over IEEE benchmark systems.

## 4.4 Data-Driven Approach for Topology Identification

Power system state estimation is usually formulated based on bus-branch network modeling. The goal is to compute nodal voltage magnitudes and phase angles based on various measurements and the grid topology. Unfortunately, topology error happens occasionally when an undetected breaker status change appears in a line, a transformer, a shunt capacitor, or a bus coupler [77]. As a result, systematic topology error identification methods have been developed based on a post-SE procedure [78]. Specifically, a topology error is detected if measurements associated with a branch or a bus are flagged as outliers by an SE-based residual test. Since the state estimation and topology estimation processes are based on the correctness of each other, it is easy to find cases where multiple conforming measurements cause incorrect topology identification results [7,40,41].

In the traditional transmission network, it is possible to use the topology identification process described above, based on the belief that no significant topology change appears in a short time with limited non-conforming errors. However, such a belief doesn't hold with a smart grid, where frequent topological changes appear, which may lead to a large amount of topology errors, making it hard on the method above to identify

them all.

Similar to the last section, where historical data is used for SE, this section aims to utilize valuable historical data resources to improve topology identification accuracy against the ever-changing hard-to-predict uncertainties in smart grids. Instead of using only a single data point (the current time), the key in this section is to use more historical data (i.e. topology and measurement) for robustness. The proposed method is based on the idea that two similar system measurement sets usually indicate two similar topology configurations. After collecting a group of similar measurements in the past, a supervised learning framework is employed to map the historical data to the current topology estimate. Specifically, we propose a Bayesian approach based on a historical data search, where a group of measurement sets and the corresponding topology information are used in combination with the current measurement in a logistic kernel regression framework [14] to pursue a good estimate of the current topology.

The performance of the data-driven method is verified by simulations on standard IEEE 14, 30, and 39-bus test cases [37, 38]. Provided with enough historical data, the data-driven topology estimate outperforms the estimate from the traditional approach.

The rest of the section is organized as follows: Subsection 4.4.1 reviews the current topology identification approach; in Subsection 4.4.2, we pose a data-driven topology estimation problem; Subsection 4.4.3 describes the nearest neighbors approach; Subsection 4.4.4 illustrates the simulation results and subsection 4.4.5 concludes the section.

#### **4.4.1 Current State-of-the-Art Power System Topology Error Identification**

Topology estimation deals with the occasionally unreported removal of customer nodes, sensor failures, and manually updated breakers. Without accurate topological estimation, SCADA systems will have the wrong topology for subsequent state estimation, resulting in erroneous consumer power cut-offs or even a blackout throughout the grid.

The well-known method [7, 40, 41] for topology estimation has two steps: the first step uses a bus branch model for AC power systems. Subsequently, Weighted Least Squares and the Chi-square test [42] are used to locate the branches suspected of abnormal measurements.

Recall that the state estimation in static WLS form is highly non-convex. Due to the non-convexity of the cost function in (2.2), currently used Newton's method is highly sensitive to the initial guess. An undesirable



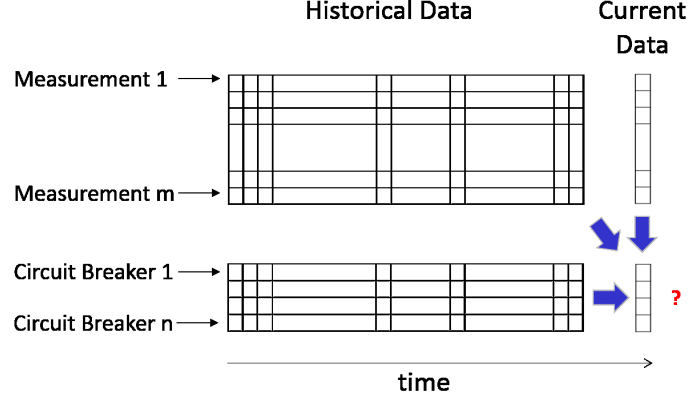


Figure 4.13: Problem definition.

local optimum may appear.

After SE, a topology error is identified if measurements associated with a branch or a bus are flagged as outliers by an SE-based residual test. Such a method is problematic due to (1) the interdependence between SE and Topology Estimation, and (2) the local optimum issue with Newton's method. In the next two sections, we are going to discuss a new systematic approach for topology estimation.

#### 4.4.2 Problem Definition

In this section, we aim to perform historical data-driven topology estimation for power systems in a manner similar to how we have performed data-driven state estimation. The major difference is that, instead of continuous state estimation, we will use discrete topology estimation. Specifically, we consider a sequence of historical SCADA measurements obtained every 2 seconds (or shorter, when faster devices such as Phasor Measurement Units are available). These measurements are represented as a discrete time sequence  $\{z_1, z_2, \dots, z_k, \dots, z_Q\}$ , where  $k \in \{1, 2, \dots, Q\}$ . For instance, each column of Fig.4.13 represents a measurement set at a particular time slot. In addition, the corresponding circuit breaker statuses are represented as  $\{s_1, s_2, \dots, s_k, \dots, s_Q\}$ , where  $s_i$  is a binary vector representing the topology connections at time slot  $i$ . For each element of  $s_i$ , 1 indicates connected and 0 otherwise. The problem that we are trying to solve here is to use historical data to identify the current topology. The formal problem definition, as shown in Fig.4.13, is as follows:

- Problem Name: Topology Estimation

- Given:
  - a sequence of historical measurement vectors  $\{z_1, z_2, \dots, z_k, \dots, z_Q\}$ ;
  - a sequence of historical topology configurations  $\{s_1, s_2, \dots, s_k, \dots, s_Q\}$ ;
  - current measurements  $z_{current}$ .
- Find: the current topology  $s_{current}$ .

### 4.4.3 The Nearest Neighbors Approach

Mathematically, the proposed data-driven topology estimation algorithm can be decomposed into two parts:

- A minimization problem to obtain a group of historically similar data;
- A Logistic Kernel Regression problem to obtain an “optimal” topology estimate from the group.

Intuitively, similar topologies usually produce similar measurements. Therefore, a smaller distance between the current measurement set  $z_{current}$  and a historical measurement set  $z_k$  at time  $k$  implies that the associated historical topology vector  $s_k$  has a high probability of staying closer to the current topology vector  $s_{current}$ . Provided with the aforesaid, we propose a historical data search method. To free the estimation from relying on a single data point, a group of nearest neighbors is obtained instead of a single historical data point. This is similar to 4.3.

### 4.4.4 Bayesian Inference: Logistic Kernel Regression

To combine prior beliefs with data in a principled way, we propose conducting Bayesian inference over the collected K-Nearest Neighbor points to obtain a data-driven topology estimate. Since the major goal is to conduct robust inference for the hidden parameters (topology) based on the labeled data (the historical measurements-topology pairs), one can invert the causality relation for a Discriminative model. Recall that, after obtaining the minimum distance index set  $u$ , this subsection aims to use the associated measurements and their corresponding topology, as well as the current measurement set, to obtain a good current topology estimate. In the following we start with a Generative model, namely Naive Bayes, to obtain the preferred Discriminative model, namely Logistic Regression.

We define the mapping from the measurements to a topology variable  $s^{(l)}$  as

$$f(\mathbf{z}) = P(s^{(l)} = 1 | \mathbf{z} = \{z^{(1)}, z^{(2)}, \dots, z^{(k)}, \dots, z^{(K)}\}),$$

where  $s^{(l)}$  represents the status of the  $l^{th}$  circuit breaker, i.e.  $s^{(l)} = 1$  if the  $l^{th}$  circuit breaker is close.

To learn  $P(s^{(l)} | \mathbf{z})$ , Naive Bayes uses the training data to estimate  $P(\mathbf{z} | s^{(l)})$  and  $P(s^{(l)})$  separately. With these estimates, the Bayes rule can then be applied to determine  $P(s^{(l)} | \mathbf{z})$  for any new instance, such as  $\mathbf{z}_{current}$  [79].

$$\begin{aligned} P(s^{(l)} = 1 | \mathbf{z}) &= \frac{P(s^{(l)} = 1)P(\mathbf{z} | s^{(l)} = 1)}{P(s^{(l)} = 1)P(\mathbf{z} | s^{(l)} = 1) + P(s^{(l)} = 0)P(\mathbf{z} | s^{(l)} = 0)} \\ &= \frac{1}{1 + \exp(\ln \frac{P(s^{(l)}=0)P(\mathbf{z}|s^{(l)}=0)}{P(s^{(l)}=1)P(\mathbf{z}|s^{(l)}=1)})} \\ &= \frac{1}{1 + \exp((\ln \frac{1-\pi}{\pi}) + \sum_i \ln \frac{P(z^{(i)} | s^{(l)}=0)}{P(z^{(i)} | s^{(l)}=1)})}, \end{aligned} \quad (4.23)$$

where  $P(s^{(l)})$  is assumed to be Bernoulli ( $\pi$ ), and the Gaussian Naive Model assumes that all  $z^{(i)}$  are conditionally independent given  $s^{(l)}$ . As we will see later, this assumption is removed in our learning phase when Logistic Regression is applied. Now, we focus on the second term in the denominator of (4.23):

$$\sum_{i=1}^m \ln \frac{P(z^{(i)} | s^{(l)} = 0)}{P(z^{(i)} | s^{(l)} = 1)} = \sum_{i=1}^m \ln \frac{\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp(\frac{-(z^{(i)} - \mu_{i0})^2}{2\sigma_i^2})}{\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp(\frac{-(z^{(i)} - \mu_{i1})^2}{2\sigma_i^2})} = \sum_{i=1}^m (\frac{\mu_{i0} - \mu_{i1}}{\sigma_i^2} z^{(i)} + \frac{(\mu_{i1}^2 - \mu_{i0}^2)}{2\sigma_i^2}).$$

Note that this expression is a linear weighted sum of  $z^{(i)}$ s. Substituting it into (4.23), we obtain

$$P(s^{(l)} = 1 | \mathbf{z}) = \frac{1}{1 + \exp((\ln \frac{1-\pi}{\pi}) + \sum_i (\frac{\mu_{i0} - \mu_{i1}}{\sigma_i^2} z^{(i)} + \frac{(\mu_{i1}^2 - \mu_{i0}^2)}{2\sigma_i^2}))} = \frac{1}{1 + \exp(w_0 + \sum_{i=1}^n w_i z^{(i)})} \quad (4.24)$$

where  $P(z^{(i)} | s^{(l)})$  is assumed to be Gaussian  $N(\mu_{il}, \sigma_i)$ . The weights  $w_1, \dots, w_n$  are given by

$$w_i = \frac{\mu_{i0} - \mu_{i1}}{\sigma_i^2}, \quad i \in \{1, 2, \dots, m\}, \quad (4.25)$$

$$w_0 = \ln \frac{1 - \pi}{\pi} + \sum_{i=1}^m \frac{\mu_{i1}^2 - \mu_{i0}^2}{2\sigma_i^2}. \quad (4.26)$$

As  $P(s^{(l)} = 0|\mathbf{z}) = 1 - P(s^{(l)} = 1|\mathbf{z})$ , we have

$$P(s^{(l)} = 0|\mathbf{z}) = \frac{\exp(w_0 + \sum_{i=1}^n w_i z^{(i)})}{1 + \exp(w_0 + \sum_{i=1}^n w_i z^{(i)})}. \quad (4.27)$$

As for our inference, the distribution of  $s^{(l)}$  is not needed, we prefer to learn  $P(s^{(l)}|\mathbf{z})$ , or  $\mathbf{w}$  directly from (4.24) and (4.27). There are also other motivations for this preference. First, the form of  $P(s^{(l)}|\mathbf{z})$  used by Logistic Regression holds under a broad variety of settings beyond the Gaussian Naive Bayes problem. Second, we may wish to estimate the  $w_i$  parameters directly from the data, rather than going through the intermediate step of estimating the Gaussian Naive Bayes (GNB) parameters that forces us to adopt its more stringent modeling assumptions [79]. For example, GNB assumes conditional independence of the measurements given one topology connectivity, which does not hold for power systems.

To train Logistic Regression, we choose to maximize the conditional data likelihood below:

$$\hat{\mathbf{w}} = \arg \max_{\mathbf{w}} \prod_{k \in \{\hat{\mathbf{u}}\}} P(s^{(l)}|\mathbf{z}_k, \mathbf{w}), \quad (4.28)$$

where  $\mathbf{w} = \{w_0, w_1, \dots, w_n\}$ . Taking the log, we obtain the conditional data log-likelihood function

$$l(\mathbf{w}) = \sum_{k \in \{\hat{\mathbf{u}}\}} \left[ s_k^{(l)} \ln P(s_k^{(l)} = 1|\mathbf{z}_k, \mathbf{w}) + (1 - s_k^{(l)}) \ln P(s_k^{(l)} = 0|\mathbf{z}_k, \mathbf{w}) \right]. \quad (4.29)$$

Note that, since  $s_k^{(l)}$  is binary, only one term appears. Plug in (4.24) and (4.27), and we have

$$l(\mathbf{w}) = \sum_{k=1}^K \left[ s_k^{(l)} (w_0 + \sum_{i=1}^n w_i z_i^{(k)}) - \ln (1 + \exp(w_0 + \sum_{i=1}^n w_i z_i^{(k)})) \right]. \quad (4.30)$$

Subsequently, the gradient ascent method can be used to estimate an optimal  $\hat{\mathbf{w}}$ .

## Kernel Logistic Regression

From the above, we learn the Discriminative model of one circuit breaker status by using historical data. Such a form is easily expanded to learn all circuit breaker statuses. Further, to deal with the nonlinearity inherent in electric power systems (5.1), we employ Logistic Kernel Regression instead of Logistic Regression.

In Kernel Logistic Regression, we want to minimize [80]:

$$H = - \sum_{k \in \{\hat{u}\}} \left[ s^l f(z_k) - \ln(1 + \exp(f(x_i))) \right] + \frac{\lambda}{2} \|f\|_{H_k}^2, \quad (4.31)$$

where  $H_k$  is the reproducing kernel Hilbert space and  $\lambda$  is the penalization parameters. The mapping  $f(z) = w_0 + \sum_{i=1}^m w_i K(z, z_i)$ . With some derivation, it can be shown that this is equivalent to the finite dimensional form:

$$H = -z(K_w \mathbf{w}) + \mathbf{1}^T \ln(1 + \exp(K_w \mathbf{w})) + \frac{\lambda}{2} \mathbf{w}^T K_c \mathbf{w}, \quad (4.32)$$

where the regressor matrix  $K_w = [K(z_i, z_j)]_{N \times N}$ ; and the regularization matrix  $K_c = K_w$ . [80] shows that the Newton-Raphson step at step  $g$  is a weighted least squares step:

$$\mathbf{w}^{((g))} = (K_\alpha^T A K_\alpha + \lambda K_q)^{-1} K_\alpha^T A (K_w \mathbf{w}^{(k-1)} + A^{-1}(\mathbf{s} - \mathbf{p})), \quad (4.33)$$

where  $\mathbf{w}^{((g))}$  is the value of  $\mathbf{w}$  in the  $q^{th}$  step and  $\mathbf{p} = p(\mathbf{z})$ . The weight matrix is  $A = \text{diag}[p(z_i)(1 - p(z_i))]_{N \times N}$ .

Notice that  $K_w = [K(z_i, z_j)]_{N \times N}$  creates the potential to reduce computation costs.

## The Kernel Trick for the Normal Discriminative Model

Recall that kernels are important building blocks for high-dimensional learning techniques. There is a trick called kernelization for improving a computationally simple classifier/regressor [14]. The idea is to embed the covariate  $z_j$  into a higher dimensional space and apply regression in the embedding space. This can yield a more flexible estimator while retaining computational simplicity. The point is that to get a richer set

of regression models, we do not need to give up the convenience of a relatively easier regression model. We simply map the covariates to a higher-dimensional space.

## Model Selection

In order to choose the best model, we need to assess the performance of various models based on different  $\lambda$ s in (4.31) and the kernels above.

Therefore, we divide the data into three phases.

- In the training phase, one applies part of the historical data to different kernel functions and  $\lambda$  pairs to calculate different  $w$ s.
- In the validating phase, another part of the historical data is used to choose the best kernel function and  $\lambda$ .
- Finally, the chosen  $\hat{s}^B$ , computed from the validated  $\lambda$  and the chosen kernel function, is used for the testing phase with topology estimation.

### 4.4.5 Simulation Results

In this section, we simulate and verify the performance of the nearest neighbors (NN) approach, and compare it to the traditional SE-based approach in standard IEEE 14, 30 and 39-bus test cases for topology estimation. We find that the two approaches perform similarly. The 30-bus result is shown.

#### Data Setup

The simulations are completed in a MATLAB environment in accordance with the MATLAB Power System Simulation Package (MATPOWER) [37,38]. Further, to simulate power system behavior in a more practical way, the online load profile from the New York ISO [76] is adopted in the simulation. Specifically, it has online load profiles from the New York ISO area recorded every five minutes. The load data used is between February 2005 and December 2013 with a consistent data format. Therefore, we use the load data between February 2005 and May 2013 in training and validation sessions. The load profiles between July 2013 and December 2013 are used in the testing session.

To generate the data for topology estimation, we first fit the normalized load data into the case file. Then we fix an integer number  $\beta \in [1, 10]$  and randomly disconnect  $\beta$  branches in the case file. Subsequently, an AC power flow is run to generate the true states of the power system, and then we create true measurement sets with Gaussian noises (the standard deviations in Table 4.1). We assume that each measurement set includes 1) power injection; 2) line power flows; 3) voltage magnitudes; and 4) some phase angle measurements.

### Training, Validation, and Testing

- **Training Phase:** By randomly selecting one data point between July 2013 and December 2013 as a test case, a group of robust nearest neighbor measurements in (??) between February 2005 and December 2012 is selected via Fig.?? in Section 4.4.3. Then  $w$  is computed for different choices of  $\lambda$  and kernel functions.
- **Validation Phase:**  $w$  is validated on the data between January 2013 and May 2013 to obtain the best choice for  $\lambda$  and the kernel pair.
- **Testing Phase:** We use  $w$  chosen in the validating phase to calculate a Bayesian topology estimate  $s^B$  in the data between July 2013 and December 2013. For comparison purposes, the industrial approach of the SE-based residual test is also applied to the same testing data.

### Robustness to Topology Changes

We use different  $\beta$  to test the performance of the proposed approach. For comparison, the traditional approach discussed in Section 2.2 is simulated as well. Fig.4.15 shows the averaged successful probability of the two approaches in the IEEE 30 bus system in Fig.4.14. We simulated up to 10 topology errors. In particular, for each value of  $\beta$ , we generate historical data with measurements and topologies. When testing a data point, these historical data are used together with the measurements of the testing data point. The goal is to identify the topology of the testing data point via the data-driven approach. In contrast, the traditional approach based on post-state estimation and residual test is run based on the original case file. The algorithm will then iteratively detect topology error and try to correct it according to Section 4.4.1 or Chapter 8 in [7]. After all the testing cases are simulated for the fixed  $\beta$ , we average the success probability of finding the correct “current” topology in both approaches. Then we change the value of  $\beta$ , and redo the process above.

The simulation results for the IEEE 30 bus systems are displayed in Fig.4.15. The  $x$  coordinate represents the topology error numbers, and the  $y$  coordinate represents the averaged success probability.

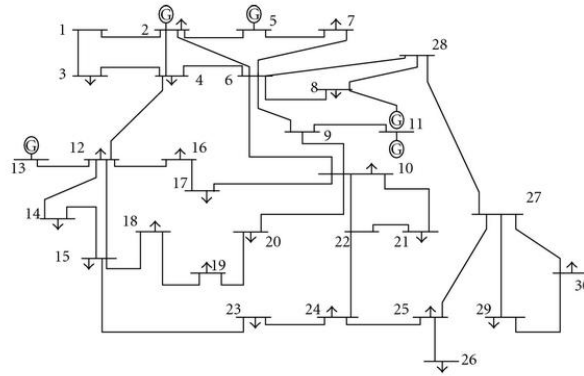


Figure 4.14: Bus branch model

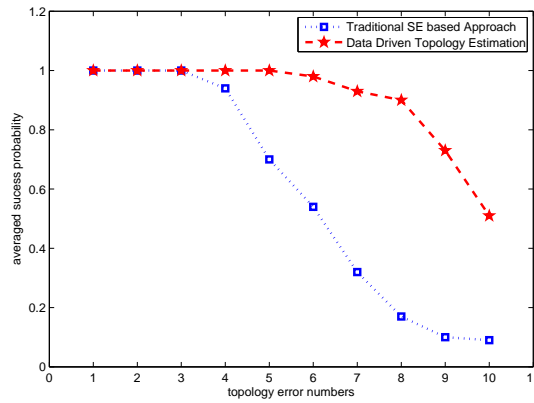


Figure 4.15: Averaged topology identification success probability

By comparing the two curves, we observe that both the traditional approach and the proposed data-driven approach can successfully identify up to three topology errors on average. When the error number increases to four and above, the success probability of the traditional approach decreases dramatically. However, for four to eight errors, the proposed nearest neighbors approach still has a success probability of close to 100%. When the topology error goes beyond 8, the proposed nearest neighbors approach starts to deteriorate, due to the increased configuration possibilities and insufficient training data. From this fact, we can reasonably conclude that, given enough training data, the proposed data-driven approach is able to handle more topology



errors than the traditional approach. Notably, since a the power grid usually has a topology error tendency at ceratin locations, the proposed data-driven approach is expected to perform even better in practice than in simulations.

## Conclusions

In this section, we propose a new historical data-driven topology estimation method based on data mining. In particular, we formulate the finding of the topology estimate as a minimum distance search problem based on the intuition that similar measurements reflect similar power system topologies. Further, a Bayesian estimate is obtained via Logistic Kernel Regression for robustness and to deal with the inherent nonlinearity of power systems. The numerical results show that our data-driven method can be a way for sustainable smart grid design to break the current model-based monitoring architecture that requires both large complex modeling and a great deal of computation time.

## 4.5 Dynamic Approach

In the non-parametric approach in the previous section with the nearest-neighbors search and regression, the static SE model limits the gain one can obtained using historical data. The dynamic approach introduced in this section aims at utilizing both historical data and the dynamic model.

### 4.5.1 Current State-of-the-Art Kalman Filter Approaches for State Estimation

Recently, several approaches have been proposed for enhancing the current static model with a linear dynamic model for quasi-static SE [15,81] via the Kalman Filter [82]. These approaches enable us to correlate current and past on-line data for SE. However, they use an Extended Kalman Filter (EKF) approach that is known to be quite suboptimal in a highly nonlinear environment such as that of an electric power system; this is caused by the fundamental flaw in the EKF of approximating non-Gaussian density function with a Gaussian density function through a closure scheme [83,84]. This causes several nontrivial problems, most importantly an unbounded error variance growth. For instance, [85] shows that EKF is difficult to tune, and is only reliable for systems that are almost linear on the time scale of the updates. Secondly, the EKF approaches in [15] only rely on active-reactive power measurement pairs, and neglect other important mea-

surement types, such as voltage magnitudes and the highly accurate phase angle measurements [8, 9] from Phasor Measurement Units (PMUs). This leads to an overall suboptimal design, significant performance loss, and poor observability; this problem is made worse in combination with unexpected missing data. Further, the computational burden is high since EKF needs to store and manipulate error covariance matrices, which becomes unmanageable for high dimensional models.

Another approach is to rely on PMU measurements [8] alone. For instance, the ongoing industry pilot experiments plan to have PMU measurements installed at all major 230 KV substations and all large generators. However, because the deploying process is gradual, the PMU measurements [9] will be insufficient and limited for a long time to come. Even if all buses are equipped with PMUs, we still need redundancy from other measurement types for robustness [10] because current experience shows that a PMU is imperfect, and its accuracy varies depending on manufacturing qualities and various sources of uncertainties (GPS synchronization, instrument transformers, A/D converters, etc.). Therefore, for future SE, we need a general dynamic SE method to deal with the combined SCADA and PMU data.

Motivated by the above systemic problems in the current and proposed SE methods, we propose a new historical data-driven SE approach. The proposed method can be implemented given recent advances in communications and computing and these advances are becoming drivers and sources of data previously unavailable in the electric power industry. Notably, learning from the data has been widely recognized as playing a critical role in achieving a key design for Wide Area Monitoring, Control and Protection (WAMPAC) systems that centers on efficient and reliable operation.

Specifically, different from earlier work [15], parameters in the model are obtained via supervised learning over the historical data with an Expectation-Maximization (EM) algorithm [65], which has been widely adopted in machine learning literature and successfully applied to economic forecasting, biological sequence studies, speech recognition, video processing, and prediction of Web-user behavior. As an iterative method, EM alternates between performing an Expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current parameter estimate, and a Maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the E-step. Then these parameter-estimates are used to determine the distribution of the latent variables (the electric power system states) in the next E-step. As a result, all unknown parameters will have closed form solutions. One possible advantage of this approach over traditional model-based static state estimation methods and dynamic state estimation [15], may come from its purely data-constructed modeling, where data can be flexibly organized

for learning according to the cyclic nature of power system operation. Further, to deal with the inherent non-linearity of an electric power system, we first employ kernels to embed the state into a higher dimensional feature space before starting the learning process with EM.

In addition to its inclusion of all measurement types for learning and estimation, we show that the proposed data-driven SE improves simulation accuracy significantly. Moreover, once the parameters are learned, the data-driven state estimator can adaptively adjust its measurement subset for possible missing data and bad data and thereby facilitate robust streaming SE. The proposed method is very attractive in practical terms because its on-line computational complexity is the same or smaller than that of the currently used static state estimation based on Newton's method, where the learning phase is done off-line due to limited computational resources. Furthermore, the algorithm can adaptive to bad data. Finally, when new historical data comes into the database, the proposed method can also be used to start a new learning round with the possibility of improvements.

#### 4.5.2 Kalman Filter Approach

Since the Data Grid maps the Power Grid in real time, it would be useful to have a view into the future evolution of the various variables. The traditional, time-tested way to simulate the dynamic behavior of a power system is with a mathematical parameter-driven model of the system. Therefore, one might support the use of dynamic equations for dynamic data-driven SE. Unfortunately, such high-order dynamic equations are highly complex, and phenomena in the power grid occur much faster than the measurements' scanning speed; i.e., the electromechanical effects of oscillations in motors and generators take place in around 10 microseconds. To use the distinct correlation between scans, past work addressed dynamic estimation problems via modifying the Kalman Filter for quasi-static state estimation. Blood et al. [16] proposed a static state estimation through Kalman filtering where power flow equations are used to derive a linear state space model. The state space modeling can be found in section 2.2. One problem with such an approach is its poor performance due to the first order Taylor approximation used in the model. To overcome this drawback, we start with a embedded kernel space approach [86] to time series modeling that preserves the framework of a linear dynamical system for our data-driven SE.

We now define the problem of historical data-driven SE as a two-stage problem as in Fig.4.16; in the first stage, historical measurements  $\{z_1, z_2, \dots, z_K\}$  are used to obtain the best model parameters in (2.9a) and

(2.9b). However, since the actual power system model (2.8a) and (2.8b) is nonlinear, we embed the state  $x_k$  into  $x_k^\phi$ , which is the feature state space proposed in [12, 13, 33, 35, 36, 87]. In particular, the feature space approach is used to extend the state space so that a maximum-likelihood estimate can be obtained in this linear model. Once the system parameters are estimated, stage two is performed via a standard online Kalman Filter based on current measurement  $z_t$  at time  $t$  to find the Minimum Mean Square Error (MMSE) estimate of  $\hat{x}_t$ . This two-stage historical data-driven SE process is described next.

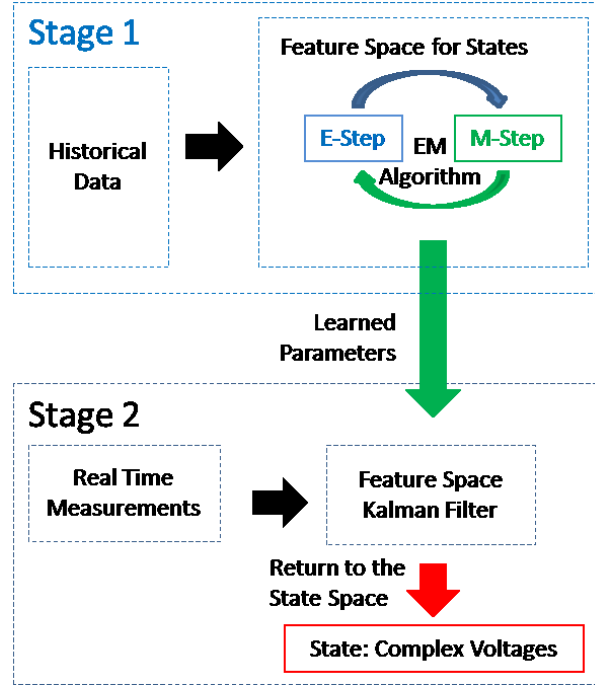


Figure 4.16: Block diagram of the proposed historical data-driven approach.

### 4.5.3 Stage 1: Historical Data-Driven Parameter Estimation

Here we assume stationarity, which means that topology changes and bad data have not been detected for the time being. In the case of topology changes, only the historical data associated with the new topology is used. In case of detected bad data, the learned measurement matrix  $C$  in (2.9b) will be changed adaptively by eliminating the corresponding rows.

## Kernel Spaces for Nonlinearity

One drawback of using (2.9) to approximate (2.8) comes from the inherent nonlinearity in electric power systems, which contain watts, vars, voltage magnitudes, and voltage angles. To overcome this drawback, we start with a kernel space approach [86] to time series modeling that preserves the framework of the linear dynamical system for our data-driven SE.

The key to the proposed approach is the embedding of state space into a feature space as  $\mathbf{x}_k \rightarrow \mathbf{x}_k^\phi : \phi = \phi(\mathbf{x}_k)$ , on which we assume the linear state space model (Fig.4.17),

$$\mathbf{x}_k^\phi = A^\phi \mathbf{x}_{k-1}^\phi + \mathbf{w}_k^\phi, \quad (4.34a)$$

$$\mathbf{z}_k = C^\phi \mathbf{x}_k^\phi + \mathbf{u}_k, \quad (4.34b)$$

where we assume  $\mathbf{w}_k^\phi \sim \mathcal{N}(\mathbf{0}, \Gamma^\phi)$ .

Using our work on Semidefinite Programming for State Estimation in Chapter 3 [33, 36, 87, 88], we now choose a polynomial kernel with order 2 for linearity in (4.34b). Since the generation power and load changes cause the system state in the power system to change, our polynomial kernel choice for (4.34a) is also preferable because it produces a quadratic form of states (voltages) rather than the direct voltage expression in (2.9).

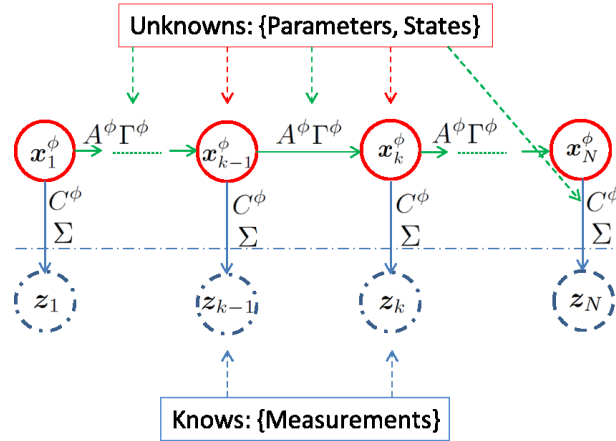


Figure 4.17: Linear feature space model.

## Expectation-Maximization Algorithm

After obtaining the linear dynamic system in the embedded feature space (4.34), we then estimate the unknown parameters ( $\theta = \{A^\phi, \Gamma^\phi, C^\phi, \Sigma\}$ ) and the hidden state sequence  $\{x_k^\phi\}$  (latent variables) in the upper part of Fig.4.17 that maximize the likelihood of the historical data sequence  $\{z_k\}$  in the lower part of Fig.4.17. The model has latent variables, but this can be addressed using the EM algorithm [65]. Such a method has become an appealing field for machine learning approaches [66, 89] over the last few decades. Specifically, the EM algorithm is an iterative method for finding maximum-likelihood or maximum-a-posteriori (MAP) estimates of parameters in statistical models that are dependent on unobserved latent variables. It starts with a certain initial selection for the model parameters, which can be denoted by  $\theta^{old} = \{A^{\phi,old}, \Gamma^{\phi,old}, C^{\phi,old}, \Sigma^{old}\}$ . Then we take these parameter values and find the posterior distribution of the latent variables  $p(x^\phi|z, \theta^{old})$  in the E-step. Subsequently, we use this posterior distribution to evaluate the expectation of the logarithm of the complete-data likelihood function, as a function of the parameters  $\theta$ , and to employ a maximum-likelihood criterion to find the  $\theta$  in the M-step before going back to the E-step. Then the M-step and E-step repeat until the algorithm converges.

### Expectation (E) Step

In this part, filtering (forward) and smoothing (backward) algorithms are conducted to obtain the hidden states. The first step is forward-moving like the Kalman Filter. Given the initial value of state mean  $\mu_0^\phi$ , state variance  $V_0$ , and “old” fixed parameters  $\theta^{old}$ , we can compute the Kalman gain, state mean, and state variance at time slot  $k = 1$  as follows:

$$K_1 = V_0(C^{\phi,old})^T (C^{\phi,old}V_0(C^{\phi,old})^T + \Sigma^{old})^{-1}, \quad (4.35a)$$

$$\mu_1 = \mu_0^{old} + K_1(z_1^{old} - C^{\phi,old}\mu_0^{old}), \quad (4.35b)$$

$$V_1 = (I - K_1C^{\phi,old})V_0, \quad (4.35c)$$

where  $I$  is an identity matrix. The algorithm then moves forward until the end of the training data at discrete time  $K$ .

$$P_{k-1} = A^{\phi,old} V_{k-1} (A^{\phi,old})^T + \Gamma^{\phi,old}, \quad (4.36a)$$

$$K_k = P_{k-1} (C^{\phi,old})^T (C^{\phi,old} P_{k-1} (C^{\phi,old})^T + \Sigma^{old})^{-1}, \quad (4.36b)$$

$$\boldsymbol{\mu}_k^{\phi, fw} = A^{\phi,old} \boldsymbol{\mu}_{k-1}^{\phi, fw} + K_k (\mathbf{z}_k - C^{\phi,old} A^{\phi,old} \boldsymbol{\mu}_{k-1}^{\phi, fw}), \quad (4.36c)$$

$$V_k^{fw} = (I - K_k C^{\phi,old}) P_{k-1}, \quad (4.36d)$$

where  $P_{k-1}$  updates the state covariance matrix based on the previous estimate.  $K_k$  is the Kalman gain for updating the prediction with the new measurement vector  $\mathbf{z}_k$ . The superscript “fw” indicates it is a “forward” step while the superscript  $T$  represents the transpose operator. Thus, given the values of  $\boldsymbol{\mu}_{k-1}^{\phi}$  and  $V_{k-1}$  one time slot back, together with the new observation  $\mathbf{z}_k$ , we can evaluate the Gaussian marginal for  $\mathbf{x}_k^{\phi}$  as having mean  $\boldsymbol{\mu}_k^{\phi, fw}$  and covariance  $V_k^{fw}$ .

The Kalman Filter above is a “forward algorithm” that is optimal in obtaining current state estimate by using past and current measurements. However, since all future measurements are more or less dependent on the past state, a better estimate can be achieved by conducting a backward (“bw”) algorithm, and obtain a smoothed estimate,

$$\boldsymbol{\mu}_k^{\phi, bw} = \boldsymbol{\mu}_k^{\phi, fw} + J_k (\boldsymbol{\mu}_{k+1}^{bw} - A \boldsymbol{\mu}_k^{\phi, fw}), \quad (4.37a)$$

$$V_k^{bw} = V_k^{fw} + J_k (V_{k+1}^{bw} - P_k) J_k^T, \quad (4.37b)$$

where  $J_k \triangleq V_k^{fw} (A^{\phi,old})^T (P_k)^{-1}$ .

Note that these recursions require a completed forward pass so that the quantities  $\boldsymbol{\mu}_k^{\phi, fw}$  and  $V_k$  will be available for the backward pass.

### Maximization (M) Step

As a summary of the E-step for the M-step, we rewrite the results above in the following form

$$E[\mathbf{x}_k^{\phi}] = \boldsymbol{\mu}_k^{\phi, bw}, \quad (4.38a)$$

$$E[\mathbf{x}_k^{\phi} (\mathbf{x}_{k-1}^{\phi})^T] = J_{k-1} V_k^{bw} + \boldsymbol{\mu}_k^{\phi, bw} (\boldsymbol{\mu}_{k-1}^{\phi, bw})^T, \quad (4.38b)$$

$$E[\mathbf{x}_k^{\phi} (\mathbf{x}_k^{\phi})^T] = V_k^{bw} + \boldsymbol{\mu}_k^{\phi, bw} (\boldsymbol{\mu}_k^{\phi, bw})^T. \quad (4.38c)$$

With the statistical estimate of  $\{\mathbf{x}_k^\phi\}$  in (4.38) and the measurement set sequence  $\{z_k\}$ , the EM algorithm can estimate parameters using the maximum-likelihood method rather than the complete-data log-likelihood function below.

$$\ln p(\mathbf{X}^\phi, \mathbf{Z} | \boldsymbol{\theta}) = \ln p(\mathbf{x}_1^\phi | \boldsymbol{\mu}_0^\phi, V_0) + \sum_{k=2}^N \ln p(\mathbf{x}_k^\phi | \mathbf{x}_{k-1}^\phi, A^\phi, \Gamma^\phi) + \sum_{k=1}^N \ln p(z_k | \mathbf{x}_k^\phi, C^\phi, \Sigma), \quad (4.39)$$

in which we make explicit the dependence on the parameters, and define  $\mathbf{X}^\phi = \{\mathbf{x}_1^\phi, \dots, \mathbf{x}_N^\phi\}$ ,  $\mathbf{Z} = \{z_1, \dots, z_N\}$ . Note that  $\{A^\phi, \Gamma^\phi, C^\phi, \Sigma\}$  are parameters in need of a new estimation in this M-step, rather than the  $\{A^{\phi,old}, \Gamma^{\phi,old}, C^{\phi,old}, \Sigma^{old}\}$  with fixed values in the E-step.

We now take the expectation of the complete-data log-likelihood regarding the posterior distribution  $\ln p(\mathbf{X}^\phi, \mathbf{Z} | \boldsymbol{\theta})$ , namely,  $E_{\mathbf{X} | \boldsymbol{\theta}^{old}}[\ln p(\mathbf{X}^\phi, \mathbf{Z} | \boldsymbol{\theta})]$ . By taking the derivative of  $E[\ln p(\mathbf{x}_1^\phi | \boldsymbol{\mu}_0^\phi, V_0)]$  and  $E[\sum_{k=2}^N \ln p(\mathbf{x}_k^\phi | \mathbf{x}_{k-1}^\phi, A^\phi, \Gamma^\phi)]$  and employing the first order condition, we reach

$$\boldsymbol{\mu}_0^{\phi,new} = E[\mathbf{x}_1^\phi], \quad (4.40a)$$

$$V_0^{new} = E[\mathbf{x}_1^\phi (\mathbf{x}_1^\phi)^T] - E[\mathbf{x}_1^\phi] E[(\mathbf{x}_1^\phi)^T], \quad (4.40b)$$

$$A^{\phi,new} = \left( \sum_{k=2}^N E[\mathbf{x}_k^\phi (\mathbf{x}_{k-1}^\phi)^T] \right) \left( \sum_{k=2}^N E[\mathbf{x}_{k-1}^\phi (\mathbf{x}_{k-1}^\phi)^T] \right)^{-1}, \quad (4.40c)$$

$$\begin{aligned} \Gamma^{\phi,new} = \frac{1}{N-1} \sum_{k=2}^N \left\{ E[\mathbf{x}_k^\phi (\mathbf{x}_k^\phi)^T] - A^{new} E[\mathbf{x}_{k-1}^\phi (\mathbf{x}_{k-1}^\phi)^T] - E[\mathbf{x}_k^\phi (\mathbf{x}_{k-1}^\phi)^T] (A^{\phi,new})^T \right. \\ \left. + A^{\phi,new} E[\mathbf{x}_{k-1}^\phi (\mathbf{x}_{k-1}^\phi)^T] (A^{\phi,new})^T \right\}. \end{aligned} \quad (4.40d)$$

Note that  $A^{\phi,new}$  must be evaluated first, and then the result can be used to determine  $\Gamma^{\phi,new}$ . Similarly, in order to determine the new values of  $C^{\phi,new}$  and  $\Sigma^{new}$ , maximizing  $E[\sum_{k=1}^N \ln p(z_k | \mathbf{x}_k^\phi, C^\phi, \Sigma)]$  in (4.39) with respect to  $C^\phi$  and  $\Sigma$ , then we have

$$C_{row\ i}^{\phi,new} = \left( \sum_{k=1}^N z_{n,i} E[(\mathbf{x}_k^\phi)^T] \right) \left( \sum_{k=1}^N E[\mathbf{x}_k^\phi (\mathbf{x}_k^\phi)^T] \right)^{-1}, \quad (4.41a)$$

$$\Sigma^{new} = \frac{1}{N} \sum_{k=1}^N \left\{ E[z_k z_k^T] - C^{\phi,new} E[\mathbf{x}_k^\phi] z_k^T - z_k E[(\mathbf{x}_k^\phi)^T] \cdot C^{\phi,new} + (C^{\phi,new})^T E[\mathbf{x}_k^\phi (\mathbf{x}_k^\phi)^T] C^{\phi,new} \right\}. \quad (4.41b)$$

These new parameters are treated as  $\boldsymbol{\theta}^{old}$  in a subsequent E-step, and a new round of EM computation is started until the parameters converge. Notably, the  $C^{\phi,new}$  matrix, because of its row-wise form, is adaptable for bad data detection.



#### 4.5.4 Stage 2: On-line Kalman Filter State Estimation

After convergence, the parameters learned and denoted by  $\hat{\theta} = \{\hat{A}^\phi, \hat{\Gamma}^\phi, \hat{C}^\phi, \hat{\Sigma}\}$ , are used in the standard Kalman Filter for dynamic power system AC SE. When conducting a Kalman Filter for real-time data with the parameters  $\hat{\theta}$  learned from the historical data, the state estimate lies in embedding feature space rather than the original state space. Therefore, we need to conduct a further step to map the state back into the state space, leading to the following optimization problem

$$\hat{\mathbf{x}}_t = \arg \min_{\mathbf{x}_t} \|C^\phi \phi(\mathbf{x}_t) - \mathbf{z}_t\|^2, \quad (4.42)$$

where subscript  $t$  is used to distinguish the real time SE from the EM algorithm over the historical data, where the subscript  $k$  is employed.

Performing Newton's method with a flat start for state recovery is not as good as using the converged values of  $\hat{C}^\phi$  in the EM algorithm and the real-time Kalman Filter result  $\hat{\mathbf{x}}^\phi$  to compute the MMSE estimate via  $h(\hat{\mathbf{x}}_t) = \hat{C}^\phi \hat{\mathbf{x}}_t^\phi = h(\mathbf{x}_t) + \mathbf{u}_t$ , which creates a recovery problem as

$$\hat{\mathbf{x}}_t = \arg \min_{\mathbf{x}_t} \left( h(\hat{\mathbf{x}}_t) - h(\mathbf{x}_t) \right)^T R^{-1} \left( h(\hat{\mathbf{x}}_t) - h(\mathbf{x}_t) \right), \quad (4.43)$$

where  $R = (\hat{C}^\phi)^T P_t^{-1} \hat{C}^\phi$ .

#### Summary

The calculation steps of the proposed data-driven SE in Fig.4.16 are the following

- embed the state space model into feature space via the polynomial kernel with order 2 to deal with the inherent nonlinearity of the electric power system;
- use historical data to learn the system parameters in feature space with the EM algorithm;
- apply the learned linear state space model in order to conduct a feature space Kalman Filter over a real-time dynamic SE;
- conduct a reverse state mapping back into state space for complex bus voltages.

### 4.5.5 Simulation Results

The simulations are implemented on the IEEE standard 14, 30, 39, 57, 118, and 300-buses test systems. 300-bus simulation results are presented here with the MATLAB Power System Simulation Package (MATPOWER) from [37, 38]. The historical data preparation is similar to Section 4.3.3. Data between February 2005 and December 2010 are used for training with EM algorithm, and the rest are used for testing various estimators for comparison.

### Numerical Results

The proposed method achieves its improvements by conducting the following SE approaches below on the 01/01/11 to 12/31/11 testing data:

- industrially used Static SE;
- the Kalman Filter Approach to Quasi-Static SE from [15], denoted by EKF;
- the proposed Data-Driven SE in this paper.

We apply in Fig.4.18 a normalization for the Euclidean distances between the various estimation results and the true system states at different time slots.

$$\text{Ratio}_{\{Static,k\}} = \frac{\|\mathbf{v}_{Static,k} - \mathbf{v}_{True,k}\|}{\|\mathbf{v}_{Static,k} - \mathbf{v}_{True,k}\|} = 1, \quad (4.44a)$$

$$\text{Ratio}_{\{EKF,k\}} = \frac{\|\mathbf{v}_{EKF,k} - \mathbf{v}_{True,k}\|}{\|\mathbf{v}_{Static,k} - \mathbf{v}_{True,k}\|}, \quad (4.44b)$$

$$\text{Ratio}_{\{DataDriven,k\}} = \frac{\|\mathbf{v}_{DD,k} - \mathbf{v}_{True,k}\|}{\|\mathbf{v}_{Static,k} - \mathbf{v}_{True,k}\|}. \quad (4.44c)$$

The  $x$  coordinate in Fig.4.18 is the time of the testing data. The  $y$  coordinate is for the metric in (4.44). We observe that the EKF SE is similar to the static SE (due to its measurement number reduction) while the proposed data-driven approach significantly reduces error by at least 20% compared to the other two approaches except in rare cases. The proposed method can sometimes achieve an error reduction of more than 50%. In addition, the small distance between the data-driven approach curve and the zero value line informs us that the proposed approach greatly increases the possibility of hitting the true state because not

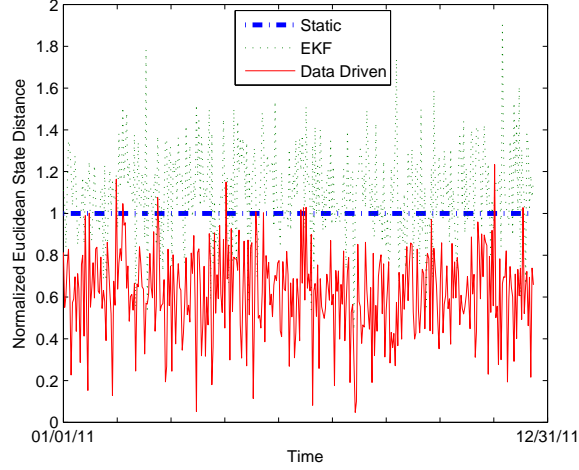


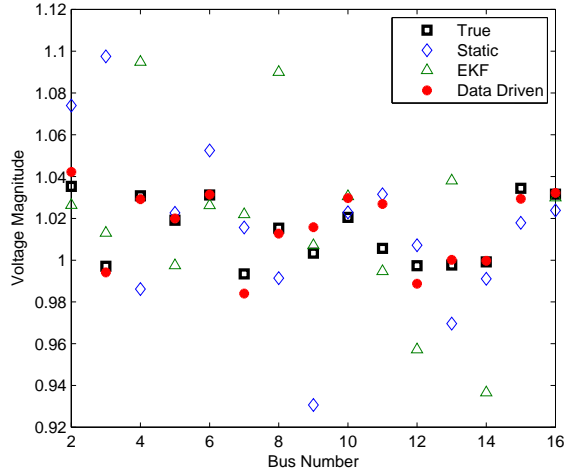
Figure 4.18: Normalized Euclidean state distance.

only does it learn patterns from the historical data, but it also deals with the inherent nonlinearity of electric power systems by using kernel spaces.

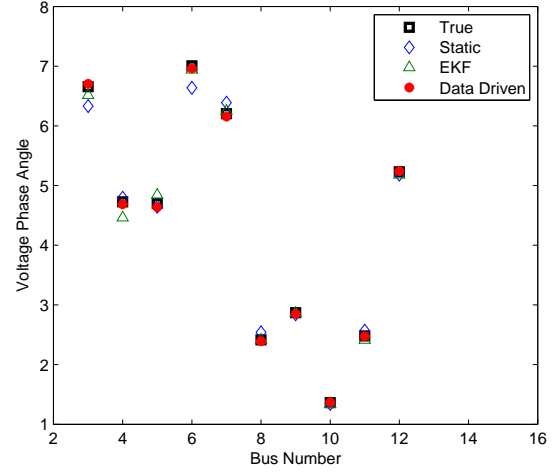
Fig.4.19a and Fig.4.19b compare the proposed method with others in terms of voltage magnitudes and phase angles and display the results in the state domain. To visualize the comparison, Fig.4.19 displays one part of the test results obtained (bus 2 to bus 16). Similar results are observed at the other buses in the 300-bus system. From Fig.4.19a and Fig.4.19b, the proposed approach (in red dots) shows superiority in performance to static SE (in blue diamonds) and EKF (in green triangle) by providing a much closer estimate to the true state (in black boxes).

## 4.6 Chapter Summary

This chapter proposed historical data-driven frameworks for solving both static and dynamic AC power system state estimation (SE) problems in EMS/SCADA systems. In particular, we used the periodicity of power system operation in a static data-driven approach for a nearest neighbors search to improve accuracy. Then, non-parametric regression was used to obtain the current state estimate, followed by using the kernel trick for state embedding. Several data filtering steps were conducted to further enhance the robustness of SE. Finally, speedup methods were introduced to make the data-driven approach online.



(a) Voltage magnitudes



(b) Voltage phase angles

Figure 4.19: IEEE 300 bus.

To use the inertia of the power systems, we also introduced a dynamic state estimation approach in the embedded kernel space. First, we formulated a linear state space model in the proper embedded space to deal with the inherent nonlinearity in power systems by adopting the Expectation-Maximization algorithm for system parameter learning. Second, we used the learned parameters in linear feature space for real-time Kalman Filter state estimation. Subsequently, the state in embedded feature space was converted back into state space via MMSE estimation to obtain complex voltages. We demonstrated by simulations that using the data-driven approach can significantly reduce error and obtain a more accurate estimate than when applying the conventional static or dynamic methods, despite the chance of missing data and measurement types. Notably, we can conduct this learning process continuously to achieve accuracy when accommodating new measurements into SCADA.

## Chapter 5

# Graphical Model-Based State Estimation

The potential gap in SE performance between the state-of-the-art methods and the desired informative data exploration can result in fragile grid behavior, such as blackouts for example. The previous chapter aims at utilizing historical data to bridge the gap.

However, the gap may grow as new unconventional resources are deployed within electric power grids, which were not initially designed to accommodate the social objectives of deploying clean resources and reducing the environmental footprint [90]. For instance, numerous distributed power plants (wind, solar, etc.) have been connected to the existing grid in the past ten years. Their presence and exponential growth have raised great concerns about the reliability of electric power grids in the future because these new plants serve primarily local areas and are hard to absorb into large power systems. Furthermore, President Obama's goal of putting one million electric vehicles on the road by 2015 will also contribute to the grid architecture shift. The robustness of these new architectures will have to be studied. Because of their unconventional characteristics, new and proper modeling of these technologies in order to account for the uncertainties of their architectures is necessary for accurate SE in the smart grid.

Besides, the SE used by the industry today is hard to scale up and is computationally complex [11]. To avoid excessive computational complexity, only the representations of Extra High Voltage (EHV), High Voltage (HV) and occasionally Medium Voltage (MV) in complex multi-voltage power grids are included here. However, in current industrial operation, low voltage (LV) distribution networks are not modeled nor supported by today's on-line SE, which in turn, makes it difficult to estimate the status and the states of diverse resources and users newly connected to LV level distribution systems. The operators of traditional

power grids are now facing inherent difficulties in managing the effects of small scale generations and loads, terms which cover not only renewable energy generators, such as wind and solar generators, but also responsive small electricity users and storage-based electricity users such as electric cars. While these new components are more environmentally friendly, they also raise tremendous concerns regarding the secure and reliable operation of backbone EHV/HV power grids. In particular, each and every component's state needs to be estimated in order to account for its effect on the state of the entire power grid. The necessity to estimate the online state of the entire electric power grid makes it even more difficult than before to manage all the data in a centralized way. For future electric energy systems, a multi-layered, distributed implementation of state estimators is likely to become the preferred approach between different layers within the complex grid, because the grid has to integrate the objectives of various users. This requires a systematic design of distributed algorithms that are capable of maintaining or improving upon the performance of the centralized methods.

Therefore, this chapter proposes using distributed system architecture as well as historical data to manage the uncertainties created by new energy providers, users, and delivery providers in a distributed way in the future smart grid. This chapter offers a formal problem formulation for modeling, decision-making, and performance evaluation, which is achieved by a proposed graphical model description [65] of the electric power grid. Such a modeling is inspired by the exciting results made available by applying graphical models to compact uncertainty representations and computationally-tractable inferences [91, 92]. Specifically, we consider a graph representing the electrical power system as a graphical model [1], and we model power grid states (the bus voltages) as random variables on the graph vertices. The edges of the graph determine the interaction of the state variables according to physical laws (i.e. Kirchhoff's laws). Viewed together, the graphical model is specified by the joint density of random variables in the network for state estimation, subject to the constraints imposed by the physical laws.

Distinguished from the traditional state estimation process that aims to minimize mean square error (MMSE) [2,4], this section aims to maximize a posteriori probability (MAP) popular in the graphical model. To achieve MAP, an exact inference on the distributed SE for trees is conducted via belief propagation (BP) [65, 93–95], which is an approach that embed the graphical model distribution into a larger probability space. This is one way of organizing the “global” computation of marginal beliefs in terms of smaller local computations in the graphical model, which are enabled by distributed computation capability and the communication capability (a wireless, telephony or Internet link) of components in the future smart grid

(such as a small generator).

To deal with meshed networks like transmission systems, we further adopt a variational belief propagation (VBP) algorithm [17, 96] with the embedding techniques introduced in Section 1.4 in sum-product form for single bus state estimation, by breaking a meshed structure into multiple spanning trees [89]. In addition to our preliminary work on sum-product algorithms, we illustrate how to conduct joint state estimation for the whole power grid via max-product algorithm, which is preferable to the sum-product in power grid analysis. To avoid bias, we propose using historical data to compute the prior information for MAP, or using uninformative prior. We further propose a modified VBP called sequential tree-reweighted message passing to solve the convergence problem and other challenges that arise out of the application of the VBP algorithm [97]. Finally, a tree agreement condition is proposed in order to check the optimality of all VBP algorithms.

Such BP-based algorithms can form the basis for enhanced system operation in smart grids by enabling numerous small system users to participate. For example, the operator of the backbone system does not need to estimate the state (power consumed, voltage) of smart meters based on the measurement provided at a given location. Instead, the state of small users can be estimated in a distributed, message-passing manner with neighboring system users where smart meters are located. Then the aggregated information is communicated to the backbone system operator in a bottom-up way. Therefore, the BP method offers a way to break the current centralized monitoring architecture that requires both a large communication overhead and tight data synchronization.

We use simulations of IEEE test systems of up to 300 buses to show that this graphical model approach can provide more accurate results than weighted least square (WLS) SE does. Further, the proposed approach features an approximately linear computational time, which has not been available in the past. Based on these preliminary results, we believe the proposed method can offer a major promise of scalable SE with high accuracy for the future smart grid.

In Section 5.1, we present state-of-the-art analysis of the Bayesian method for electric power systems. In Section 5.2, we introduce probabilistic graphical modeling. In Section 5.3, we use various systems as a proof-of-concept illustration of the proposed graphical model approach. The entire chapter is summarized in Section 5.4.

## 5.1 State-of-the-Art Bayesian Method and Probabilistic Analysis for Power Systems

In the following, we summarize various work in power system analysis related to the Bayesian method and probabilistic approach.

- **Evaluate Performances:** [98] presents an application of Bayesian networks (BN) to the problem of reliability assessment of power systems. Efficient probabilistic inference algorithms in Bayesian networks permit not only computation of the loss of load probability but also the answers to various probabilistic queries about the system. [99] presents a method for power system security assessment based on the Bayes classifier, which calculates probabilistic security indices as well as on-line security assessment.
- **Bayes Estimation:** [100] proposes a reliable and efficient methodology based on the recursive Bayesian approach and its improved version is proposed for transformer tap position estimation. In [101], three element-oriented models based on simplified Bayesian networks with Noisy-Or and Noisy-And nodes are proposed to estimate the faulty section of a transmission power system. The three models are used to test whether any transmission line, transformer, or busbar within a blackout area is faulty.
- **Probabilistic Estimation:** [102] proposes storing critical network configurations in the form of a model bank and conducting a recursive Bayesian topology reconfiguration to utilize the output of the state estimation function of each model in the bank. [103] proposes a Quasi-Monte Carlo-based method to solve the probabilistic load flow problem of a radial distribution network. Such a method is particularly suitable for high dimension problems with low effective dimensions, and has been successfully used to solve large scale problems in econometrics and statistical circuit design. With the increase in intermittent generation in power systems, [104] includes uncertainties to calculate probability density function (PDF) of reactive power at generators in order to evaluate their capability of maintaining bus voltage at the desired level for power flow solution. [105] presents a probabilistic modeling in a hybrid solar-wind power system for solar tracking, which shows annual energy gain.
- **Prediction:** In [106], a classification method combined with a simple regression model is presented to predict the discrete PDF of power market clearing prices, a prediction which is critical for decisions such as how to optimize bidding strategies, and which is very difficult to accomplish because of high



market uncertainties. In [107], a Bayesian model has been designed for Short-Term Load Forecasting, which is a very important aspect of the power system that ensures safe and economic operations and achieves a scientific management of the power system.

- Network Planning and Stability Analysis: [108] proposes a methodology and a practical tool for the study of long-term network planning under uncertainties by using a probabilistic model of micro-scenarios based on past statistics. Massive Monte-Carlo simulations are used to generate and simulate a large number of scenarios and store the detailed results in a relational database. The probability of transient stability is analyzed in [109] with a Monte Carlo simulation method to determine whether transient stability probability is feasible.

## 5.2 Probabilistic Graphical Modeling for State Estimation

In the typical mathematical description of a power grid, the key variables are usually assumed to be deterministic. However, there is an increasing recognition of sources of randomness in the grid, such as random power due to newly-built intermittent distributed wind generators, random electric vehicle charging, and random failure of transmission lines, etc., due to natural and human causes. Therefore, the traditional power system model must be extended to account for stochastic property; since these variables are related through the power grid graph, this suggests the use of a graphical model description of the electric power grid.

The basic graph representation is motivated by an abstract model of the existing U.S. electric power grid, represented as a graph  $G(V, E)$ . The vertices of the graph represent state variables  $x_i$ , which have a concrete physical interpretation as the voltage of bus  $i$  [110]. Other auxiliary variables  $z_i$  represent physical quantities, such as power flow. Known physical laws, such as Kirchoff laws determine the interaction of these variables, which are represented by the edges  $E$  of the graph [110].

Recall from Chapter 2, Section (2.1.2), a power grid is defined as a physical graph  $G(V, E)$  with vertices  $V$  that represent the buses (generators and loads) and edges  $E$  that represent the transmission lines and transformers. The graph of the physical network can be visualized as the physical layer in Fig.5.1, which describes the interactions between the electric power network variables based on electromagnetics (Kirchhoff's laws). The probabilistic measurement model of AC power system state estimation is expressed

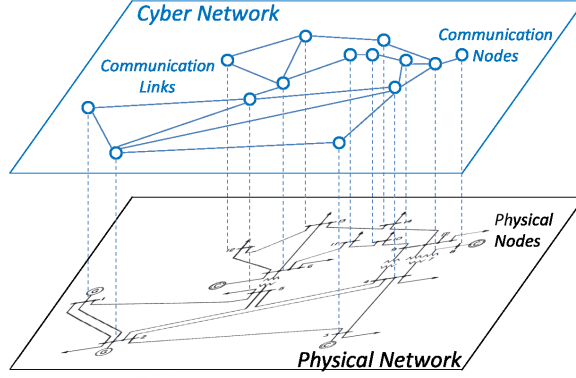


Figure 5.1: Physical network and cyber network (14 bus system).

as

$$z_i = h_i(\mathbf{v}) + u_i, \quad (5.1)$$

where the vector  $\mathbf{v} = [|v_1|e^{j\delta_1}, |v_2|e^{j\delta_2}, \dots, |v_n|e^{j\delta_n}]^T$  represents the probabilistic power system states, instead of the conventionally used deterministic states.  $u_i$  is the  $i^{th}$  additive measurement noise assumed to be independent Gaussian random variable with zero mean, i.e.,  $\mathbf{u} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ , where  $\Sigma$  is a diagonal matrix, with the  $i^{th}$  diagonal element  $\sigma_i^2$ .  $z_i$  is the  $i^{th}$  telemetered measurement, such as power flow and voltage magnitude.  $h_i(\cdot)$  is the nonlinear function associated with the  $i^{th}$  measurement.

The probabilistic power system state estimator aims to find an estimate ( $\hat{\mathbf{v}}$ ) of the true states ( $\mathbf{v}$ ) that achieves the maximum a posteriori probability (MAP), given the measurement set  $\mathbf{z}$  and the priori state information of  $\mathbf{v}$  according to the measurement model in (5.1). It is mathematically expressed as

$$\max_{\mathbf{v}} p(\mathbf{v}|\mathbf{z}) = \frac{p(\mathbf{v})p(\mathbf{z}|\mathbf{v})}{p(\mathbf{z})}, \quad (5.2)$$

where  $p(\cdot)$  represents the probability density function. Such a process is achieved via the cyber network layer as in Fig.5.1. In this work, the cyber network topology is the same as the physical network topology.

To obtain the MAP estimate in (5.2), we need to 1) obtain proper formulations for  $p(\mathbf{v})$ ,  $p(\mathbf{z}|\mathbf{v})$ , and  $p(\mathbf{z})$ ; and 2) employ efficient algorithms to conduct marginalization or maximization over  $p(\mathbf{v}|\mathbf{z})$  with respect to  $\mathbf{v}$ . While the first problem will be addressed in subsection 5.2.4, we start by introducing efficient marginal and maximization computations for trees, which will be extended to account for meshed networks such as the IEEE 14 bus system.

### 5.2.1 Belief Propagation for Tree Networks

Belief Propagation, also known as sum-product message passing, is a message passing algorithm for performing inference on graphical model [111]. Due to its ability of decompose large marginalization problems into smaller ones, it allows marginals to be computed efficiently, leading to wide applications.

Here we provide a simple illustration of four binary variables in a chain, defined by the joint probability  $p(v_1, v_2, v_3, v_4)$  in Fig.5.2. When the graph sparsity is disregarded,  $2^3 = 8$  summations (this grows exponentially) are needed to obtain the marginal distribution of  $v_1$  via  $p(v_1) = \sum_{v_2, v_3, v_4} p(v_1, v_2, v_3, v_4)$ . Instead, BP needs only  $2 \cdot 3 = 6$  summations (the number grows linearly here) when exploring the system structure via  $p(v_1) = \sum_{v_2} p(v_1|v_2) \sum_{v_3} p(v_2|v_3) \sum_{v_4} p(v_4)p(v_3|v_4)$ . In such an operation,  $M_{4 \rightarrow 3} \triangleq \sum_{v_4} p(v_4)p(v_3|v_4)$ , a function of  $v_3$ , is interpreted as a message passed from node 4 to node 3 [93]. It represents what node 4 believes node 3's probability mass function (pmf) to be, based on node 4's own pmf and the joint pmf between them.



Figure 5.2: A graph with four nodes.

As a generalization [65], BP can be conducted on an arbitrary tree-structured graph  $G(V, E)$  with a pairwise Markov Random Field factorization

$$p(v_1, v_2, \dots, v_n) = \alpha \prod_{s \in V} \phi_s(v_s) \prod_{(s,t) \in E} \phi_{st}(v_s, v_t), \quad (5.3)$$

where  $\phi_s$  and  $\phi_{st}$  are compatibility functions [93] for joint distribution  $p(v_1, v_2, \dots, v_m)$ ;  $\alpha$  denotes a positive constant chosen to ensure distribution normalization. Finally, BP conducts message updates according to

$$M_{s \rightarrow t}(v_t) \leftarrow \sum_{v_s} \phi_s(v_s) \phi_{st}(v_s, v_t) \prod_{k \in \mathcal{N}(s), k \neq t} M_{k \rightarrow s}(v_s), \quad (5.4)$$

where  $\mathcal{N}(s)$  is the set of neighboring buses of the bus  $s$ .

In such a message-passing calculation, product is taken over all messages going into node  $s$  except for the one coming from node  $t$ . In practice, we can start with the nodes on the graph edge, and compute a

message only when all necessary messages are received. Therefore, each message needs to be computed only once for a tree structured graph.

### 5.2.2 Variational Belief Propagation for Mesh Networks

As observed above, the key assumptions of the BP algorithm are: 1) each subgraph remains a tree after graph division; and 2) the subgraphs are disjointed. Such assumptions often do not hold for electric power transmission networks, such as in the test case of the IEEE 14-bus system. To overcome this problem, a variational BP approach [17] is used by randomly generating spanning trees of the meshed network. The key is to assign probability to the edges based on the edges' appearance probability  $\rho_{st}$  in the spanning trees according to (5.5). Mathematically, VBP starts by randomly generating spanning trees [112] according to the mesh network structure, leading to the probability assignment of edge appearance  $\rho_{st}$  as follows

$$\rho_{st} = \frac{\text{No. of spanning trees with edge (s,t)}}{\text{No. of all spanning trees}}. \quad (5.5)$$

Subsequently, convex combination methods are adopted to approximate the inference on the meshed networks with the BP algorithm on the trees [96]. Mathematically, the new message-passing algorithm below for marginalization is run with  $\rho_{st}$  until convergence of the states,

$$M_{t \rightarrow s}^{n+1}(v_s) = \alpha \sum_{v_t} \left\{ \exp \left( \frac{\theta_{st}(v_s, v_t)}{\rho_{st}} + \theta_t(v_t) \right) \frac{\prod_{k \in \mathcal{N}(t) \setminus s} [M_{k \rightarrow t}^n(v_t)]^{\rho_{kt}}}{[M_{s \rightarrow t}^n(v_t)]^{(1-\rho_{ts})}} \right\}, \quad (5.6)$$

where  $\theta_{st}$  and  $\theta_t$  are the exponential parameters associated with compatible functions such as  $\phi_{st}$ , and  $\phi_s$  in (5.3). Note that, if  $\rho_{st} = 1$ ,  $\forall (s, t) \in E$ , the VBP in (5.6) degrades to the BP form in (5.4) due to the tree structure implication. Further, the VBP approach relies on a special exponential family that includes many common distributions, such as the normal, exponential, gamma, and Chi-square distributions, instead of the more general arbitrary distribution in (5.3) and (5.4). However, the exponential family is sufficient for the purpose of this paper.

In power systems, when the joint maximum a posterior probabilities are preferred, the following formula can be employed according to [113]:

$$M_{t \rightarrow s}^{n+1}(v_s) = \alpha \max_{v_t} \left\{ \exp \left( \frac{\theta_{st}(v_s, v_t)}{\rho_{st}} + \theta_t(v_t) \right) \frac{\prod_{k \in \mathcal{N}(t) \setminus s} [M_{k \rightarrow t}^n(v_t)]^{\rho_{kt}}}{[M_{s \rightarrow t}^n(v_t)]^{(1-\rho_{ts})}} \right\}. \quad (5.7)$$

As we can see, the algorithm in (5.7) simply replaces the “ $\sum$ ” operator in (5.6) with the “max” operator. This is because both (5.6) and (5.7) are based on giving different weights to the spanning trees. Therefore, they are also called tree re-weighted algorithms.

### 5.2.3 Sequential Variational Belief Propagation Algorithm

The VBP algorithms introduced in the last subsection were formed by minimizing an upper bound based on a convex combination. However, there are no guarantees regarding the decrease of this bound [97], which may actually go up. This is because the tree re-weighted algorithm does not maintain the convex combination constraint  $\sum \rho_i \theta_i = \bar{\theta}$ , where  $\bar{\theta}$  and  $\theta_i$ s are parameters associated with the mesh network and the spanning trees. Without such a constraint, re-parameterizations of the original parameter vector may violate the equality constraint. Besides, VBP has no guaranteed convergence. To improve its performance, [97] constructs a modified VBP called a sequential tree re-weighted message passing algorithm.

**Algorithm: Sequential tree re-weighted algorithm.**

- 0. Generate spanning trees that satisfy  $\sum \rho_i \theta_i = \bar{\theta}$ , where  $\rho_i$  is the  $i^{th}$  spanning tree probability.
- 1. Select some order for nodes and edges in  $V \cup E$ . For each element  $w \in V \cup E$ , find all trees containing  $w$ . If there is more than one tree, then do the following:
  - For all trees re-parameterize  $\theta$  such that values  $\theta$  gives the correct min-marginals [16] as follows:

$$\phi_s(\theta) = \theta_s^n + \text{const}_s,$$

$$\phi_{st}(\theta) = \theta_s^n + \theta_{st}^n + \theta_t^n + \text{const}_{st}.$$

– “Averaging” operation:

- \* If  $w = s$  is a node in  $V$  then compute  $\theta_s^{n+1} = \frac{1}{\rho_s} \sum_{All\ Trees} \rho_i \theta_{si}^n$ , where  $\rho_s$  is the node appearing probability.
- \* If  $w = (s, t)$  is an edge in  $E$  then compute  $M_{t \rightarrow s}^{n+1} = \frac{1}{\rho_{st}} \sum_{v_t} \rho_{st} (\theta_s^{n+1}(v_s) + \theta_{st}^n(v_s, v_t) + \theta_t^{n+1}(v_t))$ ; set  $\theta_{st}^{n+1}(v_s, v_t)$  such that  $\theta_s^{n+1}(v_s) + \theta_{st}^{n+1}(v_s, v_t) + \theta_t^{n+1}(v_t) = M_{t \rightarrow s}^{n+1}$ .
- 2. Check whether the message  $M_{t \rightarrow s}$  converges; if yes, terminate, otherwise go to step 1.

Similar to the VBP approach, the sequential method in Sequential tree re-weighted algorithm works by message passing. Specifically, for each directed edge  $(t \rightarrow s) \in E$ , a message  $M_{t \rightarrow s}$  is computed and passed. The following choice is needed to run the algorithm above:

- choose spanning tree probability distribution  $\rho$  such that  $\sum_i \rho_i = 1$ . Also, each edge must be covered by at least one chain.
- select the computational order of the different nodes.

Importantly, the sequential belief propagation algorithm only requires half as much memory as VBP. This is because VBP requires bi-direction message storing, i.e.  $M_{st}$  and  $M_{ts}$ . However, the new approach needs to store either  $M_{st}$  or  $M_{ts}$  due to the node pre-ordering.

### Tree Agreement Condition

As the algorithms above reduce to the sum product or the max product algorithm for tree networks, they are exact for tree networks. However, for mesh networks, this can no longer guarantee the output of the correct MAP assignment. This is because the algorithmic derivation is based on approximating the distribution of mesh networks via upper bounds, and it is straightforward to demonstrate problems on which it specifies an incorrect MAP estimate. However, one can always use the following condition to validate the tightness of bounds after the associated VBP algorithm or sequential VBP algorithm is applied. This condition is called the tree agreement condition.

**Definition** We say the VBP decomposition satisfies the tree agreement if different spanning trees, which form the upper bound of the mesh network, share a common optimal result when the BP algorithm in 5.4 is run on each of them.

### 5.2.4 Variational Belief Propagation for AC Power System State Estimation

In this subsection, we explain how to apply the VBP algorithm in (5.6), (5.7), and in the sequential tree re-weighted algorithm to the posterior probability distribution in (5.2) in an AC power system state estimation setting. In particular, we explain what form is used for priori state distribution  $P(\mathbf{v})$  and conditional distribution  $P(\mathbf{z}|\mathbf{x})$ , so that VBP can be applied. First, we can use the uniform prior probability distribution for voltage magnitude, i.e.  $|v_i| \in [0, 10]$ , and the phase angle, i.e.  $\delta_i \in [0, 2\pi]$ , instead of historical data [92, 114, 115] in order to avoid bias. This is because maximum a posteriori probability (MAP) estimation can be viewed as a regularization of currently-used maximum likelihood estimation (MLE). By relaxing the regularization conditions on non-informative prior probability distributions, we can prevent bias in the

Bayesian framework. With  $p(\mathbf{v})$  defined above and  $p(\mathbf{z})$  as a constant, we show next the forms of  $p(\mathbf{z}|\mathbf{v})$  needed to calculate posterior probability distribution  $p(\mathbf{v}|\mathbf{z})$  in (5.2).

Since VBP is built on the exponential family, we use the additive Gaussian noise in (5.1) to represent

$$p(\mathbf{z}|\mathbf{v}) \sim \exp \left\{ - \sum_i (z_i - h_i(\mathbf{v}))^2 / \sigma_i^2 \right\}. \quad (5.8)$$

Without loss of generality, we omit the variance  $\sigma_i^2$  in the rest of the paper for simplicity. In the following, we specify each measurement type.

**The complex valued power flow (pf) measurement on branch (edge)  $s - t$  near bus  $s$**

$$p(z_i^{pf}|\mathbf{v}) \sim \exp \left\{ - \sum_i \left| z_i - (v_s - v_t) Y_{st}^* v_s^* \right|^2 \right\}. \quad (5.9)$$

This form can easily be extended into real power measurements and reactive power measurements.

**The voltage magnitude (vm) on bus  $s$**

$$p(z_i^{vm}|\mathbf{v}) \sim \exp \left\{ - \sum_i \left( z_i - (v_s v_s^*)^{\frac{1}{2}} \right)^2 \right\}. \quad (5.10)$$

**The voltage phase angle (va) on bus  $s$**

$$p(z_i^{va}|\mathbf{v}) \sim \exp \left\{ - \sum_i \left( z_i - \tan^{-1} \frac{\text{Im}(v_s)}{\text{Re}(v_s)} \right)^2 \right\}. \quad (5.11)$$

The probability distribution functions associated with the measurement types above satisfy the pairwise Markov Random Field representation requirement of the SE problem in (5.3) and (5.6), etc. Now we discuss how to deal with a power injection measurement that violates the pairwise requirement.

### The power injection into bus $s$

$$p(z_i^{\text{pinj}}|\mathbf{v}) \sim \exp \left\{ - \sum_i \left| z_i - \sum_{t \in \mathcal{N}(s)} (v_s - v_t) Y_{st}^* v_s^* \right|^2 \right\} = \exp\{-T\}, \quad (5.12)$$

where  $T$  equals

$$\begin{aligned} \sum_i \left| z_i - \sum_{t \in \mathcal{N}(s)} (v_s - v_t) Y_{st}^* v_s^* \right|^2 &= \sum_i \left\{ z_i - \sum_{t \in \mathcal{N}(s)} (v_s - v_t) Y_{st}^* v_s^* \right\} \left\{ z_i - \sum_{k \in \mathcal{N}(s)} (v_s - v_k) Y_{sk}^* v_s^* \right\}^* \\ &= \sum_i \left\{ |z_i|^2 - z_i \sum_{k \in \mathcal{N}(s)} (v_s^* - v_k^*) Y_{sk} v_s - z_i^* \sum_{t \in \mathcal{N}(s)} (v_s - v_t) Y_{st}^* v_s^* + \sum_{t \in \mathcal{N}(s)} \sum_{k \in \mathcal{N}(s)} |v_s|^2 Y_{st}^* Y_{sk} (v_s - v_t)(v_s^* - v_k^*) \right\}, \end{aligned} \quad (5.13)$$

which can be abstracted as

$$\sum_s \left\{ \theta_s(v_s) + \sum_{t \in \mathcal{N}(s)} \theta(v_s, v_t) + \sum_{t \in \mathcal{N}(s)} \sum_{k \in \mathcal{N}(s)} |v_s|^2 Y_{st}^* Y_{sk} v_t v_k^* \right\}, \quad (5.14)$$

By including the multiplication of three different state variables,  $|v_s|^2 Y_{st}^* Y_{sk} v_t v_k^*$  violates the pairwise Markov random assumption required by the VBP algorithm. To resolve the problem, dummy variable vector  $\mathbf{w}_{stk} \triangleq [w_{stk}^{(1)}, w_{stk}^{(2)}, w_{stk}^{(3)}]^T$  and the corresponding  $\phi(\mathbf{w}_{stk}) \triangleq |v_s|^2 Y_{st}^* Y_{sk} v_t v_k^*$  are defined for regularization. In this way, the original problem of either maximizing (5.12) or equivalently minimizing (5.14) can be regarded as a problem of minimizing the following regularized problem

$$\sum_s \left\{ \theta_s(v_s) + \sum_{t \in \mathcal{N}(s)} \theta_{st}(v_s, v_t) + \sum_{t \in \mathcal{N}(s), k \in \mathcal{N}(s)} \phi_{stk}(\mathbf{w}_{stk}) + 10|w_{stk}^{(1)} - v_s|^2 + 10|w_{stk}^{(2)} - v_t|^2 + 10|w_{stk}^{(3)} - v_k|^2 \right\} \quad (5.15)$$

where 10 is adopted as the penalty coefficient.  $|w_{stk}^{(1)} - v_s|^2 = |w_{stk}^{(1)}|^2 - w_{stk}^{(1)} v_s^* - w_{stk}^{(1)*} v_s + |v_s|^2$  results in a pairwise expression, with a similar extension to regularization on  $v_t$  and  $v_k$  in (5.15).

Now all measurement types can be written in a pairwise Markov form as in (5.3). Furthermore, since the prior state distribution is uniform, which is uninformative, maximizing the posterior distribution  $p(\mathbf{v}|\mathbf{z})$  is equivalent to maximizing the conditional distribution  $p(\mathbf{z}|\mathbf{v})$  according to Bayes' rule in (5.2), given  $\mathbf{z}$ . We can apply the VBP algorithm in (5.7), or the sequential tree re-weighted algorithm to (5.8) (a compact summary of different measurements types (5.9), (5.10), (5.11), and (5.12)), to obtain the maximum a posteriori joint probability distribution of  $p(\mathbf{v}|\mathbf{z})$  or the probabilistic state estimate for all buses. Similarly, if marginal distribution is preferred, one can apply (5.6) to (5.8).



**Remark** One may also be concerned about the bias of the prior information, which may cause states to look good when bad data appears. However, when using a Bayesian Approach, it is assumed that the statistical model is correct, or that bad data detection and the filtering process are conducted before the proposed inference is run.

### Algorithm Summary

We provide here a summary of the proposed graphical model-based SE in Fig.5.3 with the following steps:

- Step 1: randomly generate spanning trees of the graph based on the physical power system topology according to [112]. (If a sequential tree re-weighted method is used, add the constraint that convex combination of the tree parameters equals to the mesh parameters.) Then calculate the appearance probability  $\rho_{st}$  of each edge according to (5.5);
- Step 2: use the measurement values of  $z$  in (5.9), (5.10), (5.11), and (5.12) to obtain the joint probability function (5.8) over state  $v$ ;
- Step 3: initialize the state variables; (i.e., ones for voltage magnitudes and zeros for voltage phase angles.)
- Step 4: with the regularization (5.15) of the power injection measurements, apply the result above to the VBP algorithm in (5.6), (5.7), or the sequential tree re-weighted algorithm for state variable updates;
- Step 5: repeat Step 4 until the state variables converge.

### 5.2.5 Illustration on a Small Example

Fig.2a represents a three-bus system to which the proposed VBP is applied. In this example, we assume that we have voltage measurement  $z_1^{vm}$  on bus 1, voltage phase angle measurement  $z_2^{va}$  on bus 2, complex power flow measurement  $z_3^{pf}$  on branches 2 – 3 near bus 2, and a complex power injection measurement  $z_4^{pinj}$  on bus 3.

- Step 1: because there are three possible spanning trees shown in Fig.2b,c,d with equal edge probability,  $\rho_{12} = \rho_{23} = \rho_{13} = 2/3$ ;

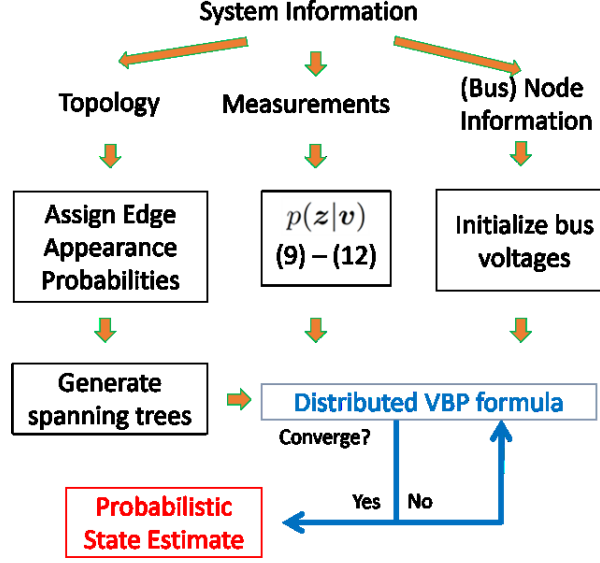


Figure 5.3: Flow chart for the proposed approach.

- Step 2: write  $p(\mathbf{v}|\mathbf{z})$  as

$$p(v_1, v_2, v_3|\mathbf{z}) \sim \exp \left\{ - \left( z_1^{\text{vm}} - (v_1 v_1^*)^{\frac{1}{2}} \right)^2 - \left( z_2^{\text{va}} - \tan^{-1} \frac{\text{Im}(x_2)}{\text{Re}(v_2)} \right)^2 - \left| z_3^{\text{pf}} - (v_2 - v_3) Y_{ij}^* v_2^* \right|^2 - \left| z_4^{\text{pinj}} - [(v_3 - v_1) Y_{31} + (v_3 - v_2) Y_{32}] v_3^* \right|^2 \right\} \quad (5.16)$$

and use the measurement value  $z_i$  and the admittance value  $Y_{ij}$ , leading to

$$p(v_1, v_2, v_3) \sim \exp \left\{ \theta_{v_1}(v_1) + \theta_{v_2}(v_2) + \theta_{v_1, v_2}(v_1, v_2) + \theta_{v_2, v_3}(v_2, v_3) + \theta_{v_1, v_3}(v_1, v_3) + \phi_{v_1, v_2, v_3}(v_1, v_2, v_3) \right\}; \quad (5.17)$$

- Step 3: initialize the voltage belief on each bus with magnitude one and phase angle zero;
- Step 4: with regularization (5.15) on  $\phi_{v_1, v_2, v_3}(v_1, v_2, v_3)$ , apply the result above to VBP algorithm in (5.6), (5.7) or the sequential tree re-weighted algorithm for message  $(M_{ij})$  passing;
- Step 5: repeat Step 4 until the state variables converge.
- Step 6: check Tree Agreement Condition for optimality.

Having formulated the VBP algorithm for SE and illustrate it in a toy example, we will test it against other methods in standard IEEE test systems in the next section.

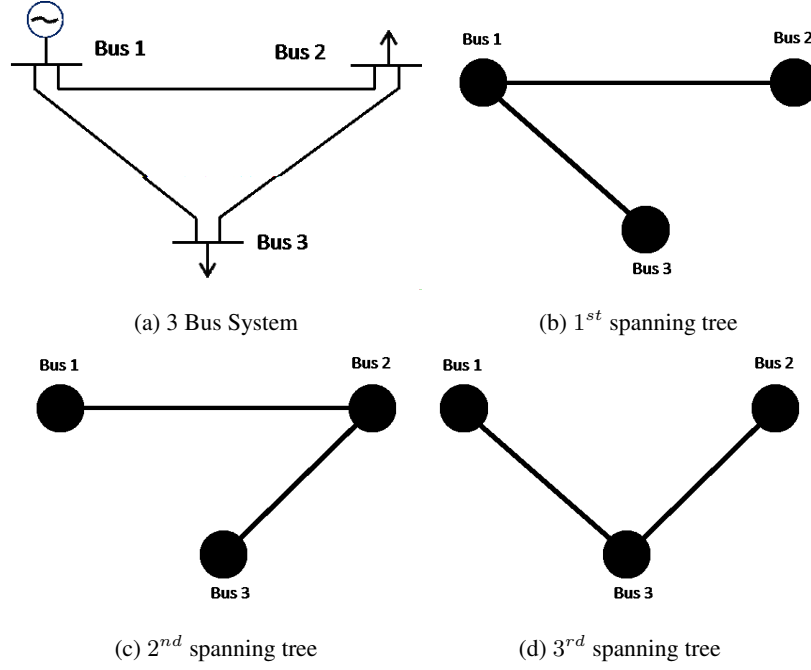


Figure 5.4: Generation of spanning trees to obtain the value of  $\rho_{ij}$

### 5.3 Simulation Results

In this section, we simulate and verify the significantly improved performance of the proposed graphical model-based SE over the WLS SE. The simulations are implemented on IEEE standard test systems for IEEE 9, 14, 30, 39, 57, 118, and 300-bus test systems. Similar performance improvements are observed in all the test systems. In particular, the 14-bus system demonstrates error domain improvements. The 30-bus system and 118-bus system are simulated for state domain comparisons. The 9-bus to 300-bus systems are simulated to show improvements in computational time. The data preparation is similar to section 3.3.6.

#### Numerical Results

We demonstrate the performance of the graphical model-based approach by conducting SE with Gaussian noise. For comparison purposes, Newton's method is used to obtain a locally optimal estimate. Also, the true states from the optimal power flow result are also displayed for reference.

## Accuracy

Fig.5.5 shows 30 simulations for the weighted residual sum of squares (WRSS) error in the 14 bus system with the metric WRSS defined as follows:

$$\text{WRSS} = \sum_{i=1}^m \left( \frac{z_i - h_i(\hat{v})}{\sigma_i} \right)^2, \quad (5.18)$$

where  $m$  is the total measurement number.

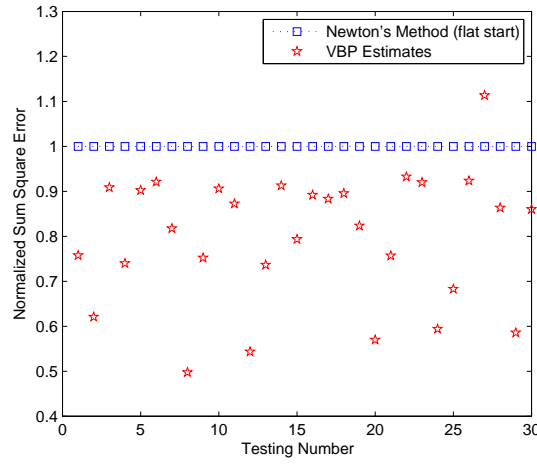


Figure 5.5: Normalized sum square errors (14 bus system).

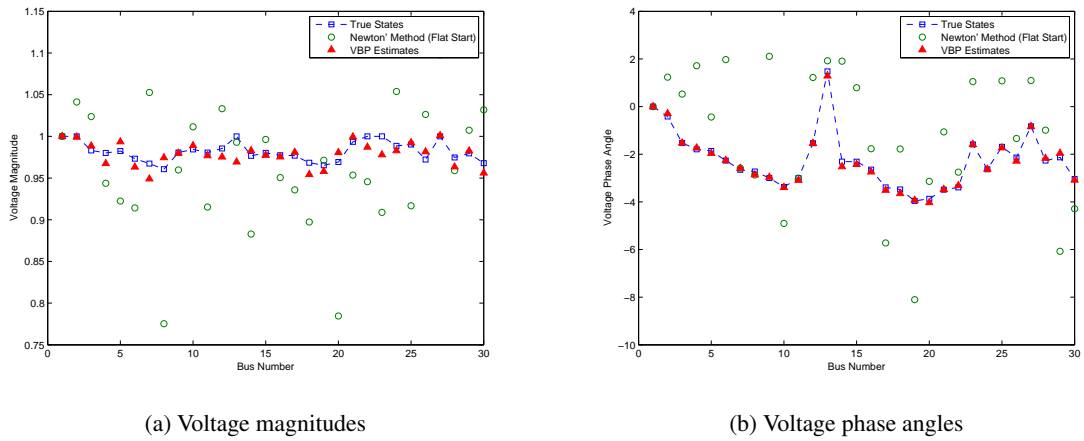


Figure 5.6: Results obtained from the IEEE 30 bus.

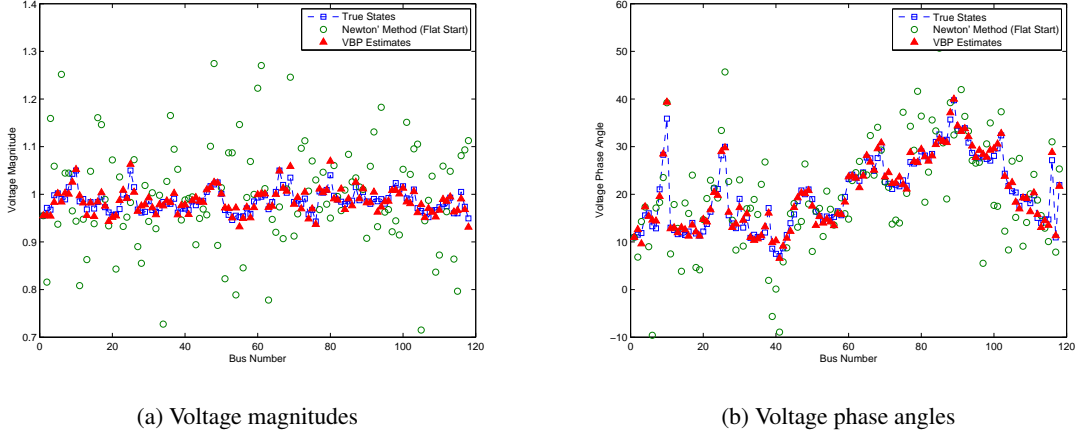


Figure 5.7: Results obtained from the IEEE 118 bus.

In this illustration, the  $x$  axis is the simulation test number. The  $y$  axis is the metric WRSS (normalized with respect to Newton's method). It can be seen that the graphical model-based approach can reduce the error by 20% on average when compared to Newton's method with a flat start. The reason for simulating a flat start is to mimic the smart grid environment with a strong state variation, making the typical initial guess method (from the last static state estimate) less informative for the new estimation process. A more than 50% improvement is achieved at simulation numbers 8, 12, 20, 24, and 29. These facts lead to a natural interpretation of the proposed SE procedure: the possibility of the graphical model-based approach coming closer to the global optimum is greatly increased since the results do not rely heavily on the initial guess. The flat start seems to perform better only in simulation 27, because it successfully helps Newton's method reach the global optimum, over which the VBP cannot improve. The VBP fails in this particular simulation because the objective of VBP is MAP, not MMSE. However, such a case rarely occurs in a true smart grid scenario where a flat start performs poorly. Note that the proposed distributed method is superior to the distributed implementation of Newton's method as well. This is because the centralized method is equal to or better than its distributed realization.

Fig.5.6a and Fig.5.6b show the estimates of voltage magnitudes and phase angles of the proposed approach for the IEEE 30-bus system. It can be seen from the two plots that the graphical model-based approach (red solid triangles) is superior to Newton's method with a flat start (green circles) since it provides an estimate much closer to the true state. Fig.5.7a and Fig.5.7b also show a more accurate state estimation for the VBP method with respect to voltage magnitudes and phase angles for the 118 bus system. In sum-

mary, since our priori state distribution is non-informative, the proposed VBP estimate is suitable for the future smart grid, in which priori information may be less helpful. In addition, we will show later that VBP has another favorite property—linear scalability—that is unavailable with Newton’s method.

## Computational Cost

A plot of CPU time is provided in Fig.5.8 for a computational time comparison of the regular WLS and the graphical model-based methods. The  $x$  axis is the test case bus number. The  $y$  axis stands for the  $\log(\text{CPU time})$ . All simulations are obtained using MATLAB on an Intel Core 2 CPU with 3GB RAM. The computational time needed by the graphical model-based method grows linearly, and is much lower than that needed by the regular WLS method from MATPOWER where the matrix inversion has a computational complexity of  $O(n^2 \log n)$ , growing exponentially. As a result, computational cost is reduced via distributed algorithm. This confirms VBP’s scalable feature, which is the key to the design of the future Wide Area Monitoring, Control and Protection (WAMPAC) systems.

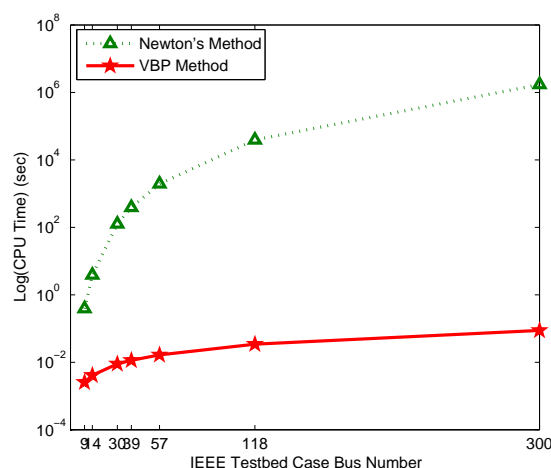


Figure 5.8: CPU time comparison.

## Improvement with Sequential Tree Re-weighted Algorithm

To illustrate the superiority of the sequential tree re-weighted algorithm (STRW) over the ordinary tree re-weighted algorithm (TRW), or VBP, we also compare the two algorithms in Table 5.1. The probability of

optimality is checked by the tree agreement condition. As one can observe, the sequential tree re-weighted

Table 5.1: Convergence

Method	Convergence probability	Probability of optimality
TRW	93%	86%
STRW	100%	97%

algorithm can solve the convergence problem for the VBP algorithm. The probability of optimality is also guaranteed with a probability of 97%. Finally, it is worth noting that the sequential algorithm needs about half the memory of the ordinary VBP approach, which is especially important for computation in large networks.

## 5.4 Chapter Summary

In this chapter, we propose for the first time a probabilistic model for the whole power grid. This type of model is valuable when labeled historical data is not available, but historical state distribution is. Unlike many cyber models that do not account for physical constraints, and unlike deterministic engineering modeling defined solely by physical laws, this chapter combines the two into a single cyber-physical graphical model. In particular, this chapter introduces a distributed graphical model approach for AC power system state estimation that uses embedding techniques; as such, it is sufficiently scalable to account for low voltage level distributed technologies. Mature graphical model inference tools, such as variational belief propagation, are subsequently applied. To improve our approach’s performance, a sequential tree re-weighted belief propagation algorithm is introduced to solve the convergence problem. We demonstrate that the proposed approach can significantly reduce SE error. Furthermore, its linear computational time is attractive due to network scalability needs, and such scalability is vital for the future large-scale smart grid. This also opens the door for other CPS networks in which scalable probabilistic analysis according to physical laws is preferred.

As a highlight of this work, the future grid structure for SE will include a mesh network (transmission grid) in the center and many tree networks (distribution grids) on the borders. Therefore, using an algorithm that combine BP and VBP algorithms to suit a partially mesh and partially tree network is vital for the future

smart grid. Fig.5.9 illustrates such a need, where local service entities are cooperating with the backbone transmission network on message passing.

### Physical and Information Network Graphs Today

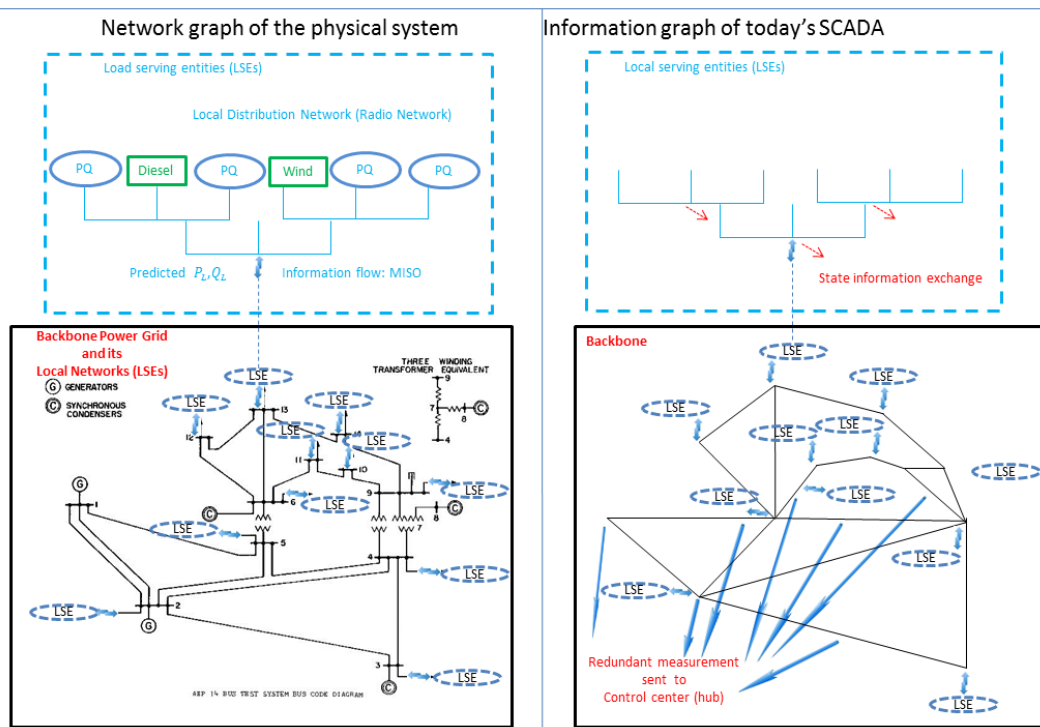


Figure 5.9: Centralized and decentralized BP



## Chapter 6

# Comparison of the Various Proposed Methods using IEEE 30-bus systems

In different SCADA centers around the world, the software, the availability of historical data, and computational resources differ dramatically. Therefore, we consider the different settings below in order to improve current static state estimation. This chapter aims to illustrate the advantage of each method in different SCADA center setups. A flow chart is drawn at the end to help decide the SE method of choice at the SCADA center.

In the following, we list various simulation scenarios. For the sake of consistency, an IEEE 30-bus system is used.

- A SCADA center without historical data where the measurement noises are regular Gaussian noise.
- A SCADA center without historical data where the noise is very small.
- A SCADA center with some historical data about the estimated states.
- A SCADA center with long labeled historical data. Here “labeled” means that the historical measurement data are either labeled or associated with a state estimation.
- A SCADA center with long labeled historical data. However, undetected bad data and malicious attacks may exist in the data set.

- A SCADA center with long labeled historical data. The data rate is exceptionally fast in order to capture dynamics.
- A SCADA center with poor historical data.

## 6.1 Simulation Scenario I: A SCADA Center without Historical Data.

Some SCADA centers do not have historical data. This happens especially with a new SCADA center. In this case, any improvement over Newton's method is desirable. Since there are no historical data, we will only compare:

- Newton's method with a flat start
- Centralized SDP approach
- Distributed SDP approach
- No-noise-SDP approach
- Lower bound

The simulation setup is similar to that in Section 3.3.6. To evaluate the performances for accuracy, we first compute the weighted residual sum of squares error (WRSS)

$$\text{WRSS} = \sum_{i=1}^m \left( (z_i - \text{tr}(T_i W) - M_i f) / \sigma_i \right)^2, \quad (6.1)$$

where  $m$  is the total measurement number. Here, we conduct such a comparison with an IEEE 30 bus system multiple times. Fig.6.1 illustrates the 30 simulation results; the  $x$  coordinate represents the test index number and the  $y$  coordinate represents the metric of WRSS metric defined above. To display the relative improvements, we normalize the WRSS with respect to Newton's method with a flat start.

From Fig.6.1, we can see that

- The centralized SDP approach (red solid line) and distributed SDP approach (dot star line) achieve a similar performance. Their curves are also very close to the lower bound, leading to approximately global optimum results.

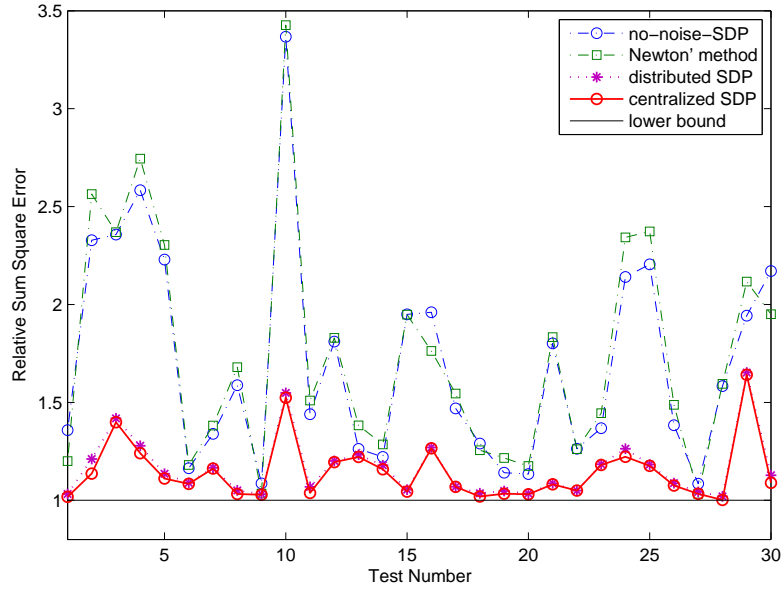


Figure 6.1: Simulation scenario II: accuracy comparison (measurement domain).

- The WLS method (green dash square line) performs worse than the centralized and distributed SDP approaches. However, WLS does hit the global optimum in tests 10 and 27. In the other tests, the WLS approach hits the local optimum and stops.
- The No-noise-SDP approach does not work well because the noise is not close to zero.

Since the centralized SDP and distributed SDP perform best in minimizing error, we compare, in Fig.6.2, their computational costs with respect to that of WLS. Great acceleration can be observed in the proposed distributed algorithm for the distributed parallel SDP approach, especially in large systems. In these systems, thanks to the fast rate of convergence, the distributed method saves much more time than the extra time needed to compute the dual variable  $\lambda$  with respect to the centralized SDP SE. As a result, although distributed SDP SE is consistently more time-consuming than the traditional WLS method, it nevertheless reduces the time dramatically from that required by centralized SDP by flattening the curve in Fig. 6.2. Admittedly, for a small system such as 4-bus system, the distributed (parallel) SDP method requires more time than the distributed one. This is because, for a small system, dividing the network into multiple sub-networks cannot substantially lower the optimization variable number. Instead, dual variables' computation may produce heavy burdens for the algorithm. Therefore, Fig. 6.2 suggests distributed computation for

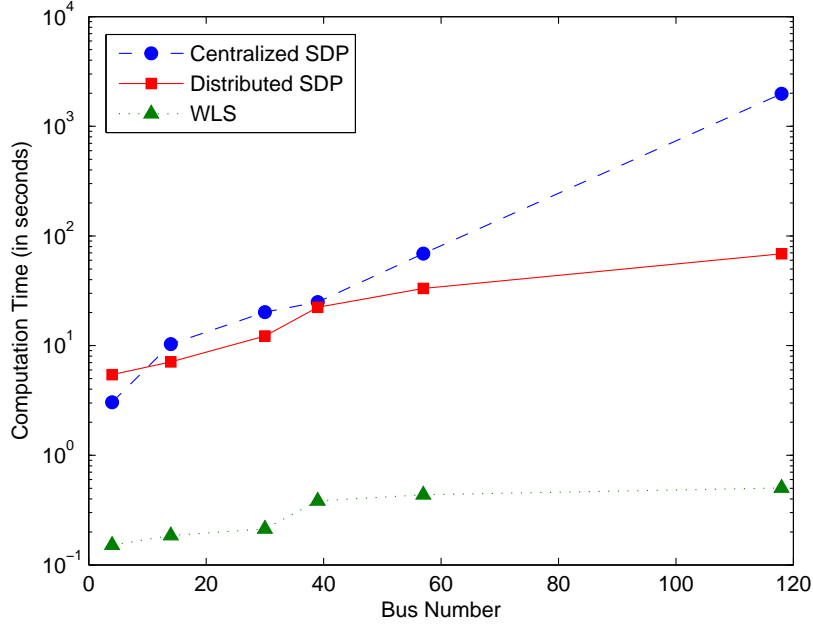


Figure 6.2: Simulation scenario II: computational time comparison.

large systems and centralized computation for small systems.

In conclusion, when no historical data are available, the distributed parallel SDP approach is preferable for large systems.

## 6.2 Simulation Scenario II: A New SCADA Center without Historical Data where the Noise is Very Small.

In this section, the simulation setup is similar to the setup in the last section except that here we use a much smaller noise: the standard deviation is 0.1% of the measurement value. Since there is no historical data, we will compare

- Newton's method with a flat start.
- Centralized SDP approach
- Distributed SDP approach

- No-noise SDP approach
- Lower bound

From Fig.3.3, we see that the No-noise SDP approach is the closest to the lower bound. Therefore, this method is preferable when the noise is small and no historical data are available.

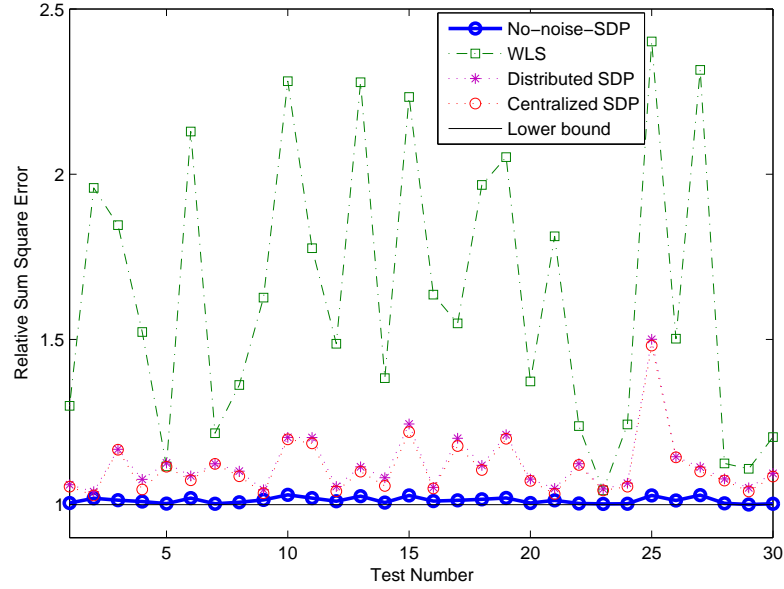


Figure 6.3: Simulation scenario II: accuracy comparison (measurement domain).

### 6.3 Simulation Scenario III: A SCADA Center with Some Historical Data about the Estimated States.

In this section, some historical data means that the SCADA center has a priori distribution of the state variables. However, no detailed information about the historical measurements and states is available.

Since the centralized and distributed SDP approaches have similar accuracy, we only show the simulation result for the centralized SDP approach in this section, and we call it the SDP approach for short. Also, since the noise level is set at ordinary (not close to zero), we already know because of the simulation results in Simulation Scenario I that the SDP approach is better than the no-noise SDP approach. Finally, data-driven

SE approach is not included here due to the unavailability of labeled measurement-state pairs. Therefore, one can neither conduct the static data-driven SE nor the dynamic data-driven SE. We compare

- Newton's method with a flat start.
- SDP approach.
- Belief Propagation approach.

In simulation scenario I and II, all the approaches being compared focus on minimizing the sum square error in the measurement domain. Therefore, it is fair to compare all of them in the measurement domain. However, the belief propagation approach in this section focuses on maximum-a-posterior distribution of the state. Since the ultimate goal of SE is to estimate state, we compare differences in state domain in this section. The simulation setup is similar to that in Section 5.3.

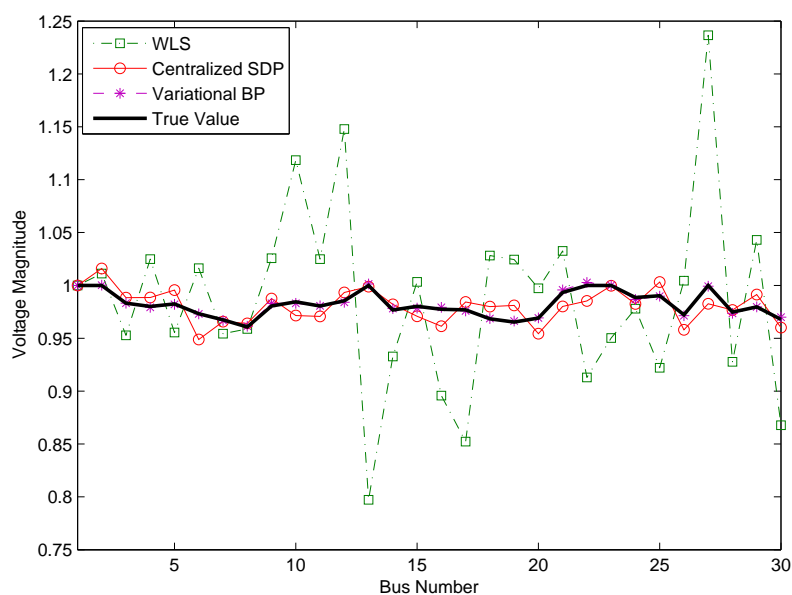


Figure 6.4: Simulation scenario III: accuracy comparison (state domain).

From Fig. 6.4 and Fig. 6.5, we see that the Belief Propagation approach generates a state that is closer to the true state, when comparing to the SDP method and WLS method with Newton's method.

Speedup by the BP approach is illustrated in Fig.6.6.

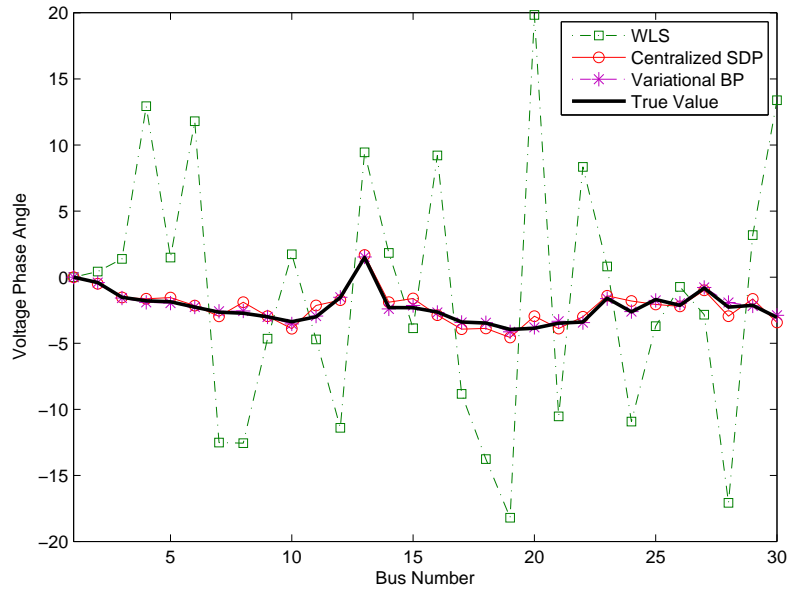


Figure 6.5: Simulation scenario III: accuracy comparison (state domain).

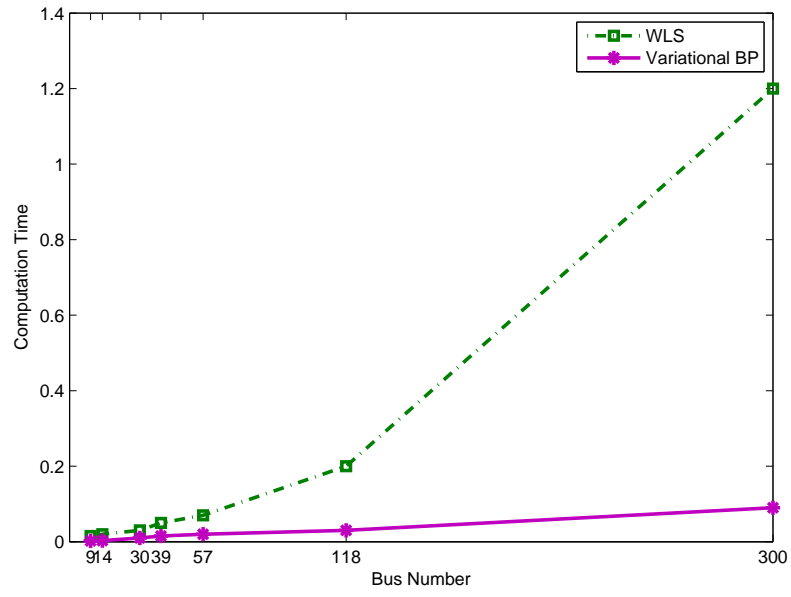


Figure 6.6: Simulation scenario III: computational time comparison.

## **6.4 Simulation Scenario IV: A SCADA Center with Long Labeled Historical Data. However, Undetected Bad Data and Malicious Attacks May Exist in the Data Set.**

The simulation setup is similar to that in Section 4.3.3 for Robust Data-Driven SE. Bad data are occasionally added into both the historical data and the current measurements. We compare:

- Newton's method with bad data detection
- SDP approach with bad data detection
- Belief Propagation approach
- Static Data-Driven SE
- Robust Static Data-Driven SE
- Dynamic Data-Driven SE

Again, we are going to make comparison in the state domain.

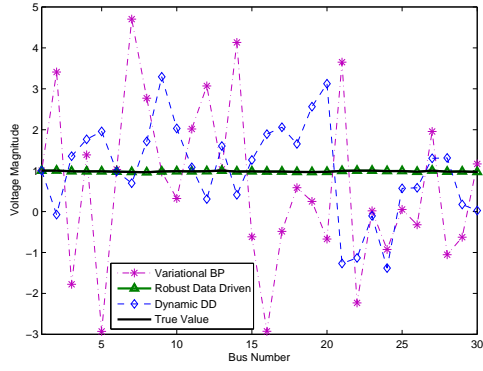
From Fig.6.7, robust data-driven SE is the best. Since the belief propagation approach, static data-driven approach, and dynamic data-driven SE are not designed to deal with bad data, they perform more poorly here. For the other approaches in the Fig.6.7, robust data-driven approach is the best.

## **6.5 Simulation Scenario V: A SCADA Center with Labeled Historical Data. The Data Rate is Very Fast.**

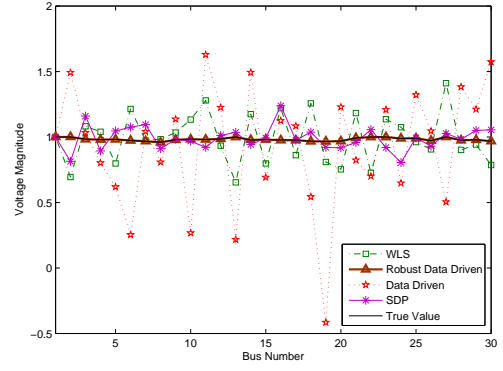
In this part, we assume that the measurement data rate is fast and at 1 result per second. We compare:

- Newton's method with a flat start.
- SDP approach.
- Belief Propagation approach.

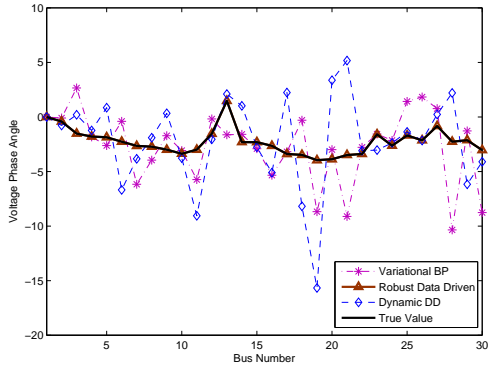




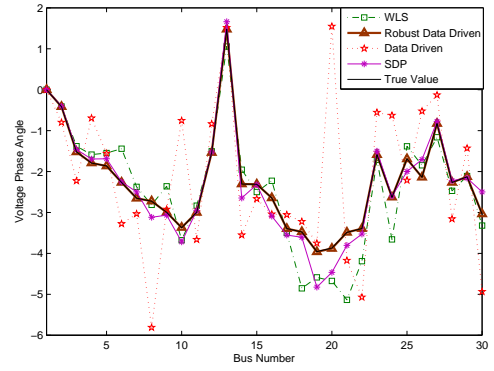
(a) Voltage Magnitude Comparison 1



(b) Voltage Magnitude Comparison 2



(c) Voltage Phase Angle Comparison 1

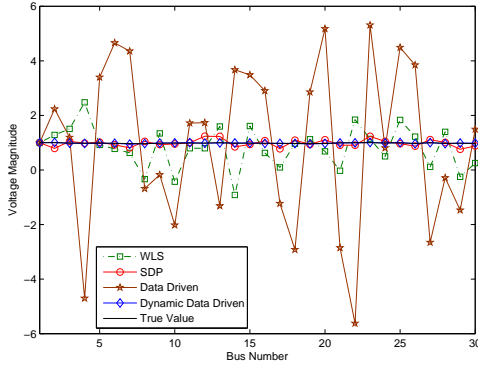


(d) Voltage Phase Angle Comparison 2

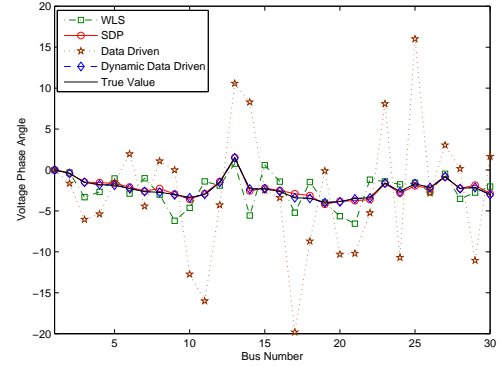
Figure 6.7: Simulation scenario IV: accuracy comparison (state domain).

- Static Data-Driven SE.
- Dynamic Data-Driven SE.

The comparison is in the state domain.



(a) Voltage Magnitude Comparison



(b) Voltage Phase Angle Comparison

Figure 6.8: Simulation scenario V: accuracy comparison (state domain).

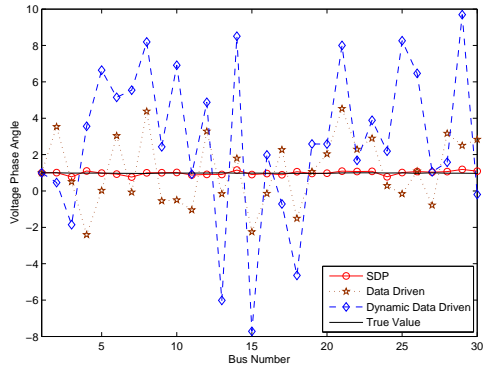
From the simulation above, we can see that the dynamic data-driven approach performs the best.

## 6.6 Simulation Scenario VI: A SCADA Center with Poor Historical Data.

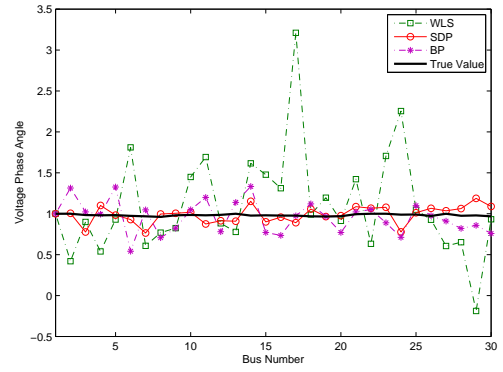
The simulation setup is similar to Section 4.3.3 for Data-Driven SE. However, we intentionally remove historical data and give wrong priori distribution on the state. we compare:

- Newton's method with a flat start.
- SDP approach.
- Belief Propagation approach
- Static Data-Driven SE
- Dynamic Data-Driven SE

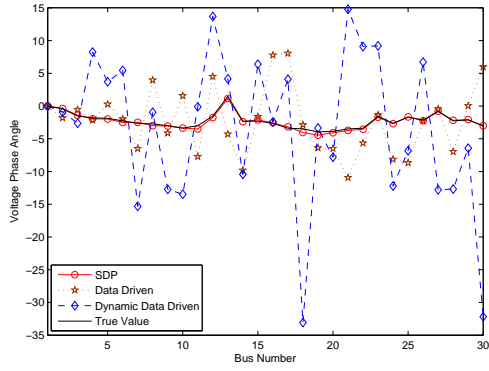
From the simulation, we can see that



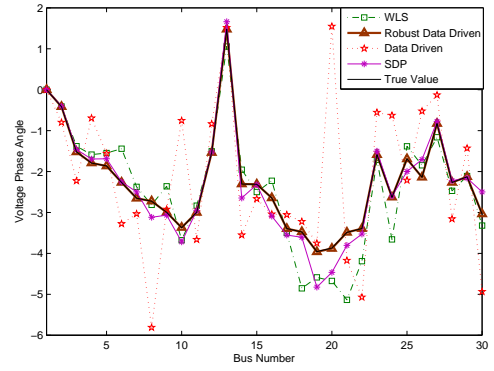
(a) Voltage Magnitude Comparison 1



(b) Voltage Magnitude Comparison 2



(c) Voltage Phase Angle Comparison 1



(d) Voltage Phase Angle Comparison 2

Figure 6.9: Simulation scenario VI: accuracy comparison (state domain).

- the Static Data-Driven approach is bad occasionally due to loss of similar historical data;
- the dynamic Data-Driven SE fails due to the inconsecutive data for learning;
- the Belief Propagation approach fails because the priori is wrong;
- the BP is not working well as the historical data constructed priori state distribution is wrong;
- the SDP approach is the best. This is because the SDP-based approach has the advantage of being independent of historical bad data or bias because it does not use historical information.

## 6.7 Chapter Summary

In this chapter, various proposed SE approaches are applied to IEEE 30-bus system for comparison. The results show that a SCADA center should use a strategies depending on: 1) noise level, 2) historical data availability, 3) historical data quality, 4) scalability need, and 5) the desire for a probabilistic state estimate.

In particular, the simulation results recommend that the flow chart in Fig.6.10 be employed to select a suitable SE approach.

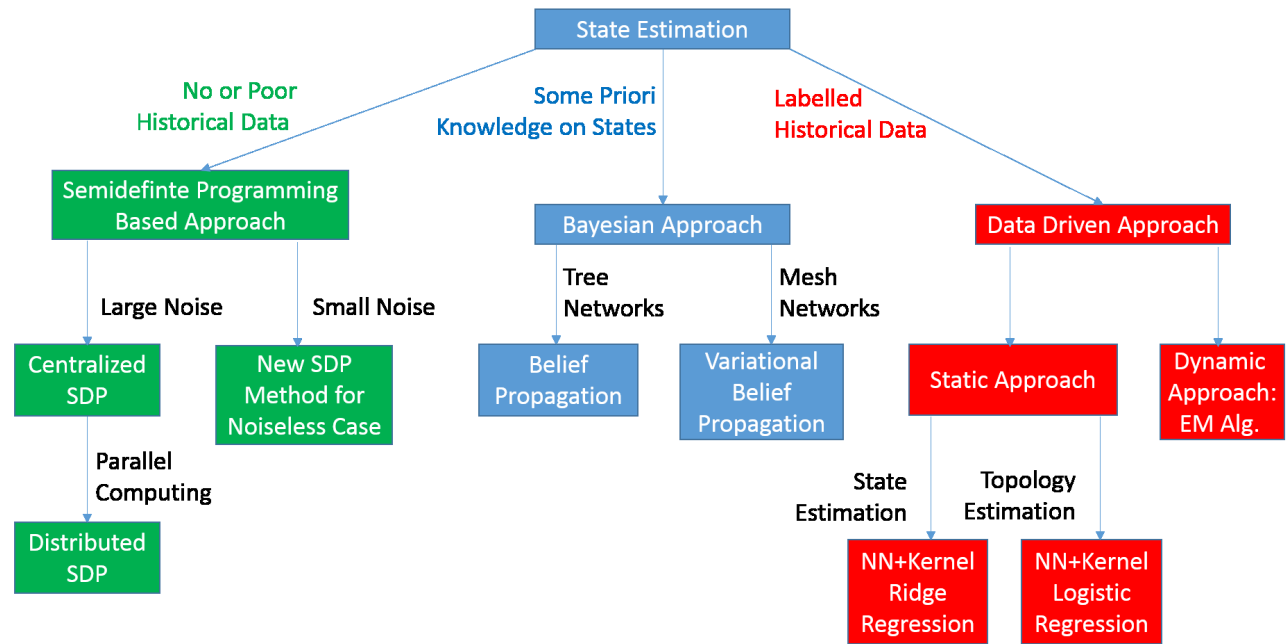


Figure 6.10: Decision tree for the various new SE approaches.

## Chapter 7

# Conclusions

This thesis attempts to resolve, via embedding techniques, the challenges produced by sources of randomness in the power grid that cause uncertainties and vulnerability in power system state estimation (SE). By exploring the convex optimization method and machine learning approaches to enhance the robustness, accuracy, and speed of current SE, we find a way to deal with new power producers such as newly built intermittent distributed wind generators, electric vehicle charging, and accidental transmission line failures.

There are three scenarios where SE can be improved for power system monitoring. First, when reliable historical data is unavailable, we propose embedding state space to a high dimensional space in order to deal with the nonconvexity of the static AC SE problem. This embedding process allows a linear model and enables a convex semidefinite programming (SDP) representation of the problem. But despite the benefit of working well for backbone power grids with a small number of buses, this method makes the SDP method increasingly time-consuming as the network size grows, so that a centralized solver will be infeasible for large networks. To address this restriction, we introduce a distributed algorithm to restructure the centralized SDP computation into local SDP problems without compromising the accuracy. Finally, we propose an alternative SDP-based approach to convexify the SE problem exactly when no noise appears. Even when noise is present, such a method still works well since the presence of noise will only mildly perturb the optimal result. Simulations show that the proposed SDP-based SE approaches can significantly reduce the large residuals common in conventional static SE for large scale electric power systems and achieve a near-global optimum. On average, our proposed SDP approach can improve on the industrial approach by 30% on average.

In the second scenario, we consider using historical data to deal with new uncertainties within grids associated with the intermittent generation of renewables, etc. The idea is motivated by the recent success of applying data-driven approaches to other cyber systems. To first refine the historical data for use, we propose conducting a nearest neighbors search due to the periodic nature of power system operation. After obtaining these neighbors, we use the associated measurements and their corresponding states, as well as the current measurement embedded in a larger feature space to obtain a good current state estimate. We conduct a parameter learning process to achieve good current SE based on a discriminative model called kernel regression that maps the historical data similar to current measurements into the current state. The training and validation stages of the mapping process lead to an efficient data-driven SE. In general, the proposed method is time-consuming when the historical data size is large, resulting in an excessive amount of time required to solve optimization problems. Therefore, the application of the proposed method can be computationally infeasible for large networks with massive amount of historical data. To reduce the time needed to search historical data, we propose utilizing the clustered nature of power system data. This enables a better streaming estimation process with real-time guarantees.

To conduct robust data-driven SE against topology changes, we notice that the proposed data-driven method above only uses current measurements and past measurement-state pairs. It does not use another important data set: current topology. Our refined idea is to use past information and current topology together for a similarity data points search. Then, as an extension to the data-driven SE approach, we propose conducting topology estimation within the same framework. Since the grid connection can be described by discrete and binary variables, we use kernel logistic regression instead of kernel ridge regression with the same embedding technique.

While historical data is useful for predicting the current state, the correlation between consecutive on-line data is another resource with which to enhance estimation accuracy. A static SE was previously preferable due to the slow updates of past measurement systems. However, we now have opportunities for dynamic tracking because faster sensors have been deployed in recent years [116, 117]. Therefore, we propose a kernel Kalman filter approach capable of dealing with nonlinearity in the measurement equations, via embedding, as well as capturing the correlation between two neighbor data points.

Our simulations show that the proposed data-driven approaches are particularly useful when reliable historical data is available. The improvement can be as high as 90% in the state domain.

Apart from utilizing historical data to deal with the uncertainties in power grid monitoring, we also need to address scalability issues as the grid grows larger. Such challenges are enormous as the industry paradigm shifts from a traditional centralized monitoring architecture-based deterministic model to a highly distributed interactive data and resource management-based probabilistic model. Therefore, to properly model a system's stochastic property and conduct efficient distributed SE, we propose a probabilistic graphical model description [65] of the electric power grid. Our graphical model is inspired by results from applying such models for uncertainty representation and computationally-tractable inferences in several other field [91, 92, 118].

We then conduct computationally-tractable distributed inferences [91, 92, 118], namely “Belief Propagation” over the graphical model and Variational Belief Propagation [17], which are based on embedding the true probability distribution into a larger space of distributions. Since the future power grid will have load serving entities with distributed renewable energy generations, an extension of the algorithm can be conducted on a partially mesh/partially tree network.

Our numerical results show that when distributed computation is available and the historical data reflects good priori on the state, the accuracy improvement is 80% on average. The improvement on computation time is higher, only  $\frac{1}{10}$  time of the current industrial approach is needed.



## .1 Other Measurement Type for Distributed Semidefinite Programming-Based State Estimation

This section discusses the inclusion of power flow measurements in the distributed algorithm.

**Power flow measurements** Similarly to the power injection expression (3.18), the power flow representation without considering substation topology ( $M_i \mathbf{f}$ ) can be formulated as:

$$s_{ij} = v_i l_{ij}^H = v_i (Y_{\text{bra}} \mathbf{v})_k, \quad (1)$$

where  $k$  indicates that branch  $ij$  is the  $k^{\text{th}}$  branch in the case file ( $M_i \mathbf{f}$  can easily be added when necessary due to its linear operation over the measurement formula.). As a result, the power flow vectors are:

$$\mathbf{s}_{\text{flow}} = \text{diag}\{D \mathbf{v} \mathbf{v}^H Y_f^H\} = \text{diag}\{T \mathbf{v} \mathbf{v}^H Y^H T^H\}, \quad (2)$$

$$\mathbf{p}_{\text{flow}} = \text{diag}\{D \mathbf{v} \mathbf{v}^H M^{\text{act}} T^H\}, \quad (3)$$

$$\mathbf{q}_{\text{flow}} = \text{diag}\{D \mathbf{v} \mathbf{v}^H M^{\text{rea}} T^H\}, \quad (4)$$

where  $D$  is a deterministic matrix mapping the bus voltages onto the branch voltages. As a result, our problem turns from (3.19) to:

$$\begin{aligned} \underset{W}{\text{minimize}} \quad & \sum_{i=1}^l \left[ (D_{i,:} T^{\text{act}} W D_{i,:}^T - z_i^{\text{act.pf}})^2 \right. \\ & \left. + (D_{i,:} T^{\text{rea}} W D_{i,:}^T - z_i^{\text{rea.qf}})^2 \right] \\ \text{subject to} \quad & W \succeq 0. \end{aligned}$$

where  $l$  is the branch power flow measurement number.  $T_{i,:}$  is the  $i^{\text{th}}$  row of matrix  $T$ .

The SDP SE above is equivalent to the following optimization, by adding equality constraints and auxiliary variables  $\mathbf{y}^{\text{act.pf}}$  and  $\mathbf{y}^{\text{rea.qf}}$ .

$$\begin{aligned} \underset{W}{\min} \quad & \sum_{i=1}^l \left[ (y_i^{\text{act.pf}})^2 + (y_i^{\text{rea.qf}})^2 \right] \\ \text{subject to} \quad & W \succeq 0, \\ & y_i^{\text{act.pf}} = D_{i,:} W M^{\text{act}} D_{i,:}^T - z_i^{\text{act.pf}}, \\ & y_i^{\text{rea.qf}} = D_{i,:} W M^{\text{rea}} D_{i,:}^T - z_i^{\text{rea.qf}}. \end{aligned}$$

Its Lagrangian can be written as

$$\begin{aligned}
& L(W, \boldsymbol{\lambda}^{\text{act.pf}}, \boldsymbol{\lambda}^{\text{rea.qf}}, \mathbf{y}^{\text{act.pf}}, \mathbf{y}^{\text{rea.qf}}) \\
&= \sum_{i=1}^n \left[ (y_i^{\text{act.pf}})^2 + \lambda_i^{\text{act.pf}} (y_i^{\text{act.pf}} - D_{i,:} W M^{\text{act}} D_{i,:}^T + z_i^{\text{act.pf}}) \right] \\
&+ \sum_{i=1}^n \left[ (y_i^{\text{rea.qf}})^2 + \lambda_i^{\text{rea.qf}} (y_i^{\text{rea.qf}} - D_{i,:} W W^{\text{rea}} D_{i,:}^T + z_i^{\text{rea.qf}}) \right].
\end{aligned} \tag{5}$$

Therefore the dual problem is

$$\begin{aligned}
& \max_{\boldsymbol{\lambda}} \min_{W, \mathbf{y}} \sum_{i=1}^l \left[ (y_i^{\text{act.pf}})^2 + (y_i^{\text{rea.qf}})^2 + \lambda_i^{\text{act}} (y_i^{\text{act}} + z_i^{\text{act}}) \right. \\
& \quad \left. + \lambda_i^{\text{rea}} (y_i^{\text{rea}} + z_i^{\text{rea}}) - D_{i,:} W T^{\text{act}} D_{i,:}^T \right. \\
& \quad \left. - D_{i,:} W M^{\text{rea}} D_{i,:}^T \right], \\
& \text{subject to } W \succeq 0.
\end{aligned}$$

Notice that  $D_{i,:} T^{\text{act}} W D_{i,:}^T$  can be converted into the simpler form below with a new matrix  $M^{\text{flow}}$  such that

$$\sum_{i=1}^l D_{i,:} (T^{\text{act}} + T^{\text{rea}}) W T_{i,:}^T = \sum_{i=1}^n \sum_{k=1}^n T_{ik}^{\text{flow}} W_{ki}. \tag{6}$$

Consequently, the above optimization becomes equivalent to

$$\begin{aligned}
& \max_{\boldsymbol{\lambda}} \min_{W, \mathbf{y}} \sum_{i=1}^l \left[ (y_i^{\text{act.pf}})^2 + (y_i^{\text{rea.qf}})^2 + \lambda_i^{\text{act}} (y_i^{\text{act}} + z_i^{\text{act}}) \right. \\
& \quad \left. + \lambda_i^{\text{rea}} (y_i^{\text{rea}} + z_i^{\text{rea}}) \right] - \sum_{i=1}^n \sum_{k=1}^n T_{ik}^{\text{flow}} W_{ki}, \\
& \text{subject to } W \succeq 0.
\end{aligned}$$

Voltage magnitude and voltage phase angle measurements can be derived similarly.

# Bibliography

- [1] F. C. Schweppe, J. Wildes, and D. B. Rom. Power system static state estimation, parts 1, 2, 3. *IEEE Transactions on Power Apparatus and Systems*, page 120, Jan. 1970.
- [2] F. F. Wu. Power system state estimation: A survey. *International Journal of Electrical and Power Engineering*, 12:80–87, 1990.
- [3] A. Monticelli. The impact of modeling short circuit branches in state estimation. *IEEE Transactions on Power Systems*, 8(1):364–370, Feb. 1993.
- [4] A. G. Exposito, A. Abur, A. V. Jaen, and C. G. Quiles. A multilevel state estimation paradigm for smart grids. *Proceedings of the IEEE*, page 952, Jun. 2011.
- [5] B. V. Tuykom, J. C. Maun, and A. Abur. Use of phasor measurements and tuned weights for unbalanced system state estimation. *North American Power Symposium (NAPS)*, page 1, Sep. 2010.
- [6] A. P. S. Meliopoulos, B. Fardanesh, and S. Zelingher. *Power system state estimation: modeling error effects and impact on system operation*. Jan. 2001.
- [7] A. Abur and A. G. Exposito. *Power system state estimation: Theory and implementation*. CRC Press, Mar. 2004.
- [8] A. G. Phadke. Synchronized phasor measurements-a historical overview. *Transmission and Distribution Conference and Exhibition*, (1):476, 2002.
- [9] Q. Li, T. Cui, Y. Weng, R. Negi, F. Franz, and M. D. Ilić. An information-theoretic approach to pmu placement in electric power systems. *IEEE Transactions on Smart Grid, Special Issue on Computational Intelligence Applications in Smart Grids*, submitted, 2011.

- [10] Y. Weng, R. Negi, Q. Liu, and M. D. Ilić. Robust state-estimation procedure using a least trimmed squares pre-processor. *IEEE Innovative Smart Grid Technologies*, pages 1–6, Jan. 2011.
- [11] A. Wood and B. Wollenberg. *Power generation, operation, and control*. 2 edition, 1996.
- [12] J. Lavaei and S. H. Low. Zero duality gap in optimal power flow problem. *IEEE Transactions on Power Systems*, 27(1):92–107, Feb. 2012.
- [13] B. Zhang and D. Tse. Geometry of feasible injection region of power networks. *2011 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1508–1515, Sep. 2011.
- [14] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, second edition, Feb. 2009.
- [15] E. Blood, B.H. Krogh, J. Ilic, and M.D. Ilic. A kalman filter approach to quasi-static state estimation in electrical power systems. *Proceedings of the North American Power Symposium (NAPS)*, pages 505–510, Sep. 2006.
- [16] E. Blood, B.H. Krogh, and M.D. Ilic. Electric power system static state estimation through kalman filtering and load forecasting. *IEEE Power Engineering Society General Meeting*, Jul. 2008.
- [17] T. Jaakkola M. J. Wainwright and A. S. Willsky. A new class of upper bounds on the log partition function. *IEEE Trans. on Information Theory*, 51:2313–2335, Jul. 2005.
- [18] Embedding.
- [19] Discriminative Model.
- [20] E. Handschin, F. C. Schweppe, J. Kohlas, and A. Fiechter. Bad data analysis for power system state estimation. *IEEE Transactions on Power Apparatus and Systems*, page 329, Apr. 1975.
- [21] A. Garcia, A. Monticelli, and P. Abreu. Fast decoupled state estimation and bad data processing. *IEEE Transactions on Power Apparatus and Systems*, page 1645, Sep. 1979.
- [22] N. Xiang, S. Wang, and E. Yu. A new approach for detection and identification of multiple bad data in power system state estimation. *IEEE Transactions on Power Apparatus and Systems*, 101(2):454–462, Feb. 1982.

- [23] A. Monticelli and A. Garcia. Reliable bad data processing for real-time state estimation. *IEEE Transactions on Power Apparatus and Systems*, page 1126, May 1983.
- [24] K. A. Clements and P. W. Davis. Multiple bad data detectability and identifiability: A geometric approach. *IEEE Transactions on Power Systems*, PWRD-1(3):355–360, Aug. 1986.
- [25] M. R. Irving, R.C. Owen, and M.J.H. Sterling. Power system state estimation using linear programming. *IEE Proceedings, Part G*, 125(9):879–885, Sep. 1978.
- [26] A. Abur and M. K. Celik. A fast algorithm for the weighted least absolute value state estimation. *IEEE Transactions on Power Systems*, 6(1):1, Feb. 1991.
- [27] S. Boyd and L. Vandenberghe. *Convex Optimization*. 2004.
- [28] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, Mar. 1996.
- [29] J. Carpentier. Contribution to the economic dispatch problem. *Bull. Soc. Franc. Elect*, 8(3):431–447, 1962.
- [30] X. Bai, H. Wei, K. Fujisawa, and Y. Wang. Semidefinite programming for optimal power flow problems. *International Journal of Electric Power & Energy Systems*, 30(6–7):383–392, Feb. 2008.
- [31] B. C. Lesieutre, A. Pinar, and S. Roy. Power system extreme event detection: The vulnerability frontier. *Proceedings of the 41st Annual Hawaii International Conference on System Sciences*, page 184, Jan. 2008.
- [32] D. K. Molzahn, B. C. Lesieutre, and C. L. DeMarco. Investigation of non-zero duality gap solutions to a semidefinite relaxation of the optimal power flow problem. *47th Hawaii International Conference on System Sciences (HICSS)*, Jan. 2014.
- [33] Y. Weng, Q. Li, R. Negi, and M. D. Ilić. Semidefinite programming for power system state estimation. *IEEE Power and Energy Society General Meeting*, Jul. 2012.
- [34] A. Y. S. Lam, B. Zhang, and D. Tse. Distributed algorithms for optimal power flow problem. Submitted for publication.
- [35] H. Zhu and G. B. Giannakis. Estimating the state of ac power systems using semidefinite programming. *Proceedings 43rd North America Power Symposium (NAPS)*, pages 1–7, Aug. 2011.

- [36] B. Fardanesh. Direct non-iterative power system state solution and estimation. *IEEE Power and Energy Society General Meeting*, pages 1–6, Jul. 2012.
- [37] R. D. Zimmerman, C. E. Murillo-Sanchez, , and R. J. Thomas. Matpower’s extensible optimal power flow architecture. *IEEE Power and Energy Society General Meeting*, pages 1–7, Jul. 2009.
- [38] R. D. Zimmerman and C. E. Murillo-Sanchez. Matpower, a matlab power system simulation package. <http://www.pserc.cornell.edu/matpower/manual.pdf>, Jul. 2010.
- [39] J. F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software (Special issue on Interior Point Methods (CD supplement with software))*, pages 625–653, 1999.
- [40] M.R. Irving and M.J. Sterling. Substation data validation. *IEE Proceedings C on Generation, Transmission and Distribution*, 129(3):119–122, May 1982.
- [41] A. Monticelli and A. Garcia. Modeling zero impedance branches in power system state estimation. *IEEE Transactions on Power Systems*, 6(4):1561–1570, Nov. 1991.
- [42] A. Abur, H. Kim, and M. K. Celik. Identifying the unknown circuit breaker statuses in power networks. *IEEE Transactions on Power Systems*, 10(4):2029–2037, Nov. 1995.
- [43] S. Boyd, L. Xiao, A. Mutapcic, and J. Mattingley. Notes on decomposition methods. 2007.
- [44] X. Bai and H. Wei. A semidefinite programming method with graph partitioning technique for optimal power flow problems. *International Journal of Electrical Power and Energy Systems*, 33:1309–1314, Sep. 2011.
- [45] R. A. Jabr. Exploiting sparsity in sdp relaxations of the opf problem. *IEEE Transactions on Power Systems*, 27(2):1138–1139, May 2012.
- [46] D. P. Bertsekas. Nonlinear programming. *Athena Scientific*, 1995.
- [47] R. Grone, C. R. Johnson, E. M Sa, and H. Wolkowicz. Positive definite completions of partial hermitian matrices. *Linear Algebra and Applications*, 58:109–124, 1984.

- [48] M. Fukuda, M. Kojima, K. Murota, and K. Nakata. Exploiting sparsity in semidefinite programming via matrix completion i: General framework. *Journal of Mathematical Analysis and Applications*, 11(3):647–674, 2000.
- [49] K. Nakata, K. Fujisawa, M. Fukuda, M. Kojima, and K. Murota. Exploiting sparsity in semidefinite programming via matrix completion ii: Implementation and numerical results. *Mathematical Programming*, 95(B):303–327, 2003.
- [50] D. J. Rose, R. E. Tarjan, and G. S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computing*, 5:266–283, 1976.
- [51] A. A. Ahmadi, A. Olshevsky, P. A. Parrilo, and John N. Tsitsiklis. Np-hardness of deciding convexity of quartic polynomials and related problems. *Mathematical Programming*, 137:453–476, Feb. 2013.
- [52] B. Mariere, Zhi-Quan Luo, and T. N. Davidson. Blind constant modulus equalization via convex optimization. *IEEE Transactions on Signal Processing*, 51(3):805–818, Mar. 2003.
- [53] Y. Nesterov. Squared functional systems and optimization problems. *High Performance Optimization*, pages 405–440, 2000.
- [54] H. Wu and J. Giri. Pmu impact on the state estimation reliability for improved grid security. *IEEE Power Energy Society Transmission and Distribution Conference and Exhibition*, page 1349, Mar. 2005.
- [55] R. M. Edwards X. Jin and A. Ray. Feature extraction for data-driven fault detection in nuclear power plants. *Transactions of the American Nuclear Society*, pages 163–164, 2010.
- [56] M. Kruger, S. X. Ding, A. Haghani, P. Engel, and T. Jeansch. A data-driven approach for sensor fault diagnosis in gearbox of wind energy conversion system. *10th IEEE International Conference on Control and Automation (ICCA)*, pages 227–232, June 2013.
- [57] Shen Yin, Guang Wang, and Hamid Reza Karimi. Data-driven design of robust fault detection system for wind turbines. *Mechatronics*, 24(4):298 – 306, 2014. Vibration control systems.
- [58] G.J. van der Veen, J.W. van Wingerden, P.A. Fleming, A.K. Scholbrock, and M. Verhaegen. Global data-driven modeling of wind turbines in the presence of turbulence. *Control Engineering Practice*, 21(4):441 – 454, 2013.

- [59] Rui Xiong, Fengchun Sun, Zheng Chen, and Hongwen He. A data-driven multi-scale extended kalman filtering based parameter and state estimation approach of lithium-ion olymer battery in electric vehicles. *Applied Energy*, 113(0):463 – 476, 2014.
- [60] 6097, 2010.
- [61] A. Kusiak and Z. Zhang. Short-horizon prediction of wind power: A data-driven approach. *Energy Conversion, IEEE Transactions on*, 25(4):1112–1122, Dec 2010.
- [62] H. Wang, B. J. Zhang, X. Z. Liu, D. Z. Luo, and S. B. Zhong. Prediction model for power transmission line icing load based on data-driven. *Advanced Materials Research*, pages 1295–1299, 2011.
- [63] Q. Sun and Y. Wang. Data-driven predictive functional control of power kites for high altitude wind energy generation. *Electrical Power and Energy Conference (EPEC), 2012 IEEE*, pages 274–279, Oct 2012.
- [64] J. Zhu, E. Zhuang, C. Ivanov, and Z. Yao. A data-driven approach to interactive visualization of power systems. *IEEE Transactions on Power Systems*, 26(4):2539–2546, Nov 2011.
- [65] C. M. Bishop. Pattern recognition and machine learning. *Springer*, 2006.
- [66] Y. Weng, R. Negi, and M. Ilic. A search method for obtaining initial guesses for smart grid state estimation. *IEEE SmartGridComm Symposium*, Nov. 2012.
- [67] L. Mili, T. V. Cutsem, and M. R. Pavella. Hypothesis testing identification: A new method for bad data analysis in power system state estimation. *IEEE Transactions on Power Apparatus and Systems*, page 3239, Nov. 1984.
- [68] Y. Liu, P. Ning, and M. K. Reiter. False data injection attacks against state estimation in electric power grids. *ACM Conference on Computer and Communications Security*, pages 21–32, Nov. 2009.
- [69] D. Hawkins. Identification of outliers. *Chapman and Hall, London*, 1981.
- [70] M. M. Breunig, H. P. Kriegel, R. T. Ng, and J. Sander. Lof: Identifying density-based local outliers. *Proc. ACM SIGMOD Conf. 2000*, pages 93–104, 2000.
- [71] E. M. Knorr and R. T. Ng. Algorithms for mining distance based outliers in large datasets. *Proceeding 24th Int. Conf. on Very Large Data Bases*, pages 392–403, 1998.



- [72] L. Wasserman. All of nonparametric statistics. *Springer*, 2007.
- [73] G. Strang. Introduction to linear algebra (section 6.7). *Wellesley-Cambridge Press*, 1998.
- [74] M. Sahlgren. An introduction to random indexing. in *Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE*, 2005.
- [75] W. B. Johnson and J. Lindenstrauss. Extensions of lipshitz mapping into hilbert space. *American Mathematical Society Conference in Modern Analysis and Probability, Contemporary Mathematics*.
- [76] NYISO. Load data profile. <http://www.nyiso.com>, May. 2012.
- [77] L. Mili. A robust estimation method for topology error identification. *IEEE TPWRS*, (14(4)), 1999.
- [78] R.L. Lugtu, D.F. Hackett, K.C. Liu, and D.D. Might. Power system state estimation: Detection of topological errors. *IEEE Transactions on Power Apparatus and Systems*, PAS-99(6):2406–2411, Nov. 1980.
- [79] Tom Mitchell. Generative and discriminative classifiers: Naive bayes and logistic regression. *Machine Learning Draft Chapter*, 2005.
- [80] J. Zhu and T. Hastie. Kernel logistic regression and the import vector machine. *Journal of Computational and Graphical Statistics*, (14):185–205, 2005.
- [81] N. R. Shivakumar and A. Jain. A review of power system dynamic state estimation techniques. *Power System Technology and IEEE Power India Conference*, pages 1–6, Oct. 2008.
- [82] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82(Series D):35–45, 1960.
- [83] E. A. Wan and R. van der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of IEEE Symposium on Adaptive Systems for Signal Processing Communications and Control*, pages 153–158, Oct. 2000.
- [84] E. H. Abed, N. S. Namachchivaya, T. J. Overbye, M. A. Pai, P. W. Sauer, and A. Sussman. Data-driven power system operations. *Proc. Int. Conf. Comput. Sci.*, pages 448–455, 2006.

- [85] S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, (92(3)):401C422, Mar. 2004.
- [86] L. Ralaivola and F. d'Alche Buc. Dynamical modeling with kernels for nonlinear time series prediction. *Proc. Neural Information Processing Systems*, pages 8–13, Dec. 2003.
- [87] Y. Weng, Q. Li, M. Ilic, and R. Negi. Distributed algorithm for sdg state estimation. *IEEE Innovative Smart Grid Technology Conference*, Aug. 2013.
- [88] Y. Weng, Bruce Fardanesh, M. Ilic, and R. Negi. Novel approaches using semidefinite programming method for power systems state estimation. *IEEE North American Power Symposium*, Sep. 2013.
- [89] Y. Weng, R. Negi, and M. Ilic. Graphical model for state estimation in electric power systems. *IEEE SmartGridComm Symposium*, Oct. 2013.
- [90] L. D. Alvaro. Development of distribution state estimation algorithms and application. *IEEE PES ISGT Europe*, Oct. 2012.
- [91] R. Negi Y. Rachlin and P. Khosla. Sensing capacity for markov random field. *Proc. IEEE Int. Symp. Information Theory*, Jun. 2005.
- [92] R. Negi B. Narayanaswamy, Y. Rachlin and P. Khosla. The sequential decoding metric for detection in sensor networks. *Proc. IEEE Int. Symp. Information Theory*, Jun. 2007.
- [93] J. Yedidia, W. T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. *International Joint Conference on Artificial Intelligence*, 2001.
- [94] Y. Hu, A. Kuh, A. Kavcic, and D. Nakafuji. Micro-grid state estimation using belief propagation on factor graphs. *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, Dec. 2010.
- [95] Y. Hu, A. Kuh, A. Kavcic, and T. Yang. A belief propagation based power distribution system state estimator. *IEEE Computational Intelligence Magazine*, 11:36C46, Aug. 2011.
- [96] M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1:1–305, 2008.

- [97] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1568–1583, Oct. 2006.
- [98] D.C. Yu, T.C. Nguyen, and P. Haddawy. Bayesian network model for reliability assessment of power systems. *IEEE Transactions on Power Systems*, 14(2):426–432, May 1999.
- [99] H. Kim and C. Singh. Power system probabilistic security assessment using bayes classifier. *Electric Power Systems Research*, 74(1):157 – 165, 2005.
- [100] Yanbo Chen, Feng Liu, Shengwei Mei, Guangyu He, Qiang Lu, and Yanlan Fu. An improved recursive bayesian approach for transformer tap position estimation. *IEEE Transactions on Power Systems*, 28(3):2830–2841, Aug 2013.
- [101] Y. Zhu, L. Huo, and J. Lu. Bayesian networks-based approach for power systems fault diagnosis. *Power Delivery, IEEE Transactions on*, 21(2):634–639, April 2006.
- [102] R. Singh, E. Manitsas, B.C. Pal, and G. Strbac. A recursive bayesian approach for identification of network configuration changes in distribution system state estimation. *Power Systems, IEEE Transactions on*, 25(3):1329–1336, Aug 2010.
- [103] C. Tao and F. Franchetti. A quasi-monte carlo approach for radial distribution system probabilistic load flow. pages 1–6, Feb 2013.
- [104] N. Gupta. A complete probabilistic power flow solution for transmission system. pages 1–6, Dec 2012.
- [105] Giuseppe Marco Tina and Salvina Gagliano. Probabilistic modelling of hybrid solar/wind power system with solar tracking system. *Renewable Energy*, 36(6):1719 – 1727, 2011.
- [106] E. Ni and P. B. Luh. Forecasting power market clearing price and its discrete pdf using a bayesian-based classification method. 3:1518–1523, 2001.
- [107] Yuan Ning, Yufeng Liu, and Qiang Ji. Bayesian - bp neural network based short-term load forecasting for power system. 2:V2–89–V2–93, Aug 2010.
- [108] S. Vassena, P. Mack, P. Rouseaux, C. Druet, and L. Wehenkel. A probabilistic approach to power system network planning under uncertainties. 2:6 pp, June 2003.

- [109] P. M. Anderson and A. Bose. A probabilistic approach to power system stability analysis. *IEEE Transactions on Power Apparatus and Systems*, PAS-102(8):2430–2439, Aug 1983.
- [110] R. Negi, M. Ilic, F. Franchetti, and O. Mengshoel. A computing framework for distributed decision making to ensure robustness of complex man-made network systems: The case of the electric power networks. *NSF Proposal*, 2009.
- [111] Belief Propagation.
- [112] A. Z. Broder. Generating random spanning trees. *30th Annual Symposium on Foundations of Computer Science*, pages 442–447, Oct. 1989.
- [113] M. J. Wainwright, T. Jaakkola, and A. S. Willsky. Map estimation via agreement on trees: message-passing and linear programming. *IEEE Trans. on Information Theory*, 51(11):3697–3717, Nov. 2005.
- [114] Y. Weng, R. Negi, and M. Ilic. A search method for obtaining initial guesses for smart grid state estimation. *IEEE SmartGridComm Symposium*, Dec. 2012.
- [115] Y. Weng, R. Negi, and M. Ilic. Historical data-driven state estimation for electric power systems. *IEEE SmartGridComm Symposium*, Oct. 2013.
- [116] J. Zhang, G. Welch, and G. Bishop. Lodim: A novel power system state estimation method with dynamic measurement selection. *IEEE Power and Energy Society General Meeting*, Jul. 2011.
- [117] EATON. Power xpert meters 4000/6000/8000.
- [118] R. Negi B. Narayanaswamy, Y. Rachlin and P. Khosla. An analysis of the computational complexity of sequential decoding of specific tree codes over gaussian channels. *Proc. IEEE Int. Symp. Information Theory*, Jun. 2008.