STUDY OF PARAMETER ESTIMATION FOR THREE PHASE UNTRANSPOSED TRANSMISSION LINES

A Thesis Presented

By

Pengxiang Ren

to

The Department of Electrical and Computer Engineering

in partial fulfillment of the requirements
for the degree of

Master of Science

in the field of

Power Systems

Northeastern University
Boston, Massachusetts

December 2015
ABSTRACT

The increased amounts of stochastic renewable generations require new strategies in the system operation dealing with the issues related to the fluctuating power flows and reliability of generation system. One critical aspect to the successful deployment of renewable resources is the consideration of transmission capacity, which largely depends on the accurate information of transmission line parameters. What’s more, for power system state estimation and other steady-state applications, such as relay settings and fault distance calculation, a precise knowledge of transmission line parameters is also indispensable.

Currently, methods for transmission line parameter identification are various but limited. First and foremost, most of the prevailing algorithms to do parameter identification is based on the assumption that transmission line is fully transposed and the corresponding components are symmetric. However, this is not true for certain low voltage systems or systems with short lines. Second, several methods that do such parameter identification use the off-line information on conductor types and tower geometry, which neglect the temperature changes and other factors that may vary the parameters during daily operation. Therefore, a detailed study on developing and evaluating the algorithms for untransposed transmission line parameter estimation is taken up in this thesis.

The case of the unknown parameters of untransposed transmission lines is investigated first. By considering the trivial Ohm’s Law, the equation containing all the measurements and unknown parameters is then constructed and transformed to desired format. By thoroughly understanding the properties of these parameters, three algorithms for estimating these parameters, one static and two dynamic algorithms are developed. The static algorithm uses the sliding window to estimate the parameters along several time scans and thereby avoids the singularity of the Jacobian matrix. The first dynamic algorithm augments the states with the parameters and then constructs a highly nonlinear measurement function. To estimate the augmented states vector, the iterated unscented
Kalman filter is introduced and evaluated. The other dynamic method combines the idea of state estimation and parameter tracking altogether. A three phase static state estimation is introduced first to estimate the states, then the unknown parameters are tracked dynamically by a Kalman filter. The uncertainties existed in the estimation function will be eliminated by iteratively processing state estimation and parameter tracking. Several tests are conducted under distinct line parameter variations and the results show the performance of the proposed approaches. The comparison of the proposed algorithms and the implementations of the algorithm on a typical power system are also provided in the last part of the thesis.
DECLARATION OF AUTHORSHIP

I, Pengxiang Ren, declare that this thesis titled, 'Study of Parameter Estimation for Three Phase Untransposed Transmission Lines' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: 

Date: 12/18/2015
ACKNOWLEDGEMENTS

It took me a long time to finish this thesis. And fortunately it is finished on time and the guidance from many people helped me to finish it successfully.

First of all, I would like to express my deep gratitude to my advisor Professor Ali Abur. I would like to thank you for providing me this opportunity to perform this research work and encouraging me all the time. It is of great fortune to work with you. Your expertise and insight have been influential and your teaching and work ethic are an inspiration. Your patience and support from the beginning to the end help overcome the challenges and progress on the research.

I would also like to express my great appreciation to Professor Hanoch Lev-Ari, my co-advisor. It is my great honor to work with you. And I am really thankful for your valuable suggestions and willingness to spend your time on this research.

I would also like to thank the member of my supervisory committee, Professor Bahram Shafai, for your impressive teaching about control problems.

I am also obliged to the coordinator Ms. Faith Crisley for her kindness help on the arrangement of the thesis defense.

Last but not least, I would like to thank all my friends, Arthur Mouco, Pignati Marco, Yuzhang Lin and Zanni Lorenzo for their constructive suggestions on the research.
TABLE OF CONTENTS

ABSTRACT ...................................................................................................................... ii

DECLARATION OF AUTHORSHIP .................................................................................. iv

ACKNOWLEDGEMENTS ................................................................................................... v

TABLE OF CONTENTS ..................................................................................................... vi

TABLE OF FIGURES ......................................................................................................... xi

TABLE OF TABLES ........................................................................................................... xv

NOMENCLATURE ............................................................................................................. xvii

CHAPTER 1  INTRODUCTION ................................................................. 1

CHAPTER 2  PARAMETERS IN OVERHEAD TRANSMISSION LINES .................. 5

2.1  Parameters in Three Phase Transmission Line ................................................... 5

2.1.1 Three Phase Transmission Line Model ........................................................... 6

2.1.2 Transmission Line Type .................................................................................... 9

2.2  Transposition of Transmission Line ................................................................. 13

2.2.1 Transposition .................................................................................................... 13

2.2.2 Untransposed Transmission Line .................................................................... 15

2.3  Simulation Model ................................................................................................. 24

2.3.1 Transmission Line Model in ATP ................................................................. 26
2.3.2 Powergui Computation of RLC Parameters ................................................... 28
2.3.3 Simulation Test Cases .................................................................................... 32
2.3.4 Simulation Procedure .................................................................................... 36

CHAPTER 3  STATIC PARAMETER ESTIMATION .................................................. 38
3.1 Formulation of Measurement Model ................................................................. 38
3.2 Static Implementation of Parameter Estimation .................................................. 39
  3.2.1 Model Construction .................................................................................. 39
  3.2.2 Sliding Window Method ........................................................................... 44
3.3 Simulation Results for Static Parameter Estimation ........................................... 46
  3.3.1 Test Case 1 Simulations ........................................................................... 46
  3.3.2 Test Case 2 Simulations ........................................................................... 51
3.4 Chapter Summary ............................................................................................. 52

CHAPTER 4  AUGMENTED STATE ESTIMATION .............................................. 53
4.1 Static Implementation of Augmented State Estimation ...................................... 53
4.2 Basic Algorithm of Augmented State Estimation .............................................. 54
4.3 Iterated Unscented Kalman Filter Implementation .......................................... 55
4.4 Simulation Results for ASE ............................................................................ 61
  4.4.1 Tuning of IUKF .................................................................................... 61
4.4.2 Test Case 1 Simulations ................................................................. 63
4.4.3 Test Case 2 Simulations ................................................................. 67
4.5 Chapter Summary ........................................................................... 70

CHAPTER 5 JOIN T STATE ESTIMATION AND PARAMETER TRACKING ....... 72
5.1 Overview ....................................................................................... 72
5.2 State Estimation Based on Extended Kalman Filter ...................... 73
  5.2.1 Measurement Model for State Estimation .................................. 73
5.3 Parameter Tracking Implementation ............................................ 76
5.4 Joint State Estimation and Parameter Tracking Implementation ...... 78
  5.4.1 Algorithm Procedure ................................................................. 78
  5.4.2 Implementation Accuracy .......................................................... 83
  5.4.3 Implementation Complexity ....................................................... 84
5.5 Simulation Results for JSEPT ....................................................... 84
  5.5.1 Tuning of KF ............................................................................ 84
  5.5.2 Test Case 1 Simulations ............................................................. 86
  5.5.3 Test Case 2 Simulations ............................................................. 89
5.6 Chapter Summary ........................................................................... 92

CHAPTER 6 COMPARISON OF THREE METHODS ................................... 94
6.1 Estimation Sensitivity to Initial Value ................................................................. 94
   6.1.1 ASE Sensitivity to Initial Value ................................................................. 95
   6.1.2 JSEPT Sensitivity to Initial Value ............................................................... 99
6.2 Run Time Analysis of the Algorithms .............................................................. 102

CHAPTER 7 IMPLEMENTATION OF JSEPT ON NETWORK SYSTEMS ................. 104
   7.1 Overview of Network Implementation ............................................................ 104
   7.2 Network Model ............................................................................................... 104
      7.2.1 Network Model Construction ................................................................. 104
      7.2.2 Simulation Model .................................................................................. 106
   7.3 JSEPT Algorithm Modification ....................................................................... 110
      7.3.1 State Estimation ..................................................................................... 110
      7.3.2 Parameter Tracking ............................................................................... 111
   7.4 Simulations of Network Model ....................................................................... 114

CHAPTER 8 CONCLUSIONS AND FUTURE WORK ............................................ 120
   8.1 Conclusions ..................................................................................................... 120
   8.2 Future Work .................................................................................................... 122

REFERENCE .................................................................................................................. 123
<table>
<thead>
<tr>
<th>Figure 2.1</th>
<th>Model of three phase transmission line</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 2.2</td>
<td>Equivalent PI model of medium-length three phase transmission line</td>
<td>9</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>Equivalent PI model of long three phase transmission line</td>
<td>12</td>
</tr>
<tr>
<td>Figure 2.4</td>
<td>An example of three phase transmission line phasing</td>
<td>13</td>
</tr>
<tr>
<td>Figure 2.5</td>
<td>Transposition of the transmission lines</td>
<td>14</td>
</tr>
<tr>
<td>Figure 2.6</td>
<td>‘T’ point of three transmission lines</td>
<td>16</td>
</tr>
<tr>
<td>Figure 2.7</td>
<td>500kV transmission line tower</td>
<td>25</td>
</tr>
<tr>
<td>Figure 2.8</td>
<td>Geometric information of 500kV transmission line tower</td>
<td>25</td>
</tr>
<tr>
<td>Figure 2.9</td>
<td>LCC line model definition</td>
<td>26</td>
</tr>
<tr>
<td>Figure 2.10</td>
<td>LCC line model data</td>
<td>27</td>
</tr>
<tr>
<td>Figure 2.11</td>
<td>Powergui computation of transmission line</td>
<td>29</td>
</tr>
<tr>
<td>Figure 2.12</td>
<td>Three phase transmission line model based on ATP</td>
<td>32</td>
</tr>
<tr>
<td>Figure 2.13</td>
<td>Actual value of (p_1) test case 1</td>
<td>34</td>
</tr>
<tr>
<td>Figure 2.14</td>
<td>Actual value of (p_2) test case 1</td>
<td>34</td>
</tr>
<tr>
<td>Figure 2.15</td>
<td>Actual value of (p_1) test case 2</td>
<td>35</td>
</tr>
<tr>
<td>Figure 2.16</td>
<td>Actual value of (p_2) test case 2</td>
<td>35</td>
</tr>
<tr>
<td>Figure 3.1</td>
<td>Sliding window illustration</td>
<td>45</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>SPE scenario 1-0 with (T = 4)</td>
<td>46</td>
</tr>
</tbody>
</table>
Figure 5.2  Process flow diagram of JSEPT ................................................................. 82
Figure 5.3  JSEPT tuning scenario 1-0............................................................... 85
Figure 5.4  JSEPT tuning scenario 2-0............................................................... 85
Figure 5.5  JSEPT scenario 1-0 ........................................................................ 86
Figure 5.6  JSEPT scenario 1-1 ........................................................................ 87
Figure 5.7  JSEPT scenario 1-2 ........................................................................ 88
Figure 5.8  JSEPT scenario 2-0 ........................................................................ 89
Figure 5.9  JSEPT scenario 2-1 ........................................................................ 90
Figure 5.10  JSEPT scenario 2-2...................................................................... 91
Figure 6.1  ASE scenario 1-0 with perfect initial ............................................. 95
Figure 6.2  ASE scenario 1-0 with biased initial .............................................. 96
Figure 6.3  ASE scenario 1-0 with biased initial EVT ........................................ 96
Figure 6.4  ASE scenario 1-0 with biased initial with optimal tuning ............ 97
Figure 6.5  ASE scenario 1-0 with biased initial EVT with optimal tuning ... 97
Figure 6.6  ASE scenario 1-0 with zero initial .................................................. 97
Figure 6.7  ASE scenario 2-0 with perfect initial .............................................. 98
Figure 6.8  ASE scenario 2-0 with biased initial .............................................. 98
Figure 6.9  JSEPT scenario 1-0 with perfect initial ........................................... 99
Figure 6.10  JSEPT scenario 1-0 with biased initial ........................................ 99
Figure 6.11  JSEPT scenario 1-0 with biased initial EVT ................................. 100
Figure 6.12  JSEPT scenario 1-0 with zero initial ........................................... 100
Figure 6.13  JSEPT scenario 1-0 with zero initial EVT ..................................... 100
Figure 6.14  JSEPT scenario 2-0 with perfect initial ........................................ 101
Figure 6.15  JSEPT scenario 2-0 with biased initial ......................................... 101
Figure 6.16  JSEPT scenario 2-0 with biased initial EVT ................................... 101
Figure 7.1   System model for parameter estimation ........................................ 106
Figure 7.2   JSEPT network scenario 1 ............................................................... 116
Figure 7.3   JSEPT network scenario 1 EVT ....................................................... 117
Figure 7.4   JSEPT network scenario 2 ............................................................... 117
Figure 7.5   JSEPT network scenario 2 EVT ....................................................... 118
Figure 7.6   JSEPT network scenario 3 ............................................................... 118
Figure 7.7   JSEPT network scenario 3 EVT ....................................................... 119
TABLE OF TABLES

Table 2.1  Size of Parameters in Untransposed Line...................................................... 21
Table 2.2  Actual value of line parameters ................................................................. 33
Table 3.1  Comparison of the norm of the measurement residuals ......................... 48
Table 4.1  Evaluation quantities of ASE scenario 1-0.............................................. 64
Table 4.2  Evaluation quantities of ASE scenario 1-2.............................................. 66
Table 4.3  Evaluation quantities of ASE scenario 2-0.............................................. 67
Table 4.4  Evaluation quantities of ASE scenario 2-1.............................................. 68
Table 4.5  Evaluation quantities of ASE scenario 2-2.............................................. 69
Table 5.1  Evaluation quantities of JSEPT scenario 1-0.......................................... 86
Table 5.2  Evaluation quantities of JSEPT scenario 1-1.......................................... 87
Table 5.3  Evaluation quantities of JSEPT scenario 1-2.......................................... 88
Table 5.4  Evaluation quantities of JSEPT scenario 2-0.......................................... 89
Table 5.5  Evaluation quantities of JSEPT scenario 2-1.......................................... 90
Table 5.6  Evaluation quantities of JSEPT scenario 2-2.......................................... 92
Table 6.1  Time Consumption for algorithm .............................................................. 102
Table 7.1  Actual values of line 1 parameters ............................................................. 114
Table 7.2  Actual values of line 2 parameters ............................................................. 114
Table 7.3  Actual values of line 3 parameters ............................................................. 115
Table 7.4  Actual values of line 4 parameters ................................................................. 115
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASE</td>
<td>Augmented State Estimation</td>
</tr>
<tr>
<td>ATP</td>
<td>Alternative Transients Program</td>
</tr>
<tr>
<td>CC</td>
<td>Coefficient Covariance</td>
</tr>
<tr>
<td>EKF</td>
<td>Extended Kalman Filter</td>
</tr>
<tr>
<td>EMS</td>
<td>Energy Management Systems</td>
</tr>
<tr>
<td>EMTP</td>
<td>Electro-Magnetic Transient Program</td>
</tr>
<tr>
<td>EV</td>
<td>Error Variance</td>
</tr>
<tr>
<td>EVS</td>
<td>Error Variance Summation</td>
</tr>
<tr>
<td>EVT</td>
<td>Error Variance along with Time</td>
</tr>
<tr>
<td>IEKF</td>
<td>Iterated Extended Kalman Filter</td>
</tr>
<tr>
<td>IUKF</td>
<td>Iterated Unscented Kalman Filter</td>
</tr>
<tr>
<td>JSEPT</td>
<td>Joint State Estimation and Parameter Tracking</td>
</tr>
<tr>
<td>KF</td>
<td>Kalman Filter</td>
</tr>
<tr>
<td>LCC</td>
<td>Line constants, Cable constants and Cable parameters</td>
</tr>
<tr>
<td>MAPE</td>
<td>Mean Absolute Percentage Error</td>
</tr>
<tr>
<td>NRMSE</td>
<td>Normalized RMSE</td>
</tr>
<tr>
<td>PMU</td>
<td>Phasor Measurement Unit</td>
</tr>
</tbody>
</table>
PE  Parameter Estimation
PT  Parameter Tracking
RECV Receive end of transmission line
RMSE Root-Mean-Square-Deviation
RMS  Root-Mean-Square
SE State Estimation
SEND Send end of transmission line
SPE Static Parameter Estimation
UKF Unscented Kalman Filter
WLS Weighted Least Squares
Chapter 1

Introduction

Renewable energy technologies, such as wind and solar power, are becoming increasingly attractive complementary resources to the existing energy supplies, because of climate change and energy diversification concerns [1]-[4]. The increased amounts of stochastic renewable generations require new strategies in the system operation dealing with the issues related to the fluctuating power flows and reliability of generation system [5]. One critical aspect to the successful deployment of renewable resources is the consideration of transmission capacity. The transmission capacity of the power system must be as unplanned dynamics caused by variable output, and can transport energy generated from emerging renewable sources in a long distance without producing system bottlenecks [6]. To lighten the impact of variable power output from renewable sources such as intermittency and surge, various methods are proposed in [7]-[8] so that the transmission capacity of a transmission line could be calculated precisely. Since the transmission system is always operated near to its limits, an accurate setting of protection relays based on exact parameters of the transmission line becomes more and more important. What’s more, relay settings [9]-[12], fault distance calculation [13], and state estimation [14]-[15] also entail the parameters of transmission lines. All in all, an accurate knowledge of transmission line parameters is a prerequisite for the realization of future smart grid.

Transmission line parameters, including series resistance, series inductance, shunt capacitance and shunt conductance, are critical to power system analysis. Transmission lines are generally
configured in three phases, and are designed to operate in an almost balanced manner. The analysis of symmetric three phase transmission line is relatively simple: a symmetrical transformation will decompose the three phase system into three independent systems, noted as the positive, negative and the zero sequence networks. There are many related studies about parameter estimation based on transposed transmission line model [16]-[23]. In these literatures, several methods are proposed based on symmetrical components to estimate sequence impedance of a live transmission line. In [20]-[22], they use the off-line geometric information to estimate line parameters, which neglect the temperature change and other factors that may vary the parameters. In [23], a method based on symmetrical component to estimate the zero sequence impedance of a live transmission line is proposed.

However, in practice, though most of high voltage systems are almost symmetrical, there are cases where the untransposed transmission lines will carry significant power flows. It has been reported in [19] that verification of line parameters for more than 50 in-service overhead transmission lines show the reduced parameters are a poor reflection of the real circumstances due to asymmetric and untransposed transmission lines. What’s more, for actual transmission lines, there are several reasons that will lead to asymmetrical spacing of each phase, which corresponds to asymmetrical system. This is detailed in Chapter 2.

Therefore, the discussion of the case of line parameters for untransposed transmission lines is necessary. To obtain the more prevailing line parameters, online estimation approaches are highly desirable and especially beneficial [24]-[26]. In [24], two sets of synchronized voltage and current phasors from two ends of the line are used to obtain ABCD parameters of the line. However, the identification of distributed parameters per unit length is not addressed. In [25], the authors solve for the line parameters based on Laplace transform technique by utilizing three sets of synchronized voltage and current phasors, however the assumption that parameters would not vary during three sample sets makes the algorithm unreliable if the sampling duration is long. Reference [26] presents a method for parameter identification of unsymmetrical transmission lines by introducing two stages, which are robust segmentation and precise phase shift calculation. Nevertheless, the neglecting measurement noise impacts the accuracy of the estimated parameters.
Thus, a parameter estimation approach based on untransposed transmission line model is proposed in this thesis. For an untransposed transmission line, the number of unknown parameters is much larger than those for the transposed case. Thus, the Jacobian matrix in the measurement model does not have full column rank, and the corresponding gain matrix will be singular. Three algorithms are developed to address this problem and still successfully carry out parameter estimation for untransposed transmission lines.

The first one is the approach of static parameter estimation (SPE), which is proposed in Chapter 3. The measurement model and the Jacobian matrix is built first. By introducing the sliding window that covers several measurement scans, the singularity problem is solved. Using the weighted least squares estimation method, parameters can be estimated from the measurements with little time latency. Simulation results along with a discussion of the foot step of sliding window and the influence of measurement noise are subsequently presented.

The second algorithm proposed in Chapter 4 is denoted as augmented state estimation (ASE). It stacks the state and the unknown parameters together as a new augmented state, which reduces the number of basic equations. By reconstructing the measurement model as a highly nonlinear equation, along with the dynamic equations of the new state, an iterated unscented Kalman filter is used to solve this problem. Numerical results are given for the tests carried out using several different scenarios.

Chapter 5 describes the third algorithm which is referred here as the joint state estimation and parameter tracking (JSEPT) algorithm. It is an online algorithm for estimating the three phase untransposed transmission line parameters. There are two loosely –coupled linear sub-problems namely state estimation and parameter tracking. State estimation eliminates the impact of measurement noise using weighted least squares (WLS) method and parameter tracking uses the Kalman filter to track the parameters with only a single snap-shot measurement at a time. Each of the sub-problems could be processed separately using the estimated value and then joined together into an iteration to calibrate the parameters and reduce the error caused by the substitution of estimated value for actual value in each part. Some simulation results and evaluation studies are reported and compared with the other two algorithms.
In Chapter 6, evaluations and comparisons of the three methods are provided. There are two kinds of evaluations: sensitivity to initial value and computation time.

In Chapter 7, a power system network model is built and tested. JSEPT algorithm is modified to solve the network problem and estimate the parameters of each line. The evaluation results are illustrated and discussed.

Chapter 8 presents the conclusions of the thesis and makes recommendations for future work.
Chapter 2

Parameters in Overhead Transmission Lines

Overhead transmission line parameters are crucial to the operation of power system. The inaccuracy of the parameters of transmission lines will result in disastrous impact on the analysis of the power system, such as the state estimation, short circuit calculation, power flow calculation, fault analysis, relay settings, fault location and so on. Usually, transmission lines are fully transposed, by the means of practically transposition. However, the untransposed transmission lines are also widely existed in the current power system. In this chapter, the properties of transmission line parameters are illustrated.

2.1 Parameters in Three Phase Transmission Line

It is fundamentally important to acquire the precise knowledge of parameters in three phase power transmission line since it is the most useful and practical way to transmit electricity in the power system. For a typical three phase transmission line, the following data indicates the geometric information that influences the characteristic line parameters.

Conductor: the body part of the transmission line which can carry electricity along its length. The major characters of conductors include: wire size, conductor material, area of conductor, number of aluminum strands, number of steel strands, number of aluminum layers, DC resistance, AC resistance, inductive reactance and capacitive reactance.
Tower Configuration: steel tower to support an overhead power line. The values of the tower configuration are the following: tower height, conductor spacing, phase spacing and conductors per bundle.

Line Length: the distance of the transmission line. The units are miles when using English system, or kilometers when using the Metric (SI) system.

Power Base: the system power base in MVA.

Voltage Base: the line-line voltage base in KV.

Impedance Base: the impedance base in Ohms. This value is computed when the power base and the voltage base are entered or modified.

Admittance Base: the admittance base in Siemens. It is also computed as the inverse of the impedance base.

2.1.1 Three Phase Transmission Line Model

Based on the geometric information above, the model of a three phase transmission line with mutual impedance is shown in Figure 2.1 [27].

![Figure 2.1 Model of three phase transmission line](image)

Current and voltage at each end of the transmission line are assumed to be measured via various measurement tools, such as PMU. In the following discussions in this section, all the voltages and currents are in complex form in order to build equivalent PI model. The measurements can be written in vector format as:
The power system model for transmission lines is developed from the conventional distributed parameter model. There are four kinds of line parameters in distributed model: \( r \) (resistance matrix per unit length), \( l \) (inductance matrix per unit length), \( c \) (capacitance matrix per unit length) and \( g \) (conductance matrix per unit length), which are in per unit length.

Usually, the conductance of the overhead transmission line is neglected since it is so small that has little effect on the calculation of the parameters, but for underground cables it is not negligible. In this thesis, the overhead transmission line is discussed, therefore the conductance is neglected. However, the proposed algorithms in the following chapters also apply for the underground cable case with proper modifications.

For a single phase line, these distributed parameters are scalars. For a three phase transmission line, \( r, l, \) and \( c \) are all matrices in the same size (usually 3 by 3). Also note that all these line parameter matrices are symmetric [28]. In such case, \( r, l, \) and \( c \) can be expressed in matrix form that

\[
\mathbf{r} = \begin{bmatrix}
    r_{1} & r_{2} & r_{3} \\
    r_{2} & r_{4} & r_{5} \\
    r_{3} & r_{5} & r_{6}
\end{bmatrix}
\]
where the \( \{r,l,c\}_{i,j} \) indicate:

- \( i = j \) self-resistance/inductance/capacitance;
- \( i \neq j \) mutual resistance/inductance/capacitance with \( \{r,l,c\}_{i,j} = \{r,l,c\}_{j,i} \).

The series impedance matrix and shunt admittance matrix per unit length can be expressed as

\[
\begin{align*}
    z &= \begin{bmatrix}
        z_1 & z_2 & z_3 \\
        z_2 & z_4 & z_5 \\
        z_3 & z_5 & z_6
    \end{bmatrix}
    = r + j\omega l
    = \begin{bmatrix}
        r_1 + j\omega l_1 & r_2 + j\omega l_2 & r_3 + j\omega l_3 \\
        r_2 + j\omega l_2 & r_4 + j\omega l_4 & r_5 + j\omega l_5 \\
        r_3 + j\omega l_3 & r_5 + j\omega l_5 & r_6 + j\omega l_6
    \end{bmatrix}
    \\
    y &= \begin{bmatrix}
        y_1 & y_2 & y_3 \\
        y_2 & y_4 & y_5 \\
        y_3 & y_5 & y_6
    \end{bmatrix}
    = j\omega c
    = \begin{bmatrix}
        c_1 & c_2 & c_3 \\
        c_2 & c_4 & c_5 \\
        c_3 & c_5 & c_6
    \end{bmatrix}
\end{align*}
\]

where

- \( z \) series impedance per unit length.
- \( y \) shunt admittance per unit length to neutral.
- \( \omega \) system frequency.

Given a transmission line, the distributed-parameter model should be introduced since any transmission line should be regarded as the combination of infinite small sections. But for the purpose of parameter identification, total impedance and total admittance of a certain length transmission line are needed. To derive such parameters, besides the distributed parameter model,
two other simplified models are also provided under the constraint of proper length. Assume the line length is denoted as \( L \).

2.1.2 Transmission Line Type

2.1.2.1 Medium-length line \( 50 \leq L < 150 \) miles

A nominal PI equivalent circuit is introduced to model the medium length transmission line. Shown in Figure 2.1, PI model is a reliable model for distributed parameter transmission line, the impedance matrix and the admittance matrix can be regarded as the series impedance of the line consisting of the series resistance and inductive reactance, and the admittance corresponding to the shunt capacitance placed at each end of the line, respectively.

![Figure 2.2 Equivalent PI model of medium-length three phase transmission line](image)

In the above PI model, the series impedance and shunt admittance satisfy:

\[
Z_{\text{series}} = Z \cdot L
\]  
\[
Y_{\text{shunt}} = Y \cdot L
\]

The relationship between the currents and voltages at both end of the line in the PI model can be expressed as:

\[
I_k + I_m = \frac{Y_{\text{shunt}}}{2} (V_k + V_m)
\]  
\[
V_k - V_m = Z_{\text{series}} \cdot I_{\text{pi}}
\]

where
\[ I_{zp} = I_k - \frac{Y_{\text{shunt}}}{2} \cdot V_k = -I_m + \frac{Y_{\text{shunt}}}{2} \cdot V_m. \]  

\[ (2.14) \]

From equation (2.12), (2.13) and (2.14), the following result can be derived:

\[
\begin{bmatrix}
I_k \\
I_m
\end{bmatrix} = \begin{bmatrix}
\frac{Y_{\text{shunt}}}{2} + \frac{1}{Z_{\text{series}}} & -\frac{1}{Z_{\text{series}}} \\
-\frac{1}{Z_{\text{series}}} & \frac{Y_{\text{shunt}}}{2} + \frac{1}{Z_{\text{series}}}
\end{bmatrix} \begin{bmatrix}
V_k \\
V_m
\end{bmatrix}.
\]

\[ (2.15) \]

In the above equation (2.15), the relationship between currents, voltages and line parameters are shown explicitly. Actually, the matrix consisted of \(Z_{\text{series}}\) and \(Y_{\text{shunt}}\) is one of the most important matrices for network description of the interconnected power system, named bus admittance matrix, and denoted as \(Y_{\text{bus}}\). The \(Y_{\text{bus}}\) represents the relationship between the injected currents at the nodes of the interconnected network and the voltages at the nodes. Typically, the \(Y_{\text{bus}}\) represents only one single phase at each node. But here, all three phases are taken into consideration. For this two nodes system, it holds

\[
Y_{\text{bus}} = \begin{bmatrix}
\frac{Y_{\text{shunt}}}{2} + \frac{1}{Z_{\text{series}}} & -\frac{1}{Z_{\text{series}}} \\
-\frac{1}{Z_{\text{series}}} & \frac{Y_{\text{shunt}}}{2} + \frac{1}{Z_{\text{series}}}
\end{bmatrix}
\]

\[ (2.16) \]

where the \(Y_{\text{bus}}\) has the following properties:

- \(Y_{\text{bus}}\) is symmetric.

- \(Y_{\text{bus}}^{(i,j)} (Y_{\text{bus}}^{(1,1)} \text{ and } Y_{\text{bus}}^{(2,2)})\) is the self-admittance matrix, which is equal to the sum of the branch admittances of all the components connected to \(i\)-th node.

- \(Y_{\text{bus}}^{(i,j)} (Y_{\text{bus}}^{(1,2)} \text{ and } Y_{\text{bus}}^{(2,1)})\) is equal to the negative of the branch admittance of all components connected between node \(i\) and node \(j\).

Note that in equation (2.15), \(I_k, I_m, V_k\) and \(V_m\) are vectors of size 3 by 1. The matrix \(Y_{\text{bus}}\) that correlates the currents and voltages is in the size of 6 by 6.
2.1.2.2 Short line, \( L < 50 \) miles

The situation is the same as for medium-length line except the ignorance of capacitance, i.e. \( Y_{\text{shunt}} = 0 \).

2.1.2.3 Long line, \( L \geq 150 \) miles

For a long three phase transmission line, the distributed parameter model is the only suitable model since the prerequisite to simplify the model of short and medium length line does not satisfy the long line. But still, the equivalent PI circuit model can still be used with appropriate modification.

The basic equations of a three phase line are written as:

\[
\frac{d^2 V}{dx^2} = zyV \tag{2.17}
\]
\[
\frac{d^2 I}{dx^2} = yzI \tag{2.18}
\]

where \( z \) and \( y \) are per unit length longitudinal impedance and shunt admittance matrices of the line, derived from equation (2.8) and (2.9), respectively. The vectors \( V \) and \( I \) are voltages and currents in the phase. Note that the above two equations are distance dependency, therefore the voltages and currents are the function of distance \( x \). Given the information:

- \( x = 0 \): \( V = V_k \) and \( I = I_k \),
- \( x = L \): \( V = V_m \) and \( I = I_m \),

general solutions of equation (2.17) and (2.18) can be determined using standard method of solving linear ordinary second order differential equation [29].

\[
V_k = V_m \cdot \cosh(\gamma L) - Z \cdot I_m \cdot \sinh(\gamma L) \tag{2.19}
\]

\[
I_k = \frac{1}{Z} \cdot V_m \cdot \sinh(\gamma L) - I_m \cdot \cosh(\gamma L) \tag{2.20}
\]

where

\( L \), line length.
The propagation function is given by \( \gamma = \sqrt{z \cdot y} \).

The characteristic impedance is \( Z_c = \frac{\bar{z}}{\sqrt{y}} \).

Therefore, introduce the PI model again, shown in the following figure.

![Figure 2.3](image)

Figure 2.3  Equivalent PI model of long three phase transmission line

With the consideration of equation (2.19) and (2.20), \( Z' \) and \( Y' \) can be expressed as:

\[
Z' = Z_c \cdot \sinh(\gamma L) = z \cdot L \cdot \frac{\sinh(\gamma L)}{\gamma L} \tag{2.21}
\]

\[
\frac{Y'}{2} = \frac{1}{Z_c} \cdot \tanh\left(\frac{\gamma L}{2}\right) = \frac{y \cdot L}{2} \cdot \frac{\tanh(\gamma L/2)}{\gamma L/2} \tag{2.22}
\]

and for the three phase \( Y_{bus} \) matrix, it holds

\[
Y_{bus} = \begin{bmatrix}
\frac{Y'}{2} + \frac{1}{Z'} & -\frac{1}{Z'} \\
-\frac{1}{Z'} & \frac{Y'}{2} + \frac{1}{Z'}
\end{bmatrix} \tag{2.23}
\]

From the above discussions under different length of the transmission line, the total impedance and admittance of a transmission line can be expressed by the \( r, l \) and \( c \) in equation (2.5), (2.6) and (2.7). This indicates that that with the precise knowledge of \( Y_{bus} \) matrix, the line parameters can be fully identified, which provides a basis for the later discussion.
2.2 Transposition of Transmission Line

2.2.1 Transposition

According to the physical transposition of the phase conductors, transmission line can be divided into two categories: transposed transmission line and untransposed transmission line.

Transposition can be partly and easily fulfilled in a transmission line by manipulating the relative position of the conductors of individual phases. In Figure 2.4, a 500-kV three phase, double-circuit transmission line is provided. The colored dot in red, blue and yellow stands for the phase a, b and c, respectively. Each side of the tower - each circuit - produces a magnetic field that oscillates backwards and forwards. For the left pylon where the phases are in the same order from top to bottom on the two sides, the two magnetic fields produced by two parallel circuits are in the same direction, so they add up. On the other hand, for the right pylon, the phases on two sides are in the opposite direction. This makes the magnetic fields neutralized from each other. This kind of cancellation is not exactly perfect because there is inevitable leakage. But this still largely reduces the magnetic field intensity, thus decreases the mutual inductance and finally make the system symmetric.

![Figure 2.4][1]

Figure 2.4 An example of three phase transmission line phasing

Besides the way of manipulating the position of phases, another way to keep system symmetric is the transposition tower. Transposition tower is a kind of transmission tower that changes the relative position of the conductors of an overhead transmission line. For a long transmission line,
transposition tower will be settled at same distance, usually 25km. A simple transposition example is shown in the following Figure 2.5.

![Figure 2.5 Transposition of the transmission lines](image)

The above figure can be regarded as three conductors in the same vertical plane. In the first section where the phases are in the sequence of 'a-b-c' from top to bottom, it is just as shown in the left circuits of the right tower of Figure 2.4. After a certain distance, these three conductors will change their relative position to ‘c-a-b’. And this change will repeat once again over a same distance. Figure 2.5 can also be regarded as three conductors in the same horizontal plane or even a view of three conductors not in the same plane.

The transposition ensures the symmetric capacitance of a three phase line, and to some extent, ensures the balanced operation of the power system by physically manipulating the three phase transmission line to be symmetric over a long distance. Also, transposing is an effective measure for the reduction of inductively linked normal mode interferences.

For transposed transmission line, line parameters in each phase are identical, which means the line parameter matrix $r$, $l$ and $c$ in equation (2.5), (2.6) and (2.7) have only two independent unknowns. One of the two unknowns in each matrix is at the diagonal and the other one is at the off-diagonal, as shown in the following:

$$
\begin{bmatrix}
    r_a & r_b & r_b \\
    r_b & r_a & r_b \\
    r_b & r_b & r_a \\
\end{bmatrix}, \quad
\begin{bmatrix}
    l_a & l_b & l_b \\
    l_b & l_a & l_b \\
    l_b & l_b & l_a \\
\end{bmatrix}, \quad
\begin{bmatrix}
    c_a & c_b & c_b \\
    c_b & c_a & c_b \\
    c_b & c_b & c_a \\
\end{bmatrix}.
$$

Correspondingly, the impedance matrix and admittance matrix can be rewritten as

$$
\begin{bmatrix}
    z_a & z_b & z_b \\
    z_b & z_a & z_b \\
    z_b & z_b & z_a \\
\end{bmatrix}
$$

(2.24)
\[ y = \begin{bmatrix} y_a & y_b & y_b \\ y_b & y_a & y_b \\ y_b & y_b & y_a \end{bmatrix} \]  

(2.25)

In this case, in the \( \mathbf{Y}_{\text{bus}} \) matrix defined in (2.16) (or (2.23)), there are 6 real unknowns (or say 3 complex unknowns with independent real and imaginary part each).

In this thesis, the conductance is ignored since when the line is not that long and voltage level is not that high, conductance is very small. For some other discussions, the conductance would not be ignored for the following reason: it gives a general solution for the untransposed three phase line parameters, but not restricted to the specific lines or models.

The simulation results in the following chapters show that using the proposed algorithm, the calculated (estimated) conductance is very small that can be regarded as zero. And this proves the correctness of the algorithms as well.

### 2.2.2 Untransposed Transmission Line

In practice, most high voltage systems are nearly balanced and can be modeled only in the positive sequence. However, in a power transmission system, due to the following reasons, the transposition of lines may not be fulfilled.

- The transmission line is not designed to have such transposition structure. Usually, the reason is that the voltage level is not high enough. And this kind of lines may also carry critical power flows in the system.
- Many old lines are not transposed but still in use.
- There are geometry reasons at the ‘T’ point of three transmission lines. For example,
From line A, the red phase is at the top. Therefore, line B also has red phase at the top on its left-hand circuit. To make line B transposed, red phase should be at the bottom on the right hand circuit. Thus line C has red phase at the bottom of its left-hand circuit and has to have red phase at the top of its right-hand circuit to keep it transposed. But this means line A has red phase at the top of both circuits - it is untransposed. So, when three lines meet at a "T" point, one can have only two of them transposed, but not all three.

• The installation of new lines or dismantle of old lines will influence the mutual inductance of electric circuits and the transposition may disappear.

• Accidental non-transposition of lines due to the weather (including temperature, rain and moisture), human factor or failure of transposition equipment.

Under the untransposed circumstance, voltage drops of different magnitudes exist in three conductors. This difference in voltage drops is caused by the unequal inductance of three phases (mutual inductance of conductors are different in untransposed lines). Also due to asymmetrical spacing the magnetic fields external to the conductors is not zero.

According to the analysis of section 2.1 , (2.5), (2.6) and (2.7) show that the line parameter matrices are symmetric. For untransposed transmission line, the six elements of each matrix (3 in diagonal, 3 in the rest position) are independent from each other. This can be explained as: the mutual resistance/inductance/capacitance/conductance are not identical between two different lines. Thus, the number of overall real value unknowns is 18. They are:

\[ r_1, r_2, r_3, r_4, r_5, r_6, l_1, l_2, l_3, l_4, l_5, l_6, c_1, c_2, c_3, c_4, c_5, c_6. \]

Compared with 6 unknown parameters for the transposed transmission line, there are 18 parameters of the untransposed line which make the parameter identification much harder. The calculation of parameters of transmission line are derived from equation (2.15). One prerequisite should be fulfilled that the current and voltage must be synchronized. In reality, GPS technique, and the introduction of PMU, can employ this synchronization approach.
To calculate parameters, the known variables are the measured currents and voltages through appropriate equipment, e.g., PMUs. Typically, the measurements are in the format of magnitudes and phases, but these can be transformed to the real and imaginary form easily. Here all the measurements are in real and imaginary form to keep consistent format with the line parameters. There are 24 quantities that can be measured at the same time

\[
\begin{align*}
V_{kar}, V_{kbr}, V_{kcr}, V_{mar}, V_{mbr}, V_{mcr}, I_{kar}, I_{kbr}, I_{kcr}, I_{mar}, I_{mbr}, I_{mcr}, \\
V_{kai}, V_{kbi}, V_{kci}, V_{mai}, V_{mbi}, V_{mci}, I_{kai}, I_{kbi}, I_{kci}, I_{mai}, I_{mbi}, I_{mci}.
\end{align*}
\]

For the line parameters, they can also be written in the complex form that

\[
\begin{align*}
z_i = r_i + j \omega l_i & \quad (i = 1 \ldots 6) \\
y_i = j \omega c_i & \quad (i = 1 \ldots 6).
\end{align*}
\]

In the following, the relationship between the parameters and elements of \( Y_{bus} \) is illustrated specifically for the untransposed case and then proved. For simplicity in the derivations, the medium line model is chosen because the whole line impedance/admittance is a linear function of distributed line parameters which needs to be specified. Re-write equation (2.15) (or equation (2.23) for an alternative line model), the following equations can be derived.

\[
\begin{bmatrix}
I_{ka} \\
I_{kb} \\
I_{kc} \\
I_{ma} \\
I_{mb} \\
I_{mc}
\end{bmatrix}
= Y_{bus}
\begin{bmatrix}
V_{ka} \\
V_{kbr} \\
V_{kcr} \\
V_{mar} \\
V_{mbr} \\
V_{mcr}
\end{bmatrix}
\]

where
\[
Y_{\text{bus}} = \begin{bmatrix}
Y_1 & Y_2 & Y_3 & Y_4 & Y_5 & Y_6 \\
Y_2 & Y_7 & Y_8 & Y_5 & Y_6 & Y_{10} \\
Y_3 & Y_8 & Y_{11} & Y_6 & Y_{10} & Y_{12} \\
Y_4 & Y_5 & Y_6 & Y_1 & Y_2 & Y_3 \\
Y_5 & Y_9 & Y_{10} & Y_2 & Y_7 & Y_8 \\
Y_6 & Y_{10} & Y_{12} & Y_3 & Y_8 & Y_{11}
\end{bmatrix}
\]

(2.29)

\[
\begin{bmatrix}
Y_4 \\
Y_5 \\
Y_6
\end{bmatrix} = -\frac{1}{Z}
\]

(2.30)

\[
\begin{bmatrix}
Y_1 \\
Y_2 \\
Y_3
\end{bmatrix} = \frac{Y + \frac{1}{Z}}{2}
\]

(2.31)

It is obvious that the number of unknowns in the \( Y_{\text{bus}} \) is equal to number of line parameters, which is 18. Since \( Y_{\text{bus}} \) is a nonlinear combination of those parameters, the following expressions illustrate the tight relationship between the elements of \( \{Z, Y\} \), \( \{z, y\} \), \( \{r, l, c\} \) and the elements of \( Y_{\text{bus}} \).

\[
Z = zL = (r + j\omega)L
\]

(2.32)

\[
Y = yL = j\omega L
\]

(2.33)

\[
\text{det}(Y) = (y_1y_4y_6 + 2y_2y_3y_5 - y_3^2y_4 - y_2^2y_6 - y_5^2y_1)L^3
\]

\[
= -j(\omega L)^3(c_1c_4c_6 + 2c_2c_3c_5 - c_3^2c_4 - c_2^2c_6 - c_1^2c_5)
\]

(2.34)

\[
\text{det}(Z) = (z_1z_4z_6 + 2z_2z_3z_5 - z_3^2z_4 - z_2^2z_6 - z_1^2z_5)L^3
\]

(2.35)

Define

\[
f(a, b, c) \triangleq f_r(a, b, c) + jf_j(a, b, c) = [r_{ab}r_{bc} - \omega^2(r_{ab}l_{bc} + l_{ab}r_{bc} + l_{ab}l_{bc})] + j\omega[l_{ab}r_{bc} + r_{ab}l_{bc} + r_{ab}l_{bc} - \omega^2l_{ab}l_{bc}]
\]

(2.36)

thus,
\[
\text{det}(Z) = \left[ f(1,4,6) + 2f(2,3,5) - f(3,3,4) - f(2,2,6) - f(5,5,1) \right] L^3 \\
= \left[ f(1,4,6) + 2f(2,3,5) - f(3,3,4) - f(2,2,6) - f(5,5,1) \right] \\
+ j \left[ f(1,4,6) + 2f(2,3,5) - f(3,3,4) - f(2,2,6) - f(5,5,1) \right] L^3 \\
= \{ \text{det}_r(Z) + j \text{det}_i(Z) \} L^3
\] (2.37)

What's more, the elements of \( Y_{bus} \) are derived as follows.

\[
Y_1 = \frac{(z_1 z_6 - z_4^2) L^2}{\text{det}(Z)} + \frac{y_1L}{2} = -Y_4 + \frac{y_1L}{2} 
\] (2.38)

\[
Y_2 = \frac{(z_2 z_5 - z_5^2) L^2}{\text{det}(Z)} + \frac{y_2L}{2} = -Y_5 + \frac{y_2L}{2} 
\] (2.39)

\[
Y_3 = \frac{(z_3 z_6 - z_6^2) L^2}{\text{det}(Z)} 
\] (2.40)

\[
Y_4 = \frac{(z_4^2 - z_4 z_6) L^2}{\text{det}(Z)} 
\] (2.41)

\[
Y_5 = \frac{(z_5^2 - z_5 z_5) L^2}{\text{det}(Z)} 
\] (2.42)

\[
Y_6 = \frac{(z_6^2 - z_6 z_6) L^2}{\text{det}(Z)} 
\] (2.43)

\[
Y_7 = \frac{(z_7 z_6 - z_7 z_7) L^2}{\text{det}(Z)} + \frac{y_7L}{2} = -Y_9 + \frac{y_7L}{2} 
\] (2.44)

\[
Y_8 = \frac{(z_8 z_5 - z_8 z_5) L^2}{\text{det}(Z)} 
\] (2.45)

\[
Y_9 = \frac{(z_9^2 - z_9 z_6) L^2}{\text{det}(Z)} 
\] (2.46)
\begin{align}
Y_{10} &= \frac{(z_1z_5 - z_2z_3)L^2}{\det(Z)} \\
Y_{11} &= \frac{(z_1z_4 - z_2^2)L^2}{\det(Z)} + \frac{y_6L}{2} = -Y_{12} + \frac{y_6L}{2} \\
Y_{12} &= \frac{(z_2^2 - z_1z_4)L^2}{\det(Z)}.
\end{align}

Equation (2.38) to (2.49) indicate the nonlinear relationship between the elements of $Y_{\text{bus}}$ and elements of \{r, l, c\}. Since \{r, l, c\} each have 6 real value unknowns, there are also 18 real value unknowns in the matrix $Y_{\text{bus}}$. Note that $Y_1$ in (2.38) and $Y_4$ in (2.41) have the common term $(z_5^2 - z_4z_6)L^2/\det(Z)$ which is complex, but the extra term $y_1L$ has only imaginary part. Therefore, the real part of $Y_1$ is identical to the negative real part of $Y_4$, which indicates there are only three independent unknowns in the pair \{Y_1, Y_4\}. This relation also holds among \{Y_2, Y_5\}, \{Y_3, Y_6\}, \{Y_7, Y_9\}, \{Y_8, Y_{10}\} and \{Y_{11}, Y_{12}\}.

According to the discussion above, a parameter vector $p$ is formulated containing some of the real part and imaginary parts of the elements \{Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7, Y_8, Y_9, Y_{10}, Y_{11}, Y_{12}\}. To distinguish the transmission line parameters \{r, l, c, z, y\} and the parameters $p$, we would name $p$ as modified parameters. From the next chapter, only modified parameters are discussed and analyzed, therefore, $p$ is abbreviated as parameter. The use of real part and imaginary part separately meets the requirement of the decoupled model in state estimation. It also holds the consistent form as the line parameters which are real values. Similarly, $I_k$ and $I_m$ can formulate a real vector $I$, and $V_k$ and $V_m$ can formulate a real vector $V$. In the following expressions, ‘r’ and ‘i’ in the subscript of each element denotes the real part and imaginary part of this element, respectively.

\begin{align}
p &= [Y_{1r}, Y_{1i}, Y_{2r}, Y_{2i}, Y_{3r}, Y_{3i}, Y_{4r}, Y_{4i}, Y_{5r}, Y_{5i}, Y_{6r}, Y_{6i}, Y_{7r}, Y_{7i}, Y_{8r}, Y_{8i}, Y_{9r}, Y_{9i}, Y_{10r}, Y_{10i}, Y_{11r}, Y_{11i}, Y_{12r}, Y_{12i}]^T \\
V &= [V_{kar}, V_{kai}, V_{kbr}, V_{kbi}, V_{kcr}, V_{cki}, V_{mar}, V_{mai}, V_{mnr}, V_{mni}, V_{mcr}, V_{mci}]^T
\end{align}
\[ I = \begin{bmatrix} I_{kar} & I_{kai} & I_{kbr} & I_{kbi} & I_{kcr} & I_{kci} & I_{mar} & I_{mai} & I_{mbr} & I_{mbi} & I_{mcr} & I_{mci} \end{bmatrix}^T \] (2.52)

The size of the vector \( p \), \( I \) and \( V \) are shown in Table 1.1 as follows.

<table>
<thead>
<tr>
<th>Vector</th>
<th>Size</th>
<th>Illustration</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>12 * 1</td>
<td>currents’ 12 real &amp; 12 imaginary</td>
</tr>
<tr>
<td>V</td>
<td>12 * 1</td>
<td>voltages’ 12 real &amp; 12 imaginary</td>
</tr>
<tr>
<td>( p )</td>
<td>18 * 1</td>
<td>( Y_{bus} )’s 6 real &amp; 12 imaginary</td>
</tr>
</tbody>
</table>

\[ Y_{bus} = \begin{bmatrix} Y_1 & Y_2 & Y_3 & Y_4 & Y_5 & Y_6 \\ Y_7 & Y_8 & Y_9 & Y_{10} & Y_{11} & Y_{12} \\ Y_3 & Y_8 & Y_{11} & Y_6 & Y_{10} & Y_{12} \\ Y_4 & Y_5 & Y_6 & Y_7 & Y_8 & Y_3 \\ Y_5 & Y_9 & Y_{10} & Y_2 & Y_7 & Y_8 \\ Y_6 & Y_{10} & Y_{12} & Y_3 & Y_8 & Y_{11} \end{bmatrix} \]

\[ Y_{bus} = \begin{bmatrix} p_1 + jp_2 & p_3 + jp_4 & p_5 + jp_6 & -p_1 + jp_3 & -p_3 + jp_5 & -p_5 + jp_7 & -p_1 + jp_3 & -p_3 + jp_5 & -p_5 + jp_7 & -p_1 + jp_3 & -p_3 + jp_5 & -p_5 + jp_7 \\ p_3 + jp_4 & p_{10} + jp_{11} & p_{12} + jp_{13} & -p_3 + jp_8 & -p_{10} + jp_{14} & -p_{12} + jp_{15} & -p_3 + jp_8 & -p_{10} + jp_{14} & -p_{12} + jp_{15} & -p_3 + jp_8 & -p_{10} + jp_{14} & -p_{12} + jp_{15} \\ p_5 + jp_6 & p_{12} + jp_{13} & p_{16} + jp_{17} & -p_5 + jp_9 & -p_{12} + jp_{15} & -p_{16} + jp_{18} & -p_5 + jp_9 & -p_{12} + jp_{15} & -p_{16} + jp_{18} & -p_5 + jp_9 & -p_{12} + jp_{15} & -p_{16} + jp_{18} \\ -p_1 + jp_3 & -p_3 + jp_5 & -p_5 + jp_7 & p_1 + jp_2 & p_3 + jp_4 & p_{10} + jp_{11} & p_3 + jp_4 & p_{10} + jp_{11} & p_3 + jp_4 & p_{10} + jp_{11} & p_3 + jp_4 & p_{10} + jp_{11} \\ -p_3 + jp_4 & -p_{10} + jp_{11} & -p_{12} + jp_{15} & p_3 + jp_4 & p_{10} + jp_{11} & p_{12} + jp_{13} & p_3 + jp_4 & p_{10} + jp_{11} & p_{12} + jp_{13} & p_3 + jp_4 & p_{10} + jp_{11} & p_{12} + jp_{13} \\ -p_5 + jp_6 & -p_{12} + jp_{13} & -p_{16} + jp_{17} & p_5 + jp_6 & p_{12} + jp_{13} & p_{16} + jp_{18} & p_5 + jp_6 & p_{12} + jp_{13} & p_{16} + jp_{18} & p_5 + jp_6 & p_{12} + jp_{13} & p_{16} + jp_{18} \end{bmatrix} \] (2.53)

where

\[ p_1 = \frac{1}{L} \frac{\det_i(Z) \left( r_5^2 - r_5^2 - \omega^2 l_r^2 + \omega^2 l_s^2 \right) + \det_i(Z) \left( \omega l_r^2 r_6 + \omega l_s^2 r_4 - 2 \omega l_s r_5 \right)}{\left( \det_i(Z) \right)^2 + \left( \det_i(Z) \right)^2} \] (2.54)

\[ p_2 = -p_1 + \frac{\omega c_L}{2} \] (2.55)
\[ p_3 = \frac{1}{L} \frac{\det_r(Z) r_5 r_6 - \omega^2 l_3 l_5 + \omega^2 l_3 l_6}{(\det_r(Z))^2 + (\det_i(Z))^2} + \frac{\det_i(Z) (\omega l_3 r_5 + \omega r_3 l_5 - \omega l_3 r_6 - \omega r_3 l_6)}{(\det_r(Z))^2 + (\det_i(Z))^2} \] (2.56)

\[ p_4 = -p_8 + \frac{\omega c_y L}{2} \] (2.57)

\[ p_5 = \frac{1}{L} \frac{\det_r(Z) (r_5 r_6 - \omega^2 l_3 l_6 + \omega^2 l_3 l_5)}{(\det_r(Z))^2 + (\det_i(Z))^2} + \frac{\det_i(Z) (\omega l_3 r_5 + \omega r_3 l_5 - \omega l_3 r_6 - \omega r_3 l_6)}{(\det_r(Z))^2 + (\det_i(Z))^2} \] (2.58)

\[ p_6 = -p_9 + \frac{\omega c_z L}{2} \] (2.59)

\[ p_7 = \frac{1}{L} \frac{-\det_r(Z) (\omega l_3 r_5 + \omega r_3 l_5 - 2 \omega l_3 r_6 + \omega r_3 l_6) + \det_i(Z) (r_5 r_6 - r_3^2 - \omega^2 l_4 l_5 + \omega^2 l_3 l_4)}{(\det_r(Z))^2 + (\det_i(Z))^2} \] (2.60)

\[ p_8 = \frac{1}{L} \frac{-\det_r(Z) (\omega l_3 r_5 + \omega r_3 l_5 - \omega l_3 r_6 - \omega r_3 l_6) + \det_i(Z) (r_5 r_6 - r_3^2 - \omega^2 l_4 l_5 + \omega^2 l_3 l_4)}{(\det_r(Z))^2 + (\det_i(Z))^2} \] (2.61)

\[ p_9 = \frac{1}{L} \frac{-\det_r(Z) (\omega l_3 r_5 + \omega r_3 l_5 - \omega l_3 r_6 - \omega r_3 l_6) + \det_i(Z) (r_5 r_6 - r_3^2 - \omega^2 l_4 l_5 + \omega^2 l_3 l_4)}{(\det_r(Z))^2 + (\det_i(Z))^2} \] (2.62)

\[ p_{10} = \frac{1}{L} \frac{\det_r(Z) (r_5 r_6 - r_3^2 - \omega^2 l_3 l_6 + \omega^2 l_3 l_5)}{(\det_r(Z))^2 + (\det_i(Z))^2} + \frac{\det_i(Z) (\omega l_3 r_5 + \omega r_3 l_5 - 2 \omega l_3 r_6 + \omega r_3 l_6)}{(\det_r(Z))^2 + (\det_i(Z))^2} \] (2.63)

\[ p_{11} = -p_{14} + \frac{\omega c_y L}{2} \] (2.64)

\[ p_{12} = \frac{1}{L} \frac{\det_r(Z) (r_5 r_6 - r_3^2 - \omega^2 l_3 l_6 + \omega^2 l_3 l_5)}{(\det_r(Z))^2 + (\det_i(Z))^2} + \frac{\det_i(Z) (\omega l_3 r_5 + \omega r_3 l_5 - \omega l_3 r_6 - \omega r_3 l_6)}{(\det_r(Z))^2 + (\det_i(Z))^2} \] (2.65)

\[ p_{13} = -p_{15} + \frac{\omega c_z L}{2} \] (2.66)

\[ p_{14} = \frac{1}{L} \frac{-\det_r(Z) (\omega l_3 r_5 + \omega r_3 l_5 - 2 \omega l_3 r_6 + \omega r_3 l_6) + \det_i(Z) (r_5 r_6 - r_3^2 - \omega^2 l_4 l_5 + \omega^2 l_3 l_4)}{(\det_r(Z))^2 + (\det_i(Z))^2} \] (2.67)
\[ p_{18} = \frac{1}{L} \frac{-\det_r(Z) \left( \omega l_r r_3 + \omega r_5 l_3 - \omega l_r r_5 - \omega r_5 l_5 \right) + \det_r(Z) \left( r_5 r_5 - r_5 r_5 - \omega^2 l_2 l_3 + \omega^2 l_5 l_5 \right)}{(\det_r(Z))^2 + (\det_r(Z))^2} \]  

(2.68)

\[ p_{16} = \frac{1}{L} \frac{\det_r(Z) \left( r_5 r_5 - r_5^2 - \omega^2 l_2 l_4 + \omega^2 l_5 l_5 \right) + \det_r(Z) \left( \omega l_r r_4 + \omega r_7 l_4 - 2\omega l_r r_2 \right)}{(\det_r(Z))^2 + (\det_r(Z))^2} \]  

(2.69)

\[ p_{17} = -p_{18} + \frac{\omega c L}{2} \]  

(2.70)

\[ p_{18} = \frac{1}{L} \frac{-\det_r(Z) \left( \omega l_r r_4 + \omega r_7 l_4 - 2\omega l_r r_2 \right) + \det_r(Z) \left( r_5 r_5 - r_5^2 - \omega^2 l_2 l_4 + \omega^2 l_5 l_5 \right)}{(\det_r(Z))^2 + (\det_r(Z))^2}. \]  

(2.71)

From the above equations (2.53) to (2.71), the transmission line parameters can be transformed into the parameter vector \( p \), whose elements consist the \( Y_{bus} \). It is obvious that the elements of \( p \) are independent from each other. For parameter identification, (2.28) is a good model since it contains all the measurement information as well as the parameter information. But this only gives the estimate of the parameter vector \( p \). To obtain the line parameters \( r, l \) and \( c \), it is also necessary to clarify the function of line parameters with respect to the parameter vector \( p \).

\[
\begin{bmatrix}
  r_1 + j \omega l_1 \\
  r_2 + j \omega l_2 \\
  r_3 + j \omega l_3 \\
  r_5 + j \omega l_5 \\
  r_6 + j \omega l_6
\end{bmatrix}
= \frac{1}{L} \cdot \text{inv}
\begin{bmatrix}
  p_1 - j p_7 \\
  p_3 - j p_8 \\
  p_5 - j p_9 \\
  p_7 - j p_{10} \\
  p_9 - j p_{11} \\
  p_11 - j p_{12} \\
  p_{13} - j p_{14} \\
  p_{15} - j p_{16} \\
  p_{17} - j p_{18}
\end{bmatrix}
\]  

(2.72)

The above two equations show that how to derive line parameters from the modified parameters. Usually, the admittance matrix and impedance matrix of the transmission line are more useful for further system topology analysis, these can be derived as

\[
\begin{bmatrix}
  c_1 \\
  c_2 \\
  c_3 \\
  c_4 \\
  c_5 \\
  c_6
\end{bmatrix}
= \frac{2}{\omega L} \cdot \begin{bmatrix}
  p_2 + p_7 \\
  p_4 + p_8 \\
  p_6 + p_9 \\
  p_{11} + p_{14} \\
  p_{13} + p_{15} \\
  p_{17} + p_{18}
\end{bmatrix}.
\]  

(2.73)
\[
Z = \begin{bmatrix}
p_1 - jp_7 & p_3 - jp_8 & p_5 - jp_9 \\
p_3 - jp_8 & p_{10} - jp_{14} & p_{12} - jp_{15} \\
p_5 - jp_9 & p_{12} - jp_{15} & p_{16} - jp_{18}
\end{bmatrix}^{-1}
\]

\[
\frac{Y}{2} = j \begin{bmatrix} p_2 + p_7 \\ p_4 + p_8 \\ p_6 + p_9 \\ p_{11} + p_{14} \\ p_{13} + p_{15} \\ p_{17} + p_{18} \end{bmatrix}.
\]

Therefore, from the discussion above, we have the following conclusions.

- Transmission line parameters and the modified parameters (i.e., \( p \)) can be transformed to each other easily.

- The number of independent variables for the untransposed line parameters is the same as in the modified parameter vector \( p \).

Therefore, the problem of line parameter identification can be transformed to the problem of estimating the modified parameter \( p \).

### 2.3 Simulation Model

The construction of simulation model includes two steps: the construction of the transmission line model, and the system model.

For the transmission line model, the case of untransposed transmission tower in Figure 2.4 is used. It is a 200 km long, 500-kV three-phase, double-circuit line. 6 bundles of 3 ACSR conductors along with 2 steel ground wires. A practical image for this kind of transmission tower is shown in Figure 2.7.
To specify the parameters of this model, the following figure gives the basic geometric information of the transmission tower.

Figure 2.8 Geometric information of 500kV transmission line tower
Two kinds of software are introduced, ATP-EMTP and Simulink, are introduced for the transmission line model construction. The model is primarily built based on ATP and the actual values of the parameters are then figured out using Simulink.

The system model of a long transmission line with the sources and loads is also built in ATP as well.

2.3.1 Transmission Line Model in ATP

Alternative transients program (ATP) is a universal program system for digital simulation of transient phenomena of electromagnetic as well as electromechanical nature. With this digital program, complex networks and control systems of arbitrary structure can be simulated. ATP has extensive modelling capabilities and additional important features besides the computation of transients.

LCC (Line constants, Cable constants and Cable parameters) is a template of ATP that can build multi-phase, multi-circuit long transmission line with appropriate line parameter inputs. The parameters in LCC are defined in the following figures.

![Figure 2.9 LCC line model definition](image)

---

**Figure 2.9** LCC line model definition
In Figure 2.9, the system model is defined and constructed based on the following data.

**System type:** Overhead lines.

**Transposed:** The overhead line is transposed if button is checked.

**Auto bundling:** When checked, each phase is assumed to be bundled with several conductors, which will reduce the power loss caused by Corona effect.

**Skin effect:** Skin effect is the tendency of an AC to become distributed within a conductor such that the current density is largest near the surface of the conductor, and decreases with depths in the conductor.

**Seg. ground:** Segmented ground wires. If button is unchecked then the ground wires are assumed to be continuously grounded.

**Real trans. matrix:** If checked, the transformation matrix is assumed to be real. The eigenvectors of the transformation matrix are rotated closer to the real axis so that their imaginary part is assumed to become negligible.

**Rho:** The ground resistivity measured in ohm*m of the homogeneous earth (Carson's theory). 100 ohm*m indicates a slight corrosiveness of the soil.

![Table of LCC line model data](image-url)
Freq. init: Frequency at which the line parameters will be calculated. Set the same value as the standard power frequency.

Length: Length of overhead line in (m/km/miles).

In Figure 2.10, the geometrical and material data for overhead line conductors are specified.

Phase no.: Phase number. Ground wire = 0 (eliminated).

Rin: Inner radius of one conductor. Also defined as the geometric mean radius (GMR) of bundle.

Rout: Outer radius (cm or inch) of one conductor.

RESIS: Conductor resistance at DC (with skin effect) or at Freq. Init. (no skin effect).

Horiz: Horizontal distance (m or foot) from the center of bundle to a user selectable reference line.

VTower: Vertical bundle height at tower (m or foot).

VMid: Vertical bundle height at mid-span (m or foot). The height \( h = \frac{2}{3} \times \text{VMid} + \text{VTower} \) is used in the calculations.

Separ: Distance between conductors in a bundle (cm or inch)

Alpha: Angular position of one of the conductors in a bundle, measured counter-clockwise from the horizontal line.

NB: Number of conductors in a bundle.

From the above conductor and geometry information, the distributed parameters of the transmission line can be computed.

### 2.3.2 Powergui Computation of RLC Parameters

The Matlab Powergui provides the command to compute RLC parameters of overhead transmission line from its conductor characteristics and tower geometry. Figure 2.11 provides the parameters in Powergui command window.
The computed RLC parameters are shown below, where the susceptance is neglected.

\[
\begin{bmatrix}
0.097816 & 0.084442 & 0.082491 \\
0.084442 & 0.089701 & 0.079190 \\
0.082491 & 0.079190 & 0.086803 \\
\end{bmatrix} \text{ (ohm / km)}
\]

\[
\begin{bmatrix}
1.20630 & 0.72393 & 0.65795 \\
0.72393 & 1.16120 & 0.72815 \\
0.65795 & 0.72815 & 1.21510 \\
\end{bmatrix} \text{ (mH / km)}
\]

\[
\begin{bmatrix}
18.5951 & -5.4928 & -2.3424 \\
-5.4928 & 20.9222 & -4.7463 \\
-2.3424 & -4.7463 & 20.5130 \\
\end{bmatrix} \text{ (nF / km)}
\]

According to the medium-length line PI model, the \( Y_{bus} \) matrix can be calculated as

\[
Z_{series} = z \cdot L = (r + jol) \cdot L
\]

\[
\begin{bmatrix}
19.5633 + j90.9528 & 16.8884 + j54.5829 & 16.4981 + j49.6082 \\
16.8884 + j54.5829 & 17.9401 + j87.5549 & 15.8379 + j54.9009 \\
16.4981 + j49.6082 & 15.8379 + j54.9009 & 17.3606 + j91.6166 \\
\end{bmatrix}
\]
\[ Y_{\text{shunt}} = y \cdot L = (j \omega c) \cdot L \]
\[ = j \begin{bmatrix} 1.4020 & -0.4141 & -0.1766 \\ -0.4141 & 1.5774 & -0.3579 \\ -0.1766 & -0.3579 & 1.5467 \end{bmatrix} \times 10^3 \]

Therefore,
\[
Y_{\text{bus}} = \begin{bmatrix} \frac{Y_{\text{shunt}} + 1}{2} \frac{1}{Z_{\text{series}}} - \frac{1}{Z_{\text{series}}} \\ -\frac{1}{Z_{\text{series}}} \frac{Y_{\text{shunt}} + 1}{2} \end{bmatrix}
\]
\[
= \begin{bmatrix} 0.1450 - j1.8034 & -0.0122 + j0.8497 & 0.0172 + j0.5133 \\ -0.0122 + j0.8497 & 0.1486 - j2.1274 & -0.0226 + j0.8521 \\ 0.0172 + j0.5133 & -0.0226 + j0.8521 & 0.1152 - j1.7962 \end{bmatrix}
\]
\[
= \begin{bmatrix} -0.1450 + j1.8735 & 0.0122 - j0.8704 & -0.0172 - j0.5201 \\ 0.0122 - j0.8704 & -0.1486 + j2.2063 & 0.0226 - j0.8700 \\ -0.0172 - j0.5201 & 0.0226 - j0.8700 & -0.1152 + j1.7962 \end{bmatrix} \times 10^2
\]

If the long line circuit is introduced, which is proposed in equation (2.21), (2.22) and (2.23), the following results can be derived:

\[
\gamma = \sqrt{z \cdot y} = \begin{bmatrix} 0.1176 + j1.5179 & 0.0729 + j0.2348 & 0.0904 + j0.3090 \\ 0.0790 + j0.2349 & 0.1035 + j1.5096 & 0.0824 + j0.3332 \\ 0.0788 + j0.2308 & 0.0662 + j0.2529 & 0.1121 + j1.6437 \end{bmatrix} \times 10^3
\]

\[
Z_c = \sqrt{z \cdot y} = \begin{bmatrix} 2.6587 - j0.2548 & 1.2218 - j0.2088 & 0.9237 - j0.1828 \\ 1.2300 - j0.2023 & 2.5027 - j0.2293 & 1.0890 - j0.1707 \\ 1.0396 - j0.1982 & 1.1993 - j0.1907 & 2.4862 - j0.2028 \end{bmatrix} \times 10^2
\]
\[ Z' = Z_e \cdot \sinh(yL) \]
\[
= \begin{bmatrix}
19.1584 + j89.7650 & 16.6802 + j54.0482 & 16.2542 + j49.0633 \\
16.6572 + j54.0402 & 17.6036 + j86.4481 & 15.5801 + j54.2560 \\
16.2960 + j49.1658 & 15.6450 + j54.3918 & 16.9295 + j90.1897 \\
\end{bmatrix}
\]

\[ Y' = \frac{2}{Z_c} \cdot \tanh\left(\frac{\gamma L}{2}\right) \]
\[
= \begin{bmatrix}
0.0022 + j1.4139 & -0.0010 - j0.4186 & -0.0007 - j0.1795 \\
-0.0010 - j0.4186 & 0.0023 + j1.5910 & -0.0010 - j0.3627 \\
-0.0008 - j0.1795 & -0.0010 - j0.3627 & 0.0026 + j1.5621 \\
\end{bmatrix} \times 10^{-3}
\]

\[ \mathbf{Y}_{bus}' = \begin{bmatrix}
\frac{Y'}{2} + \frac{1}{Z'} \\
\frac{1}{Z'} & \frac{Y'}{2} + \frac{1}{Z'}
\end{bmatrix} \]
\[
= \begin{bmatrix}
0.1408 - j1.8034 & -0.0092 + j0.8658 & 0.0197 + j0.5225 \\
-0.0095 + j0.8659 & 0.1442 - j2.1667 & -0.0201 + j0.8701 \\
0.0198 + j0.5233 & -0.0197 + j0.8727 & 0.1105 - j1.8353 \\
-0.1408 + j1.907 & 0.0092 - j0.8868 & -0.0197 - j0.5314 \\
0.0198 + j0.5323 & 0.0197 + j0.8868 & 0.1105 + j1.8353 \\
\end{bmatrix} \times 10^{-2}
\]

The above calculations show that, the result of \( \mathbf{Y}_{bus} \) matrix using the medium line model is quite close to the one using the long line model. However, these calculations are not precise because all of these models are an approximation of the practical scenario, and there exists a difference between the actual distributed line and the pi-model.

To obtain the actual \( \mathbf{Y}_{bus} \), one method is to calculate the positive, zero and negative sequence parameters by using the measured voltages and currents. But this only uses data from one scan, i.e. one time shot, which is not convincible for all the data point and it is more applicable for the
transposed transmission line. Therefore, for the following scenarios, the actual values are obtained from

- If the parameters are constant, the parameters are calculated by formulating the least-squares problem for the measurements of all the time shots.
- If the parameters are changing, for each value of the parameters, we simulate more measurements under the same parameters and obtain the actual values as above.

### 2.3.3 Simulation Test Cases

A three phase transmission line model, noted as model 1, is shown in Figure 2.12 (see Appendix A for interpretations of symbols/figures in ATP). It shows one LCC (Line constants, Cable constants and Cable parameters) transmission line in the middle, with the sending end noted as ‘SEND’ and receiving end noted as ‘RECV’. The measurements on these two nodes are the ones that simulation cares about. Besides the transmission line, there is an ideal AC voltage source in one end of the system, along with some three phase RLC components in order to simulate the loss in the generator and transformers. Also, on the other side of system there are also several RLC components to simulate the loads.

![Three phase transmission line model based on ATP](image)

**Figure 2.12** Three phase transmission line model based on ATP

Based on the simulation model above, there are two test cases to be discussed. These test cases simulate three different circumstances in real power system

- Constant line parameters;
- Continuous changing parameters.

The following illustrations, figures and tables show how each test case will work as well as the actual values of parameters. To illustrate the results simply, the real and imaginary part of the first
element of $\mathbf{Y}_{\text{bus}}$ (i.e. $p_1$ and $p_2$ in (2.53)) are figured separately. The changing parameter case is made by manipulation of the geometry information in Figure 2.10. Note that the values following are in per unit representation with proper voltage and current base.

2.3.3.1 Test case 1: constant line parameters

Under this circumstance, the transmission line parameters do not change with time. However, the loads, line voltages and currents are changing rapidly, which simulates the realistic scenarios happens in a system. The actual values of test case 1 are shown in the following table. Note that these values are in per unit with the voltage base 500kV and current base 500A.

<table>
<thead>
<tr>
<th></th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$p_3$</th>
<th>$p_4$</th>
<th>$p_5$</th>
<th>$p_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>1.5963</td>
<td>-19.2150</td>
<td>-0.2044</td>
<td>9.3367</td>
<td>0.1641</td>
<td>5.4220</td>
</tr>
<tr>
<td></td>
<td>$p_7$</td>
<td>$p_8$</td>
<td>$p_9$</td>
<td>$p_{10}$</td>
<td>$p_{11}$</td>
<td>$p_{12}$</td>
</tr>
<tr>
<td>Value</td>
<td>19.9661</td>
<td>-9.5670</td>
<td>-5.5170</td>
<td>1.6780</td>
<td>-22.9671</td>
<td>-0.3143</td>
</tr>
<tr>
<td></td>
<td>$p_{13}$</td>
<td>$p_{14}$</td>
<td>$p_{15}$</td>
<td>$p_{16}$</td>
<td>$p_{17}$</td>
<td>$p_{18}$</td>
</tr>
</tbody>
</table>

In Figure 2.13 and Figure 2.14, the values of $p_1$ and $p_2$ are given with respect to time.
2.3.3.2 Test case 2: continuous change parameters

In this case, line parameters change smoothly. By arbitrarily changing the specific parameters in Figure 2.10 (here the changing parameter is the $V_{mid}$ in the $7^{th}$ column of that table), the parameters in $p$ are changing with time. There are 20 time shot measurements available, with the $V_{mid}$ value of phase 1, 2 and 3 continuously decreasing.

The actual values of such changing parameters are obtained by

- At one time shot, taking several different set of measurements corresponding to the same $V_{mid}$ value. Then calculating the parameters needed.
- Re-do the above procedure for all the time scans.

The reason to take several different measurements in step 1 is that, the calculation will meet singularity problem if there is only one set of measurements. This is also the largest obstacle to estimate the parameters for untransposed transmission line.

The following two figures show the trajectory of the changing parameter $p_1$ and $p_2$, with respect to time.

Figure 2.15  Actual value of $p_1$ test case 2

Figure 2.16  Actual value of $p_2$ test case 2
2.3.4 Simulation Procedure

By simulating in ATP, all the measurements are obtained. These measurements are then modified by

- Transformed to per unit value.
- Adding appropriate white noise with zero mean but different standard deviation. This would be categorized into 6 scenario.
  
  - Scenario 1-0: test case 1 without adding measurement noise;
  - Scenario 1-1: test case 1 with measurement noise with $10^{-4}$ standard deviation;
  - Scenario 1-2: test case 1 with measurement noise with $10^{-5}$ standard deviation;
  - Scenario 2-0: test case 2 without adding measurement noise;
  - Scenario 2-1: test case 2 with measurement noise with $10^{-4}$ standard deviation;
  - Scenario 2-2: test case 2 with measurement noise with $10^{-5}$ standard deviation;

Such white noises added to the measurements are fixed when tested by different algorithms in order to guarantee the consistence of the simulated cases and scenarios.

In the following chapters, these scenarios will be discussed as the simulation results. Several algorithms are proposed to solve the parameter estimation problem under the six scenarios. The simulations aim to evaluate the performance of each algorithm, according to the following kinds of figures or quantities.

- The values of $p_1$ and $p_2$ are figured out, comparing with the actual values of each case in Figure 2.13, Figure 2.14, Figure 2.15 and Figure 2.16.

- Mean absolute percentage error (MAPE) which is defined by

$$MPE^{(i)} = \frac{1}{M} \sum_{k=1}^{M} \left| \frac{\hat{p}_{k}^{(i)} - p_{\text{actual},k}^{(i)}}{p_{\text{actual},k}^{(i)}} \right|$$

(2.78)

where the superscript $(i)$ indicates the $i$-th parameter in vector $p$, $k$ indicates the time, and $M$ is the total number of sampling measurements.
• Normalized root-mean-squared-error (NRMSE) which is

\[
NRMSE^{(i)} = \frac{RMSE^{(i)}}{\frac{1}{M} \sum_{k=1}^{M} P_{\text{actual},i,k}^{(i)}} = \sqrt{\frac{1}{M} \sum_{k=1}^{M} (\hat{P}_{k}^{(i)} - P_{\text{actual},i,k}^{(i)})^2} \quad (2.79)
\]

• Nonlinear correlation coefficient (CC) for test case 2 only (since it is \( \infty \) for test case 1)

\[
CC = \frac{\text{cov} (\hat{p}, p)}{\sigma_{\hat{p}} \sigma_{p}}
\]

where \( \sigma \) stands for standard deviation, is a criterion of how two variables are linearly related. If CC is close to 1, it indicates the estimated variables track the actual values closely.
Chapter 3

Static Parameter Estimation

3.1 Formulation of Measurement Model

The general model for the power system network relating the measurement vector $z$, state vector $x$ and parameter vector $p$ is shown in the following equation

$$z = h(x, p) + e$$

(3.1)

where

- $z$ vector including all measurements.
- $x$ state vector consisting of bus voltages of interest.
- $p$ vector consists of parameters.
- $e$ vector of measurement noise, $E[e] = 0$ and $\text{cov}(e) = R$.
- $h$ nonlinear function correlates the states, parameters and measurements.

Equation (3.1) also illustrates the nonlinear relationship between the states, parameters and measurements. Note the states here are the bus voltages.
3.2 Static Implementation of Parameter Estimation

3.2.1 Model Construction

One straightforward way to calculate the transmission line parameters is to take these parameters as the ones to be estimated using WLS (weighted least square) technique. This requires us to linearize (3.1) and then choose appropriate values as the measurements.

Revisiting (2.28) that

\[
\begin{bmatrix}
I_{ka} \\
I_{kb} \\
I_{kc} \\
I_{ma} \\
I_{mb} \\
I_{mc}
\end{bmatrix} = \mathbf{V}_{bus}^{-1} \cdot 
\begin{bmatrix}
V_{ka} \\
V_{kb} \\
V_{kc} \\
V_{ma} \\
V_{mb} \\
V_{mc}
\end{bmatrix},
\]

it is not difficult to find the following relation by rewriting the above equation as

\[
I = H_v p
\]  

(3.2)

where

\(H_v\) matrix consists of voltages, \(H_v \in \mathbb{R}^{m \times n}\).

\(p\) vector consists of line parameters, as shown in (2.50), \(p \in \mathbb{R}^{n \times 1}\).

\(I\) vector consists of currents, \(I \in \mathbb{R}^{m \times 1}\).

\[
I = [I_{kAr}, I_{kAi}, I_{kBr}, I_{kBi}, I_{kCi}, I_{mAa}, I_{mAi}, I_{mBa}, I_{mBi}, I_{mCi}]^T
\]  

(3.3)

\[
p = [p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, p_{16}, p_{17}, p_{18}]^T.
\]  

(3.4)

The subscript here denotes:

\(k/m\) the k-end or the m-end.

\(A/B/C\) the three phase.
In equation (3.2), if the matrix $H_v$ and $I$ are available, the parameters $p$ can be calculated. However, this needs two prerequisites.

**a) The voltage measurements must be reliable.**

According to (3.1), the measurements are not only a function of line parameters, but also a function of states. However, if we make the assumption that the measured voltages are reliable without any noise, the voltage measurements can be regarded as the states directly. By choosing the current measurements as the measurements in (3.1), the following equation can be derived

$$ I = h(V, p) + e $$

(3.5)

where

$$ V = \begin{bmatrix} V_{kAr} & V_{kAi} & V_{kBr} & V_{kBi} & V_{kCr} & V_{kCi} & V_{mA} & V_{mA} & V_{mBr} & V_{mBi} & V_{mCr} & V_{mCi} \end{bmatrix}^T $$

(3.6)

Considering the relationship between the parameters in (3.2), (3.5) is rewritten as

$$ I = H_v p + e $$

(3.7)

which indicates the measurements are linear functions of the parameters. The coefficient matrix $H_v$ consists of the voltage measurements and is derived from (2.28), (2.53), (3.2) and (3.3). It can also be regarded as the Jacobian matrix in the weighted-least-square (WLS) estimation algorithm. Here gives one example of the derivation of $H_v$.

$$ I_{ka} = \mathbf{Y}_{bus}(1,:) \cdot \begin{bmatrix} V_{ka} & V_{kb} & V_{kc} & V_{ma} & V_{mb} & V_{mc} \end{bmatrix}^T $$

$$ \begin{bmatrix} p_1 + j p_2 & p_3 + j p_4 & p_5 + j p_6 & -p_1 + j p_7 & -p_3 + j p_8 & -p_5 + j p_9 \end{bmatrix}^T $$

$$ = \left[ V_{kar} + j V_{kai} - V_{kbr} + j V_{kbi} - V_{kcr} + j V_{kci} - V_{mar} + j V_{mai} - V_{mbr} + j V_{mbi} - V_{mcr} + j V_{mci} \right]^T $$

$$ = \left[ (V_{kar} - V_{mar}) p_1 - V_{kai} p_2 + (V_{kbr} - V_{mbr}) p_3 - V_{kbi} p_4 + (V_{kcr} - V_{mcr}) p_5 - V_{kci} p_6 $$

$$ -V_{mai} p_7 - V_{mbi} p_8 - V_{mci} p_9 \right] $$

$$ + j \left[ (V_{kai} - V_{ma}) p_1 + V_{kbr} p_2 + (V_{kbi} - V_{mbi}) p_3 + V_{kcr} p_4 + (V_{kci} - V_{mbi}) p_5 + V_{kci} p_6 $$

$$ + V_{mai} p_7 + V_{mbi} p_8 + V_{mci} p_9 \right] $$

Therefore,
\[ I_{kar} = (V_{kar} - V_{mai} - V_{mar}) p_1 - V_{kar} p_2 + (V_{kbi} - V_{mbi}) p_3 - V_{mbi} p_4 + (V_{kci} - V_{mbi}) p_5 - V_{kbi} p_6 + V_{mai} p_7 - V_{mbi} p_8 - V_{mci} p_9 \]

\[
\begin{bmatrix}
V_{kar} & -V_{mai} & V_{mbi} & V_{kbi} & -V_{mbi} & V_{kci} & -V_{mbi} & -V_{mci} & 0 \\
-V_{kai} & V_{kar} & V_{kbi} & -V_{mbi} & V_{kbi} & -V_{mbi} & V_{kci} & -V_{mbi} & 0 \\
V_{kbr} & -V_{mbi} & V_{kbr} & V_{kci} & -V_{mbi} & V_{mbr} & V_{mbr} & V_{mcr} & 0 \\
0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} \\
\end{bmatrix}
\]

\[ I_{kai} = (V_{kar} - V_{mai}) p_1 + V_{kar} p_2 + (V_{kbi} - V_{mbi}) p_3 + V_{mbi} p_4 + (V_{kci} - V_{mbi}) p_5 + V_{kbi} p_6 + V_{mai} p_7 + V_{mbi} p_8 + V_{mci} p_9 \]

\[
\begin{bmatrix}
V_{kar} & -V_{mai} & V_{mbi} & V_{kbi} & -V_{mbi} & V_{kci} & -V_{mbi} & -V_{mci} & 0 \\
V_{kai} & V_{kar} & V_{kbi} & -V_{mbi} & V_{kbi} & -V_{mbi} & V_{kci} & -V_{mbi} & 0 \\
V_{kbr} & -V_{mbi} & V_{kbr} & V_{kci} & -V_{mbi} & V_{mbr} & V_{mbr} & V_{mcr} & 0 \\
0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} & 0_{9x1} \\
\end{bmatrix}
\cdot p \]

By doing such calculations for all the elements of \( I \) in (3.3), \( H_\text{f} \) can be written as follows.

\[
H_\text{f}^T =
\begin{bmatrix}
V_{kar} & V_{kar} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
V_{mar} & V_{mai} & V_{mai} & V_{mai} & 0 & 0 & 0 & 0 & 0 \\
-V_{kai} & V_{kar} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
V_{kbi} & V_{kbi} & V_{kar} & V_{kai} & 0 & 0 & 0 & 0 & 0 \\
V_{mbi} & V_{mbi} & V_{mar} & V_{mai} & 0 & 0 & 0 & 0 & 0 \\
-V_{kbi} & V_{kbi} & V_{kbr} & V_{kar} & V_{kai} & 0 & 0 & 0 & 0 \\
-V_{mbi} & V_{mbi} & -V_{mai} & V_{mar} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]
\begin{align*}
V_{kcr} & \quad V_{kci} & 0 & 0 & V_{kar} & V_{kai} & V_{mcr} & V_{mci} & 0 & 0 & V_{mar} & V_{mai} \\
V_{mcr} & \quad V_{mci} & 0 & 0 & V_{mar} & V_{mai} & V_{kcr} & V_{kci} & 0 & 0 & V_{kar} & V_{kai} \\
-\ V_{kci} & \quad V_{kcr} & 0 & 0 & -V_{kai} & V_{kar} & -V_{mci} & V_{mcr} & 0 & 0 & -V_{mai} & V_{mar} \\
-\ V_{mai} & \quad V_{mar} & 0 & 0 & 0 & 0 & -V_{kai} & V_{kar} & 0 & 0 & 0 & 0 \\
-\ V_{mci} & \quad V_{mcr} & 0 & 0 & -V_{mai} & V_{mar} & -V_{kci} & V_{kcr} & 0 & 0 & -V_{kai} & V_{kar} \\
0 & 0 & V_{kbi} & V_{kbi} & 0 & 0 & 0 & 0 & -V_{kbi} & V_{kbi} & 0 & 0 \\
0 & 0 & -V_{kci} & V_{kcr} & -V_{kbi} & V_{kbr} & 0 & 0 & -V_{mci} & V_{mcr} & -V_{mbi} & V_{mbr} \\
0 & 0 & -V_{mci} & V_{mcr} & -V_{mbi} & V_{mbr} & 0 & 0 & 0 & 0 & -V_{kbi} & V_{kbr} & 0 & 0 \\
0 & 0 & -V_{mci} & V_{mcr} & -V_{mbi} & V_{mbr} & 0 & 0 & -V_{kci} & V_{kcr} & -V_{kbi} & V_{kbr} \\
0 & 0 & 0 & 0 & V_{kci} & V_{kcr} & 0 & 0 & 0 & 0 & V_{mci} & V_{mcr} \\
0 & 0 & 0 & 0 & -V_{kci} & V_{kcr} & 0 & 0 & 0 & 0 & -V_{mci} & V_{mcr} \\
0 & 0 & 0 & 0 & -V_{mci} & V_{mcr} & 0 & 0 & 0 & 0 & -V_{kci} & V_{kcr}
\end{align*}

b) \textit{H}, must be full column rank.

\(H_r\), built in the previous section, needs to be a full column rank matrix. This can be explained from two aspects.
From the point view of linear algebra, equation (3.2) can be regarded as a non-homogeneous linear equation. Therefore, the rank, especially the column rank of the coefficient matrix $H_v$ decides whether the unknown $p$ would have a single unique solution, no solution or infinitely many solutions. If and only if $rank(H_v) = n$ where $H_v \in \mathbb{R}^{mn}$, it has the unique solution.

From the point view of regression analysis, the approximate solution of the parameters can be decided by the least squares technique, but this requires the matrix $H_v^T H_v$ is invertible, which is equivalent to $rank(H_v) = n$.

Under different type of transmission lines, the circumstances are different.

For transposed transmission line, the number of parameters is 6, as illustrated in section 2.2.1. Therefore, based on the following model

$$I = H_v p_{\text{transposed}} + e$$

where $p \in \mathbb{R}^{6 \times 1}$, $I \in \mathbb{R}^{12 \times 1}$, $H_v \in \mathbb{R}^{12 \times 6}$ and $e \in \mathbb{R}^{12 \times 1}$ denotes the measurement noise.

Mathematically speaking, $p$ could be solved by applying weighted least squares technique.

**Weighted Least squares Technique:**

The weighted least square (WLS) [30] state estimation procedure can make use of the redundant measurements in order to filter out the measurement noise. Given the measurement model

$$z = Hx + e$$

To obtain the ‘best’ fits of the data, the parameter values can be determined by minimizing

$$J = (z - Hx)^T R^{-1} (z - Hx)$$

where $R = \text{cov}(e)$. To find the minimum of $J$, it holds

$$\frac{\partial J}{\partial x} = 0$$

Apply Karush–Kuhn–Tucker conditions,
Thus,

\[ \hat{x} = (H^T R^{-1} H)^{-1} H^T R^{-1} z. \]

Therefore, for the transposed line, the parameters are calculated as

\[ \hat{p}_{\text{transposed}} = (H_1^T R^{-1} H_1)^{-1} H_1^T R^{-1} I. \] (3.9)

For untransposed transmission line, the number of parameters is 18 as discussed in the previous chapter. In such case, (3.7) is used as the measurement model, with \( p \in \mathbb{R}^{18 \times 1}, I \in \mathbb{R}^{12 \times 1}, e \in \mathbb{R}^{12 \times 1} \) and \( H_v \in \mathbb{R}^{12 \times 18} \). Since the \( H_v \) matrix is not in full column rank, such WLS method cannot be applied directly to solve the parameters. Therefore, some other techniques are required.

### 3.2.2 Sliding Window Method

In a single measurement sample, the number of measurements is 12, while the number of parameters to be estimated is 18. It is mathematically unsolvable as discussed before.

Usually, the measurements are measured from appropriate measure tools in a very high frequency. If during some time interval from time \( n - T \) to time \( n \), the parameters are considered as constants, then the use of multi-scan measurements to estimate parameters is achievable.

In this case, static parameter estimation can be solved based on a sliding window method. It constructs a window that contains the measurements measured from time \( n - T + 1 \) to time \( n \), denotes as \( z(t_{n-T+1}) \) to \( z(t_n) \). The index \( T \) is referred to as the foot step of the estimation. Using the measurements inside this sliding window, new measurement vector can be formulated as

\[
\begin{bmatrix}
z(t_{n-T+1}) \\
\vdots \\
z(t_{n-1}) \\
z(t_n)
\end{bmatrix}
\]
Therefore, new measurement model can be derived.

\[
\begin{bmatrix}
I(t_{n-T+1}) \\
\vdots \\
I(t_{n}) \\
I(t_{n+1})
\end{bmatrix}_{n=12T+1} = \begin{bmatrix}
H_x(t_{n-T+1}) \\
\vdots \\
H_x(t_n) \\
H_x(t_{n+1})
\end{bmatrix}_{12T+1} \cdot p(t_n) + e(t_n)
\]

(3.10)

In order to apply the WLS method to estimate \( p \), the following equation should be satisfied.

\[ 12T > 18 \]

Therefore, if the foot step satisfies \( T > 2 \), the results at time \( n \), i.e. \( \hat{p}(t_n) \), can be calculated as

\[
\hat{p}(t_n) = \left( \begin{bmatrix}
H_x(t_{n-T+1}) \\
\vdots \\
H_x(t_n)
\end{bmatrix}_{12T+1} \right)^T \cdot R^{-1} \cdot \left( \begin{bmatrix}
H_x(t_{n-T+1}) \\
\vdots \\
H_x(t_n)
\end{bmatrix}_{12T+1} \right)^{-1} \cdot \begin{bmatrix}
H_x(t_{n-T+1}) \\
\vdots \\
H_x(t_n)
\end{bmatrix}_{12T+1} \cdot p(t_n) + e(t_n)
\]

(3.11)

Then, the window moves forward one step, as shown in the measurement model

\[
\begin{bmatrix}
I(t_{n-T+2}) \\
\vdots \\
I(t_{n+1})
\end{bmatrix}_{12T+1} = \begin{bmatrix}
H_x(t_{n-T+2}) \\
\vdots \\
H_x(t_{n+1})
\end{bmatrix}_{12T+1} \cdot p(t_{n+1}) + e(t_{n+1})
\]

(3.12)

This process is to some extent the same as a stack to pop the oldest measurement set and push a new set into it. Figure 3.1 shows a sliding window with foot step \( T = 2 \) that covers the measurements from time 1 to time 2.

---

**Figure 3.1** Sliding window illustration

The value of foot step largely influences the calculation results and the ability to track the variation of parameters. The differences between measured values in consecutive time, i.e. \( z(t_{n-1}) \) and
could only be eliminated by increasing the value of foot step. However, the disadvantage of increasing foot step is obvious: it will cause a delay of real-time results and reduce the sensitivity to parameter variations. This will be discussed specifically in section 3.3 numerically.

3.3 Simulation Results for Static Parameter Estimation

For static parameter estimation (SPE), the simulation contains 20 measurement sets and the sliding window method is introduced. This method is evaluated from three aspects

- Different scenarios;
- Different foot step T;
- Measurement with and without noise.

3.3.1 Test Case 1 Simulations

The following figures show the implementation static parameter estimation (SPE) algorithm on test case 1.

- **Scenario 1-0, T = 4**

![Figure 3.2 SPE scenario 1-0 with T = 4](image)

- **Scenario 1-0, T = 8**
From the above three figures, Figure 3.2, Figure 3.3 and Figure 3.4, it is obvious that with the increasing of foot step, the estimation accuracy is improved. However, the results are still not the expected constant values, even though the measurement are directly measured from the simulation. The reason for that is as follows.

Rewrite equation (3.12) as

$$z_N = H_{v,N} p + e$$  \hspace{1cm} (3.13)
\[
H_{v,N} = \begin{bmatrix}
H_v(t_{n-T+2}) \\
\vdots \\
H_v(t_n) \\
H_v(t_{n+1})
\end{bmatrix}_{n=12T+18}^{n=12T+18}
\]
and
\[
z_N = \begin{bmatrix}
I(t_{n-T+2}) \\
\vdots \\
I(t_n) \\
I(t_{n+1})
\end{bmatrix}
\]
It is obvious from the above table that the estimation results are not as good as the actual constant value of the parameters. But considering the small squared residual of the estimated parameters, static estimation still provides a considerably good estimate. Note that the above discussions and results are all based on the fact that measurements are free of noise. In the following several results, scenario 1-2 is chosen as the noisy situation since this noise is smaller the 1-1. The parameter estimates are provided.

- **Scenario 1-2, T = 4**

![Figure 3.5 SPE scenario 1-2 with T = 4](image)
• Scenario 1-2, T = 8

![Figure 3.6](image1.png)

**Figure 3.6** SPE scenario 1-2 with T = 8

• Scenario 1-2, T = 12

![Figure 3.7](image2.png)

**Figure 3.7** SPE scenario 1-2 with T = 12

In the above Figure 3.5, Figure 3.6 and Figure 3.7, the results indicate that with the increasing of the foot step, the estimation accuracy is improved, which shares the same conclusion as the cases without measurement noise. But for the these scenarios that measurement are not free of noise, the estimates of parameters are much farther from the actual constant value, which indicates this static parameter estimation is not that tolerable with noise. Since the noise of scenario 1-2 has less standard deviation than scenario 1-1, there is no need to try on the latter scenario and the results of that would be much worse than scenario 1-1.
As a summary to the results given before, some conclusions can be made:

- Without measurement noises, the performance of SPE is acceptable;
- SPE can be used to deal with the cases with little measurement noise.
- Foot step largely decides the accuracy of the results.

### 3.3.2 Test Case 2 Simulations

For the introduction of foot step, though it overcomes the obstacle of singularity, it sacrifices the timeliness, i.e. the results will suffer latency to the change of the parameters. However, the fact is that static parameter estimation performs very badly for the parameter changing case. The following results give an example for this.

- **Scenario 2-0, T = 4**

  ![SPE scenario 2-0 with T = 4](image)

  Figure 3.8 SPE scenario 2-0 with T = 4

- **Scenario 2-0, T = 12**
Not surprising, the SPE gives totally wrong results for the changing parameter case, as shown in Figure 3.8 and Figure 3.9.

### 3.4 Chapter Summary

In this chapter, a static method to estimate parameters is proposed. It takes the measured voltages as the actual ones without noise, and uses them directly to formulate the measurement-to-parameter coefficient matrix, which can also be regarded as measurement Jacobian. Assume parameters are constant during a time interval, the sliding window method is introduced by augmenting the measurement vector with multi-scan data, which solves the problem of the singularity of the coefficient matrix. However, this method does not have good performance facing the measurement noise. What’s more, the use of parameter estimation alone ignores the uncertainties lie in the states, which is a practical problem in real world.

This section gives a general idea that how parameter estimation works and the obstacles that the parameter estimation of untransposed transmission line would have, and it somehow leads to the algorithm in chapter 5.
Chapter 4

Augmented State Estimation

4.1 Static Implementation of Augmented State Estimation

Revisiting the general model in (3.1) that

\[ z(t_k) = h(x(t_k), p(t_k)) + e(t_k) \]

(4.1)

one prevailing method to estimate parameter \( p \) is to augment the state \( x \) with the parameter \( p \). The measurements are the measured currents and voltages and the new state vector consists of the states and parameters. A static implementation has no information of how states vary with respect to time, but taking measurement model as the only equation for estimation. This process is shown explicitly by the following equations.

\[ z(t_k) = f(c(t_k)) + e(t_k) \]

(4.2)

where \( \text{cov}[e(t_k)] = R \). Note that the subscript \( k \) indicates the values at time \( t_k \). Also note, in the previous chapters, \( k \) denotes the one end of the transmission lines and exists in the subscripts of \( V \), \( I \) and \( x \). In the following discussion, subscript \( k \) is only used as the time index, except when there is specific illustration.

The new nonlinear function \( f(\cdot) \) has the Jacobian matrix \( F \) as
By introducing the WLS method, we have

$$
F(c(t_k)) = 2 \frac{\partial f}{\partial c(t_k)} = \begin{bmatrix} \frac{\partial f}{\partial x(t_k)} & \frac{\partial f}{\partial p(t_k)} \end{bmatrix}
$$

Only when the matrix \( G(c) = T^T(c)R^{-1}F(c) \) is non-singular, (4.2) is solvable. However, for untransposed transmission line, \( F(c) \in \mathbb{R}^{24 \times 30} \) which is not in full column rank. What’s more, the sliding window method does not make sense, since the states are always changing and cannot be regarded as constant. Another explanation is that, if the states are estimated using multi scan data, the states signal, with respect to time, are ‘flattened’. Therefore, the accuracy of these estimated states may be even lower than that of the measurements themselves.

Consequently, the static implementation of augmented state estimation could not achieve the desirable results. However, the idea of augmenting the states with parameters is helpful and will be modified in the later sections.

### 4.2 Basic Algorithm of Augmented State Estimation

Static augmented state estimation method has been discussed in previous section. In this section, a dynamic state model is provided, along with the measurement model, to deal with augmented state estimation. The dynamic state model is as follows.

$$
c_k = g(c_{k-1}) + v_k
$$

One considerable idea of constructing such function \( g(\cdot) \) is learning from the empirical data. In this thesis, we simply choose

$$
g(c) = c.
$$

Thus,

$$
c_k = c_{k-1} + v_k
$$
The general measurement model is given in (4.1) and the augmented measurement model is in (4.2). According to (4.2), function $f(\cdot)$ is highly nonlinear since the measurements equal to the product of states and parameters.

Therefore, given the dynamic state model (4.4) and the measurement model (4.2), a dynamic state estimator is applied to estimate the augmented states.

There are several methods ([31], [32]) that can be chosen as such state estimator:

- Extended Kalman filter.
- Iterated extended Kalman filter.
- Unscented Kalman filter [33].
- Iterated unscented Kalman filter.

For the first one, the augmented measurement function $f(\cdot)$ is highly nonlinear; therefore, in the measurement update process of the Kalman filter, the Jacobian matrix $F(\cdot)$ contains large error due to the linearization. Considering the general cases, EKF is not a good choice.

For the iterated extended Kalman filter (IEKF), the same linearization problem also exists.

For the unscented Kalman filter, it is always chosen as the optimal filter for the nonlinear system. UKF picks a set of sigma points and propagates them through the nonlinear function. There is no need to calculate the Jacobian, which is a difficult and inaccurate task. However, when applied to this augmented system, UKF performs not very well. One reason would be the weak observability of the system. Therefore, for specific application to the parameter estimation problem, UKF is not selected.

As a consequence, iterated unscented Kalman filter (IUKF) is proposed to solve this augmented parameter estimation problem.

### 4.3 Iterated Unscented Kalman Filter Implementation

An iterated unscented Kalman filter (IUKF) is proposed in this section to address the augmented parameter estimation problem. The main idea to do IUKF is to decide sigma points iteratively until some constrains are fulfilled. The outer iteration, whose index is $k$, processes every time when there
is measurement. The inner iteration with index \( j \), iteratively estimates the states and produces sigma points among the iteration \( k \).

IUKF can be realized by the following steps.

- **Step 1:**

  Revisit the state model and the measurement model that

  \[
  c_k = c_{k-1} + v_k \tag{4.5}
  \]

  \[
  z_k = f(c_k) + e_k \tag{4.6}
  \]

  where

  - \( c_k \) \quad \text{augmented state that} \quad \mathbf{c}_k = \begin{bmatrix} x_k \\ p_k \end{bmatrix}, \text{ where } x \in \mathbb{R}^{12 \times 1}, \quad p \in \mathbb{R}^{18 \times 1} \quad \text{and} \quad c \in \mathbb{R}^{30 \times 1}.

  - \( z_k \) \quad \text{measurement vector that} \quad \mathbf{z}_k = \begin{bmatrix} I_k \\ V_k \end{bmatrix}, \text{ where } I \in \mathbb{R}^{12 \times 1}, \quad V \in \mathbb{R}^{12 \times 1} \quad \text{and} \quad z \in \mathbb{R}^{24 \times 1}.

  - \( e_k \) \quad \text{measurement noise, } \quad e \in \mathbb{R}^{24 \times 1}, \text{ with covariance } \text{cov}(e) = R \in \mathbb{R}^{24 \times 24}.

  - \( v_k \) \quad \text{process noise, } \quad v \in \mathbb{R}^{30 \times 1} \text{ with } \text{cov}(v) = Q \in \mathbb{R}^{30 \times 30}.

  Actually, according to the structure of the \( \mathbf{c}_k \), and assuming the state \( x \) and the parameter \( p \) are uncorrelated, then the covariance of the process noise has the structured as follows.

  \[
  Q = \begin{bmatrix} Q_x & 0 \\ 0 & Q_p \end{bmatrix}
  \]

  where \( Q_x \in \mathbb{R}^{12 \times 12} \quad \text{and} \quad Q_p \in \mathbb{R}^{18 \times 18} \).

  Under the general UKF framework, the prediction can be obtained by following equations.

  \[
  \mathbf{x}_{0,k-\delta k-1} = \hat{\mathbf{c}}_{k-\delta k-1} \tag{4.7}
  \]
\[ x_{i,k-1|i-1} = \hat{c}_{k-1|i-1} + \left( \sqrt{(L + \gamma)} P_{k-1|i-1} \right)_{i=1} \]

\[ x_{i,k-1|i} = \hat{c}_{k-1|i} - \left( \sqrt{(L + \gamma)} P_{k-1|i} \right)_{i=1} \]

where \( x_{i,k-1|i-1} \) represents sigma point that (4.8) satisfies \( i = 1, 2, \cdots, L \) and (4.9) satisfies \( i = L + 1, L + 2, \cdots, 2L \).

L: dimension of the augmented state vector

\( \gamma = \alpha^2 (L + \kappa) - L \).

\( \alpha \): a determination of the spread of sigma points around the states, usually very small.

\( \kappa \): a secondary scaling parameter which is usually set to 0.

\( \left( \sqrt{Q} \right)_i \): the \( i \)th row of the square root of matrix \( Q \).

Now, according to the state equation \( (4.5) \), the time update of the UKF is

\[ x_{i,k|i-1} = x_{i,k-1|i-1} \]  \hspace{1cm} (4.10)

The weights for the states and covariance are given by

\[ w_0^{(m)} = \frac{\gamma}{L + \gamma} \]  \hspace{1cm} (4.11)

\[ w_0^{(c)} = \frac{\gamma}{L + \gamma} + \left( 1 - \alpha^2 + \beta \right) \]  \hspace{1cm} (4.12)

\[ w_i^{(m)} = w_i^{(c)} = \frac{\gamma}{2(L + \gamma)} \]  \hspace{1cm} (4.13)

where \( \kappa \) control the spread of the sigma points. \( \beta \) is related to the distribution of the states. The normal values are \( \alpha = 10^{-3} \), \( \beta = 2 \) and \( \kappa = 0 \). The weighted sigma points are recombined to produce the predicted state and covariance

\[ \hat{c}_k = \sum_{i=0}^{2L} w_i^{(m)} x_{i,k|i-1} \]  \hspace{1cm} (4.14)
\[ P_i^k = \sum_{i=0}^{2L} w_i^{(c)} \left[ \chi_{i,j} - \hat{c}_k^- \right] \left[ \chi_{i,j} - \hat{c}_k^- \right]^T + Q \] (4.15)

Then, measurement update process is

\[ \chi_{0,k|\text{\textendash}1} = \hat{c}_k^- \] (4.16)

\[ \chi_{i,k|\text{\textendash}1} = \hat{c}_k^- + \left( \sqrt{(L+\gamma)} P_{i,k|\text{\textendash}1} \right)_{i-L} \] (4.17)

\[ \chi_{i,k|\text{\textendash}1} = \hat{c}_k^- - \left( \sqrt{(L+\gamma)} P_{i,k|\text{\textendash}1} \right)_{i-L} \] (4.18)

\[ \gamma_{i,k|\text{\textendash}1} = f \left( \chi_{i,k|\text{\textendash}1} \right) \] (4.19)

\[ \hat{z}_k^- = \sum_{i=0}^{2L} w_i^{(m)} \gamma_{i,k|\text{\textendash}1} \] (4.20)

\[ P_{zz,k} = \sum_{i=0}^{2L} w_i^{(c)} \left[ \gamma_{i,k|\text{\textendash}1} - \hat{z}_k^- \right] \left[ \gamma_{i,k|\text{\textendash}1} - \hat{z}_k^- \right]^T + R \] (4.21)

\[ P_{xz,k} = \sum_{i=0}^{2L} w_i^{(c)} \left[ \gamma_{i,k|\text{\textendash}1} - \hat{c}_k^- \right] \left[ \gamma_{i,k|\text{\textendash}1} - \hat{z}_k^- \right]^T \] (4.22)

\[ K_k = P_{xz,k} P_{zz,k}^{-1} \] (4.23)

\[ \hat{c}_k = \hat{c}_k^- + g \cdot K_k \left( z_k - \hat{z}_k^- \right) \] (4.24)

\[ P_k = P_k^- - K_k P_{zz,k} K_k^T \] (4.25)

**Step 2:**

Define the iteration initial as

\[ \hat{c}_{k,0} = \hat{c}_k^- \] (4.26)

\[ P_{k,0} = P_k^- \] (4.27)
\[ \hat{c}_{k,1} = \hat{c}_k \]  
\[ P_{k,1} = P_k \]  

and \( g = 1 \), \( j = 2 \).

**Step 3:**
Generate new sigma points around the \( \hat{c}_{k,j-1} \) and do UKF again. A set of deterministic sample points with associated weights are generated as

\[ \chi_{0,j} = \hat{c}_{k,j-1} \]  
\[ \chi_{i,j} = \hat{c}_{k,j-1} + \left( \sqrt{(L + \gamma)P_{k,j-1}} \right) \]  
\[ \chi_{i,j} = \hat{c}_{k,j-1} - \left( \sqrt{(L + \gamma)P_{k,j-1}} \right) \]  

where \( \chi_{i,j} \) denotes the new sigma point. The weights of the states are the same as in (4.11) to (4.13).

**Step 4:**
The following time update and measurement update process are derived

\[ \tilde{c}_{k,j} = \sum_{i=0}^{2L} w_i^{(m)} \chi_{i,j} \]  
\[ \gamma_{i,j} = h(\chi_{i,j}) \]  
\[ \tilde{z}_{k,j} = \sum_{i=0}^{2L} w_i^{(m)} y_{i,j} \]  
\[ P_{zz,k,j} = \sum_{i=0}^{2L} w_i^{(c)} \left[ \gamma_{i,j} - \tilde{z}_{k,j} \right]\left[ \gamma_{i,j} - \tilde{z}_{k,j} \right]^T + R \]  
\[ P_{wc,k,j} = \sum_{i=0}^{2L} w_i^{(c)} \left[ \chi_{i,j} - \tilde{c}_{k,j} \right]\left[ \gamma_{i,j} - \tilde{z}_{k,j} \right]^T \]
\[ K_{k,j} = P_{zz,k,j} P_{zz,k,j}^{-1} \quad (4.38) \]

\[ \hat{c}_{k,j} = \hat{c}_{k,j}^- + g \cdot K_{k,j} \left( z_k - \hat{z}_{k,j}^- \right) \quad (4.39) \]

\[ P_{k,j} = P_{k,j-1} - K_{k,j} P_{zz,k,j} K_{k,j}^T \quad (4.40) \]

**Step 5:**

The difference between the estimated states at iteration \( j \) and \( j-1 \) is given as

\[ \bar{c}_{k,j} = \hat{c}_{k,j} - \hat{c}_{k,j-1} \quad (4.41) \]

And the difference between the measurements and the estimated measurements, i.e. residuals, is defined as

\[ \bar{z}_{k,j} = z_k - \hat{z}_{k,j} \quad (4.42) \]

where

\[ \hat{z}_{k,j} = h(\hat{c}_{k,j}) \quad (4.43) \]

**Step 6:**

If the following inequality holds

\[ \bar{c}_{k,j}^T P_{k,j-1}^{-1} \bar{c}_{k,j} + \bar{z}_{k,j}^T R_k^{-1} \bar{z}_{k,j} < \bar{z}_{k,j-1}^T R_k^{-1} \bar{z}_{k,j-1} \quad (4.44) \]

for the \( j \leq N \), then set \( g = \eta \cdot g \) and \( j = j + 1 \). and go back to Step 3.

If the inequality is not satisfied, or if \( j \) is too large, i.e. \( j > N \), then let

\[ \hat{c}_k = \hat{c}_{k,j} \quad (4.45) \]

\[ P_k = P_{k,j} \quad (4.46) \]
Then set \( k = k + 1 \) and go back to Step 1.

4.4 Simulation Results for ASE

4.4.1 Tuning of IUKF

To apply ASE properly, the covariance of the noise are assumed to be constant that

\[
Q = \begin{bmatrix} Q_e & 0 \\ 0 & Q_p \end{bmatrix} = \begin{bmatrix} aI_{12 \times 12} & 0 \\ 0 & bI_{18 \times 18} \end{bmatrix}
\]

and

\[
R = cI_{24 \times 24}.
\]

These parameters \( \{a, b, c\} \) need to be tuned carefully to obtain the optimal results. Actually, the tuning of the Kalman filter is always a problem in practice.

To evaluate the performance of the results, error variance summation (EVS) of the parameters are introduced as

\[
EVS = \sum_{i=1}^{M} \sum_{k=1}^{N} \left( \hat{p}_{ik,i} - p_{actual,i,k,i} \right)^2
\]

where

- \( i \) the index of the parameter in the parameter vector.
- \( M \) the number of elements consist the parameter vector \( p \), which is 18 for the typical untransposed transmission line.
- \( k \) index of the iteration.
- \( N \) the number of sampling measurements, i.e. iterations, which is 20 for scenario 1 and scenario 2.

The EVS defines the overall differences from the estimated to the actual, sum up this difference with all the samplings and all the parameters. To determine the value of these three values, the easiest way is to test all the possible, as well as reasonable, covariance of these three and pick the one that would minimize the error variance summation. There are several things to be noted.
• This process is based on test case 2, where the parameters are changing. It is because for test case 1, the parameters does not change, then Kalman filter would treat the state dynamic covariance $Q$ to be very small, which is not applicable to test case 2.

• In practice, this method is not helpful since one does not know the actual value, therefore there is no criterion to determine which estimate is better. However, there are many other algorithms proposed in several literatures, with the topic of Kalman filter tuning.

• Such optimal set of $\{Q_x, Q_y, R\}$ ($\{a, b, c\}$) can be found as $a = 10^{-6}$, $b = 10^{-8}$ and $c = 10^{-9}$. Since the 4-dimensional graph is not easy to plot, the following three figures, Figure 4.1, Figure 4.2 and Figure 4.3 are provided with fixing each of the value and exploring the local minimum of the other two, with respect to the error variance summation.

![ASE tuning with fixed $Q_x$](image)

Figure 4.1 ASE tuning with fixed $Q_x$
For scenario 1-0, it can be proved numerically from the following results that  $a = 10^{-6}$,  $b = 10^{-8}$ and $c = 10^{-9}$ is also an optimal set for the covariance.

### 4.4.2 Test Case 1 Simulations

To demonstrate the performance of the proposed algorithm, all scenarios are evaluated, using IUKF. First, scenario 1 is simulated under the condition that with and without measurement noise. Besides the figures of some of the parameters, the evaluation quantities MAPE (mean absolute percentage error) and NRMSE (normalized root-mean-squared-error) is also given for test case 1.

- **Scenario 1-0**
The results of the above scenario show that ASE performs well on the constant case without measurement noise. However, when it goes to the noisy cases, things become complicated.

- **Scenario 1-1**
This is obviously not the optimal results. This is because we do not use the optimal tuned covariance. However, since for the practical situation, the value, or even the distribution of measurement noise is not achievable. In this case, there is no way to know the optimal covariance before the optimal results are derived.

Here, the optimal results are still given with different covariance from the ones previously talked about. Now they are: \( \{a, b, c\} = \{10^{-2}, 10^{-9}, 10^{-6}\} \).

- **Scenario 1-2**
For scenario 2, the error is larger than that in scenario 1. However, the results are much better. This is because the set of covariance chosen before is kind of best choice for the most number of situations, with or without noise, but not for all the situations.

From the above three set of results, the performance of ASE is much better than the static parameter estimation method discussed in the previous chapter. For the noise free scenario, ASE provides the
estimate very close to the actual value. For the second scenario, when there is measurement noise, ASE still estimates parameters properly with acceptable results.

4.4.3 Test Case 2 Simulations

Besides NRMSE and MAPE, CC (correlation coefficient) is provided also as the quantified results.

- **Scenario 2-0**

![ASE scenario 2-0](image)

**Table 4.3** Evaluation quantities of ASE scenario 2-0

<table>
<thead>
<tr>
<th>p #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0106</td>
<td>0.0028</td>
<td>0.256</td>
<td>0.0118</td>
<td>0.1774</td>
<td>0.0116</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0133</td>
<td>0.0044</td>
<td>0.2757</td>
<td>0.0146</td>
<td>0.2629</td>
<td>0.0153</td>
</tr>
<tr>
<td>CC</td>
<td>0.2356</td>
<td>0.076</td>
<td>0.4947</td>
<td>0.8298</td>
<td>0.44</td>
<td>0.7226</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p #</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0024</td>
<td>0.0049</td>
<td>0.0153</td>
<td>0.016</td>
<td>0.0036</td>
<td>0.2251</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.004</td>
<td>0.0077</td>
<td>0.0219</td>
<td>0.0226</td>
<td>0.0053</td>
<td>0.3643</td>
</tr>
<tr>
<td>CC</td>
<td>0.1323</td>
<td>0.3466</td>
<td>0.8387</td>
<td>0.1119</td>
<td>0.8349</td>
<td>0.694</td>
</tr>
</tbody>
</table>
Presenting the results of test case 2 without measurement noise in Figure 4.8 and Table 4.3, the results accuracy decreases a lot comparing with the results of scenario 1-1.

- Scenario 2-1

<table>
<thead>
<tr>
<th>p #</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0077</td>
<td>0.0028</td>
<td>0.0079</td>
<td>0.0142</td>
<td>0.0048</td>
<td>0.0028</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0103</td>
<td>0.0035</td>
<td>0.0112</td>
<td>0.0167</td>
<td>0.0069</td>
<td>0.0036</td>
</tr>
<tr>
<td>CC</td>
<td>0.3198</td>
<td>0.7623</td>
<td>0.4796</td>
<td>0.3103</td>
<td>0.346</td>
<td>0.8045</td>
</tr>
</tbody>
</table>

Table 4.4  Evaluation quantities of ASE scenario 2-1

<table>
<thead>
<tr>
<th>p #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.1554</td>
<td>0.0115</td>
<td>0.3334</td>
<td>0.0166</td>
<td>0.6487</td>
<td>0.0199</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.1792</td>
<td>0.014</td>
<td>0.4041</td>
<td>0.0206</td>
<td>0.773</td>
<td>0.0265</td>
</tr>
<tr>
<td>CC</td>
<td>0.7639</td>
<td>0.8027</td>
<td>0.8985</td>
<td>0.9114</td>
<td>0.4635</td>
<td>0.7543</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p #</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0023</td>
<td>0.0096</td>
<td>0.024</td>
<td>0.032</td>
<td>0.0055</td>
<td>0.2095</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0029</td>
<td>0.0126</td>
<td>0.0304</td>
<td>0.0422</td>
<td>0.0071</td>
<td>0.2432</td>
</tr>
</tbody>
</table>
It is not easy to draw the conclusion from the above results. Though the results’ NRMSE and MAPE is not too large for most of the parameters, it is not acceptable that one or two parameters are much worse than others (such as p3, NRMSE>0.1).

**Scenario 2-2**

![Figure 4.10 ASE scenario 2-2](image)

<table>
<thead>
<tr>
<th>p #</th>
<th>MAPE</th>
<th>NRMSE</th>
<th>CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>0.018</td>
<td>0.0061</td>
<td>0.933</td>
</tr>
<tr>
<td>15</td>
<td>0.0148</td>
<td>0.0139</td>
<td>0.8629</td>
</tr>
</tbody>
</table>

Table 4.5 Evaluation quantities of ASE scenario 2-2
<table>
<thead>
<tr>
<th>p #</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0032</td>
<td>0.0108</td>
<td>0.0046</td>
<td>0.0098</td>
<td>0.0034</td>
<td>0.0278</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0039</td>
<td>0.0141</td>
<td>0.0062</td>
<td>0.0119</td>
<td>0.0045</td>
<td>0.0323</td>
</tr>
<tr>
<td>CC</td>
<td>0.4774</td>
<td>0.8956</td>
<td>0.8395</td>
<td>0.1249</td>
<td>0.9314</td>
<td>0.9297</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p #</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0071</td>
<td>0.0029</td>
<td>0.0085</td>
<td>0.0095</td>
<td>0.0047</td>
<td>0.003</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0102</td>
<td>0.0041</td>
<td>0.011</td>
<td>0.0106</td>
<td>0.0062</td>
<td>0.004</td>
</tr>
<tr>
<td>CC</td>
<td>0.0549</td>
<td>0.0309</td>
<td>0.7227</td>
<td>0.6898</td>
<td>0.845</td>
<td>0.9639</td>
</tr>
</tbody>
</table>

For test case 2, ASE performs much better than static parameter estimation proposed in Chapter 3, but still needs further improvement. One possible way to improve is to adaptively tune the covariance. But this is beyond the topics of this thesis, so it is not further studied here.

### 4.5 Chapter Summary

In this chapter, augmented state estimation (ASE) is developed to estimate both the states and the parameters. Since the transfer function for ASE is highly nonlinear as discussed in section 4.1, dynamic model is necessary. In such case, iterated unscented Kalman filter (IUKF) have been developed as the dynamic estimator for parameter estimation. For IUKF, tuning is always a hard problem. By appropriately simulating under different cases, the optimal error covariance parameter $Q$ and $R$ is found.

Considering the results in Figure 4.4 and Figure 4.7, when the parameters are constant, ASE performs well and the error variance of the results is small enough to say that ASE is a good estimator. However, for scenario 2 where the parameters are changing, the performance of IUKF deteriorated, especially when noise is introduced to the measurements. Some of the parameters’ NRMSE and MAPE are so large that can be regarded as absolutely wrong answer.

What’s more, one major theoretical weakness of ASE, as discussed before, is the assumption that the state dynamic is available, i.e.
\[ x_k = x_{k-1} + \eta_k. \]

This equation is part of the equation (4.4) that is used for the IUKF. The noise \( \eta_k \) above is assumed to be white noise with zero means. However, this is not practically true since the change of system states are more complicated and hard to predict. Considering the performance of ASE and its theoretical weakness, it is not the desired estimator.
Chapter 5

Joint State Estimation and Parameter Tracking

5.1 Overview

Revisiting the measurement model which has been discussed in previous chapters that

$$z = h(x, p) + e$$  \hspace{1cm} (5.1)

Here, \( h(x, p) \) is nonlinear if the states are augmented with parameters, which has already been discussed in the previous chapter.

In this chapter, a new approach is proposed to treat this nonlinear problem as two sub-problems, namely, state estimation and parameter tracking. Each of the two parts can be processed by certain methods separately with different prediction and measurement equations and then coupled together. These two parts are processed iteratively until convergence. In [34] and [35], a similar work is proposed for the transposed transmission line case. Here in this thesis, the proposed algorithm takes the advantage that separates state estimation and parameter tracking into two parts. And it also applies to the untransposed line case where the number of unknowns is larger than the number of measurements.

More specific, the proposed approach is named joint state estimation and parameter tracking (JSEPT). The measurement function is decoupled into two functions: one for state estimation and the other for parameter tracking, and both functions are linear. In state estimation, \( x \) is roughly
estimated using an approximated parameter, namely \( p_1 \), since it is uncertain rather than unknown. In parameter tracking, parameter, denoted as \( p_2 \), is tracked with dynamic equation using the previously estimated \( x \). Only when these two set of parameters are close enough, the system goes to its equilibrium point. In the following sections, this approach is fully detailed.

### 5.2 State Estimation Based on Extended Kalman Filter

#### 5.2.1 Measurement Model for State Estimation

The standard measurement model for state estimation at time \( k \) with known admittance matrix can be expressed as

\[
z_{s,k} = H_{p,k} x_k + v_{s,k}
\]

(5.2)

where

\[
z_{s,k} \quad \text{measurement vector for state estimation at time } k.
\]

\[
I_k \quad \text{current measurements at time } k.
\]

\[
V_k \quad \text{voltage measurements at time } k.
\]

\[
v_{s,k} \quad \text{vector of measurement noise with } E[v_{s,k}] = 0 \quad \text{and} \quad \text{cov}(v_{s,k}) = R_v.
\]

\[
H_{p,k} \quad \text{coefficient matrix of state at time } k, \text{ defined as}
\]

\[
H_{p,k} = \begin{bmatrix} I_{12 \times 12} \\
\tilde{H}_{p,k} \end{bmatrix}
\]

where \( \tilde{H}_{p,k} \) is constructed by a roughly approximated parameter, noted as \( \hat{p}_k \). \( I_{12 \times 12} \) is a 12 by 12 identity matrix. And \( \tilde{H}_{p,k} \in \mathbb{R}^{12 \times 12} \).

From the discussion in Chapter 2, (2.28) gives the fundamental relation between the measurements and the parameters. \( \tilde{H}_{p,k} \) indicates the coefficient matrix between the current measurements \( I \) and
the states \( x \), and can be derived from (2.53). The time index \( k \) is neglected for simplicity in the following derivation and calculation, instead, the subscript \( k \) in the following variables denote the sending end of the transmission line. Therefore, from (2.28), (2.50), (2.51) and (2.52), the following results are provided.

\[
I_{ka} = I_{kar} + jI_{kai} \\
= (p_1 + jp_2)V_{ka} + (p_3 + jp_4)V_{kb} + (p_5 + jp_6)V_{kc} \\
+ (-p_1 + jp_2)V_{ma} + (-p_3 + jp_4)V_{mb} + (-p_5 + jp_6)V_{mc} \\
= (p_1 + jp_2)(V_{kar} + jV_{ka}) + (p_3 + jp_4)(V_{kbr} + jV_{kb}) + (p_5 + jp_6)(V_{kcr} + jV_{kc}) \\
+ (-p_1 + jp_2)(V_{mar} + jV_{ma}) + (-p_3 + jp_4)(V_{mbr} + jV_{mb}) + (-p_5 + jp_6)(V_{mcr} + jV_{mc}) \\
= \left[ p_1 - p_2, p_3 - p_4, p_5 - p_6, p_7 - p_3, p_8 - p_5, p_9 - p_3 \right], x \\
+ j\left\{ p_2, p_1, p_4, p_3, p_6, p_5, p_7, p_1, p_8, p_2, p_3, p_9 \right\}, x \right]
\]

\[
I_{kb} = I_{kbr} + jI_{kbi} \\
= (p_3 + jp_4)V_{ka} + (p_{10} + jp_{11})V_{kb} + (p_{12} + jp_{13})V_{kc} \\
+ (-p_3 + jp_4)V_{ma} + (-p_{10} + jp_{11})V_{mb} + (-p_{12} + jp_{13})V_{mc} \\
= (p_3 + jp_4)(V_{kar} + jV_{ka}) + (p_{10} + jp_{11})(V_{kbr} + jV_{kb}) + (p_{12} + jp_{13})(V_{kcr} + jV_{kc}) \\
+ (-p_3 + jp_4)(V_{mar} + jV_{ma}) + (-p_{10} + jp_{11})(V_{mbr} + jV_{mb}) + (-p_{12} + jp_{13})(V_{mcr} + jV_{mc}) \\
= \left[ p_3 - p_4, p_{10} - p_{11}, p_{12} - p_{13}, p_5 - p_6, p_7 - p_8, p_9 - p_{10}, p_{14} - p_{12}, p_{15} - p_{13} \right], x \\
+ j\left\{ p_4, p_3, p_{11}, p_{10}, p_{13}, p_{12}, p_8 - p_3, p_{14} - p_{10}, p_{15} - p_{13} \right\}, x \right]
\]

\[
I_{kc} = I_{kcr} + jI_{kci} \\
= (p_5 + jp_6)V_{ka} + (p_{12} + jp_{13})V_{kb} + (p_{16} + jp_{17})V_{kc} \\
+ (-p_5 + jp_6)V_{ma} + (-p_{12} + jp_{13})V_{mb} + (-p_{16} + jp_{17})V_{mc} \\
= (p_5 + jp_6)(V_{kar} + jV_{ka}) + (p_{12} + jp_{13})(V_{kbr} + jV_{kb}) + (p_{16} + jp_{17})(V_{kcr} + jV_{kc}) \\
+ (-p_5 + jp_6)(V_{mar} + jV_{ma}) + (-p_{12} + jp_{13})(V_{mbr} + jV_{mb}) + (-p_{16} + jp_{17})(V_{mcr} + jV_{mc}) \\
= \left[ p_5 - p_6, p_{12} - p_{13}, p_{16} - p_{17}, p_9 - p_5, p_7 - p_3, p_8 - p_{12}, p_9 - p_{16}, p_{18} - p_{16} \right], x \\
+ j\left\{ p_6, p_5, p_{13}, p_{12}, p_7, p_{17}, p_{16}, p_9 - p_3, p_{15} - p_{12}, p_{18} - p_{16} \right\}, x \right]
\]

\[
I_{ma} = I_{mar} + jI_{mai} \\
= (-p_1 + jp_2)V_{ka} + (-p_3 + jp_4)V_{kb} + (-p_5 + jp_6)V_{kc} \\
+ (p_1 + jp_2)V_{ma} + (p_3 + jp_4)V_{mb} + (p_5 + jp_6)V_{mc} \\
= (-p_1 + jp_2)(V_{kar} + jV_{ka}) + (-p_3 + jp_4)(V_{kbr} + jV_{kb}) + (-p_5 + jp_6)(V_{kcr} + jV_{kc}) \\
+ (p_1 + jp_2)(V_{mar} + jV_{ma}) + (p_3 + jp_4)(V_{mbr} + jV_{mb}) + (p_5 + jp_6)(V_{mcr} + jV_{mc}) \\
= \left[ -p_1, -p_2, -p_3, -p_5, -p_3, -p_2, -p_3, -p_4, -p_5, -p_6 \right], x \\
+ j\left\{ -p_2, -p_1, -p_3, -p_5, -p_3, -p_2, -p_3, -p_4, -p_5, -p_6 \right\}, x \right]
\]
\[ I_{na} = I_{nbi} + jI_{nbi} \]
\[ = (-p_5 + jp_9)V_{ka} + (-p_{10} + jp_{14})V_{kb} + (-p_{12} + jp_{15})V_{kc} + (p_3 + jp_7)V_{ma} + (p_{10} + jp_{11})V_{ma} + (p_{12} + jp_{13})V_{ma} \]
\[ = (-p_5 + jp_9)(V_{kar} + jV_{ka}) + (-p_{10} + jp_{14})(V_{kbr} + jV_{kb}) + (-p_{12} + jp_{15})(V_{kcr} + jV_{kc}) + \]
\[ + (p_3 + jp_7)(V_{mar} + jV_{ma}) + (p_{10} + jp_{11})(V_{mar} + jV_{ma}) + (p_{12} + jp_{13})(V_{mar} + jV_{ma}) \]
\[ = \left\{ \begin{array}{c}
-p_3 \quad -p_5 \\
-p_{10} \quad -p_{14} \\
-p_{12} \quad -p_{15} \quad p_3 \\
-p_4 \quad p_{10} \quad -p_{11} \quad p_{12} \quad -p_{13} \\
\end{array} \right\} \cdot x \]
\[ + j\left\{ \begin{array}{c}
p_8 \\
p_{14} \\
p_{10} \quad p_{15} \\
p_{12} \\
p_{16} \quad -p_{13} \quad p_{12} \quad p_{17} \\
\end{array} \right\} \]

\[ I_{nc} = I_{nbi} + jI_{nbi} \]
\[ = (-p_5 + jp_9)V_{ka} + (-p_{10} + jp_{14})V_{kb} + (-p_{16} + jp_{18})V_{kc} + (p_3 + jp_7)V_{ma} + (p_{12} + jp_{13})V_{ma} \]
\[ = (-p_5 + jp_9)(V_{kar} + jV_{ka}) + (-p_{12} + jp_{15})(V_{kbr} + jV_{kb}) + (-p_{16} + jp_{18})(V_{kcr} + jV_{kc}) + \]
\[ + (p_3 + jp_7)(V_{mar} + jV_{ma}) + (p_{12} + jp_{13})(V_{mar} + jV_{ma}) + (p_{16} + jp_{17})(V_{mar} + jV_{ma}) \]
\[ = \left\{ \begin{array}{c}
p_{-5} \quad -p_9 \\
p_{-12} \quad -p_{15} \\
p_{-16} \quad -p_{18} \quad p_5 \\
p_{-6} \quad p_{12} \quad -p_{13} \quad p_{16} \quad -p_{17} \\
\end{array} \right\} \cdot x \]
\[ + j\left\{ \begin{array}{c}
p_9 \\
p_{-5} \\
p_{-15} \\
p_{-12} \quad p_{16} \quad -p_{13} \quad p_{12} \quad p_{17} \\
p_{-17} \\
\end{array} \right\} \]

Therefore, \( \tilde{H}_p = \)
Note that in the scalars, vectors and matrix constructed above, all the elements are time variant with index $k$. (5.2) shows a linear function relating the measurements and the state variables. According to the basic knowledge of WLS state estimation, $\hat{x}_k$ can be determined by minimizing the objective function

$$ J(\hat{x}_k) = \frac{1}{2} \left( z_{s,k} - H_{p,k} \hat{x}_k \right)^T R_k^{-1} \left( z_{s,k} - H_{p,k} \hat{x}_k \right) $$

and the solution of minimizing (5.3) is

$$ \hat{x}_k = \left[ H_{p,k}^T R_k^{-1} H_{p,k} \right]^{-1} H_{p,k}^T R_k^{-1} z_{s,k}. $$

### 5.3 Parameter Tracking Implementation

As discussed above, states from untransposed transmission line are estimated. Here, we assume the states as constants and do parameter tracking (PT). The prediction model is,

$$ p_k = p_{k-1} + w_k. $$

The measurement model is

$$ z_{p,k} = H_{x,k} p_k + v_{p,k} $$

where,

- $z_{p,k}$ vector of measurements consisted of currents, $z_{p,k} = I_k$, $z_{p,k} \in \mathbb{R}^{12\times 1}$.
- $w_k$ vector of zero-mean white process noise, covariance is $Q_p$.
- $v_{p,k}$ measurement noise, vector of zero-mean white process noise, covariance is $R_p$. $v_{p,k} \in \mathbb{R}^{12\times 1}$ and $R_p \in \mathbb{R}^{12\times 12}$.
- $p_k$ parameter vector, $p_k \in \mathbb{R}^{18\times 1}$.
- $H_{x,k}$ coefficient matrix (Jacobian Matrix) derived by treating all states as constants, consisted of state $x$ with time index $k$. $H_{x,k} \in \mathbb{R}^{12\times 18}$.

By re-constructing (2.28), it can be derived that
\[ H_i^T = \]

\[
\begin{array}{cccccc}
0 & 0 & x_{kar}^* & x_{kai}^* & 0 & 0 \\
x_{mar}^* & x_{mai} & 0 & 0 & 0 & 0 \\
-x_{kai} & x_{kar} & 0 & 0 & 0 & 0 \\
x_{kbr}^* & x_{kbi}^* & x_{kar}^* & x_{kai}^* & 0 & 0 \\
x_{mbr} & x_{mbi} & x_{mar} & x_{mai} & 0 & 0 \\
x_{kbi} & x_{kbr} & -x_{kai} & x_{kar} & 0 & 0 \\
x_{kcr}^* & x_{kci}^* & 0 & 0 & x_{kar}^* & x_{kai}^* \\
x_{mcr} & x_{mci} & x_{mbr} & x_{mbi} & 0 & 0 \\
x_{kci} & x_{kcr} & 0 & 0 & -x_{kai} & x_{kar} & 0 & 0 \\
x_{mai} & x_{mar} & 0 & 0 & 0 & 0 \\
x_{mbi} & x_{mbr} & -x_{mai} & x_{mar} & 0 & 0 \\
x_{mbi} & x_{mcr} & 0 & 0 & -x_{mai} & x_{mar} & 0 & 0 \\
-x_{mci} & x_{mcr} & 0 & 0 & 0 & 0 \\
-x_{kci} & x_{kcr} & 0 & 0 & -x_{kai} & x_{kar} & 0 & 0 \\
0 & 0 & x_{kbr}^* & x_{kbi}^* & 0 & 0 \\
x_{mbr} & x_{mbi} & 0 & 0 & 0 & 0 \\
0 & 0 & -x_{kbi} & x_{kbr} & 0 & 0 \\
x_{kcr}^* & x_{kci}^* & x_{kbr}^* & x_{kbi}^* & 0 & 0 \\
x_{mcr} & x_{mci} & x_{mbr} & x_{mbi} & 0 & 0 \\
0 & 0 & -x_{kci} & x_{kcr} & -x_{kbi} & x_{kbr} & 0 & 0 \\
0 & 0 & -x_{mbi} & x_{mbr} & 0 & 0 \\
0 & 0 & -x_{mci} & x_{mcr} & -x_{mbi} & x_{mbr} & 0 & 0 \\
0 & 0 & -x_{mci} & x_{mcr} & -x_{kbi} & x_{kbr} & 0 & 0 \\
0 & 0 & x_{kcr}^* & x_{kci}^* & 0 & 0 \\
x_{mcr} & x_{mci} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
-x_{kci} & x_{kcr} & 0 & 0 & 0 & 0 \\
-x_{mci} & x_{mcr} & 0 & 0 & 0 & 0 \\
\end{array}
\]
where ‘x’ stands for the state. The subscript ‘m/k’ indicate the transmission line terminal. ‘a/b/c’ indicate the bus phase. ‘r/i’ indicate the real or imaginary part of the voltage. Given the models in (5.5) and (5.6), the implementation of the Kalman filter is in the following.

Time update:

\[
\hat{p}_{k|k-1} = \hat{p}_{k-1|k-1}
\]

\[
P_{k|k-1} = P_{k-1|k-1} + Q_p
\]  

(5.7)  
(5.8)

Measurement update:

\[
\Delta z_{p,k} = z_{p,k} - H_{x,x} \hat{p}_{k|k-1}
\]

\[
\Phi_k = H_{p,k} P_{k|k-1} H_{p,k}^T + R_p
\]

(5.9)  
(5.10)

\[
K_k = P_{k|k-1} H_{p,k}^T \Phi_k^{-1}
\]

(5.11)

\[
\hat{p}_{k|k} = \hat{p}_{k|k-1} + K_k \Delta z_{p,k}
\]

(5.12)

\[
P_{k|k} = P_{k|k-1} - K_k \Phi_k K_k^T.
\]

(5.13)

### 5.4 Joint State Estimation and Parameter Tracking Implementation

#### 5.4.1 Algorithm Procedure

The two major parts of joint state estimation and parameter tracking (JSEPT) are presented before. From the viewpoint of SE, there is uncertainty that caused by the estimated voltages. A similar uncertainty holds when discussed from the viewpoint of PT. These two uncertainties will make the results inaccurate and unconvinceable.
Actually, in the process of state estimation, the Jacobian matrix \( H_{p,k} \) is consisted of estimated parameters, which are \( \hat{p}_{k-1} \) but not \( \hat{p}_k \) since the latter one is not yet available. While in the parameter tracking, \( H_{x,k} \) is consisted of \( \hat{x}_k \) which is obtained from the prior state estimate.

The overall procedure of the JSEPT can be expressed as:

- **Step 1:**
  Initialization: \( \hat{p}_{0|0}, P_{0|0} \) and \( k = 1 \).

- **Step 2:**
  For each instant \( k (k \geq 1) \), Let \( \hat{p}_{k,j-1,k-1, j} = \hat{p}_{k,j-1,k-1} \), \( P_{k,j-1,k-1, j} = P_{k,j-1,k-1} \) (the value at time \( k-1 \) is available and aim to estimate the value at time \( k \)) and \( j = 1 \). The subscript \( j \) indicates the \( j \)-th iteration.

- **Step 3:**
  Process the ‘Time update’ part of parameter tracking using \( \hat{p}_{k,j-1,k, j} \) and \( P_{k,j-1,k, j} \).

- **Step 4:**
  Formulate \( H_{p,k} \) using the predicted parameters \( \hat{p}_{k,j-1,k, j} \), then process state estimation and derive \( \hat{x}_{k,j} \).

- **Step 5:**
  Formulate \( H_{x,k} \) using \( \hat{x}_{k,j} \), then process the ‘Measurement update’ part of parameter tracking. Record the estimate \( \hat{p}_{k,j} \).

- **Step 6:**
  If \( j > 1 \) and \( \| \hat{p}_{k,j} - \hat{p}_{k,j-1} \| < \varepsilon \), go to Step 7;
  If \( j \) is too large \( j > N \), go to Step 7;
  Otherwise, repeat Step 3 to Step 6 with new \( j = j + 1 \) and \( \hat{p}_{k,j} = \hat{p}_{k,j} \), \( P_{k,j} = P_{k,j} \).

- **Step 7:**
The iteration of \( j \) is terminated. The estimated value at this instant \( k \) is \( \hat{p}_{jk}^k = \hat{p}_{jk,j}^k \), \( P_{jk}^k = P_{jk,j}^k \) and \( \hat{x}_k = \hat{x}_{k,j}^k \).

If \( k \leq M \), let \( k = k + 1 \), go back to Step 2;

Else, go to Step 8.

- **Step 8:**

  Terminate.

Note the following quantities in the above procedure:

- \( \varepsilon \)  
  predetermined threshold to decide the termination of internal iteration;

- \( N \) 
  the iteration limit of \( j \) in each instant \( k \);

- \( M \) 
  the number of sampling measurements.

The overall procedure of the JSEPT is shown in the following figure.
Initialization

If \( k \leq M \)

\( j = 1 \)

If \( j \leq N \)

Assign \( j \)-th iteration value

Predict of parameter tracking

State Estimation

Update of parameter tracking

If \( j = 1 \)

\( j = j + 1 \)

If norm < \( \varepsilon \)

Y

N

k-th instant finished

END

Figure 5.1 Illustration of overall procedure

More general, the method can be illustrated as:

- Do state estimation
- Do parameter tracking using the states estimated in A
- Do state etimation again using the parameters estimated in B
- Do parameter tracking using the states estimated in C
- Compare the results of A, B and C, D,
- Get the solution in time k when the results in E are convergent

And the correspongding flowchart is given as follows.
Furthermore, the dynamic behaviors of the voltages and the parameters are different. The former may jump abruptly while the latter may increase (or decrease) slowly but persistently. Correspondingly, appropriate detection and adaptation techniques have been developed.

The overview for the joint state estimation and parameter tracking is shown in Figure 5.1. Process Flow Diagram of Joint State Estimation and Parameter Tracking.
5.4.2 Implementation Accuracy

In parameter tracking, the Jacobian matrix is consisted of estimated states which calculated from state estimation. And the estimated parameters are also very critical to state estimation. This makes the estimation inaccurate.

The method discussed in the above section is to check the convergency of the estimated value to narrow down this inaccuracy. By introducing the iteration, only when a new equilibrium point is constructed, the iterative process can be ended and move to the next step. Thus, there is no need to consider much about the errors between the prediction value and measurement, since the iteration can always bring the result to a convergent one.

Another method to deal with it is to take the prediction-measurement error into account.

\[ z_{x,k} = H_{p,k} x_k + v_{x,k} = H_{p,k}^p x_k + v_{x,k} \]  \hspace{1cm} (5.14)

where,

\[ v'_{x,k} = (H_{p,k} - H_{p,k}^p) x_k + v_{x,k} . \]  \hspace{1cm} (5.15)

\( H_{p,k} \) is the coefficient matrix obtained from \( H_x \) by replacing \( p \) with an estimated \( \hat{p} \).

\[ \Delta H_{p,k} = H_{p,k} - H_{p,k}^p \]  \hspace{1cm} (5.16)

\[ \Delta x_{k|k-1} = x_k - \hat{x}_{k|k-1} \]  \hspace{1cm} (5.17)

\[ R_{x,k} = \Delta H_{p,k} \left( P_{k|k-1} + Q_p \right) \Delta H_{p,k}^T + R_p \]  \hspace{1cm} (5.18)

\[ \bar{v}_{x,k} = E(v'_{x,k}) = \Delta H_{p,k} \hat{x}_{k|k-1} . \]  \hspace{1cm} (5.19)

In Kalman filter, it holds

\[ \hat{z}_{x,k|k-1} = H_{p,k} \hat{x}_{k|k-1} + \bar{v}_{x,k} \]  \hspace{1cm} (5.20)
\[ \Phi_k = H_{x,k} P_{k|k-1} H_{x,k}^T + R_{x,k} \]  

(5.21)

\[ K_k = \left[ P_{k|k-1} H_{x,k}^T \right] \Phi_k^{-1} \].  

(5.22)

From above equations(5.14) to (5.22), the parameters that consist \( H_{p,k} \) are substituted by estimated value, with a transformation based on the errors between prediction and measurement.

5.4.3 Implementation Complexity

The filter is based on the Kalman filter, dealing with the eighteen parameters and twelve states. In the reactangular form, the number of parameters and states doubled because of the separation of real and imaginary component of the complex data.

The total complexity of this algorithm is hard to decide because there are lots of uncertain iterations to be processed according to the input data. In practice, the average time is less than 0.1 second to process 20 time scan data. It is available to do real-time tracking and implement in practice with the PMU.

5.5 Simulation Results for JSEPT

For the simulation of JSEPT, test case 1 and test case 2 are introduced as before. Since it is based on Kalman filter, tuning is also important. There are two covariance to be decided: \( Q_p \) and \( R_p \).

5.5.1 Tuning of KF

Structure the covariance as

\[ Q_p = a I_{18\times 18} \]

\[ R_p = b I_{12\times 12} \],

and recall the definition of error variance summation (EVS) that

\[ \text{EVS} = \sum_{i=1}^{18} \sum_{k=1}^{20} \left( \hat{p}_{k|k,i} - p_{\text{actual},k,i} \right)^2, \]

the following results hold.
For scenario 1, with the changing of $R$ and $Q$, the error variance summation (EVS) is always small since the value of the z-axis is in the negative half axis. The largest value of error variance summation is about 0.001, which is small enough to say that JSEPT performs well whatever the covariance is. In such case, the tuning for scenario 2 is also provided.

For scenario 2, JSEPT also works in a wide range of $R$ and $Q$, i.e., JSEPT can operate normally and obtain relatively accurate result. However, the smallest EVS is much larger than that of scenario 1, which indicates that the estimation on the changing parameters cannot be as precise as the constant parameters.

By comparing the intersection of these two figures, the set $\{a,b\} = \{10^{-7},10^{-7}\}$ is chosen.
5.5.2 Test Case 1 Simulations

- Scenario 1-0

Figure 5.5 JSEPT scenario 1-0

<table>
<thead>
<tr>
<th>p #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
<tr>
<td>p #</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>MAPE</td>
<td>0.0002</td>
<td>0.0001</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0002</td>
<td>0.0001</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
<tr>
<td>p #</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td>MAPE</td>
<td>0.0001</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0001</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0.0002</td>
</tr>
</tbody>
</table>

Table 5.1 Evaluation quantities of JSEPT scenario 1-0
It is obvious that JSEPT performs much better comparing with the results of ASE, as well as SPE. For the constant parameter case with small measurement noise, JSEPT still provide acceptable results with relatively small NRMSE.
• Scenario 1-2

![Figure 5.7 JSEPT scenario 1-2](image)

Table 5.3 Evaluation quantities of JSEPT scenario 1-2

<table>
<thead>
<tr>
<th>p #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p #</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0</td>
<td>0</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0</td>
<td>0</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p #</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0001</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

The above three scenarios for test cases 1 indicates the following properties of JSEPT.

• JSEPT is better than ASE and SPE from the point view of estimation accuracy, i.e. the error of JSEPT is smaller than other two.
The chosen of covariance is appropriate and helpful.

5.5.3 Test Case 2 Simulations

For test case 2, since its parameters are varying with time, it is much harder to track the parameters perfectly. Static parameter and augmented state estimation didn’t perform well for this situation because of the singularity of the Jacobian matrix. The following results provide the results of JSEPT.

Note that, though the tuning of the covariance is still a problem, but from the results of the previous scenario, the covariance is still $R_p = 10^{-7}$ and $Q_p = 10^{-7}$.

Scenario 2-0

![JSEPT scenario 2-0](image)

Figure 5.8 JSEPT scenario 2-0

<table>
<thead>
<tr>
<th>p #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0001</td>
<td>0</td>
<td>0.0002</td>
<td>0</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>CC</td>
<td>1</td>
<td>1</td>
<td>0.9998</td>
<td>1</td>
<td>0.9999</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p #</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0002</td>
<td>0.0002</td>
<td>0</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

Table 5.4 Evaluation quantities of JSEPT scenario 2-0
<table>
<thead>
<tr>
<th>NRMSE</th>
<th>0.0002</th>
<th>0.0002</th>
<th>0</th>
<th>0.0001</th>
<th>0</th>
<th>0.0001</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>1</td>
<td>0.9993</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>p #</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td>MAPE</td>
<td>0</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0</td>
<td>0.0001</td>
<td>0</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
</tr>
<tr>
<td>CC</td>
<td>1</td>
<td>0.9998</td>
<td>1</td>
<td>1</td>
<td>0.9999</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 5.8 shows that the reference parameter follows the actual value well. Table 5.4 indicates that all other parameters relatively have small MAPE and NRMSE, which stands for the small differences between the calculated values and the actual ones. What’s more, all of the correlation covariance are around 1, which indicates that calculated values have a strong ability to track the variation. It is safe to say the performance of JSEPT for scenario 2 without measurement noise is good.

- **Scenario 2-1**

![Figure 5.9 JSEPT scenario 2-1](image)

**Figure 5.9 JSEPT scenario 2-1**

<table>
<thead>
<tr>
<th>p #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
</table>

**Table 5.5 Evaluation quantities of JSEPT scenario 2-1**
<table>
<thead>
<tr>
<th></th>
<th>MAPE</th>
<th>NRMSE</th>
<th>CC</th>
<th>MAPE</th>
<th>NRMSE</th>
<th>CC</th>
<th>MAPE</th>
<th>NRMSE</th>
<th>CC</th>
<th>MAPE</th>
<th>NRMSE</th>
<th>CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>p #</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0126</td>
<td>0.0017</td>
<td>0.0957</td>
<td>0.0059</td>
<td>0.158</td>
<td>0.0125</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0191</td>
<td>0.0021</td>
<td>0.113</td>
<td>0.0077</td>
<td>0.1946</td>
<td>0.0157</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.2611</td>
<td>0.993</td>
<td>0.7227</td>
<td>0.9727</td>
<td>0.3468</td>
<td>0.9917</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0025</td>
<td>0.0033</td>
<td>0.003</td>
<td>0.0166</td>
<td>0.0022</td>
<td>0.1271</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0031</td>
<td>0.0037</td>
<td>0.004</td>
<td>0.0197</td>
<td>0.0027</td>
<td>0.1688</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.9846</td>
<td>0.9789</td>
<td>0.9592</td>
<td>0.7112</td>
<td>0.976</td>
<td>0.4807</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0071</td>
<td>0.0014</td>
<td>0.0016</td>
<td>0.012</td>
<td>0.0022</td>
<td>0.0021</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0091</td>
<td>0.0017</td>
<td>0.0019</td>
<td>0.0146</td>
<td>0.0027</td>
<td>0.0027</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.9889</td>
<td>0.9897</td>
<td>0.9919</td>
<td>0.3042</td>
<td>0.987</td>
<td>0.9895</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

With measurement noise, JSEPT’s performance is still promising.

- **Scenario 2-2**

![JSEPT scenario 2-2](image)

Figure 5.10  JSEPT scenario 2-2
Table 5.6 Evaluation quantities of JSEPT scenario 2-2

<table>
<thead>
<tr>
<th>p #</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0048</td>
<td>0.0014</td>
<td>0.0096</td>
<td>0.0026</td>
<td>0.0132</td>
<td>0.0055</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0053</td>
<td>0.0017</td>
<td>0.0108</td>
<td>0.0031</td>
<td>0.0157</td>
<td>0.0065</td>
</tr>
<tr>
<td>CC</td>
<td>0.9825</td>
<td>0.9664</td>
<td>0.9664</td>
<td>0.96</td>
<td>0.9746</td>
<td>0.9804</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p #</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.0014</td>
<td>0.0032</td>
<td>0.0043</td>
<td>0.0021</td>
<td>0.0011</td>
<td>0.0137</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0016</td>
<td>0.0037</td>
<td>0.0052</td>
<td>0.0024</td>
<td>0.0013</td>
<td>0.0158</td>
</tr>
<tr>
<td>CC</td>
<td>0.9693</td>
<td>0.9807</td>
<td>0.9689</td>
<td>0.9817</td>
<td>0.9708</td>
<td>0.9806</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>p #</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPE</td>
<td>0.003</td>
<td>0.0012</td>
<td>0.0027</td>
<td>0.0011</td>
<td>0.0015</td>
<td>0.0013</td>
</tr>
<tr>
<td>NRMSE</td>
<td>0.0036</td>
<td>0.0015</td>
<td>0.0032</td>
<td>0.0015</td>
<td>0.0017</td>
<td>0.0016</td>
</tr>
<tr>
<td>CC</td>
<td>0.9805</td>
<td>0.9804</td>
<td>0.9789</td>
<td>0.9676</td>
<td>0.9906</td>
<td>0.9849</td>
</tr>
</tbody>
</table>

The results of test case 2 indicates that

- JSEPT tracks the parameters well, whereas ASE and SPE don’t.
- When there is noise in the measurements, JSEPT still performs well with the pre-determined covariance.

### 5.6 Chapter Summary

In this chapter, the joint state estimation and parameter tracking (JSEPT) method is proposed. It contains two decoupled parts: SE and PT. For state estimation, the roughly estimated parameters are used to construct the measurement Jacobian of the state function. This can be explained as: inaccurate information rather than unknown. Then, after state estimation, the estimated states can
be used to construct the measurement Jacobian of the parameter function. The difference between parameters estimated just now and the ones used to construct the state Jacobian can be regarded as the threshold. After running SE and PT iteratively, the threshold decreases to some small enough value, which indicates this iteration converges.

Numerical results are given in the latter sections. For the constant parameter cases, JSEPT works very well even if there is measurement noise. For the changing parameter cases, JSEPT also provides acceptable results, which is much better than the performance of SPE and ASE. As a summary, all the results indicate that JSEPT performs well in estimating parameters for untransposed transmission line.
Chapter 6

Comparison of Three Methods

In the previous chapters, three methods are proposed to solve the parameter estimation problem: static parameter estimation (SPE), augmented state estimation (ASE) and joint state estimation and parameter tracking (JSEPT). By applying these methods to several scenarios under different noise, the performance of JSEPT is the best of three. In this chapter, two more evaluation criterions are pointed out to compare the performance of three methods.

6.1 Estimation Sensitivity to Initial Value

The initial condition, especially the initial value of the states, of the Kalman filter is important to the convergence of the results, as well as the tracking speed. In the previous chapters, all the simulations are based on the perfect initial, i.e. the actual initial values. In this section, two more kinds of initial values, beside the perfect initial, are provided to check whether and how will the initials influence the results.

- Zero initials.
- Biased initials: initials that have a 10% bias (less) of the perfect initials.

For simplicity, all the test cases in this chapter are the scenarios without measurement noise, i.e. scenario 1-0 and scenario 2-0.
For SPE, it is obvious that there is no requirement of initial value since it uses the static WLS to calculate the parameters, rather than estimating according to the dynamics.

Before the discussion of ASE and JSEPT, another evaluation quantity, named error variance along with time (EVT) is defined as follows.

$$EVT(k) = \sum_{i=1}^{M} \left( \hat{p}_{k,i} - p_{actual,k,i} \right)^2$$

(6.1)

where

- $i$ the index of the parameter in the parameter vector.
- $M$ the number of elements consist the parameter vector $p$, which is 18 for the typical untransposed transmission line.
- $k$ index of the iteration, which illustrates that EVT is time varying.

### 6.1.1 ASE Sensitivity to Initial Value

For ASE, the following results are provided with different initial values.

![ASE scenario 1-0 with perfect initial](image)

**Figure 6.1** ASE scenario 1-0 with perfect initial

For this situation, the results are exactly the same as in Figure 4.4.
Figure 6.2  ASE scenario 1-0 with biased initial

The EVT for this situation is shown as follows.

Figure 6.3  ASE scenario 1-0 with biased initial EVT

The results in Figure 6.2 and Figure 6.3 indicate that ASE does not perform well when there is initial bias.

Even if ASE is tuned properly, which is $\{Q_x, Q_y, R\} = \{10^{-6} I_{12=12}, 10^{-7} I_{18=18}, 10^{-7} I_{24=24}\}$, shown in Figure 6.4 and Figure 6.5, the results are better the previous but still not acceptable. Figure 6.6 shows the result that with zero initials, and it is absolutely wrong.
Figure 6.4  ASE scenario 1-0 with biased initial with optimal tuning

Figure 6.5  ASE scenario 1-0 with biased initial EVT with optimal tuning

Figure 6.6  ASE scenario 1-0 with zero initial
For the test case 2, when there is biased initial, the performance is even worse.

And for the situation with zero initial, the results are thoroughly corrupted. From the above discussions, it is not hard to make the summation that ASE depends much on the correct initials.
6.1.2 JSEPT Sensitivity to Initial Value

For the situation that there is biased initial, if the covariance is \( \{Q_p, R_p\} = \{10^{-7} I_{18 \times 18}, 10^{-7} I_{12 \times 12}\} \), the result is as follows.

In Figure 6.9, the parameter \( p_1 \) and \( p_2 \) ‘attempt’ to track the actual values along with time. Figure 6.10 shows that this attempt is true for all of the parameters.

However, for the zero initial situations, JSEPT seems not work very well, as shown in Figure 6.12 and Figure 6.13.
Figure 6.11  JSEPT scenario 1-0 with biased initial EVT

Figure 6.12  JSEPT scenario 1-0 with zero initial

Figure 6.13  JSEPT scenario 1-0 with zero initial EVT
Figure 6.14  JSEPT scenario 2-0 with perfect initial

Figure 6.15  JSEPT scenario 2-0 with biased initial

Figure 6.16  JSEPT scenario 1-0 with biased initial EVT
The above three figures, Figure 6.14, Figure 6.15 and Figure 6.16 give the results under the test case 2. Figure 6.14 indicates the perfect initial case where JSEPT performs very well. The next two figures show that when the initial values are not precise, the estimation result also has bias. But it has the trend to be close to the actual trajectory along with time, and the corresponding error variance also decreases with time.

Now, to summarize the estimation sensitivity to initial values, the following several conclusions hold.

- ASE is not applicable for the incorrect initial case.
- JSEPT performs well for the biased initial, but for the zero initial, it still needs improvement.
- JSEPT is better than ASE from the point view of estimation sensitivity to initial values.

### 6.2 Run Time Analysis of the Algorithms

The analysis of an algorithm is the determination of the amount of resources it needs, such as time, that necessary to execute them. There are three algorithms, SPE, ASE and JSEPT, to be evaluated. Table 6.1 shows the time consumption of the estimation process. All six scenarios proposed in section 2.3.4 are tested using the three algorithms proposed before.

- SPE: foot step $T = 4$;
- ASE: $\{Q, Q_p, R\} = \{10^{-6}I_{12 \times 12}, 10^{-8}I_{18 \times 18}, 10^{-9}I_{24 \times 24}\}$;
- JSEPT: $\{R_p, Q_p\} = \{10^{-7}I_{12 \times 12}, 10^{-7}I_{18 \times 18}\}$ and $\varepsilon = 10^{-7}$ (threshold, see section 5.4.1).

<table>
<thead>
<tr>
<th>Time consumption (s)</th>
<th>SPE</th>
<th>ASE</th>
<th>JSEPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario 1-0</td>
<td>0.253</td>
<td>16.917</td>
<td>0.462</td>
</tr>
<tr>
<td>Scenario 1-1</td>
<td>0.255</td>
<td>15.645</td>
<td>15.04</td>
</tr>
<tr>
<td>Scenario 1-2</td>
<td>0.274</td>
<td>16.000</td>
<td>2.076</td>
</tr>
<tr>
<td>Scenario 2-0</td>
<td>0.279</td>
<td>13.485</td>
<td>14.353</td>
</tr>
<tr>
<td>-------------</td>
<td>-------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>Scenario 2-1</td>
<td>0.272</td>
<td>14.641</td>
<td>12.816</td>
</tr>
<tr>
<td>Scenario 2-2</td>
<td>0.273</td>
<td>12.480</td>
<td>16.351</td>
</tr>
</tbody>
</table>

In Table 6.1, the time consumption of SPE is the smallest, and ASE is the greatest. But for JSEPT, it largely depends on the measurements, noise and parameter variations. So it is hard to decide

Analytically speaking, the time consumed in SPE, ASE and JSEPT is as follows.

- For SPE, it only executes the WLS estimation where contains the inverse of the Jacobian matrix. What’s more, it runs less number of iterations than other twos, since the foot step reduce the timeliness of SPE.

- For ASE, it not only do cholesky factorization of the covariance matrix, which is time consuming, but also iterative executes this process in order to meet the termination condition proposed in section 4.3 .

- JSEPT has more free parameters that would influence the time consumption. the threshold $\varepsilon$ to decide the termination of internal iteration, and the covariance of the parameter tracking part that will influence the convergence of the results. In such case, it is not easy to determine an average running time for JSEPT algorithm.
Chapter 7

Implementation of JSEPT on Network Systems

7.1 Overview of Network Implementation

In the previous chapters, three kinds of methods to do parameter estimation are developed and compared. All the test cases before are based on one three phase untransposed transmission line with one sending end and one receiving end. However, since the state estimation part of JSEPT is always processed in the network, it is necessary to expand the JSEPT algorithm to the network level.

It is obvious that the equations and calculations will become more complicated when dealing with a power system network. In this chapter, the algorithm is rebuilt according to a new network model, and the simulation results will be presented.

7.2 Network Model

7.2.1 Network Model Construction

Considering the construction of $Y_{bus}$ in section 2.2 , and the simulation model in section 2.3 , $Y_{bus}$ can be written in the following form:

$$
Y_{1,\text{bus}} = \begin{bmatrix}
Y_{1,1} & Y_{1,2} \\
Y_{1,2} & Y_{1,1}
\end{bmatrix}
$$

(7.1)
In the subscript, the first number before the comma indicates this is for line \( n \) (it is line 1 in (7.1)), and the second number denotes the different block matrices in \( Y_{bus} \). Revisiting (2.23), it has

\[
Y_1 = \frac{Y'}{2} + \frac{1}{Z'}, \quad Y_2 = -\frac{1}{Z'}
\]

For the measurement equation of one line (e.g. line 1), it is

\[
\begin{bmatrix}
I_{1,SEND} \\
I_{1,RECV}
\end{bmatrix}
= \begin{bmatrix}
Y_{1,1} & Y_{1,2} \\
Y_{1,2} & Y_{1,1}
\end{bmatrix}
\begin{bmatrix}
V_{1,SEND} \\
V_{1,RECV}
\end{bmatrix}
\tag{7.2}
\]

Given a network with \( n \) transmission lines, the measurement equation can be constructed by following steps.

- Augment all the currents together as a vector, as well as all the voltages, it holds

\[
I = \begin{bmatrix}
I_{1,SEND} \\
I_{1,RECV} \\
\vdots \\
I_{n,SEND} \\
I_{n,RECV}
\end{bmatrix} \in \mathbb{C}^{(6n \times 1)} \tag{7.3}
\]

\[
V = \begin{bmatrix}
V_{1,SEND} \\
V_{1,RECV} \\
\vdots \\
V_{n,SEND} \\
V_{n,RECV}
\end{bmatrix} \in \mathbb{C}^{(6n \times 1)} \tag{7.4}
\]

- Construct a square matrix with the diagonal block to be \( Y_{bus} \).

\[
Y = \begin{bmatrix}
Y_{1,1} & Y_{1,2} & 0 \\
Y_{1,2} & Y_{1,1} & \ddots \\
0 & \ddots & \ddots
\end{bmatrix}
\in \mathbb{C}^{(6n \times 6n)} \tag{7.5}
\]
and the relationship between $I$, $V$ and $Y$ is

$$I = YV$$  \hspace{1cm} (7.6)

- At a single bus, all the voltages measured for its connecting lines are the same, and this will not happen for the currents. Therefore, for two voltages that are the same, e.g. $V_i$ and $V_j$, delete the $j$-th row of the vector of voltages.

- Delete the $j$-th column of $Y$ in (7.5). And add this column to $i$-th column.

### 7.2.2 Simulation Model

The following network model contains four buses with four transmission lines.

![System model for parameter estimation](image)

Figure 7.1  System model for parameter estimation

The above model is untransposed transmission line with several sending points and receiving points. To distinguish the model in section 2.3.3, this model above is denoted as model 2.

In model 2, there are three untransposed transmission line. And the parameters to be estimated are from

- Line 1: SEND1 to RECV1;
- Line 2: SEND2 to RECV2;
Lien 3: SEND3 to RECV3;

Line 4: SEND4 to RECV4.

According to previous discussion, there should be six sets of current measurements and six sets of voltage measurements. However, note that the voltages at four buses are the same, which will change the structure of the admittance matrix.

For model 2, the following equations hold:

\[
\begin{bmatrix}
I_{1,\text{SEND}} \\
I_{1,\text{RECV}}
\end{bmatrix} =
\begin{bmatrix}
Y_{1,1} & Y_{1,2} \\
Y_{1,2} & Y_{1,1}
\end{bmatrix}
\begin{bmatrix}
V_{1,\text{SEND}} \\
V_{1,\text{RECV}}
\end{bmatrix}
\]  \hspace{1cm} (7.7)

\[
\begin{bmatrix}
I_{2,\text{SEND}} \\
I_{2,\text{RECV}}
\end{bmatrix} =
\begin{bmatrix}
Y_{2,1} & Y_{2,2} \\
Y_{2,2} & Y_{2,1}
\end{bmatrix}
\begin{bmatrix}
V_{2,\text{SEND}} \\
V_{2,\text{RECV}}
\end{bmatrix}
\]  \hspace{1cm} (7.8)

\[
\begin{bmatrix}
I_{3,\text{SEND}} \\
I_{3,\text{RECV}}
\end{bmatrix} =
\begin{bmatrix}
Y_{3,1} & Y_{3,2} \\
Y_{3,2} & Y_{3,1}
\end{bmatrix}
\begin{bmatrix}
V_{3,\text{SEND}} \\
V_{3,\text{RECV}}
\end{bmatrix}
\]  \hspace{1cm} (7.9)

\[
\begin{bmatrix}
I_{4,\text{SEND}} \\
I_{4,\text{RECV}}
\end{bmatrix} =
\begin{bmatrix}
Y_{4,1} & Y_{4,2} \\
Y_{4,2} & Y_{4,1}
\end{bmatrix}
\begin{bmatrix}
V_{4,\text{SEND}} \\
V_{4,\text{RECV}}
\end{bmatrix}
\]  \hspace{1cm} (7.10)

Since there is equality that:

\[
V_{1,\text{SEND}} = V_{2,\text{RECV}}
\]

\[
V_{2,\text{SEND}} = V_{3,\text{RECV}}
\]

\[
V_{3,\text{SEND}} = V_{4,\text{RECV}}
\]

\[
V_{4,\text{SEND}} = V_{1,\text{RECV}}
\]

Thus, according to the construction method proposed before, it holds
\[
\begin{bmatrix}
I_{1,\text{SEND}} \\
I_{1,\text{RECV}} \\
I_{2,\text{SEND}} \\
I_{2,\text{RECV}} \\
I_{3,\text{SEND}} \\
I_{3,\text{RECV}} \\
I_{4,\text{SEND}} \\
I_{4,\text{RECV}}
\end{bmatrix} = \begin{bmatrix}
Y_{1,1} & Y_{1,2} & 0 & 0 \\
Y_{1,2} & Y_{1,1} & 0 & 0 \\
0 & 0 & Y_{2,1} & Y_{2,2} \\
0 & 0 & Y_{2,2} & Y_{2,1} \\
0 & 0 & Y_{3,1} & Y_{3,2} \\
0 & 0 & Y_{3,2} & Y_{3,1} \\
0 & Y_{4,1} & 0 & Y_{4,2} \\
0 & Y_{4,2} & 0 & Y_{4,1}
\end{bmatrix} \begin{bmatrix}
V_{1,\text{SEND}} \\
V_{1,\text{RECV}} \\
V_{2,\text{SEND}} \\
V_{2,\text{RECV}} \\
V_{3,\text{SEND}} \\
V_{3,\text{RECV}} \\
V_{4,\text{SEND}} \\
V_{4,\text{RECV}}
\end{bmatrix} = Y_{\text{modified}} V_{\text{modified}}, \quad (7.11)
\]

Actually, (7.11) can also be obtained from the following derivations with the introduction of transfer matrix \( M \).

\[
M_1 = \begin{bmatrix}
I_{3\times3} & 0 & 0 & 0 \\
0 & I_{3\times3} & 0 & 0 \\
0 & 0 & I_{3\times3} & 0 \\
0 & 0 & 0 & I_{3\times3} \\
0 & I_{3\times3} & 0 & 0 \\
0 & 0 & 0 & I_{3\times3}
\end{bmatrix}
\]

Rewrite (7.6) by combining (7.7) to (7.10) that
Note that such $M$ can be generalized by considering the topology information of the system and such function as (7.11) can be easily derived.

From the above equations, we can see that the larger the number of network nodes is, the larger the size of $Y$ matrix is. This will cause a minor change of the JSEPT algorithm.
7.3 JSEPT Algorithm Modification

7.3.1 State Estimation

For the state estimation part of JSEPT algorithm, the measurement model in (5.2) needs to be modified according to (7.11). The modified measurement model for system state estimation at time \( k \) with known admittance matrix can be expressed as

\[ z_{N,x,k} = H_{N,p,k} x_{N,k} + v_{N,x,k} \quad (7.13) \]

where

\[ z_{N,x,k} \] measurement vector for state estimation at time \( k \).

\[ z_{N,x,k} = \begin{bmatrix} V_{N,k} \\ I_{N,k} \end{bmatrix} \] that \( V_{N,k} = \begin{bmatrix} V_{1,SEND,k} \\ V_{1,RECV,k} \\ V_{2,SEND,k} \\ V_{4,RECV,k} \end{bmatrix} \) and \( I_{N,k} = \begin{bmatrix} I_{1,SEND,k} \\ I_{1,RECV,k} \\ I_{2,SEND,k} \\ I_{2,RECV,k} \\ I_{3,SEND,k} \\ I_{3,RECV,k} \\ I_{4,SEND,k} \\ I_{4,RECV,k} \end{bmatrix} \)

Also note that the measurements now are decoupled in real and imaginary part. Thus,

\[ z_{N,x,k} \in \mathbb{R}^{72\times1} \]

\( I_{N,k} \) system current measurements at time \( k \), \( I_{N,k} \in \mathbb{R}^{48\times1} \).

\( V_{N,k} \) system voltage measurements at time \( k \), \( V_{N,k} \in \mathbb{R}^{24\times1} \).

\( v_{N,x,k} \) vector of measurement noise, \( v_{N,x,k} \in \mathbb{R}^{72\times1} \) with \( E[v_{N,x,k}] = 0 \) and \( \text{cov}(v_{N,x,k}) = R_v \).

\( x_{N,k} \) vector of states, \( x_{N,k} \in \mathbb{R}^{24\times1} \).

\( H_{N,p,k} \) Jacobian matrix of state in measurement equation at time \( k \), \( H_{N,p,k} \in \mathbb{R}^{72\times24} \). It is defined as

\[ H_{N,p,k} = \begin{bmatrix} I_{24\times24} \\ \hat{H}_{N,p,k} M_2 \end{bmatrix} \quad (7.14) \]
where \( \hat{H}_{N,p,k} \in \mathbb{R}^{48 \times 48} \) is constructed by a roughly approximated parameter, noted as \( \hat{p}_{N,k} \). \( I_{24 \times 24} \) is identity matrix.

And \( \hat{H}_{N,p,k} \) is as follows.

\[
\hat{H}_{N,p,k} = \begin{bmatrix}
\hat{H}_{1,p,k} & 0 \\
0 & \hat{H}_{3,p,k} \\
\hat{H}_{2,p,k} & \hat{H}_{4,p,k}
\end{bmatrix}
\]

(7.15)

where \( \hat{H}_{i,p,k} \) (\( i = 1,2,3 \)) \( \in \mathbb{R}^{12 \times 12} \) denotes the same structure in the matrix of section 5.2.1.

\( M_2 \in \mathbb{R}^{48 \times 24} \) is constructed as:

\[
M_2 = \begin{bmatrix}
I_{6 \times 6} & 0 & 0 & 0 \\
0 & I_{6 \times 6} & 0 & 0 \\
0 & 0 & I_{6 \times 6} & 0 \\
0 & 0 & 0 & I_{6 \times 6} \\
0 & I_{6 \times 6} & 0 & 0 \\
0 & 0 & 0 & I_{6 \times 6}
\end{bmatrix}
\]

7.3.2 Parameter Tracking

There are two ways to realize parameter tracking. One way to solve this problem is to do state estimation and parameter tracking all in the network level, in other word, globally. This can be realized easily by introducing (7.11) as the very equation that generates the state dynamic equation and measurement equation in the parameter tracking part of JSEPT. This will lead the parameter vector to be 18 times the line number \( n \), and the Jacobian matrix will also be enlarged \( n \) times. And this finally will influence the calculation complexity and accuracy of the Kalman gain, which is the multiplication of the Jacobian matrix and its transpose. For a very large system, this gain matrix would be in the size of thousands. Therefore, this method is not applicable.

The other way to solve this problem is to do state estimation globally and do parameter tracking locally. A brief illustration of the procedure is as follows.
• Do three phase state estimation for the whole network, based on (7.13).

• For each transmission line with estimated states and given current or power measurements, do parameter tracking.

• Re-do state estimation using the estimated parameters.

• Iteratively do the above three steps until the estimated parameters converge.

This method is kind of transformation of the joint state and parameter tracking method, since it separates of state estimation and parameter estimation. For a very large network, this method is a good choice since it avoids large amount of calculations for the parameter tracking.

The overall procedure is shown as follows.

• **Step 1: (Initial)**

  Initialization: \( \hat{P}_{r,0|0}, \ P_{r,0|i} \ (i = 1, 2, 3, 4) \) and \( k = 1 \), where \( i \) indicates the line number.

• **Step 2:**

  For each sampling instant \( k (k \geq 1) \), Let \( \hat{P}_{r,k-1|k-1,j} = \hat{P}_{r,k-1|k-1} \), \( P_{r,k-1|k-1,j} = P_{r,k-1|k-1} \) \((i = 1, 2, 3, 4)\) (the value at sampling time \( k-1 \) is available and aim to estimate the value at time \( k \)) and \( j = 1 \).

  The subscript \( j \) indicates the \( j \)-th iteration.

• **Step 3: (PT-time update)**

  Process the ‘Time update’ part of parameter tracking using \( \hat{P}_{r,k-1|k-1,j} \) and \( P_{r,k-1|k-1,j} \)

  \( (i = 1, 2, 3, 4) \).

  Time update:

  \[
  \hat{P}_{r,k-1|k-1,j} = \hat{P}_{r,k-1|k-1}\n  \]

  \[
  P_{r,k-1|k-1,j} = P_{r,k-1|k-1,j} + Q_{r,j}\n  \]

• **Step 4: (SE)**
Formulate $H_{N,p,k,j}$ in (7.14) using the predicted parameters $\hat{p}_{i,k|k-1,j}$ that $\hat{p}_{i,k|k-1,j}$ constructs the $\hat{H}_{i,p,k,j}$ ($i = 1,2,3$) in $\mathbb{R}^{12 \times 12}$ by applying the matrix in section 5.2.1.

Then process state estimation and derive $\hat{x}_{N,k,j}$ by the following equations.

$$
\hat{x}_{N,k,j} = \left[H_{N,p,k,j}^T R_{N,x}^{-1} H_{N,p,k,j} \right]^{-1} H_{N,p,k,j}^T R_{N,x}^{-1} z_{N,x,k}.
$$

**Step 5: (PT-measurement update)**

Formulate $H_{l,x,k,j}$ using $\hat{x}_{l,k,j}$ ($i = 1,2,3,4$) , and then process the ‘Measurement update’ part of parameter tracking.

Measurement update:

$$
\Delta z_{l,p,k,j} = z_{l,p,k} - H_{l,x,k,j} \hat{p}_{l,k|k-1,j}
$$

$$
\Phi_{l,k,j} = H_{l,p,k,j} P_{l,k|k-1,j} H_{l,p,k,j}^T + R_{l,p}
$$

$$
K_{l,k,j} = P_{l,k|k-1,j} H_{l,p,k,j}^T \Phi_{l,k,j}^{-1}
$$

$$
\hat{p}_{l,k|k,j} = \hat{p}_{l,k|k-1,j} + K_{l,k,j} \Delta z_{l,p,k,j}
$$

$$
P_{l,k|k,j} = P_{l,k|k-1,j} - K_{l,k,j} \Phi_{l,k,j} K_{l,k,j}^T
$$

**Step 6: (iteration of internal loop)**

If $j > 1$ and $\|\hat{p}_{l,k|k,j} - \hat{p}_{l,k|k,j-1}\| < \varepsilon_i$, ($i = 1,2,3,4$) , go to Step 7;

If $j$ is too large $\ l > N_{\text{threshold}}$, go to Step 7;

Otherwise, repeat Step 3 to Step 6 with new $j = j + 1$ and $\hat{p}_{l,k|k-1,j} = \hat{p}_{l,k|k,j-1}$ , $P_{l,k|k-1,j} = P_{l,k|k,j-1}$.

**Step 7: (iteration of measurement loop)**

The iteration of $j$ is terminated. The estimated value at this instant $k$ is $\hat{p}_{l,k} = \hat{p}_{l,k|k,j}$ , $P_{l,k} = P_{l,k|k,j}$ and $\hat{x}_{N,k} = \hat{x}_{N,k,j}$.
If \( k \leq N_{\text{sampling}} \), let \( k = k + 1 \), go back to Step 2;

Else, go to Step 8.

- **Step 8:**

  Terminate.

\( \varepsilon_i \)  
  predetermined threshold of line \( i \) to decide the termination of internal iteration;

\( N_{\text{threshold}} \)  
  the iteration limit of \( j \) in each instant \( k \);

\( N_{\text{sampling}} \)  
  the number of sampling measurements.

### 7.4 Simulations of Network Model

For the simulation model in Figure 7.1, the line parameters are constant. The actual values of 4 lines can be calculated as the followings. Note that line 1 is the same as the model for single line parameter estimation discussed before.

<table>
<thead>
<tr>
<th>( p_1 )</th>
<th>( p_2 )</th>
<th>( p_3 )</th>
<th>( p_4 )</th>
<th>( p_5 )</th>
<th>( p_6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5963</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( p_7 )</td>
<td>( p_8 )</td>
<td>( p_9 )</td>
<td>( p_{10} )</td>
<td>( p_{11} )</td>
<td>( p_{12} )</td>
</tr>
<tr>
<td>Value</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19.9661</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( p_{13} )</td>
<td>( p_{14} )</td>
<td>( p_{15} )</td>
<td>( p_{16} )</td>
<td>( p_{17} )</td>
<td>( p_{18} )</td>
</tr>
<tr>
<td>Value</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.3487</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( p_1 )</th>
<th>( p_2 )</th>
<th>( p_3 )</th>
<th>( p_4 )</th>
<th>( p_5 )</th>
<th>( p_6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7.1  Actual values of line 1 parameters

<table>
<thead>
<tr>
<th>( p_1 )</th>
<th>( p_2 )</th>
<th>( p_3 )</th>
<th>( p_4 )</th>
<th>( p_5 )</th>
<th>( p_6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19.96</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( p_1 )</th>
<th>( p_2 )</th>
<th>( p_3 )</th>
<th>( p_4 )</th>
<th>( p_5 )</th>
<th>( p_6 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.3487</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7.2  Actual values of line 2 parameters
<table>
<thead>
<tr>
<th>Value</th>
<th>3.9915</th>
<th>-49.6192</th>
<th>-0.5102</th>
<th>23.8167</th>
<th>0.4092</th>
<th>13.7663</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$p_7$</td>
<td>$p_8$</td>
<td>$p_9$</td>
<td>$p_{10}$</td>
<td>$p_{11}$</td>
<td>$p_{12}$</td>
</tr>
<tr>
<td>Value</td>
<td>49.9197</td>
<td>-23.9088</td>
<td>-13.8044</td>
<td>4.1969</td>
<td>-59.2151</td>
<td>-0.7888</td>
</tr>
<tr>
<td></td>
<td>$p_{13}$</td>
<td>$p_{14}$</td>
<td>$p_{15}$</td>
<td>$p_{16}$</td>
<td>$p_{17}$</td>
<td>$p_{18}$</td>
</tr>
<tr>
<td>Value</td>
<td>23.8171</td>
<td>59.5543</td>
<td>-23.8975</td>
<td>3.178</td>
<td>-49.5979</td>
<td>49.9286</td>
</tr>
</tbody>
</table>

Table 7.3  Actual values of line 3 parameters

<table>
<thead>
<tr>
<th>Value</th>
<th>3.1932</th>
<th>-39.5602</th>
<th>-0.4082</th>
<th>19.0119</th>
<th>0.3274</th>
<th>10.9959</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$p_7$</td>
<td>$p_8$</td>
<td>$p_9$</td>
<td>$p_{10}$</td>
<td>$p_{11}$</td>
<td>$p_{12}$</td>
</tr>
<tr>
<td>Value</td>
<td>39.9357</td>
<td>-19.127</td>
<td>-11.0435</td>
<td>3.3575</td>
<td>-47.2194</td>
<td>-0.631</td>
</tr>
<tr>
<td></td>
<td>$p_{13}$</td>
<td>$p_{14}$</td>
<td>$p_{15}$</td>
<td>$p_{16}$</td>
<td>$p_{17}$</td>
<td>$p_{18}$</td>
</tr>
</tbody>
</table>

Table 7.4  Actual values of line 4 parameters

<table>
<thead>
<tr>
<th>Value</th>
<th>1.2773</th>
<th>-15.0351</th>
<th>-0.1636</th>
<th>7.3637</th>
<th>0.131</th>
<th>4.299</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$p_7$</td>
<td>$p_8$</td>
<td>$p_9$</td>
<td>$p_{10}$</td>
<td>$p_{11}$</td>
<td>$p_{12}$</td>
</tr>
<tr>
<td>Value</td>
<td>15.9742</td>
<td>-7.6512</td>
<td>-4.418</td>
<td>1.3419</td>
<td>-17.9963</td>
<td>-0.2528</td>
</tr>
<tr>
<td></td>
<td>$p_{13}$</td>
<td>$p_{14}$</td>
<td>$p_{15}$</td>
<td>$p_{16}$</td>
<td>$p_{17}$</td>
<td>$p_{18}$</td>
</tr>
<tr>
<td>Value</td>
<td>7.3973</td>
<td>19.0566</td>
<td>-7.6484</td>
<td>1.0174</td>
<td>-14.9423</td>
<td>15.9759</td>
</tr>
</tbody>
</table>
As the discussion in section 2.3.4, the scenarios with different noise are tested.

- **Scenario 1**: no adding noise to the measurements;
- **Scenario 2**: measurements are added with noise that has $10^{-4}$ standard deviation.
- **Scenario 3**: measurements are added with noise that has $10^{-2}$ standard deviation.

The following results hold.

![JSEPT network scenario 1](image)

**Figure 7.2** JSEPT network scenario 1
Figure 7.3  JSEPT network scenario 1 EVT

Figure 7.4  JSEPT network scenario 2
Figure 7.5  JSEPT network scenario 2 EVT

Figure 7.6  JSEPT network scenario 3
The above four figures give the results under the situation that the initial is precise. For both scenarios, JSEPT performs very well to track the parameters. Compared with the results in Figure 5.6 and Table 5.2 where the noise is $10^{-4}$ standard deviation, Figure 7.4, Figure 7.5, Figure 7.6 and Figure 7.7 indicate that the accuracy of parameter estimation improves much when it is introduced to the system level.

As a brief summary, the proposed modified JSEPT algorithm did work for the network system under some trivial scenarios. But it still requires further analysis and evaluation.
Chapter 8

Conclusions and Future Work

8.1 Conclusions

This thesis aims at developing appropriate methods to estimate parameters for untransposed transmission lines. First, a static method using sliding window is introduced, tested and assessed by simulating on two different system models. Then two dynamic methods are proposed: augmented state estimation (ASE) and joint state estimation and parameter tracking (JSEPT). Their theoretical implementations are formulated; their performances are assessed based on numerical results under appropriate scenarios; their advantages and disadvantages are discussed by comparing against each other. Finally, JSEPT is implemented to the system network.

Detailed conclusions arrived in this thesis are listed below:

In Chapter 2, structures and properties of the transmission line parameters are investigated. A comparison of transposed and untransposed transmission line is given with respect to the number and position of the unknown parameters. Specific line models are constructed and evaluated numerically. The measurement equation of the line model is formulated then. One simulation model is built with two test cases and six kinds of scenarios to be estimated in the following chapters.
In Chapter 3, static parameter estimation (SPE) is developed and simulated. It is implemented by introducing the sliding window method to estimate the parameters using several time scans. Weighted least squares is the major technique. Simulation results are provided by comparing the scenarios with different foot step and with different kinds of measurement noise. The conclusion is that the larger the foot step is, the more accurate the results are for the constant parameter case. For the noisy and parameter changing cases, SPE does not perform well.

In Chapter 4, augmented state estimation (ASE) algorithm is proposed. It augments the state vector with 18 unknown parameters. And then the nonlinear measurement equation and state dynamic equation are built according to the line information. It is possible to use iterated unscented Kalman filter to solve this problem with proper running time and relatively accurate results. Simulation results show that ASE is much better than SPE for parameter estimation, especially for the noisy cases and parameter changing cases. One of the shortcomings of ASE is the analytical assumption that system states have dynamics. Actually it is not practically applicable. Another one is its low robustness against measurement noise.

In Chapter 5, joint state estimation and parameter tracking (JSEPT) algorithm is developed. State estimation problem and parameter tracking problem are constructed and solved iteratively. For state estimation, it uses roughly estimated parameters to build state Jacobian. Then, by constructing the parameter Jacobian using the estimated states, the parameters are tracked by Kalman filter. SE and PT are processed iteratively until the difference between the newly estimated parameters and the previously estimated parameters is small enough. Simulation results indicate that the performance of JSEPT performs well for all the scenarios, whether or not there is measurement noise or the parameters are changing. What’s more, it realizes the online parameter tracking instead of offline, which is helpful to the dynamic analysis of the power system.

In Chapter 6, comparisons of the proposed algorithms are given. From the point view of initial values, both ASE and JSEPT are influenced by incorrect initials. But for JSEPT, the impact is much less and it can overcome the bias of the initial values. For the comparison of running time, SPE is the best while ASE is the worst. Considering the overall performance of the three algorithms, JSEPT is the best one to do parameter estimation for untransposed transmission lines.
In Chapter 7, JSEPT is modified to be suitable for the system level parameter estimation. A new model is given with 4 transmission lines and 4 buses. The modified algorithm runs state estimation globally and parameter tracking locally. Preliminary results show that JSEPT works fine for the network problem.

8.2 Future Work

The future work is mainly to improve the algorithm of JSEPT from the following aspects.

- Tuning of the covariance;

  Tuning technique is needed for JSEPT to obtain a more accurate result. It is necessary to find the technique, such as self-tuning regulator, to realize the tuning of JSEPT automatically.

- Convergence condition evaluation.

  The value of the threshold to determine the convergence is to be evaluated carefully. What’s more, new termination conditions are under investigation that would help the algorithm converge sooner.

- Data from real world needs to be tested under the proposed method.
REFERENCE


The effects of incorrect measurement nansimultaneity, bias and parameter uncertainty on power system State estimation,


APPENDIX

Interpretation of Symbols in ATP:

- Voltage/Current source
- Transmission line/cable object which allows users to input
- 3-phase independent RLC load in Y connection
- Voltage probe: output node voltage
- Current probe: output branch current