AN ABSTRACT OF THE DISSERTATION OF

Tao Lyu for the degree of Doctor of Philosophy in Electrical and Computer Engineering presented on March 20, 2020.

Title: The Partitioning Technique of Power System to Improve The Observability of Sub-System for Multi-area State Estimation

Abstract approved: _____

Mario Edgardo Magaña

An accurate state estimation plays an essential role in power system operation and planning in energy management systems. However, existing multi-area state estimation researches have not focused on the importance of system clustering. The clustering mechanism divides or partitions a system according to user-defined criteria. Few published research works have mentioned the importance of considering the electrical properties of a power system while devising their partitioning methods.

To the best of our knowledge, these publications have not considered the application of such a concept to multi-area state estimation. This research attempts to model a partitioning technique of the power system whose purpose is to ensure the sub-system observability prior to the multi-area state estimation. Hence, the accuracy of the multi-area state estimation could be improved. A modified genetic algorithm based phasor measurement unit (PMU) placement is introduced in this thesis, which includes the electrical distance based additional PMU installation technique. The modified partitioning method is introduced in this thesis based on a genetic algorithm partitioning algorithm. In the modified partitioning method, a proposed genetic index is proposed aiming to include the consideration of system observability, which employs the proposed PMU placement technique to represent the sub-system observability.

In addition to the partitioning method, few publications have considered the state estimation by only employing PMU measurements. Furthermore, few publications have considered employing the noise statistics estimation technique to state estimation to improve the convergence

of the estimation process. A cubature Kalman filter based algorithm (CKF) is used in the thesis to solve the state estimation problem where only PMU measurements data are employed. An online noise statistic estimation technique is incorporated into the CKF to improve convergence. A modified two-level MASE is introduced to implement the modified CKF. The modified partitioning method is applied to the multi-area state estimation algorithm. By employing all the techniques introduced in this thesis, a considerable improvement of accuracy and convergence can be achieved.

The Partitioning Technique of Power System to Improve The Observability of Sub-System for Multi-area State Estimation

by

Tao Lyu

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented March 20, 2020
Commencement June 2020

Doctor of Philosophy dissertation of Tao Lyu presented on March 20, 2020.

APPROVED:

Major Professor, representing Electrical and Computer Engineering

---

Head of the School of Electrical Engineering and Computer Science

---

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

Tao Lyu, Author

ACKNOWLEDGEMENTS

First of all, I would like to acknowledge my major advisor Prof. Magaña. He gave me needed support and guidance during my Ph.D. studies at OSU. I sincerely appreciate that and it is hard to find words to express my gratitude.

Secondly, I would like to express my deep appreciation to Prof. Eduardo Cotilla-Sanchez. He gave me guidance when I met difficulties in the power systems area and he was always willing to schedule an appointment with me. He helped me overcome my lack of confidence during difficult moments.

I would like to thank my manager Michael from DEA. Inc. His support helped me draw a clear path to pursue my Ph.D. degree and his encouragement helped me overcome difficulties either in life or with work. I would also like to thank my colleagues Frances, Patrick, Ben, and Aaron.

I would also like to thank my committee members, Prof. Ted Brekken and Prof. Jinsub Kim.

Lastly, I would like to thank my family and my lovely girlfriend. Thank you so much for your strong support. You are the most important motivation that keeps me moving forward.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

Figure Page

LIST OF FIGURES (Continued)

LIST OF TABLES

LIST OF TABLES (Continued)

Table Page

LIST OF TABLES (Continued)

Chapter 1: Introduction

With the increasing scale of power system networks and uncertainty brought about by distributed generators (DGs), the power system state estimation (PSSE), as one of the core components of energy management system (EMS), has become an extremely challenging task to meet real-time analysis and security control requirements due to the complexity of large scale networks [76]. In order to resolve the challenge, researchers have investigated multi-area state estimation (MASE) or distributed state estimation (DSSE) to replace the traditional centralized state estimation (CSE) [95][76]. However, current research mainly focuses on the improvement of the algorithm for "state-estimation" rather than focusing on the the effect of "multi-area". In other words, the accuracy of the MASE is not only affected by the state estimation algorithm, but also affected by how proper the sub-systems are constructed for the involved power networks. This lack of knowledge in this area attracts our attention, and it becomes the major focus of this research.

The mechanism to divide a network or system is called partitioning or clustering. As for the partitioning methods of data and networks, it has a long history in the scientific literature [53, 38], from which it can be seen that the clustering of data applies to a variety of problems. During the past years, many approaches have been proposed for dividing power networks into clusters [19]. [46, 47] employed simulation methods to divide a power network. [60] developed a hierarchical clustering method for the clustering, which takes the active and reactive power mismatch between areas into consideration. [92, 4] used degree distribution based on the geographical theory of a large scale network. However, few physical laws are neglected, such as Ohm's law and Kirchoff's law, if those network methods are solely employed. This means that the fundamental characteristics of an electrical network, such as electrical connections or power flows between components are not considered to be enough [69]. The issues were illuminated by the method introduced in [19, 18], whose method mainly focuses on the electrical characteristic by developing a measurement named electrical distance. The electrical distance relates the network topology to active-power sensitivities based on the information found in the system admittance matrix Y_{bus} [19]. Thus, the method introduced in [19, 18] is the basis of our research contribution in this thesis. However, the objective in [19, 18] is to reduce the transactions

leakage, which is also called a loop flow, for the power system transmission planning. In other words, the factors related to state estimation are not taken into considerations in [19, 18]. As for the reason that the PSSE is the area we want the partitioning method to be applied to, the factors related to the PSSE need to be incorporated into the partitioning method.

In state estimations, the observability of the system is the most important concern. Network observability analysis must be performed prior to state estimation execution. The reason is that state estimation tries to assign values to unknown system state variables based on obtained measurements from the system [32, 52]. If there are enough measurements well distributed throughout the network in such a way that state estimation is possible, the system is said to be observable. The observability of a system is significantly related to the location of data measuring devices and the way they are distributed. Four main types of algorithms are used for observability analysis, which are algebraic, numerical, topological and hybrid [16, 54, 61]. In this thesis, the topological observability is the main area that we focus on. In the topological observability, two major types can be classified, namely, the conventional and PMU based observability algorithms. In PMU-based observability, it is mainly affected by the PMU placement in the power network. It implies that the system observability would be ensured with proper PMU placement solutions.

A PMU is a device that measures electrical waves employing synchronization signals, which is faster than traditional measurements. It can improve the accuracy of measurements and is also is beneficial to many other applications like system protection and control assessment [72]. Besides, the introduction of PMUs makes it possible to directly measure the phase angle between the phasors at different locations. Traditionally, state estimation is formulated as a weighted least squares problem due to the absence of phasors measurements. This problem can be solved by involving the PMUs, which significantly increases the efficiency of solving state estimation problem with proper placement. However, the implementation of PMU solutions are costlier than SCADA based systems due to the high cost of the PMU devices, communication facilities cost, and maintenance cost. It makes the PMU placement problem a challenging one [44]. The problem of solving the PMU placement problems for network observability is addressed in [28]. The study mentions that it is possible to fully monitor the system by using relatively few PMUs than the number of system buses based on the properties of the PMU associated with the connectivity of the involved network. This problem is solved by using a graph theoretic observability analysis and its proposed optimization method. However, the possibility of the failure of PMUs (or losing PMU) is not considered in the majority of research papers [44, 72, 17, 70, 74], which

would turn a completely observable system into an unobservable one.

In this research, we propose a PMU placement algorithm which can not only ensure the complete system observability but also maximize the system observability value when failures of PMUs are considered. The proposed PMU placement algorithm is modified by inserting additional PMUs into the results obtained by the genetic algorithm based PMU placement technique introduced in [7, 87]. The additional PMU location is evaluated according to the electrical distance based adjacent matrix. The system observability is represented by the measurement redundancy value produced from the PMU placement solutions. With the proper setting, the redundancy value becomes one of the important measures in the partitioning method using, for example, a genetic algorithm.

In order to evaluate the performance of the partitioning method, the state estimation should be applied. The typical state estimation algorithm is weighted least square (WLS) algorithm [64]. However, the nonlinear measurement function will increase the computation burden of WLS algorithm which would not satisfy the real-time requirement of today's power system state estimation. Kalman filter has been widely used as it is a nonlinear filtering method having high accuracy on the application of nonlinear system state estimation [97, 98]. Three kinds of kalman filters are used including the extended kalman filter (EKF), the unsected kalman filter, and the cubature kalman filter (CKF). The CKF is currently the most popular one among all three kalman filters, especially for nonlinear systems having Gaussian distributed noise models [31, 3, 30]. The reason is that it does not require the approximation of the nonlinear model as EKF and it has a stronger nonlinearity due to the use of cubature points [3]. However, the main drawback for the CKF is the requirement of the knowledge of the system and measurement noises statistical models. To the best of our knowledge, the majority of the CKF applications assumed prior knowledge of the system and measurement noise models whose initial values are evaluated based on the observation from large amount of experiments [81, 3]. In order to improve the estimation accuracy without increasing the computational burden, [31, 30] introduces an online noise statistic estimation technique. It updates the noise at each step of the estimation process.

In our research, the online noise estimation technique will be employed in the CKF state estimation algorithm to improve convergence. A modified two-level multi-area state estimation will be introduced as well, which takes advantage of the features of PMUs. Performance evaluations are presented in the latter part of the thesis to show the improvement on the system observability by using additional PMUs and the partitioning method with the proposed index. Furthermore, it also presents the performance evaluation when the modified CKF algorithm and modified two-

level multi-area state estimation are used.

Chapter 2 will present the fundamental ideas that the research is based on. Chapter 3 discusses the proposed techniques and methods utilized in the research. In order to present the improvement due to the employment of the proposed technique and the modification, chapter 4 presents our simulation results for the performance evaluations.

Chapter 2: Preliminaries

2.1 Infrastructure of Power Grids

The electric power infrastructure is vital to modern society. Therefore, its reliable operation is of the utmost importance [40]. The work in [62] is the earliest paper that emphasized the value of visualizing a power system. After that, many researches started focusing on the representation of the power grids by a proper infrastructure. Two kinds of models exist in current research papers to characterize the structure of the complex power system. One mainly focuses on using the topological information to construct the model and the other one considers the electrical properties on top of the geography characteristics. This section gives a brief overview of the structure analysis based on the graph theory for the reason that it is the basis of the electrical based power system structure.

2.1.1 Topological Structure Based Power Grids

The topological based power grids structure is developed based on graph theory. The main reason for choosing graph theory is that the graph structure of complex networks can provide valuable information about the performance of the network [21, 46, 90, 92, 23]. [41] provides a review on the typical construction of the power system based on the graph theory.

2.1.1.1 Basic Measures of Graph Topology

Some basic concepts of graph theory are now introduced in this brief review. According to [91], a graph G can be defined as a triple consisting of a vertex set $V(G)$, an edge set $E(G)$, and a relation that associates with each edge two vertices called its endpoints. $V(G)$ is used to denote the order of a graph in G, and the number of edges $E(G)$ denotes the size of a graph. In addition to that, the adjacency matrix is the important matrix that represents the connectivity of a network. The adjacency matrix A is a $n \times m$ matrix whose elements take on binary values, in which n denotes the number of nodes, and m denotes the number of edges. An entry of A will be 1 if the nodes n_i and n_j are adjacent (or connected), 0 otherwise. Thus, the matrix can be

expressed as:

$$a_{ij} = \begin{cases} 1, \forall\ n_i \textit{ and } n_j \textit{ that are connected} \\ 0, \textit{otherwise} \end{cases} \tag{2.1}$$

The adjacency matrix of a simple graph is symmetric, which means $a_{ij} = a_{ji}$ [91]. The reason is that both of these entries are 1 when n_i and n_j are adjacent, and a simple graph has no loops.

Five typical statistical measures are used to characterize the topology of complex networks, which are degree distribution, clustering coefficient, characteristic path length, graph diameter, and degree assortativity [91]. These measures provide a useful set of statistics for comparing power grid with other graph structure [18, 23, 41, 46, 92].

The degree distribution is used to describe the diversity of connectivity in a graph. The degree k_i is the number of edges connecting to vertex i. Its expression is associated with the pre-defined adjacency matrix A having entries a_{ij}, which is denoted as 1 if vertexes i and j are connected, and 0 otherwise. This is denoted by [41]:

$$k_i = \sum_{j=1}^{n} a_{ij} \tag{2.2}$$

The degree distribution can then be evaluated by a distribution function $P(k)$, which gives the probability that a randomly selected vertex has k edges. It can be expressed as [41]:

$$P(x = k) = \frac{n_k}{n} \tag{2.3}$$

where n_k is the number of nodes with degree k and n is the total number of nodes. Recent studies show that the degree distribution of some complex networks follow a power law function [91, 41]. That is, some vertices have a very large number of edges, but the majority of the vertices only have a few edges [41]. The power law function can be expressed as [18]:

$$P(k \geq x) = \left(\frac{x}{x_{min}} \right)^{-\alpha+1} \tag{2.4}$$

where x_{min} is a minimum value for the power-law tail of the degree distribution.

Many complex networks exhibit an inherent tendency to cluster. The clustering coefficient is a local property capturing "the density" of triangles in the graph, i.e., two vertices that both are connected to a third vertex are also directly connected to each other [91], which can be expressed

as [41]:

$$C = \frac{1}{n}\sum_{i=1}^{n} c_i \tag{2.5}$$

c_i is described by [41]:

$$c_i = \frac{2e_i}{k_i(k_i-1)} \tag{2.6}$$

where $e_i = \sum_{\forall j,k \in \{N_i \cup i\}} \frac{a_{jk}}{2}$ is the number of links within the cluster of nodes including node i and its immediate neighbors N_i.

In terms of the characteristic path length, it is also called the average path length. It is used to measure how the network is scattered [41]. The concept of the average path length would involve the concept of the path length. The path length is the shortest path connecting two vertices i and j. It gives the minimum number of links that one would need to traverse to arrive from one node to the other, denoting as d_{ij}. The path length can then be expressed as:

$$L = \frac{1}{n(n-1)}\sum_{\forall i \neq j} d_{ij} \tag{2.7}$$

The graph diameter d_{max} is defined as the maximum path length between any two connected vertices in the graph,which can be expressed as:

$$d_{max} = \max_{ij} d_{ij} \tag{2.8}$$

The last measure is the degree assortativity r. It is defined as the extent to which nodes connect to nodes with similar degree [18]. It is the correlation in degree for the nodes on opposite ends of each link [91]:

$$r = \frac{m^{-1}\sum_{i=1}^{m} j_i \cdot k_i - [m^{-1}\sum_{i_1}^{m} 0.5(j_i + k_i)]^2}{m^{-1}\sum_{i=1}^{m} 0.5(j_i^2 + k_i^2) - [m^{-1}\sum_{i_1}^{m} 0.5(j_i + k_i)]^2} \tag{2.9}$$

.

2.1.1.2 Theoretical Network Models

Three main kinds of network models are commonly used for complex network studies, which are born based on the interest in the statistical measures of the topology of networks that are

introduced above. The three main models include random graph model, small world model, and the scale-free network model [91]. In addition to that, references [18, 92, 4, 46, 47] state that the power system can be represented as an undirected, unweighted graph with n nodes representing the buses involved, and m edges representing the branches of the system.

According to [91], the random graph model starts with n isolated vertices. In this model, each pair of vertices are independently connected by an edge with probability p. The average number of edges in the graph is $p \cdot \frac{n(n-1)}{2}$ and the distribution degree would follow the binomial distribution. However, the degree distribution would have a Poisson distribution when the system becomes extremely large and the probability p is approaching to zero. In this scenario, the degree distribution can be expressed as:

$$P(k) = \frac{e^{-\gamma} \cdot \gamma^k}{k!} \tag{2.10}$$

where γ is approximately equal to the average degree $p \cdot n$. In this degree distribution scenario, the degree distribution value has a strong peak at the mean value $\bar{k}$ and it will exponentially decay when the value moves far away from the mean value. Besides, the clustering coefficient will be $C = \frac{\bar{k}}{n-1}$ under this scenario. In any scenario of the random graph model, the average path length is proportional to $\log n$ for the n independent involved nodes, which is the main reason that the random model would fit for many complex network structures.

[71] introduces few characteristics of the small-world model. First of all, the nodes and links of this model form a ring lattice layout, which specific number can be denoted as n for the nodes and m for the links. It means that node a and neighbor node $a-1$ are connected to each other and a second link is then created from node a to node $a-2$ with probability $p = \frac{m}{n-1}$. In addition to that, the clustering coefficient and the average path length values will grow only logarithmically with n for the small-world network model, and the probability p can be adjusted to obtain a network with the same diameter as that for the real-world network.

If the degree distribution $P(k)$ follows a power-law for large k having no pronounced peak, and there is no intrinsic scale in these networks, the network model is called scale-free network [41]. A scale-free network can be constructed by progressively adding nodes to an existing network and introducing links to existing nodes with preferential attachment so that the probability of linking to a given node i is proportional to the number of existing links k_i that node has probability as $\frac{k_i}{\sum k_j}$, $i \neq j$ and $i, j \in n$ [71]. The main property of the scale-free network is that some nodes have a tremendous number of connections (or called links/branches) to other nodes, whereas most nodes have just a few [41, 71]. The scale-free networks have certain robustness

against accidental failures but vulnerable to coordinated attacks [41].

2.1.2 Comparison of Topological and Electrical Based Power Structures

Paper [18] shows some disagreement over the topological structure representation of power networks by using a larger, more accurate model of the North American power grid than has been used in the past. In order to do that, [18] carried out a comparison of the topology of the three US interconnections to similarly-sized small-world, scale-free and random graphs. The comparison in [18] includes the statistical description of power grids topology and their counterpart canonical graphs. The statistical description includes the five measures that we introduced above.

By using the degree distribution, [18] points out that there are more high-degree nodes in a random graph or a small world network model than that in scale-free network model even if the size of three models are similar. It means that the power network degree distribution is not the same as that for the three commonly used network models having similar size. In addition to that, [18] shows that the power-law exponents in the simulation results show that it is not what would generally be considered notable power-law distribution for all of the North American power network degree distribution. It means that the power network does not follow the power-law degree distribution as the graph theory [91] assumed to be. Another assumption from [71, 11] has been rejected by [18] as well, which is the average path length of complex power networks will increase under the upper bound as $\ln(n)$ with the increasing number of nodes n according to the graph theories. [18] shows that the path lengths of a large power network increases substantially faster than $\ln(n)$. The proofs presented in [18] safely conclude that either the small-world model, or the scale-free and random network model has inappropriate properties to represent the complex power networks.

Besides that, [40] states that the topological models to assess vulnerability in electricity system can lead to provocative, but ultimately misleading conclusions. Three measures are considered in [40] for the comparison between real system structure and topological based power networks. They are average path length, connectivity loss, and blackout sizes. The average path length in this paper is used to present the average among nodes in the power networks, which will increase as more components fail than nodes the network has. The evaluation expression is the same as that from the graph theory. The connectivity loss is a way to incorporate the locations of generators and loads into a measure of network vulnerability. Its expression is constructed based

on the properties from the graph theory, which is:

$$C_{loss} = 1 - \frac{n_g^i}{n_g} \tag{2.11}$$

where n_g is the number of generators in the network and n_g^i is the number of generators that can be reached by traveling from node i across non-failed links. The blackout size is the total amount of load evaluated by a balance process based on the model of power flow to capture the effects of Ohm's and Kirchhoff's laws, which is used to calculate changes in network flow pattern caused by component(s) failure.

In order to do the vulnerability analysis, five virtual attack scenarios are set for the test, which are named as random failure, degree attack, maximum-traffic attack, minimum-traffic attack, and betweenness attack. Random failure means randomly selecting nodes in the power netowrk with an equal probability for each node. Degree attack is removing nodes incrementally, starting with the highest connection nodes in the power networks. The maximum-traffic attack is defined as removing incrementally starting with those that transport the highest amounts of power, in which the "traffic" is used to describe the quantity of power being consumed at a node, expressed as:

$$T_i = \|P_i\| + \sum_{j=1}^{n} \|\frac{(\theta_i - \theta_j)}{X_{ij}}\| \tag{2.12}$$

where $\|P_i\|$ is the absolute value of net power injection into node i by generators and loads, θ is the voltage phase angle at the node, and X_{ij} is the series reactance of the links between node i and j. The minimum-traffic attack is the inverse of the max-traffic attack defined above. The betweenness attack is defined as removing the nodes incrementally starting with those that have the highest number of shortest paths that pass through a node.

To compare the vulnerability measures, [40] reports results from the simulation of random failures and directed attacks defined above for the North American Eastern power grids. Results show that the betweenness attacks appear to have the greatest impact on the perspective of path length measure, the degree-based attacks are the most dangerous from the stand point of connectivity loss, and the max-traffic attacks appear to contribute mostly to vulnerability if the blackout size is the focus. In other words, the results from [40] indicate that the topological measures can provide some indication of general vulnerability trends but they might result in an erroneous conclusion while the physics-based concepts are involved.

2.1.3 Electrical Structure Based Connection Distance

From the previous discussion, it can be concluded that the topological structure is not sufficient to describe the performance of power networks. Thus, a new method for representing the electrical structure would be more appropriate. [58, 62] might be the earliest works the dealt with the connectivity between components in an electrical system for the structure representation, which is described by the properties of sensitivity matrices. The definition of sensitivity matrices is the measurement for the amount of affects that one component in the system has on the other ones [8] when that component has been lost or has failed. For the electrical system, the complement of the sensitivity matrix is the distance matrix [8]. Zero values for the sensitivity matrix means that a perfect connection between a paired of nodes/buses, which means there is almost no impact from one node to another in this pair. Based on this sensitivity matrix definition, recent researches show two major kinds of definitions of the distance between nodes in power networks, and they are both modeled by considering the electrical properties. These two distances are defined as resistance distance and electrical distance.

2.1.3.1 Resistance Distance

To the best of our knowledge, the earliest work on resistance distance was developed in paper [51] in 1993, whose theories have been cited by many electrical based power network papers such as [6, 18, 40]. The theory of the resistance based power network is that the effective resistance between pairs of nodes/buses is a graphical distance if the resistors are constant on each edge of a connected graph [51]. The idea of the resistance distance is evaluated as effective resistances for either patterns of connection of the battery between other pairs of vertices [6, 51], examples are shown in Fig. 2.1.

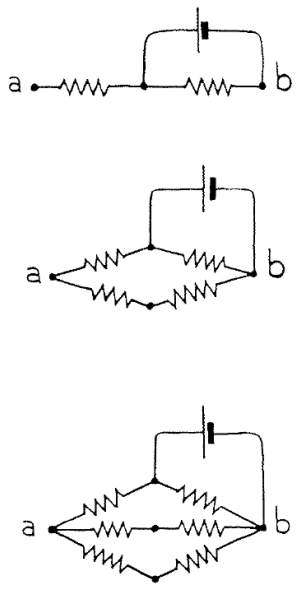


Figure 2.1: Battery Connected to Different Pairs of Nodes

Assigning 1 Ohm resistors for these three kinds of connections, the distance in Ohms can be found as 1 Ohms, $\frac{3}{4}$ Ohms and $\frac{2}{3}$ Ohms. It can be seen that the distance between two nodes is not necessarily the sum of the distance in between. Let G be the conductance matrix, then as suggested in [18]:

$$I_a = \sum_{b=1}^{n} g_{ab} V_b \tag{2.13}$$

where I_a is the injection current at node a, V_b is the potential difference at connected node b, and g_{ab} is the conductance value between nodes a and b from the conductance matrix G.

At this point, the properties of the Laplacian matrix can be associated with [6, 18, 51, 93]. The Laplacian matrix for an undirected unweighted network is the difference between the degree matrix D and adjacency connectivity matrix A, expressed as:

$$\begin{aligned} L &= D - A \\ l_{ij} &= \begin{cases} d_i, \, if \; i = j \\ -1, \, if \; i \neq j \; and \; i, j \; connected \\ 0, \, otherwise \end{cases} \end{aligned} \tag{2.14}$$

where the l_{ij} is the (i, j) entry of L, d_i is the degree value at node i. The properties of the Laplacian matrix are [6, 91]:

- L is a symmetric, positive semi-definite matrix;
- The rank of L is $n-k$, where k is the number of connected components of a graph;
- The co-factors of any two elements of L are equal;

Based on the basic concepts of the Laplacian matrix, the conductance matrix G can be regarded as a Laplacian matrix for the resistive network [6, 18]. As for the properties of the power grid, the rank of G is $n-1$ when there is no voltage reference specified. In other words, if a reference node n_0 is defined, with the voltage magnitude as 0, the rest of the sub-matrix $\overline{G}$ is a full rank matrix. The sub-matrix $\overline{G}$ will still follow Ohm's laws, which means

$$\overline{V} = \overline{G}^{-1} \cdot \bar{I} \tag{2.15}$$

where $\overline{V}$ is the voltage matrix and $\bar{I}$ is the current matrix. Based on the last equation, the diagonal elements of $\overline{G}^{-1}$ are assumed to be associated with node a, which can be expressed as $g_{a,a}^{-1}$. The value of $g_{a,a}^{-1}$ represents the change in voltage between n_a and n_0 due to the current injection at n_a [18, 51]. In terms of resistance distance [51], the sensitivity matrix is defined as the voltage differences caused by the current injection, expressed as $g = \frac{\Delta I}{\Delta V}$. Hence, the resistance distance between a pair of nodes a and b, $n_a \neq n_b \neq n_0$ can be evaluated by [18, 93]:

$$r_{a,b} = g_{a,a}^{-1} + g_{b,b}^{-1} - g_{a,b}^{-1} - g_{b,a}^{-1} \tag{2.16}$$

which presents the voltage difference between node a n_a and node b n_b after injecting unity current at one node and withdrawing unity current from the other node. The development of resistance distance acts as the starting point for much research on power systems aiming at considering the electrical properties of power networks.

2.1.3.2 Electrical Distance

The electrical distance is developed in [19], whose starting point is to have a measure of electrical connectedness so that the power networks can be clustered by considering the electrical properties. It is built based on the information found in the system admittance Y_{bus} matrix. The

concept of the electrical distance is constructed by changing the emphasis of the sensitivity matrix. The focus for the sensitivity matrix in [19] is to obtain the phase angle difference after injecting real power at an arbitrary node in the power network [18]. The modification of the sensitivity matrix starts from the Jacobian matrix of power flow. According to [33], the basic idea of Jacobian matrix of power flow can be expressed as:

$$J_{P,Q} = \begin{bmatrix} \frac{\Delta P}{\Delta \theta} & \frac{\Delta P}{\Delta V} \\ \frac{\Delta Q}{\Delta \theta} & \frac{\Delta Q}{\Delta V} \end{bmatrix} \tag{2.17}$$

Hence, the modification in [18] starts with the top half of the Jacobian matrix, which can be expressed as:

$$\Delta P = \left[\frac{\partial P}{\partial \theta}\right] \Delta \theta + \left[\frac{\partial P}{\partial V}\right] \Delta V \tag{2.18}$$

From this point on, only DC power flow is considered. Thus, the voltages are held constant, expressing as $\Delta V = 0$. Eq. (2.18) can be further expressed as:

$$\Delta P = \left[\frac{\partial P}{\partial \theta}\right] \Delta \theta \tag{2.19}$$

Hence, $J_{P,\theta} = \left[\frac{\partial P}{\partial \theta}\right]$ is a Laplacian matrix, which is analogous to the conductance matrix G in Eq. (2.15). All branches in the network are assumed to be symmetric, and the shunt capacitance in the transmission line is neglected in [18]. The power system at this point can be regarded as a weighted graph. Thus, the electrical distance can be constructed based on the resistance distance expression Eq. (2.16), which can be described by [69]:

$$E_{a,b} = (J_{P\theta}^{-1})_{a,a} - (J_{P\theta}^{-1})_{a,b} - (J_{P\theta}^{-1})_{b,a} + (J_{P\theta}^{-1})_{b,b} \tag{2.20}$$

Eq. (2.20) measures the change in phase angle between nodes a and b, $\Delta\theta_{a,b} = \Delta\theta_a - \Delta\theta_b$ by carrying out an average power transaction between nodes a and b. In order to independent from the reference bus chosen for the network, the author in [18] applies the pseudo-inverse theory [36] in Eq. (2.20), which can be further expressed as [19]:

$$E_{a,b} = (J_{P\theta}^{+})_{a,a} - (J_{P\theta}^{+})_{a,b} - (J_{P\theta}^{+})_{b,a} + (J_{P\theta}^{+})_{b,b} \tag{2.21}$$

In the definition of the electrical distance from [19], it also indicates that the incremental angle differences and incremental reactive power dissipation are the same since they assume that the angle differences are small and that the voltages are nominal. The author of [19] also proved that the proposed electrical distance E satisfies the conditions for a proper distance metric under the assumptions, which has considered the electrical properties, Kirchoff's laws and Ohm's laws as well.

2.2 Electrical Structure Based Partitioning Method of Power Grids

Clustering or partitioning methods are useful techniques used in complex power system planning or control applications, which has been indicated in [10, 38, 77, 79]. It is an intentional reaction mechanism that helps power system operators and planners maintain the system stable with respect to severe situations. It is also called controlled islanding, which will switch the grid into islanding operation mode to isolate the faults and avoids the cascading outage [57].

The author of [19] proposes a partitioning method by using the electrical distance between each bus as the critical measurement, which was demonstrated in the previous section. In that proposed partitioning method, the electrical distance is used to demonstrate that buses within a zone are strongly connected and buses between zones are weakly connected by using the proposed method. Besides, the electrical distance is also sensitive to the active-power related to the network topology. The proposed method in [19] can partition the power system according to the network topology rather than to specific operating points. Moreover, the proposed method reduces the extra-zonal power flows, which is the transaction leakage in other words. Based on the advantages introduced by [19], the proposed algorithm will be the basis of this thesis.

2.2.1 Algorithm of The Partitioning Method

The proposed partitioning algorithm in [19] is implemented with custom coded toolboxes merged from the genetic algorithm, which is a type of evolutionary algorithm. The reason for choosing the genetic algorithm as the building block is that the algorithm can be effective at solving non-linear or non-convex optimization problems. Its solutions are usually represented as strings of numbers, which satisfy a user-defined objective function that evaluates each generation of solutions. This kind of representation would result in high computational complexitiy for a large network. Thus, many other encoding approaches are being developed [25]. The major steps of

the proposed algorithm in [19] approximately follow the typical genetic algorithm, which are initialization step, selection step, genetic operation step, and termination decision step. The flow chart of the algorithm in [19] is reproduced as Fig. 2.2 [19, 42].

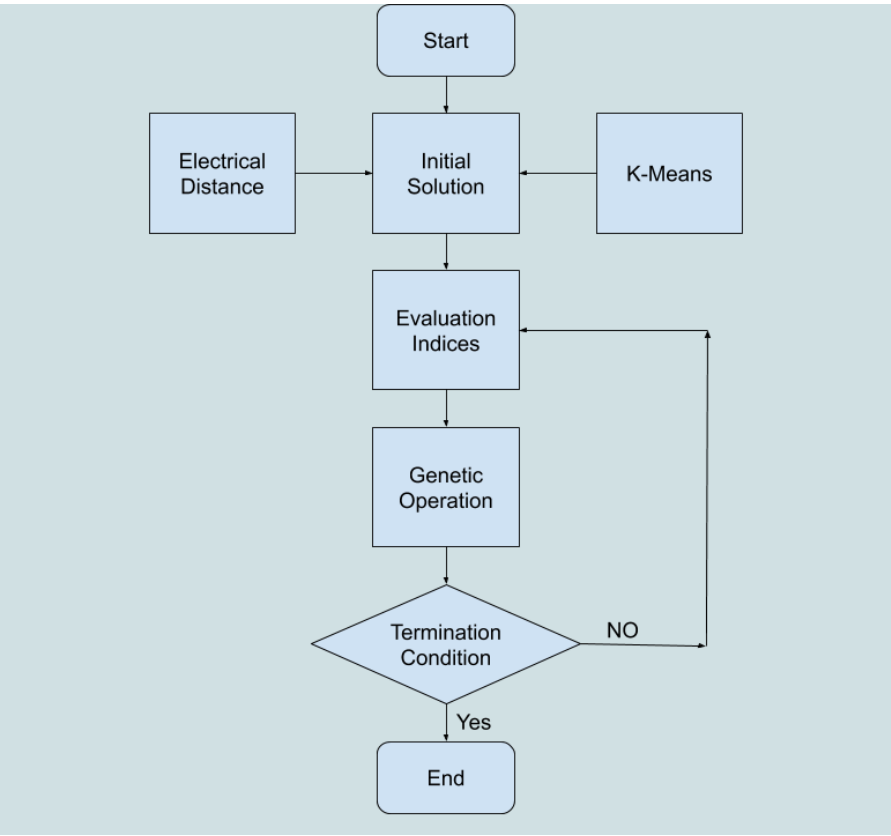


Figure 2.2: Flowchart of The Partitioning Algorithm

For the initial population part, the solution is typically randomly generated. The goal is to have large range of coverage on all the possible solutions so that the most relative optimal solution can be realized. In [19], k-means algorithm is involved as the technique for the initialization step. The purpose of involving k-means algorithm is to divide the buses in the power network into certain number of clusters. The algorithm in [19] starts from randomly selecting K nodes as the centroids within the total nodes (buses) in the power network. The rest of nodes will be assigned to the closest of the k initial centroids subsequently. The previous selected centroids are then relocated to the node in each cluster based on the principle of minimizing the mean distance

between the centroid and other nodes with the cluster. Hence, each node in the network will be evaluated according to the distance between itself and the selected k centroids. The reassignment is implemented based on the whether they are in the cluster associated with the nearest centroid. The entire process is iterated until it reaches the iteration upper bound or obtains a stable set of clusters.

2.2.1.1 Evaluation Metrics For The Partitioning Solutions

The second step of the algorithm is the selection step. The selection process is implemented through the evaluation indices defined according to a specific attribute, which is the evaluating process on each individual. Five evaluation indices are obtained in [19], and an overall fitness score is used to represent the goodness of the evaluation indices as well as the clustering solution. The evaluation indices are Electrical Cohesiveness Index (ECI), Between-Cluster Connectedness Index (BCCI), Cluster Count Index (CCI), Cluster Size Index (CSI) and Cluster Connectedness (CC). Each index ranges from 0 to 1, representing the worst and the best result with respect to the specific evaluation factor. Generally speaking, the first two indices, ECI and BCCI, are evaluated based on the electrical distance. The third and the fourth indices, CCI and CSI, are used to determine the number of clusters and their relative sizes. The specific descriptions of the indices are now given:

1. **ECI**: It is a variable used to measure the connection of the chosen nodes within a cluster. If it equals one, it means all nodes are close to each other and perfectly connected. If it equals zero, it means the nodes are inside a large cluster with distant connection.

$$ECI = 1 - \frac{\sum_{a=1}^{n} \sum_{b \in M_a} e_{ab}}{\sum_{a=1}^{n} \sum_{b=1}^{n} e_{ab}} \tag{2.22}$$

2. **BCCI**: Unlike ECI, BBCI measures the overall connection strength between nodes connection across zonal boundaries. The value of BCCI denotes that the 'weaker' it is connected to each other the closer to 1 it would be.

$$BCCI = 1 - \frac{\sum_{a=1}^{n} \sum_{b \notin M_a} \frac{1}{e_{ab}}}{\sum_{a=1}^{n} \sum_{b=1, a \neq b}^{n} \frac{1}{e_{ab}}} \tag{2.23}$$

3. **CCI**: CCI is a factor that measures the proximity of the clustering solution compared with

the ideal cluster numbers. The closer it gets to the ideal number of clustering, the close to 1 the value would be.

$$CCI = e^{\frac{-(\ln p - \ln p_+)^2}{2\sigma^2}}, \sigma = \omega \ln(n) \tag{2.24}$$

where p_+ is the user defined preference number.

4. **CSI**: CSI is a variable that corresponds to CCI. It denotes the cluster size compared with the ideal cluster size deviated from the ideal numbers of clustering.

$$CSI = e^{\frac{-(\ln(\sum_i^n \frac{s_i}{n}) - \ln \frac{n}{p_+})^2}{2\sigma^2}} \tag{2.25}$$

where s_i is the size of the cluster.

5. **CC**: CC is a binary index that evaluates whether the node(s) is/are connected the involved power network. The in-connected node(s) can be eliminated during the clustering process.

$$CC = \begin{cases} 1, \, connected \\ 0, \, in-connected \end{cases} \tag{2.26}$$

The overall fitness score function is defined in [19], which is evaluated by aggregating all clustering indices solutions with the assigned exponential factors. The fitness equation can be expressed as:

$$f = ECI^{\alpha} \cdot BCCI^{\beta} \cdot CCI^{\gamma} \cdot CSI^{\zeta} \cdot CC \tag{2.27}$$

where $\alpha, \beta, \gamma, \zeta$ are the user-defined weight coefficients with values from 0 to 1. They represent the importance of the corresponding index. 1 means the highest importance, and 0 means the lowest importance. In [19], all weight coefficients are set to 1 aiming to give equal importance during the process of simulation. By calculating the fitness score using the equation above, the higher score solution set can be chosen as the preferred solutions.

2.2.1.2 Representation of Solutions

In order to reduce the computational complexity caused by large size complex networks, the solutions generated by the genetic algorithm are usually simplified as a string of abstract numbers by certain encoding method, which is known as the genotype. In [19], the solutions are encoded

by integer representation instead of the binary representation. [19] provides the proof that the integer representation of genotype can achieve substantially better solution than that with the binary representation. Generally speaking, integer genotype would require a much smaller number of mutations to produce substantially different solutions [19].

Let the vector $\mathbf{g} = [g_1 \dots g_n]$ be used to denote the representation of the solution, where n is the number of nodes in the network. The nodes in the network are assigned to an integer number sequentially starting from 1, such as 1 to 5 for 5-bus example system shown as Fig. 2.3.

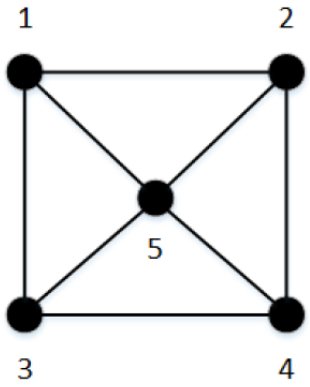


Figure 2.3: Five Bus Example System

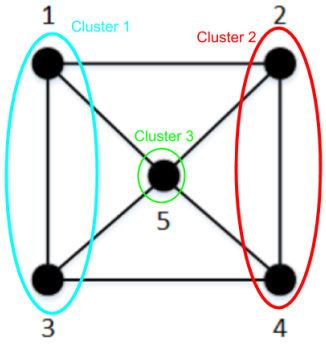


Figure 2.4: Clustered Five Bus Example System

Each element of the vector $\mathbf{g}$ is used to denote the sequential neighborhood bus that is in the same cluster as the specific bus, which has the integer starting from 0. For example, $g_i = 1$ means that bus i and its first neighborhood are located in the same cluster; $g_i = 0$ means there

are two scenario for bus i: either bus i is the only bus in the cluster, or bus i has been counted in the assignment of its previous bus in the same cluster.

Fig. 2.4 provides the specific illustration based on the system in Fig 2.3. Shown as Fig. 2.4, bus 1 and bus 3 are located in the cluster 1, bus 2 and bus 4 are located in the cluster 2, and bus 5 is in cluster 3. Starting from bus 1, the original neighborhood bus numbers are 2, 3 and 5, which will have the neighborhood index as 1, 2 and 3. Bus 3 is the first neighborhood bus of bus 1, $g_1 = 2$ as the result. $g_2 = 2$ is because bus 4 is the second neighborhood. The reason for $g_3 = 0$ and $g_4 = 0$ is that they have been included by bus 1 for bus 3 and bus 2 for bus 4. $g_5 = 0$ is because bus 5 is the only bus in the cluster. Hence, the overall genotype for the 5 bus system is $\mathbf{g} = [2,2,0,0,0]$.

From the example illustrated above, each individual genotype matrix represents an unique clustering result and its individual bus connectivity in different cluster by simplified data. Dividing or merging single bus during the clustering process will cause tremendous change on the genotype matrix. Besides, it can be seen that the integer representation approach increases the efficiency of the mutation and crossover genetic operation in the next step.

2.2.1.3 Genetic Operation

Genetic operation is to generate second generation of solution sets based on the evaluation result from the evaluation indices [25]. The operators include crossover (or recombination), mutation, regrouping, colonization-extinction, and migration in the genetic algorithms.The purpose of this part is to generate sets of solution having a higher fitness score. [19] employs crossover and mutation genetic operator to achieve better fitness score. The mutation or the recombination is implemented in each iteration based on the overall fitness score, who aims to change the clustering size or numbers without changing the network topology [25].

Crossover operator recombines two selected parents individuals, which are selected through the standard tournament selection method without replacement [25]. Tournament selection method is a selection mechanism used for crossover in the genetic algorithm. In the tournament selection, the number of selected individuals x are defined as tournament size, which will be acted as the selection pressure. The selection pressure is the degree to which the better individuals are favored. The tournament will be held among this x individual solutions by comparing their fitness score. The first x_{opt} will be the winner and selected. After the pair of parents is selected, a single random point in the pair is chosen and both parents are split. After the splitting,

the new individual will be generated by having the head vector of the first parent and the tail vector of the second parent. The process of generating new individual can be illustrated by Fig. 2.5. The new individual will replace the existing solution with certain probability by using the roulette wheel method. In the crossover operation, the top three individuals will be remained without any modification.

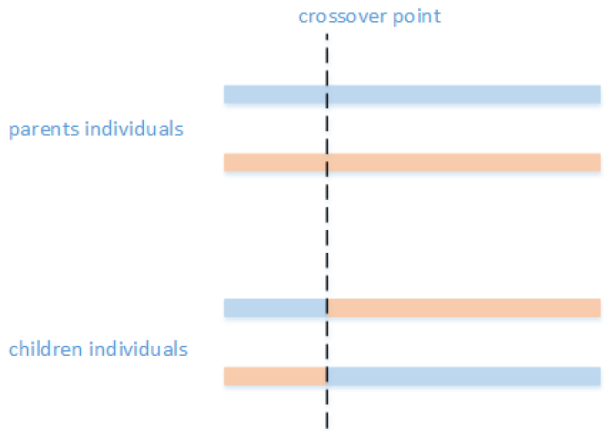


Figure 2.5: Crossover Example [65]

The mutation operation is implemented after the crossover operation at the end of each generation. It occurs on the rest of the population except the top three individuals. It is used to maintain the diversity in the genetic population with a low probability. The diversity means the cluster size and the topological structure in the case of [19]. The probability is set to $\frac{1}{n}$, where n is the number of buses. The probability is defined to maintain the mutation by implementing one time per individual per generation. The selected unit of the individual is randomly reset to an integer belonging to $\mathbf{g}$ that located inside the genotype limits. This process can be illustrated by the example shown in Fig. 2.6.

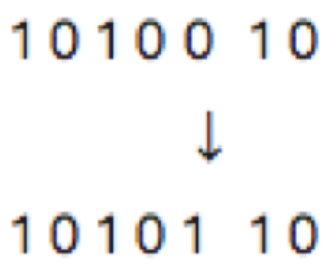


Figure 2.6: Mutation Example

According to Fig. 2.2, the termination condition needs to be decided after the genetic operation. The definition of the termination is relatively flexible and mainly depends on the user's application. Typically, the algorithm is terminated when it reaches the threshold or the criteria defined in the termination part. Common terminating conditions are:

- A solution is found that satisfies minimum criteria
- Fixed number of generations reached
- Allocated budget (computation time/money) reached
- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results
- Manual inspection

In [19], two termination conditions are employed. The first one is the convergence of the fitness score, which is evaluated according to Eq. (2.27). It means this whole heuristic process will be terminated when no better fitness score occurs. The second one is the user-defined iteration duration and times. When the upper limit time or duration is reached, the result of the heuristic process will be the final result regardless of whether or not it is the best.

2.3 Network Observability

Network observability is a very important characteristic in power system state estimation, whose analysis must be performed prior to state estimation execution. For a given set of measurements and their location in the system, the observability analysis contributes on deciding whether the

state estimation is possible or not [54]. Lacking of the measurement makes it barely possible to have accurate state estimation results, which might be caused by error in the installation of the measurement, changing on the topology structure, or error on the communication signal transmission [52]. There are two types of typical observability analysis method, which are numerical observability analysis, and topological observability analysis. Prior to the descriptions of these two kinds of observability analysis method, the general state estimation background needs to be introduced as it is the basis framework of both algorithms.

2.3.1 State Estimation Fundamentals

A state estimator aims to reconstruct the system state based on the set of given measurements. The measurement model for the power system can be expressed as [32, 52]:

$$\mathbf{z} = \mathbf{h}(x) + \mathbf{e} \tag{2.28}$$

where $\mathbf{z}$ is the measurement vector with the assumed dimension of $m-1$. $\mathbf{h}(\cdot)$ is a nonlinear vector function which maps the state variables of the system to the measurement. The state variable vector $\mathbf{x}$ is typically formed by the bus voltage angles θ_i and bus voltage magnitudes V_i in power system. Assuming there are n buses in the power system, the system consists n voltage magnitude and n voltage phase variables. Considering the reference node, one of the phase angles is assigned to be zero $\theta_1 = 0$. Thus, the state vector $\mathbf{x}$ has dimension $(2n-1)$, expressed as: $\mathbf{x}^T = [\theta_2, \ldots, \theta_n, V_1, \ldots, V_n]$. Lastly, $\mathbf{e}$ is the $m \times 1$ measurements noise vector.

Assuming a static system, the objective function in the WLS algorithm can be expressed as:

$$Obj = [\mathbf{z} - \mathbf{h}(\mathbf{x})]^T Cov^{-1} [\mathbf{z} - \mathbf{h}(\mathbf{x})] \tag{2.29}$$

where Cov is the diagonal $m \times m$ covariance matrix of measurement errors. The minimum of the last equation can be obtained iteratively using the algorithm found in [76], i.e.:

$$[H^T(\mathbf{x}_k) Cov^{-1} H(\mathbf{x}_k)] \Delta \mathbf{x}_{k+1} = H^T(\mathbf{x}_k) Cov^{-1} \Delta \mathbf{z}_k \tag{2.30}$$

where $H(\mathbf{x}_k)$ is the Jacobian matrix of $\mathbf{h}(x)$ at time k, which is given by $H(\mathbf{x}) = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}}$.

The Jacobian matrix $H(\mathbf{x})$ is very important for observability analysis since the number of rows in the matrix is the number of measurement, and the number of columns is the number

of state variables of the system; $\mathbf{x}_k$ denotes the state vector at the time stamp k; $\Delta\mathbf{z}_k$ represents the measurement difference, $\Delta\mathbf{z}_k = \mathbf{z}_k - \mathbf{h}(\mathbf{x}_k)$; $\Delta\mathbf{x}_k$ is the state vector change from k to $k+1$, expressed as $\Delta\mathbf{x}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$. $[H^T(\mathbf{x}_k)Cov^{-1}H(\mathbf{x}_k)]$ is called as gain matrix [76], namely, $G(\mathbf{x}_k) = H^T(\mathbf{x}_k)Cov^{-1}H(\mathbf{x}_k)$.

2.3.2 Numerical Observability Analysis

The study of general observability was motivated by the desire to know the entire state of the system without measuring all the system states, but rather having measurements that are functions of the system states. The early investigation on the observability property of systems focused on linear time-invariant systems. It provides the observability analysis based on the system matrix and the output distribution matrix. In general, a linear time-invariant system model can be expressed as:

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u} \\ \mathbf{z} &= C\mathbf{x} + D\mathbf{u} \end{aligned} \tag{2.31}$$

where $\dot{\mathbf{x}}$ is the time derivative of the state vector $\mathbf{x}$, A is the system dynamic matrix, B is the input distribution matrix, $\mathbf{z}$ is the measurement vector, C is the output distribution matrix, and D is the measurement transmission matrix. If there are n measurements, the linear time-variant system can be regarded as observed system if $rank(J) = n$, where $J = [C\ CA \cdots\ CA^{n-1}]^T$. The common approach to finding the rank of a matrix is to reduce the matrix to a simple form by using the elementary row operation, which is also called row echelon form. The detail observability analysis can be found in [24].

However, the power system numerical observability analysis is not the same as the classic observability analysis for linear dynamical system [32]. The inverse function theory is developed for numerical observability analysis [32]. For a no measurement error power system $\mathbf{z}_i = \mathbf{h}(\mathbf{x}_i)$, the basis of the observability analysis focuses on the determinant of the Jacobian matrix $H(\mathbf{x})$ based on Eq. (2.28) and Eq. (2.32), which can be described by [52]:

$$H(\mathbf{x}) = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m}{\partial x_1} & \cdots & \frac{\partial h_m}{\partial x_m} \end{bmatrix} \tag{2.32}$$

The system can be regarded as observable as long as the determinant of the Jacobian matrix $H(x)$ is not zero. Consequently, all the state variables can be recovered from the measurement if the Jacobian matrix $H(\mathbf{x})$ is of full rank.

The authors of [32] develop an observability analysis method regardless of the number of the measurement. Using the proposed method in [32], the unobserved branch can be found so that the situation, a sub-system full of unobserved branches, can be avoided during the partitioning. The starting point is the power flow simplification. The traditional power flow equations between bus i and bus j can be expressed by:

$$\begin{aligned} P_{ij} &= \frac{V_i[R(V_i - V_j\cos\theta_{ij}) + XV_j\sin\theta_{ij}]}{R^2+X^2} \\ Q_{ij} &= \frac{V_i[-RV_j\sin\theta_{ij} + X(V_i - V_j\cos\theta_{ij})]}{R^2+X^2} \end{aligned} \tag{2.33}$$

where X and R are the imaginary and the real part of the line impedance, respectively. [32] makes the assumption that $X \approx R$. Using orthogonal linear rotational transformation matrix, the modified power flow equations can be expressed as:

$$\begin{bmatrix} P'_{ij} \\ Q'_{ij} \end{bmatrix} = \begin{bmatrix} \sin\theta_{ij} & -\cos\theta_{ij} \\ \cos\theta_{ij} & \sin\theta_{ij} \end{bmatrix} \begin{bmatrix} P_{ij} \\ Q_{ij} \end{bmatrix} \tag{2.34}$$

where $\sin\theta = \frac{X}{Z}$ and $\cos\theta = \frac{R}{Z}$. By using the simplified DC model the relationship between measurements can be denoted as:

$$\begin{bmatrix} z_{P'} \\ z_{Q'} \end{bmatrix} = \begin{bmatrix} H_{P'\theta}\cdot\theta \\ H_{Q'V}\cdot V \end{bmatrix} \tag{2.35}$$

Since the active power and reactive power measurements may be performed in pairs in practice, the observability analysis turns to the analysis of Jacobian matrix $H_{P'\theta}$ [32]. However, the matrix $H_{P'\theta}$ is a non-square matrix, which is not invertible for inverse function theory. A gain matrix $G_{P'\theta}$ is built based on $\mathbf{z} = H\mathbf{x}$, which is obtained by multiplying both sides by the transpose of $H_{P'\theta}$, namely:

$$H^T_{P'\theta}\cdot z_{P'\theta} = H^T_{P'\theta}\cdot H_{P'\theta}\cdot\theta \tag{2.36}$$

in which $H^T_{P'\theta}\cdot H_{P'\theta} = G_{P'\theta}$. The gain matrix can be further used for observability analysis by calculating its rank: if the rank of the gain matrix is $n-1$, with n denoting the number of buses,

the distribution system is observable. After the evaluation of the gain matrix, the algorithm for finding the unobserved branch is applied. First of all, [32] starts to solve the phase value θ using the gain matrix. The Cholesky method is used to decompose the gain matrix since it is a symmetric singular matrix [32]. The way that [32] resolves it is by replacing the zero pivot by 1 and giving different arbitrary values to the corresponding entry. The solved vector θ will then be used to calculate the unobserved branch with the equation:

$$\mathbf{P}_{branch} = A \cdot \theta$$

$$a_{ij} = \begin{cases} 1, \text{ if j is the begin using bus of branch i} \\ -1, \text{ if j is the end bus of branch i} \\ 0, \text{ otherwise} \end{cases} \tag{2.37}$$

where a_{ij} is the (i, j) entry of matrix A. In the solved vector $\mathbf{P}_{branch}$ the unobserved bus will be the one having value differing from zero [32]. In [32], the whole process is iterated until no observable branches can be found.

2.3.3 Topological Observability

Generally speaking, the topological observability algorithms determines the network observability according to the location of the measurements in the entire system, which is evaluated based on the utilization of the graph theory [80]. According to the demonstration we mentioned in an earlier section, the N-bus power system can be represented by a undirected graph $G = (V, E)$, where V includes the buses involved in the analysis, and E are the corresponding branches of the system. The related sub-graph can be denoted as $G_T = (V_T, E_T)$. If the sub-graph contains all of the graph nodes (all the involved buses) and $N-1$ branches, it will have a full rank spanning tree of graph (loop free-graph) G. Under this condition, the power system could be considered as topologically observable [54].

In the conventional topology observability, there are two kind of observabilities, which are $P-\theta$ observability and $Q-V$ observability. The $P-\theta$ observability and $Q-V$ observability concepts are based on the well known decoupling principle of high-voltage power networks. They essentially imply that, if the same conditions that underlie the fast decoupled power flow apply, the observability problem can be split into two separated sub-problems [29, 16]. A N-

bus power system is said to be algebraically observable with respect to the active power flow and active power flow injection, if and only if the rank of the respective Jacobian sub-matrix $rank(H_{P\theta})$ is equal to $N-1$. On the other hand, the same system is algebraically observable with respect to active power injection and voltage magnitude measurement, if and only if the rank of the respective Jacobian sub-matrix $rank(H_{QV})$ is equal to N. According to the fact that the bus voltage magnitudes can be directly measured whereas the bus voltage angles cannot, the problem of $P-\theta$ observability is more critical than the $Q-V$ observability.

The sub-graph created by the measurement set would be the next investigation item. There are three remarks that could help on the implementation of the branch assignment, which can be concluded as [29, 16]:

- A branch flow measurement can produce an equation in the measurement model, shown as Eq. (2.28), which would incorporate the voltage angles with the magnitudes corresponding to the terminal buses of the monitored branch.
- The effect of an injection measurement is to relate the complex voltage of the measured bus with the complex voltages of the buses that are directly connected to it.
- A bus voltage measurement only carries information about the voltage magnitude.

Based on these three remarks, the following expresses the conditions for the measurement that can be assigned to the corresponding branch:

- The branch is assigned to its flow measurement when the branch flow is measured.
- The branch is assigned to the injection if an injection is measured at a terminal node of a branch.
- The branch is assigned to only one kind of measurements.

The assigned branches and nodes are regarded as edges and vertices of a measurement sub-graph. The spanning tree method is the typical method for topology observability problems. In general, a spanning tree is a sub-graph of a no-cycle graph G containing all vertices with minimum edges. The rank of a spanning tree measures the minimum number of edges that must be removed from the graph to make the remaining edges form a tree. It can be calculated by using the equation:

$$rank = E - V + L \tag{2.38}$$

where E stands for the number of edges of the graph, V stands for the vertices, and L stands for the connected components number. The general construction of the spanning tree is an ongoing research topic in many areas, but is not applied in this research work.

[1] introduces a simple topology observability method to have a general understanding of the spanning tree method, which is coined as branch coloring method. Fig. 2.7 shows branch coloring spanning tree example. 7 measurements are assigned to this 6-bus system. Measurement 1 is assigned red, 2 is assigned blue, 3 is assigned green, 4 is assigned yellow, 5 is assigned black, 6 is assigned brown, and 7 is assigned grey. The upper part of the figure shows the locations of each measurement which are shown by a number surrounded by a square. As the focus on the branch measurement, each measurement on the branch is assigned a color regardless of the node measurement. The lower part shows the spanning tree of the measurement of the system. The spanning tree shown in Fig. 2.7 is called observable spanning tree. A power network is said to be topologically observable with respect to the measurement set, if and only if at least one observable spanning tree can be found in the corresponding network graph.

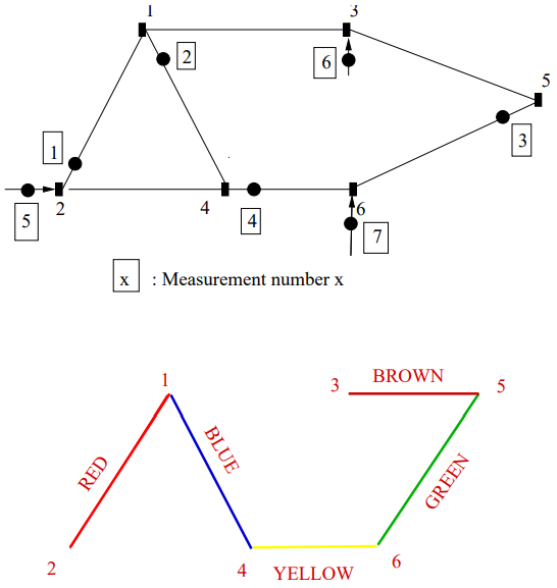


Figure 2.7: Coloring Measurement Spanning Tree Example[80]

[66] introduces a basic numerical formula for topological observability analysis. According to [66], evaluating a spanning tree of full rank implies finding (n-1) branches with meters that

connect n nodes without any loops. In [66], it uses a weighted graph $G_{P\theta}$ to represent a network with placed meters. The branch weights of $G_{P\theta}$ are determined by whether meters are located at the branches, which can be evaluated by the expression shown as:

$$w_{b_i} = \begin{cases} 1, \textit{branch has no meter} \\ 0, \textit{branch has meter} \end{cases} \tag{2.39}$$

where b_i mean the i^{th} branch of the power network. With the definition of the weight equation, the problem of topological observability could be translated into evaluating the minimum spanning tree, which means the spanning tree who has minimum sum of the branch weights among the existing spanning trees on the graph. The formulation of the problem is

$$\begin{gathered} \min \sum_{i=1}^{N-1} w_{b_i} \\ \text{subjective to } b_i \in \textit{Spanning tree } T_{P\theta} \textit{ of the graph} \\ w(T_{P\theta}) = \min \sum_{i=1}^{N-1} w_{b_i} \end{gathered} \tag{2.40}$$

With the definition of Eq. (2.34), $w_{P\theta} = 0$ means it is topologically observable. On the other hand, if it is not topologically observable, the value will be $w_{P\theta} > 0$. Specifically, if $w_{P\theta} = c, c > 0$, c will be the value standing for the number of lacking measurements to be topologically observable.

The augmented node method is developed to have the topological observability analysis [66]. The method is built based on the fact that voltage measurements are transformed into the equivalent reactive power flow. Generally, this method analyzes the topological observability by adding one node to the power network regarding as the equivalent power flow measurement from the ground. Five steps are included in this method to do the observability analysis, which are [32, 29, 66]:

- **Step 1**: The voltage and injection measurements need to be transformed into the equivalent flow measurement.
- **Step 2**: Eq. (2.39) is used to evaluate the weights of the tree.
- **Step 3**:If $\sum_{i=1}^{N-1} w_{b_i} = 0$ (or $\sum_{i=1}^{N-1} w_{b_i} = 1$ for $P-\theta$ observability analysis), the iteration

can be terminated. Otherwise, the isolated nodes and the neighboring nodes would be obtained.

- **Step 4**: The sum of weight of the graph will be updated if other nodal measurement can be assigned to the isolated nodes. Otherwise, The analysis is terminated.

2.4 PMU Placement Problem

The observability analysis normally includes three parts, namely, determining network observability, identifying the observable islands, and measurement placing for observability restoration. The first two parts can be resolved by algorithms, which are developed in many existing papers [52, 61, 16, 80]. For the third part, conventional measurements are typically employed, such as zero injection, active power or reactive power. With the development of Phasor Measurement Unit (PMU), PMU-based observability algorithms are getting the attention of many researchers, which typically involve PMU placement algorithms. PMUs are devices with synchronization signals from GPS satellites and provide positive sequence phasor voltages and currents measured at a given bus [17]. It means that a PMU installed at a bus can observe the bus where it is installed and the current phasors of the branches connecting to that bus. The assumption for realizing that is the PMU should have a sufficient number of channels. However, it is not practical to have PMUs installed on every bus in a system. It is not only because of the high cost of PMUs but also because of the high cost of installation and communication facilities cost [44, 72]. Hence, the PMU placement problems need to be addressed appropriately.

2.4.1 Theories For PMU Placement Algorithms

The objective of the PMU placement problem is a strategic choice of the minimum number of PMUs and the optimal location the PMUs installed in a system to ensure the complete observability and satisfy a given redundancy criterion [70]. Since topological observability is a relatively practical and employable method to implement for power system observability, the PMUs placement problems are typically solved based on the topological observability [70, 94]. Once the topological observability is obtained, the PMU placement problems could be regarded as the problem of finding an observable spanning tree with minimum number of branches based

on Eq. (2.40). The placement problem can be stated as:

$$\begin{aligned} &\min \sum_i \omega_i x_i \\ &\text{subject to } \; O_b(S(\sum x_i)) = 1 \end{aligned} \tag{2.41}$$

where ω stands for the cost of PMU installation, x_i is a binary variable representing the PMU installation on each bus, O_b denotes the observability evaluation logical function, and $S(\cdot)$ is the location set of the PMUs.

The solution of Eq. (2.41) always involves in a highly non-linear, non-convex and non-differentiable computational task. There are few conditions that are commonly employed for determining the observable areas during the process of selecting the placement of PMUs, which can be summarized [70, 74]:

- For a PMU installed bus, voltage and current phasors of all lines incident to the bus are known.
- Voltage phasors at the buses connecting to the PMU bus can be calculated by Ohm's law if the voltage and current phasors at a bus are known.
- If the buses at two ends of a branch are known, the current phasor at the branch can be directly obtained.
- The current phasor of the a unknown branch can be evaluated by KCL if there is a zero-injection bus without PMU and the current phasors of the connected branches are all known but one.
- If the voltage phasor of a zero-injection bus is unknown and the voltage phasors of all connected buses are given, the voltage phasor of the zero-injection can be obtained.
- For a group of connected zero-injection buses, if the voltage phasors of all buses connected to this group of buses are known, the voltage phasors of this group of buses can be calculated by KVL

For the above conditions, the measurements in the first one are called direct measurements. The measurements in other conditions are called pseudo-measurements or extension-measurements. In practical scenarios, these conditions can be summarized as that if there is a PMU is installed

at a bus, the bus and its adjacent buses are observable [70].

There have two typical types of PMU placement algorithms which are mathematical programming methods and heuristic methods. Conventionally, the mathematical programming methods include linear programming (LP), nonlinear programming (NLP), dynamic programming, or combinatorial optimization. However, the pure mathematical techniques are easily trapped at local optima, and they might have difficulties in handling numerical constraints [44, 74]. Heuristic techniques based on the graph theory are developed to overcome the numerical difficulties.

The graph theoretic heuristic techniques are based on the system's topology, whose procedure builds a measurement spanning sub-graph across the system. Similar to Eq. (2.41), the graph theoretic search can be mathematically expressed as:

$$\begin{aligned} &\min \sum_i \omega_i x_i \\ &Subject\ to\ \ \mathbf{F}(\mathbf{x}) \geq \mathbf{C} \end{aligned} \tag{2.42}$$

The objective here is to minimize the cost of the PMU installation. **F(x)** is a vector function that represents the observability constraint form by the x_i. The constraint function not only represents the observability of the system after PMU installation but also builds the constraint for the system observability redundancy, which is represented by the constant vector **C**. Note that the constraint is a component-wise inequality. It is first introduced by the author of [5], which can be expressed as:

$$\mathbf{F(x)} = A\mathbf{x} \tag{2.43}$$

where A is the binary adjacency matrix. The elements of the matrix A can be defined as [74, 5]:

$$a_{ij} = \begin{cases} 1, \text{ nodes i,j are connected} \\ 0, \text{ otherwise} \end{cases} \tag{2.44}$$

where a_{ij} means the element located at the i^{th} row and j^{th} column in the matrix A. For bus i in a n-bus system, the corresponding observability can be represented as:

$$\mathbf{f}_i(\mathbf{x}) = \sum_{j=1}^{n} a_{ij} x_j \tag{2.45}$$

If the value of the observability $\mathbf{f}_i(\mathbf{x})$ is a non-zero value, the bus i can be regarded as a observable bus. The value of $\mathbf{f}_i(\mathbf{x})$ not only reflects the observability of the involved bus, but also reflects the redundancy observability value. It must be a value greater or equal to 1. It is a constraint based on the specific request of the owner of the corresponding bus. A wide range of methods can be cited based on the existing papers such as depth first search (DFS), the minimum spanning tree (MST), simulated annealing (SA), and recursive security N algorithm (RSN) [5].

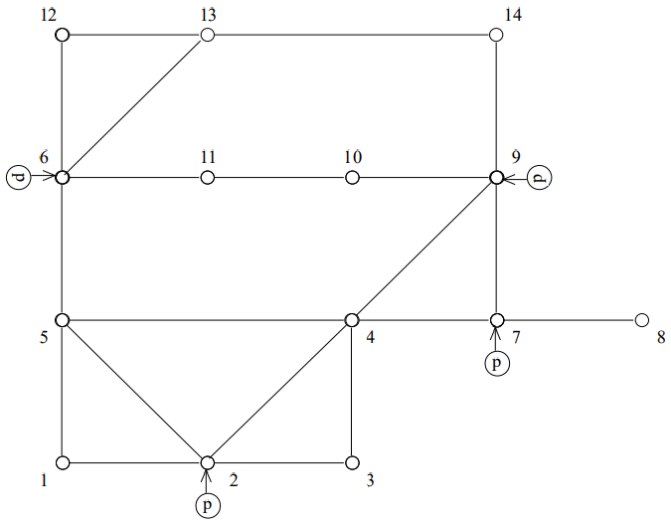


Figure 2.8: 14 Bus System with 3 PMUs [94]

Fig. 2.8 provides an example that provides a straightforward illustration of the PMU placement problems. In the 14 bus example system, there have 4 PMUs placed at buses 2, 6, 7 and 9. The PMU at bus 2 can measure the observability of bus 2 and also the branches 1-2, 2-3, 2-4 and 2-5. The buses 1, 3, 4 and 5 can be observed by using Ohm's law because of the placement of the PMU on bus 2. Hence, the current phasors of branches 1-5, 3-4 and 4-5 can be calculated since the voltage phasors at bus 1, 2, 3, 4 and 5 are determined. Similarly, PMU at bus 6 can observe the branch 5-6, 6-11, 6-12 and 6-13, which can indirectly measure the phasors at buses 5, 11, 12, 13 and the branch 12-13. Using the known current phasors of branches 4-7 and 7-9 and the PMU measurement at bus 7 can make the current phasor of branch 7-8 being measured by Kirchhoff's law. The current phasors of branches 4-9, 7-9, 9-10 and 9-14 can be measured by

the PMU at bus 9. Besides, the buses 4, 7, 10, 14 and the current phasors of branches 4-7 can be measured as well. The voltage phasors of buses 10, 11, 13 and 14 are known, which can help the current phasors of branches 10-11 and 13-14 becoming known measurements. Summarizing the above illustration, it can be seen that it is possible to use the PMU only measurements to ensure complete system observability by placing PMUs at buses 2, 6, 7 and 9.

2.4.2 Depth First Search with Heuristic Algorithm

The DFS algorithm employs the first and the third condition that we previously mentioned, which are:

- For a PMU installed bus, voltage and current phases of all lines connecting to the bus are known.
- If the buses at two ends of a branch are known, the current phasor at the branch can be directly obtained.

It is a method built based on the tree search technique. As a general idea, DFS searches a tree from a parent node to one of its direct child nodes, and does the searching repeatedly until no child node exist. The same process is implemented on all parents nodes. Based on the illustration in paper [70, 28], Fig. 2.9 shows the order of the nodes being visited when DFS is employed in a tree.

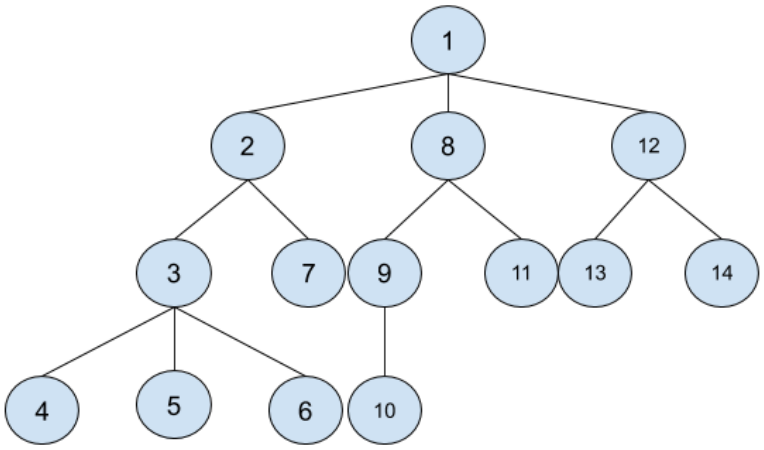


Figure 2.9: DFS Nodes Order Example

Applying the idea of DFS to the PMU placement problems, any node in the tree of DFS represents a state of a given power network. The state means the buses where PMUs have been installed. It will be represented by the variable value x_i of the vector $\mathbf{x}$ in Eq. (2.43). The child of either node represents the candidate locations of PMUs. The DFS method could traverse all states of a represented power network, which has a finite amount of buses and branches, if no special termination constraint is constructed during the searching process. In other words, DFS could find out all possible PMU placement scenarios, theoretically speaking. Hence, DFS has the potential to find the optimal solutions of a PMU placement problem. However, it is a time consuming method if the process obtains all possibilities of a power network according to the combinations calculation. Based on the illustration of the DFS in [70], the searching process can be improved by combining the main idea of minimum spanning tree, which could be named as DFS heuristic algorithm. Generally speaking, this improved algorithm improves the DFS weak convergence by changing the optimization rules from finding a slip road linking the bus to searching for the maximum coverage of the bus. Specifically, the amount of states can reach $\sum_{n=1}^{118} C_{118}^{n}$ for a 118 buses system, which is 3.3231×10^{35}. Thus, a proper constraint needs to be set up during the PMU placement searching process to have the optimal or sub-optimal solution with limited search times. The constraint is obtained during the process of choosing which a child node should be visited next time [74]. Specifically, it mentions that only the child node

that has the most connected branches will be chosen as the next to be visited node. By applying this condition, it decreases the number of states being traversed by only traversing the ones that have the largest possibility to increase the number of buses to become observable buses.

We can conclude that the DFS with heuristic algorithm can contribute to the PMU placement algorithm with the possible minimum searching time. During the placement set searching process, every bus will be assigned an unique identity for recording purpose. The bus where the PMU will be placed will be recorded by using its corresponding identity. The solution set will contain all the buses identities where the PMUs are placed. Hence, the system observability will be evaluated by employing Eq. (2.43). The logic flow of the DFS heuristic algorithm can be summarized as Fig. 2.10.

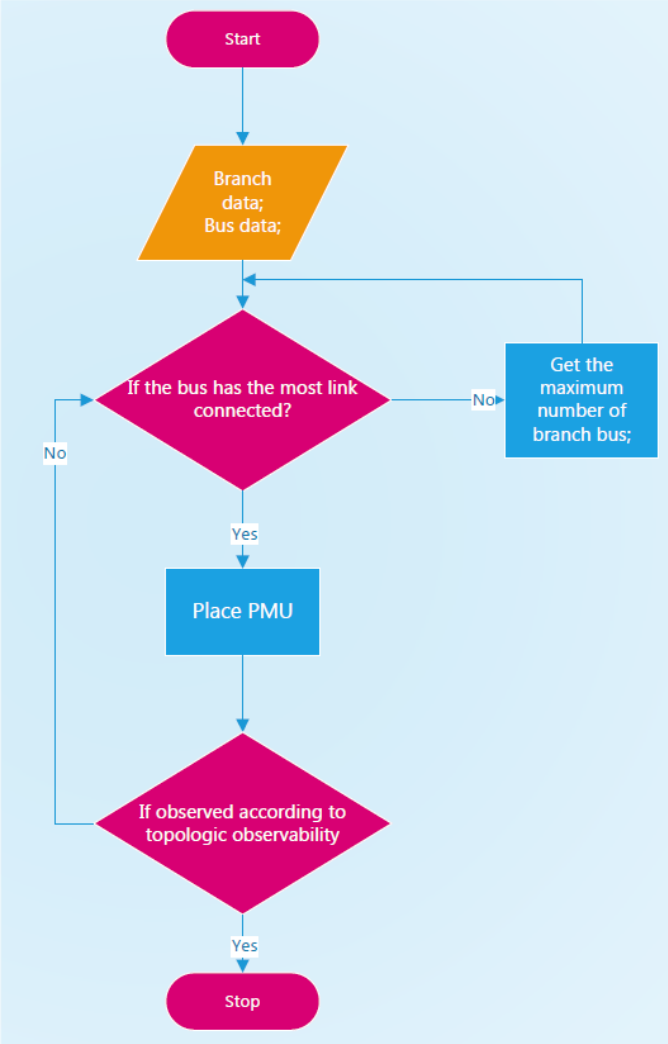


Figure 2.10: DFS Heuristic Algorithm Method Flow Chart

2.4.3 Genetic Algorithm Based Technique For PMU Placement

According to the genetic algorithm (GA) concept, the fitness function helps the algorithm to specify certain selection rules for multi-variable population for an optimized result [65]. Unlike the classical optimization problem, genetic algorithm, introduced in the previous section is a multi-point searching solution for the global optimum value instead of single point conventional

searching method [34]. According to the techniques introduced in [7] and [87], a PMU placement technique is modified based on the BILP and GA algorithm. The modified PMU placement technique has three main steps for the algorithm introduced in this thesis, which are:

- Step 1: Implementing the BILP-based PMU placement algorithm, the total number of installed PMUs from BILP will be set as the minimum number of total employed PMU. In other words, a linear constraint will be set.
- Step 2: Using the genetic algorithm to generate sets of solutions, which will be encoded by a binary string. The solutions stand for the PMU placement.
- Step 3: Comparing the system observability values of the solutions from Step 1. The solution that has the largest system observability will be derived.

The PMU placement algorithm in the first step that aims to set up the maximum number of PMUs allowed for the genetic algorithm. The initial PMU placement algorithm in this step is a user defined algorithm, which is to ensure that the minimum number of PMUs can be obtained and system observability will be ensured at the same time. In our research, a Toolbox developed by IBM will be implemented, whose details can be found in the Chap. 4. The linear equality constraint can be expressed as: $A_{eq} \cdot x_i = b_{eq}$, where A_{eq} will be 1 for each bus regarding as the cost of the installation of PMU, and b_{eq} will be the minimum number PMU from user defined method.

In Step 2, the number of variables in the genetic algorithm will be the same as the total number of buses, which means the PMU installation of each bus will be the gene of the solution. According to the concept of the genetic algorithm, the aim of the GA is to find the largest fitness score solution. With this conventional concept, the fitness function expressed as: $fit_{fcn} = nBus - \sum_i x_i$, where $nBus$ stands for the total number of system buses and $\sum x_i$ stands for the total number of PMUs. Since the total number of system buses is a constant, the fitness function score will increase when the total number of PMUs $\sum_i x_i$ decrease. Thus, it will have the same functionality as $\min \sum_i x_i$.

As we mentioned previously, the GA will search for the optimum value via multiple points, which means it can generate different sets of PMU placements. In order to find the relatively optimum placement solution, an extra constraint is defined in Step 3, namely, system observability value. The solution that has the largest observability value will be selected after multiple solutions are generated.

2.5 State Estimation of Power Systems

2.5.1 Conventional Weighted Least Squares

State estimation (SE) is the main function of the Energy Management Systems (EMS). An accurate state estimation process can provide reliable monitoring and control of the system toward the control centers [97]. The main objective of the SE is to estimate the current power system states by processing the measurements available from various substations in the power system network. In other words, the SE uses limited measurement data to estimate all state variables as accurate as possible. Conventionally, the state estimation is realized by using the Weighted Least Square (WLS) algorithm. The basic state estimation model was described in the previous section. For a system having m number of measurements and N_{bus} number of buses ($m > N_{bus}$), the state estimation basic model can be further expressed as [49]:

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} = \begin{bmatrix} h_1(x_1, x_2, \ldots, x_n) \\ h_2(x_1, x_2, \ldots, x_n) \\ \vdots \\ h_m(x_1, x_2, \ldots, x_n) \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_m \end{bmatrix} \tag{2.46}$$

where $[h_1(\cdot), h_2(\cdot), \ldots, h_m(\cdot)]$ is the measurement vector with m nonlinear transformation functions, n is the number of states ($n = 2N_{bus}$). The nonlinear transformation relates the state variable $x_i, i = 1, \ldots, n$ to its corresponding measurement. It usually employs power flow functions as shown in Eqn (2.33), where voltage magnitude and its angle V and θ are the state variables, real power and the reactive power P and Q are the measurements. $[e_1, e_2, \ldots, e_m]$ is the noise vector.

The goal of the WLS estimator is to find the state vector that minimizes the objective function, which is expressed as:

$$J(\mathbf{x}) = \sum_{i=1}^{m} \frac{(\mathbf{z}_i - \mathbf{h}_i(\mathbf{x}_i))^2}{R_i} = [\mathbf{z} - \mathbf{h}(\mathbf{x})]^{\mathbf{T}} \mathbf{R}^{-\mathbf{1}} [\mathbf{z} - \mathbf{h}(\mathbf{x})] \tag{2.47}$$

where R is the covariance matrix of the measurement noise vector $\mathbf{e}$, $R = Cov(\mathbf{e}) = E[\mathbf{e}\mathbf{e}^T]$. In order to find the minimum value, the derivative is set to zero, i.e.[64]:

$$g(\mathbf{x}) = \frac{\partial J(\mathbf{x})}{\partial \mathbf{x}} = -H^T(\mathbf{x})R^{-1}[\mathbf{z} - \mathbf{h(x)}] = 0 \tag{2.48}$$

where $H(\mathbf{x})$ is called measurement Jacobian matrix which is denoted as $H(\mathbf{x}) = \frac{\partial \mathbf{h(x)}}{\partial \mathbf{x}}$. To solve the minimization problem, the first order Taylor expansion is employed, namely,

$$\begin{aligned} \Delta \mathbf{x}_i^k &= \mathbf{x}_i^{k+1} - \mathbf{x}_i^k \\ &= G^{-1}(\mathbf{x}_i^k)H_i^T(\mathbf{x}_k)R_i^{-1}(\mathbf{z}_i - \mathbf{h}_i(\mathbf{x}_i^k)) \end{aligned} \tag{2.49}$$

where $G(\mathbf{x}_k)$ is called gain matrix derived from the derivative of $\mathbf{g(x)}$, expressed as [64]:

$$G(\mathbf{x}_i^k) = H_i^T(\mathbf{x}_i^k)R_i^{-1}H(\mathbf{x}_i^k) \tag{2.50}$$

The WLS estimator begins from a flat start, i.e. with a voltage magnitude 1 p.u. and a voltage phase angle 0 degrees. It is iterated until its tolerance reaches the desired value, or the delta value in Eqn. (2.49) converges [64]. The main steps of the WLS estimator are [64]:

1. Flat start as the initial value.
2. Calculate the gain matrix by using Eqn. (2.50).
3. Calculate the delta $\mathbf{x}$ value from Eqn. (2.49).
4. Iteratively run the last three steps until $\Delta\mathbf{x}$ converges to the desired value.

Although WLS estimator performs well on small systems, it is not widely used in current research as its computational complexity increases tremendously for large power networks.

2.5.2 Cubature Kalman Filter Based State Estimation

In order to solve the computational disadvantage of WLS, the Kalman Filter (KF) is introduced to improve efficiency [50, 55, 85, 81]. The KF is an efficient recursive algorithm for state estimation in nonlinear systems. The nature of this mathematical tool is based on minimizing the covariance of the squared error between physical and estimated states [55]. Current KF estimators include

Extended Kalman Filter (EKF), Unscented Kalman Filter (UKF), and Cubature Kalman Filter (CKF) [50, 55, 81]. EKF is a traditional nonlinear filtering method, which utilizes the first order approximation of the Taylor series to represent nonlinear functions. However, the EKF neglects high-order terms of the nonlinear system model leading to the degradation of estimation accuracy. The drawback of EKF is caused by the linearization and Jacobian matrix calculation. In order to overcome this drawback, UKF is introduced to by-pass the high order linearization process based on the unscented transform (UT) [50, 45]. The main idea of the UKF is to utilize the deterministic sigma points to capture the mean and covariance of the original distribution of state variables [50]. Compared to the EKF, the UKF has relative simplicity and good convergence with high accuracy [31]. However, UKF requires precise prior system knowledge, especially noise statistics. If the noise statistics are inaccurately initialized, it would lead to biased or divergent estimation solutions [50, 31].

Recently, cubature Kalman filter (CKF) has become popular in nonlinear filtering. Similar to UKF, CKF it does not require the linearization of the nonlinear function or the Jacobian matrix, which overcomes the problem of the computational burden and inaccuracy caused by the linear truncation. It means that the performance of CKF will not be degraded with the increase of the system dimension. Different from the UKF, the CKF has a stronger nonlinearity due to the use of the cubature points, which is the approximation points evaluated via nonlinear functions for the Gaussian integration [31, 3]. It could potentially increase the accuracy of the estimation result if the system model satisfies the Gaussian model assumption. The CKF intuition can be started from the nonlinear system model. The typical nonlinear discrete-time system model can be expressed as:

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \mathbf{w}_k \tag{2.51}$$

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k \tag{2.52}$$

where $\mathbf{x}_k$ is the state variable at time k, $\mathbf{z}_k$ is the measurement vector at time k, $\mathbf{w}_k$ is the system process noise, $\mathbf{v}_k$ is the measurement noise, $\mathbf{f}(\cdot)$ is the nonlinear system function, and $\mathbf{h}(\cdot)$ is the nonlinear measurement function. As mentioned before, one of the prior conditions for CKF is the assumption of Gaussian model including the system noise. Thus, the two noise terms are assumed to be uncorrelated Gaussian white noise with constant mean and variance. Their means

and variances are:

$$\begin{aligned} \mu_{\mathbf{w}_k} &= 0 \\ \mu_{\mathbf{v}_k} &= v \\ \sigma^2_{\mathbf{w}_k} &= Q \\ \sigma^2_{\mathbf{v}_k} &= R \end{aligned} \tag{2.53}$$

Based on the nonlinear system model previously described, there are three main steps for the CKF estimator. The three steps are initialization, prediction, and correction and update steps. For an n-bus power system, the CKF state estimation process can be denoted as [31, 3]:

1. Initialization:

   - Initialize state vector $\hat{\mathbf{x}}_0$ (flat start will be the typical initial value);
   - Initialize the error covariance of the state vector $P_{0|0}$;
   - Evaluate the cubature points ξ_i [37]:

   $$\xi_i = \sqrt{\frac{m}{2}}[\mathbf{1}]_i,\ i = 1, \ldots, 2n \tag{2.54}$$

   where $m = 2n$, n is the dimension of the state vector $\mathbf{x}$, $[\mathbf{1}]_i$ is the ith column of the matrix $[I_n| - I_n]$ and I_n is the $n \times n$ identity matrix.

   - Assign the weighting factor value ω_i:

   $$\omega_i = \frac{1}{m},\ i = 1, \ldots, 2n \tag{2.55}$$

2. Prediction:

   - Obtain the posterior error covariance $P_{k|k}$ using Cholesky decomposition:

   $$P_{k|k} = S_{k|k} S^T_{k|k} \tag{2.56}$$

   where $S_{k|k}$ is a diagonal matrix at time epoch k.

   - Calculate the cubature points χ_i:

   $$\chi_{i,k|k} = S_{k|k}\xi_i + \hat{\mathbf{x}}_{k|k},\ i = 1, \ldots, 2n \tag{2.57}$$

where $\chi_{i,k|k}$ stands for the cubature point for i^{th} state at time epoch k, $\hat{\mathbf{x}}_{k|k}$ is the state vector at time epoch k.

- Update the cubature points from time epoch k to time epoch $k+1$:

$$\chi_{i,k+1|k} = \mathbf{f}(\chi_{i,k|k}),\ i = 1,\ldots,2n \tag{2.58}$$

where $\mathbf{f}(\cdot)$ is the system nonlinear transform vector function.

- Update the mean and covariance based on the updated cubature points:

$$\hat{\mathbf{x}}_{k+1|k} = \sum_{i=1}^{2n} \omega_i \chi_{i,k+1|k} \tag{2.59}$$

$$P_{k+1|k} = \sum_{i=1}^{2n} \omega_i \left[\chi_{i,k+1|k} - \hat{\mathbf{x}}_{k+1|k}\right]\left[\chi_{i,k+1|k} - \hat{\mathbf{x}}_{k+1|k}\right]^T + Q_k, \tag{2.60}$$

3. Correction and Update:

- Re-evaluate the error covariance $P_{k|k}$ by performing Cholesky decomposition:

$$P_{k+1|k} = S_{k+1|k} S_{k+1|k}^T \tag{2.61}$$

- Re-evaluate the cubature points at $k+1$ time epoch:

$$\chi_{i,k+1|k} = S_{k+1|k}\xi_i + \hat{\mathbf{x}}_{k+1|k},\ i = 1,\ldots,2n \tag{2.62}$$

- Based on the updated cubature point, the estimated measurement of the cubature point $\mathbf{z}_{i,k+1|k}$ can be shown as:

$$\mathbf{z}_{i,k+1|k} = \mathbf{h}(\chi_{i,k+1|k}) + \mathbf{v},\ i = 1,\ldots,2n \tag{2.63}$$

- The predicted measurement of the update estimated cubature points measurement can be expressed as:

$$\hat{\mathbf{z}}_{k+1} = \sum_{i=1}^{2n} \omega_i \mathbf{z}_{i,k+1|k} \tag{2.64}$$

- Update the auto-covariance of measurement $\mathbf{z}_{k+1}$ and cross-covariance of measure-

ment and state vector:

$$P_{\mathbf{z}_{k+1}\mathbf{z}_{k+1}} = \sum_{i=1}^{2n} \omega_i \left[\mathbf{z}_{i,k+1|k} - \hat{\mathbf{z}}_{k+1}\right] \left[\mathbf{z}_{i,k+1|k} - \hat{\mathbf{z}}_{k+1}\right]^T + R_k \tag{2.65}$$

$$P_{\mathbf{x}_{k+1}\mathbf{z}_{k+1}} = \sum_{i=1}^{2n} \omega_i \left[\chi_{i,k+1|k} - \hat{\mathbf{x}}_{k+1|k}\right] \left[\mathbf{z}_{i,k+1|k} - \hat{\mathbf{z}}_{k+1}\right]^T \tag{2.66}$$

- Calculate the Kalman gain based on the updated auto-covariance and cross-covariance:

$$K_{k+1} = P_{\mathbf{x}_{k+1}\mathbf{z}_{k+1}} P^{-1}_{\mathbf{z}_{k+1}\mathbf{z}_{k+1}} \tag{2.67}$$

- Update the final estimated state vector:

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + K_{k+1}(\mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1}) \tag{2.68}$$

- Update the error covariance for the next iteration:

$$P_{k+1|k+1} = P_{k+1|k} - K_{k+1} P_{\mathbf{z}_{k+1}\mathbf{z}_{k+1}} K^T_{k+1} \tag{2.69}$$

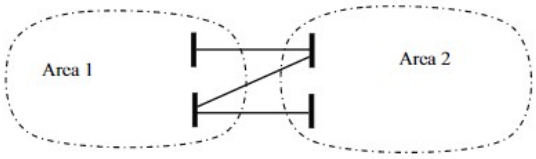
Clearly, the complete statistical information is evaluated according to the posterior probability density of the state [31]. The posterior probability density of the state will be updated using the time update and the measurement update steps on the new measurement. Based on the updated probability density of state and measurement, the estimated value of state vector evaluated recursively using predictive (time update) and corrective (measurement update) steps [3].

2.5.3 Multi-area State Estimation

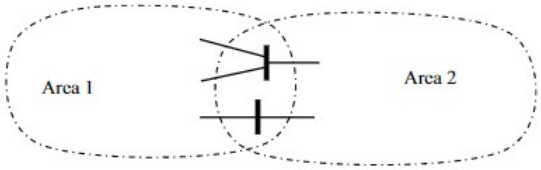
To the best of our knowledge, multi-area state estimation (MASE) research started in the early 1970s right after the state estimator was developed [35]. Recently, the MASE has been brought back to attention. MASE techniques can satisfy the need for a better real-time visibility of the operating state of the grid. What is more, the system reliability can be improved because of the increased computational efficiency. Except for the computational efficiency and system reliability, there are three additional desired MASE features, which are robustness, accuracy, and limited amount of exchanged data [35]. In terms of the robustness feature, it means the capabil-

ity of a sub-system to achieve estimation convergence. The convergence could be impacted by a wide range of possibilities, such as the decomposition/partitioning result of the global system, the amount of bad data, and the measurement configuration [35]. As for the accuracy feature, the sub-system should at least have the same level of accuracy according to the application of the system. The last feature is the amount of data to be exchanged. The sub-system is desired to have less amount of data to be exchanged between the individual sub-systems. The reason for having this feature is to avoid the possible time delays and the possible demand of large communication bandwidth. Besides, the amount of sharing data would change the complexity and security of the data acquisition process [35, 67].

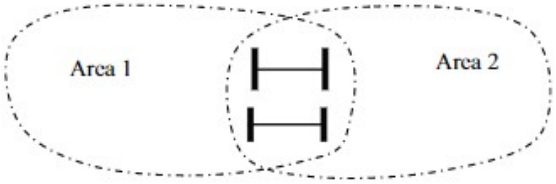
There are four levels based on the overlapping area, which are non-overlapping areas level, boundary-bus overlapping areas level, tie-line overlapping areas level, extended overlapping areas level [35, 67, 89, 27]. These four different levels can be explained with the following example:



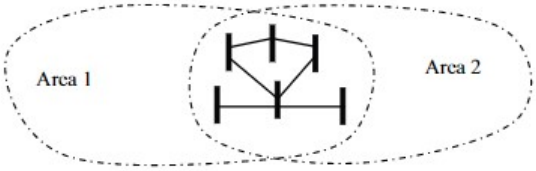
(a) Non-overlapping areas level



(b) Boundary-bus overlapping areas level



(c) Tie-line overlapping areas level



(d) Extended overlapping areas level

Figure 2.11: Examples of Area Overlapping Levels

There are two typical MASE computing architectures, which are two-level hierarchical architecture and decentralized architecture. For either architecture, the main idea is to find a good compromise that would keep their existing state estimators and to have a central entity to coordinate their results to determine the system-wide state [67].

The two-level state estimator was first introduced by [89], which contains the local state estimator level and central coordinate state estimator level. At the first level, the local state estimator level, the state estimation will be employed independently on each individual sub-system according to the partitioning strategy. At this stage, the overlapping level need to be determined

first, which is to determine the boundary bus state variables. After the first level, the central coordinator will receive the estimated state variables of each area to evaluate the global system estimated result [89]. Conventional MASE will re-estimate two kinds of information collected from the first level, which are boundary buses state variables and sub-system slack bus voltage angle. The overall procedure can be generally shown as the following figure:

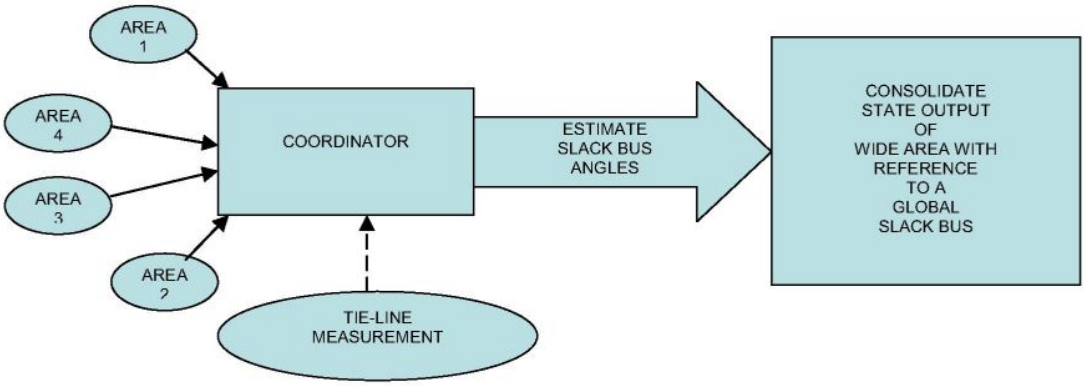


Figure 2.12: Two-Level State Estimation Procedure[59]

Generally speaking, the decentralized MASE eliminates the central coordination step from two-level hierarchical architecture. In other words, each local processor communicates only with those processors in charge of neighboring areas, exchanging border information without central computer. The general idea of this architecture is shown in the next figure:

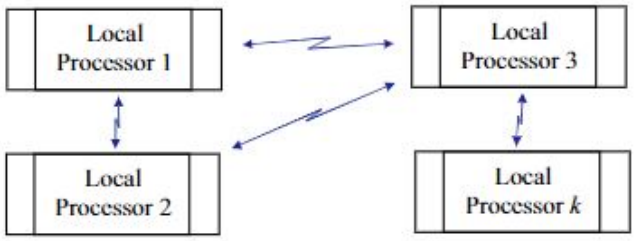


Figure 2.13: Decentralized MASE Architecture[35]

The first decentralized technique was introduced by Schweppe whose method is called spatial sweep. Its main idea is to sequentially implementing the local SE in adjacent areas with

several iterations [27]. [48] introduces a method that can solve the decentralized MASE either calculating the local SE sequentially or the local SE in parallel. It formulates the decentralized MASE by turning the SE into an optimization problem based on the object function of WLS shown as Eqn. (2.47). Lagrangian relaxation technique is employed to solve the optimization problem. Compared with Scheweppe's method, [48] can ensure the decentralized estimation result converges to the global optimal result. [27] proposed a method that incorporates PMU measurements with SCADA measurements, and the boundary bus information. The constrained WLS algorithm is employed in parallel to calculate the local SE.

Chapter 3: Proposed Partitioning Method and Multi-Area State Estimation Based on the CKF

Motivated by the method proposed in [19] where each index mentioned in the paper can be regarded as a criterion or a constraint of the partitioning method. The specific criteria are built based on the area that the method aims to apply. As an example, [19] attempts to use the proposed method to reduce the transaction leakage between each cluster after applying the proposed partitioning method. Based on this understanding, the research in this thesis introduces a new criterion or a constraint [19] that attempts to improve the accuracy of the state estimation. Instead of focusing on the algorithm of state estimation directly, the aim of the research is to find the most important factor that would affect the estimation result among either means of MASE algorithms. After researching the literature related to how to improve the accuracy of state estimation, the observability factor is the first priority that to be mentioned among the MASE algorithms [95, 76, 73]. Hence, this thesis aims to model a constraint that would consider the observability of each cluster during the partitioning process.

In this chapter, we describe the process of the proposed index modelling in order to build the partitioning method applied to multi-area state estimation. The index modeling is started from the PMU placement and the partitioning method is based on the method proposed in [19].

3.1 PMU Placement Based on Electrical Distance

Typically, the PMU placement problem is cast into the mathematical programming framework whose objective is to minimize the number of PMUs needed subject to various constraints. The most commonly employed criteria are complete and incomplete network observability [44, 72]. In fact, the observability of a system can be evaluated by analyzing the number of installed PMUs. The number of PMUs needed can be evaluated via existing PMU placement algorithms employing the criterion of full network observability [44, 72, 17].

As discussed in Section 2.4, the adjacency matrix A is the connectivity matrix of the power system, which is characterized by physical or geographic properties. Per [69], a newly defined adjacent matrix B is proposed, which is a virtual adjacent matrix modeled based on the electrical

structure over its topological structure. The proposed matrix not only considers the structure of the power grid used for the PMU placement problem, but also considers the electrical structure of a complex power network. Thus, the first step is to model a PMU placement algorithm related to the electrical distance to satisfy the globally fully observed requirement.

At the initial stage of the research, PMU measurements are the only factors for system observability. Starting from the virtual connectivity matrix B defined by [69]:

$$b_{ij} = \begin{cases} 1, \forall e(i,j) < \tau \\ 0, \forall e(i,j) \geq \tau \end{cases} \tag{3.1}$$

where b_{ij} represents the (i,j) entry of matrix B, $e(i,j)$ stands for the electrical distance between bus i and bus j, τ is the pre-defined threshold value. It means that bus i and bus j can be regarded as connected as long as the electrical distance between them is smaller than the pre-defined threshold value τ, outputting 1 as the entry value. Otherwise, the entry value will be 0, meaning disconnected.

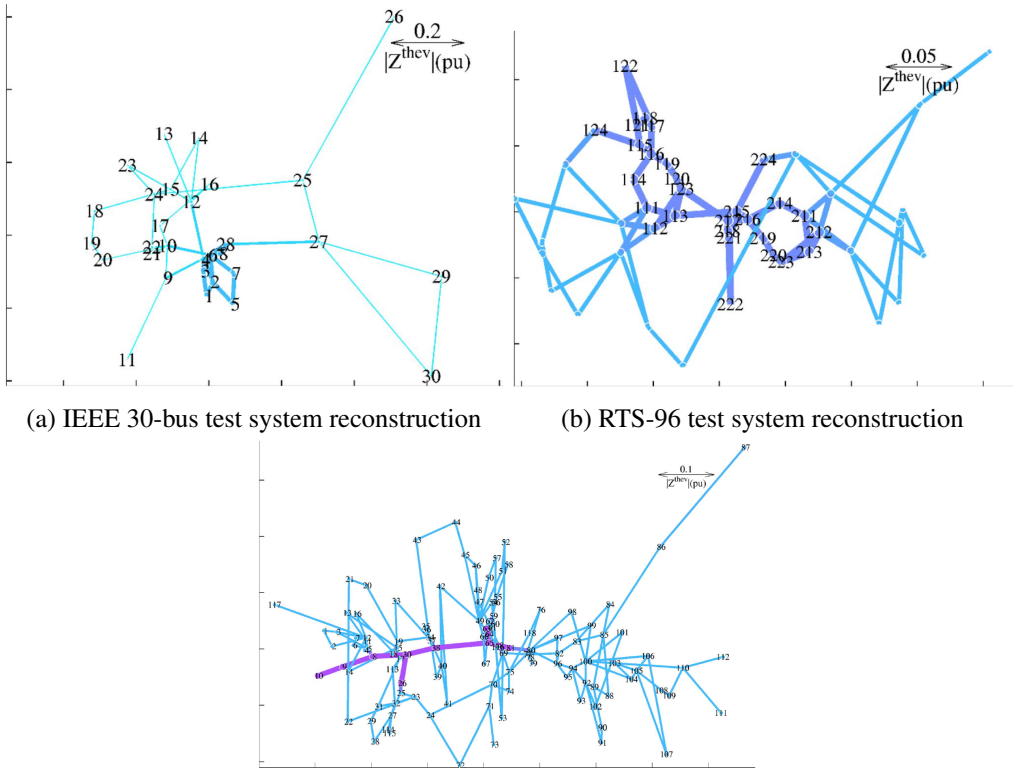
At this point, the τ criterion is the most essential part of the evaluation of the matrix B. We propose a search algorithm in this thesis based on the criteria from the topological observability analysis theory [4, 41, 69]. There are 5 main steps in the proposed B construction algorithm, namely,

1. Calculated the electrical distance $e(i,j)$, $i,j \in [1, num_{bus}]$ and $j \neq i$;
2. Sort the electrical distance matrix for each bus in row getting $e_{sort}(i,j)$;
3. Load the number of branches according to the defined case. Set the numbers as the constraint for each bus;
4. Set the initial value as $\tau = 0.1$;
5. Search the threshold value τ_i according to the constraint in Step 3 for each bus, after which the number of branches for each bus will have the same number of branches as in the real topology;

Step 1 and step 2 are the initial steps for the threshold searching algorithm. In the step 3, the constraint is set according to the topology observability analysis theory to ensure the observability on each iteration of threshold searching [41, 69]. The reason is that the level of

observability has a direct connection with the number of branches linked on each bus according to the property of the adjacency matrix A. Without this constraint, it would result in unbalanced observability for each bus over the global system.

The initial value of Step 4 is set based on the theory in [22]. [22] reconstructs the layout of different IEEE test cases by weighting the voltage on each bus. Figure 3.1(a) shows the layout when we use the IEEE 30 bus case system. The reconstructed layout shows the impedance distance as the distance between each node in the figure 3.1(a). As can be seen in the figure, buses sit within a core with a diameter of about 0.1 p.u. The diameter property, which is within a diameter of 0.1 p.u, can also be seen on RTS-96 case, and IEEE 118 test case, shown as figure 3.1(b, c). [22] employs the Sammon stress function to verify the performance of different kinds



(a) IEEE 30-bus test system reconstruction

(b) RTS-96 test system reconstruction

(c) IEEE 118-bus test system reconstruction

Figure 3.1: Bus test systems reconstruction [22]

of definitions related to the electrical structure. In terms of the distance related to the electrical

structure, the general idea can be expressed as [51, 40]:

$$E_{i,j} = g_{i,i} + g_{j,j} - g_{i,j} - g_{j,i} \tag{3.2}$$

where g is the impedance first introduced by [51]. As the concept of electrical structure becoming more and more popular, more variables in the power system have been applied to this form by using the Laplacian matrix concept in graph theory. In [22], 8 kinds of variables are involved in the distance evaluation process, which are Thevenin impedance Z^{thev}, Jacobian matrix $\frac{\Delta\theta}{\Delta P}$, Jacobain matrix $\frac{\Delta V}{\Delta Q}$, Jacobian matrix $\frac{\Delta V}{\Delta P}$, Jacobain matrix $\frac{\Delta\theta}{\Delta Q}$, power transfer PT, and mutual impedance Z^{mut}.

In terms of the Sammon stress function, the Sammon stress function can be generally explained as an error function E between the target variable and the true variable. This function is used to evaluate the goodness fit for the different distance definitions. The Sammon stress function is defined by [22]:

$$E = \frac{1}{\sum_{i<j} d'_{ij}} \sum_{i<j}^{N} \frac{(d'_{ij} - d_{ij})^2}{d'_{ij}} \tag{3.3}$$

d'_{ij} is the target variable, which will be embedded in the 8 variables mentioned before by using the iterative gradient descent methods [22]. The result is shown in Table 3.1.

		rts_1_area **[83]**	***case30*** **[84]**	***case39*** **[85]**	***rts_2_area*** **[83]**	***case57*** **[86]**	***case118*** **[86]**	***case300*** **[86]**
Thevenin Impedance	$\lvert Z^{thev}\rvert$	0.0261	0.0328	0.045	0.0229	0.0256	0.0333	0.0316
Jacobian	$\Delta\theta/\Delta P$	0.0317	0.0319	0.0377	0.0306	0.0294	0.0409	0.0195
Shortest Path	d^{geo}	0.0267	0.0457	0.0153	0.0165	0.059	0.0281	0.0349
Power Transfer	PT	0.0275	0.0447	0.0203	0.0295	0.0485	0.0383	0.0453
Jacobian	$\Delta V/\Delta Q$	0.0446	0.0769	0.0437	0.0293	0.0284	0.0289	0.0187
Jacobian	$\Delta V/\Delta P$	0.0243	0.1462	0.0417	0.1847	0.0298	0.1073	0.1769
Mutual Impedance	$\lvert Z^{mut}\rvert$	0.1267	0.1453	0.1266	0.0914	0.1457	0.0397	0.3624
Jacobian	$\Delta\theta/\Delta Q$	0.1939	0.1329	0.2558	0.3363	0.1489	0.291	0.0728

Table 3.1 Sammon Stress Value [22]

In this thesis, the electrical distance is evaluated by defining g in Eqn. (3.2) as $g = \frac{\Delta\theta}{\Delta P}$. According to the result shown in Table 3.1, the electrical distance and the the impedance distance have similar Sammon stress values [22]. In other words, it implies that the value of electrical

distance would be similar to the value of impedance distance. At this point, the initial value of 0.1 is regarded as acceptable [22]. In summary, the construction of the matrix B is illustrated by the flow chart of figure 3.3.

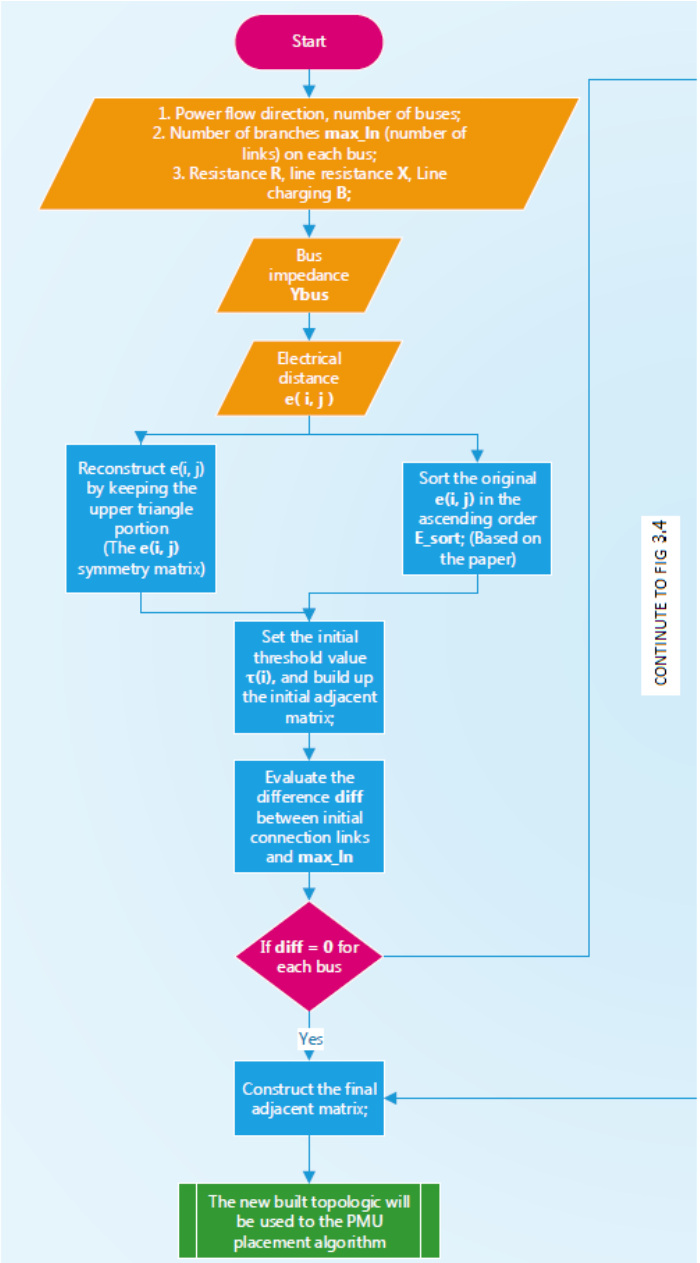


Figure 3.3: Matrix B constructing algorithm flow chart (PART ONE)

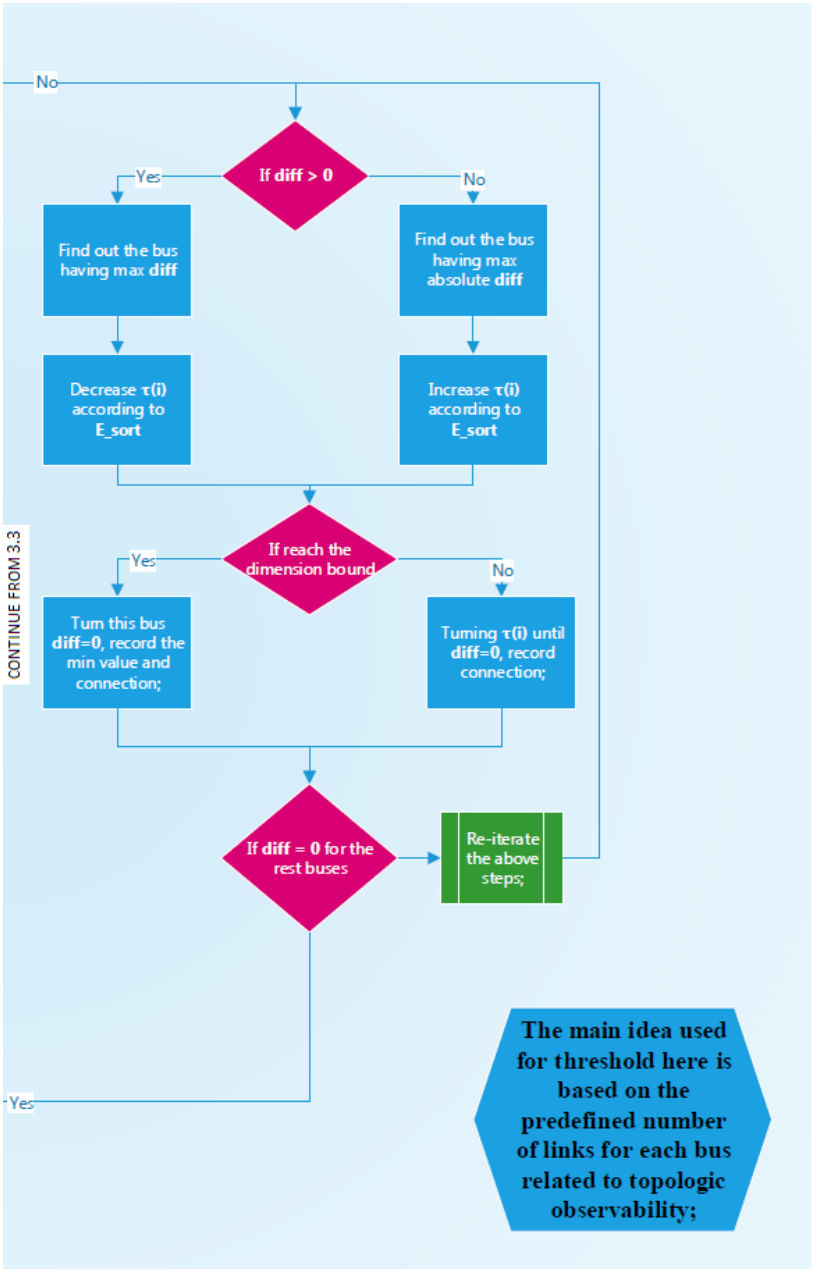


Figure 3.4: Matrix B constructing algorithm flow chart (PART TWO)

The algorithm gathers all the necessary data for the electrical distance calculation at the first step. Sorting the electrical distance is the second step. Only the upper triangle part of the matrix is used during the sorting because of symmetry. The initial connectivity matrix is built based on the initial threshold value, from which the number of branches can be evaluated. This number will be used to calculate the difference between its value and the number of branches from the real topology. An iteration is implemented if the difference does not equal to zero. Inside the iteration, threshold value is increased according to the electrical distance if the difference value is a positive number. Otherwise, the threshold value is decreased. The iteration is terminated if the difference value is zero, or at the moment it reaches the boundary.

By using the above proposed construction algorithm of adjacency matrix, the electrical based connectivity can be represented by using the matrix B. The following step of that would be the PMU placement algorithm is based on the electrical adjacent matrix. The PMU placement algorithm we propose is an extra constraint on top of the PMU placement algorithm. The extra constraint we propose is modeled based on a statistical measurement in graph theory and the electrical structure based connectivity property.

As the electrical distance definition in [19], it defines the distance between two buses by considering the realistic electrical connectivity. It implies that two graphically close buses would have higher probability of losing connection than two graphically far away buses if the two geographically close buses have large electrical distance. Based on this understanding, the bus has the graphical connection with the PMU located bus would have the possibility losing the observability if their electrical distance is relatively large. [18, 40, 69] introduce an useful statistical measurement based on the graph theory. The statistical measurement is call average nodal distance, which can be expressed:

$$d_i^{avg} = \sum_{j=1}^{N} \frac{d_{ij}}{N-1} \tag{3.4}$$

where d_i^{avg} is the average nodal distance from each node i to other nodes j in the network. N is the total number of nodes involved. d_{ij} are the entries of the distance matrix D, which gives the minimum number of links that one would need to traverse to get from node i to node j.

Inspired by the Eq. (3.4), the coefficient d_{ij} can be replaced by any factor that needs to be represented. As for the performance of the PMU placement algorithm using adjacency matrix B, the main point that needs to be paid attention to is the connectivity represented by the matrix B.

As for this purpose, d_{ij} can be replaced by the entries of the connectivity matrix B, denoted as:

$$Gamma_i^B = \sum_{j=1}^{N} \frac{b_{ij}}{N-1} \tag{3.5}$$

where $Gamma_i^B$ is the average nodal distance of using matrix B, and b_{ij} are the entries of matrix B, and N is the number of buses in the system. According to Eq. (3.5), the minimum value of $Gamma_i$ represents the lowest connectivity between bus i and the rest of buses. This means that the lower the $Gamma_i$ value is, the higher demand of having a PMU placed in.

The DFS heuristic algorithm for the PMU placement algorithm can be expressed as:

$$\begin{aligned} &\min \sum_{i=1}^{N} n_i^{pmu} \\ &\text{subject to } A \cdot \mathbf{x} \geq \mathbf{b} \end{aligned} \tag{3.6}$$

This equation says that the objective of DFS is to find the minimum number of PMUs n^{pmu} under the constraint that the observability of the system $A \cdot \mathbf{x}$ should greater than a pre-defined constant level $\mathbf{b}$.

The purpose of defining electrical distance is to build a comprehensive model on top of the power system geography structure, which will have an extra constraint by considering the electrical characteristic of the power system. Based on this consideration, the DFS PMU placement algorithm should be regarded as the first step since it only considers the geographical property of the power system. In our proposed algorithm, an extra PMU will be installed on the bus which has lowest *Gamma* value since this kind of bus has the possibility of losing electrical connectivity.

3.2 Measurement Redundancy Index

When combining the electrical distance with the PMU placement algorithm, the idea is to use the PMU placement algorithms as the link between system observability and the new index modeling. By taking a look at the PMU placement algorithm, we found out that the main constraint of the algorithm is the observability value, or the measurement redundancy value defined in [70, 72]. The measurement redundancy value is evaluated by an equation based on the topologi-

cal observability analysis, which can be expressed as [17, 70]:

$$\mathbf{f}_i = A \cdot \mathbf{x}_i,\ i = [1, num_{buses}] \tag{3.7}$$

where A is the adjacency matrix based on the power system topology. $\mathbf{x}_i$ is the binary vector denoting the observability of each bus, which means that 1 is observed and 0 is unobserved.

The first hypothesis of the index is that the measurement redundancy value would be zero after applying the partitioning method without a proper constraint related to observability. The algorithm of this hypothesis can be illustrated via the IEEE 14-bus system example.

Starting from the topology of the IEEE 14-bus system, the topology is shown as Figure 3.4. According to Eq. (3.6), the measurement redundancy value can be evaluated for each bus as follows:

$$\begin{aligned}
f_1 &= x_1 + x_2 + x_5 \\
f_2 &= x_1 + x_2 + x_3 + x_4 + x_5 \\
f_3 &= x_2 + x_3 + x_4 \\
f_4 &= x_2 + x_3 + x_4 + x_5 + x_7 + x_9 \\
&\vdots \qquad \vdots \\
&\vdots \qquad \vdots \\
f_{13} &= x_6 + x_{12} + x_{13} + x_{14} \\
f_{14} &= x_9 + x_{13} + x_{14}
\end{aligned} \tag{3.8}$$

The x_i will be 1 in Eq. (3.8) if the corresponding bus is observed by the PMU according to the previous explanation. The f_i is the corresponding bus measurement redundancy value. In the first hypothesis, the value of x_i can be expressed as:

$$x_i = \begin{cases} 1, \textit{being observed AND in the same cluster} \\ 0, \textit{unobserved OR outside the cluster} \end{cases} \tag{3.9}$$

Assuming that bus 9 is connected with a PMU and bus 13, 14 are not connected with PMU after implementing the placement algorithm globally, the value of f_{14} is 1 without the partitioning, which means bus 14 is observed. The first hypothesis is that the measurement redundancy of bus 14 could probably be expressed as $f_{14} = x_{13} + x_{14}$ after the partitioning method, whose value will be 0. It means that an observed bus would be an unobserved bus after the partitioning.

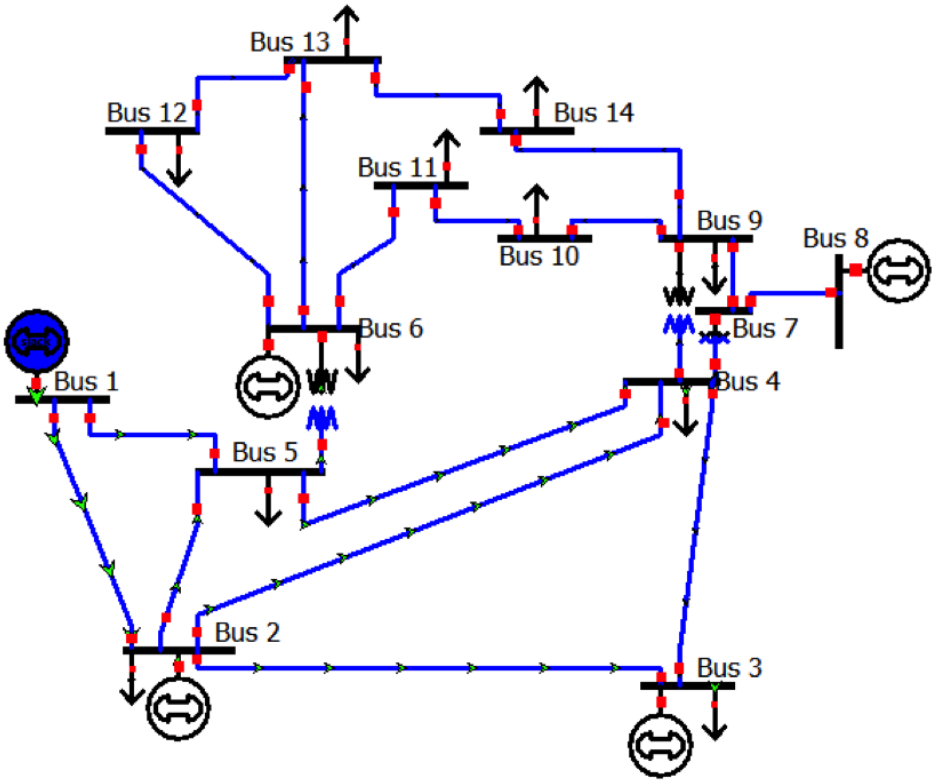


Figure 3.5: IEEE 14-bus System Topology [39]

As the new modeling constraint aims to highlight that the constraint of observability is definitely needed, the original partitioning method setting is the basis of the verification process of the first hypothesis. IEEE 14-bus system, IEEE 30-bus system, and IEEE 57-bus system are the test systems for the verification. The results are evaluated by using the partitioning method in [19] and equations (3.7) and (3.8). Unfortunately, the outputs show that no zero f_i value occurs for each bus among all the test cases. At this point, this idea needs to be improved to have a comprehensive index model.

Another approach is based on the initial idea illustrated above. Although no zero f_i occurs for each bus during the verification process of the above idea, an unbalanced value occurs for each cluster during this process. It is the average measurement redundancy value (AMRV) of each cluster after the partitioning method. The evaluation of the AMRV combines the idea of

equations (3.7) and (3.8), which can be expressed as:

$$AMRV_i = \frac{\sum_{j \in C_i} f_j}{n_{c_i}} \tag{3.10}$$

where f_j is the measurement redundancy of each bus j within the cluster C_i ($j \in C_i$), $\sum_{j \in C_i} f_j$ is the overall measurement redundancy of cluster C_i, n_{c_i} is the total number of bus in the same cluster. Eq. (3.10) denotes the average observability value of a cluster.

The $AMRV$ value of each cluster will be different. This difference value denotes the different degree of each cluster being observed, which is caused by unbalanced observation measurements. Considering the observability influence on the accuracy of MASE, the new constraint could be defined as the necessity of the balanced observability. In other words, the AMRV difference should be minimized between each cluster after applying the partitioning method. The new index can be expressed as:

$$\begin{aligned} \min \;\; & diff_{ij},\ i \neq j \in cluster_{nums} \\ S.t \;\; & diff_{ij} = AMRV_i - AMRV_j \\ & AMRV_a = \frac{\sum_{b \in C_a} f_b}{n_{c_a}} \end{aligned} \tag{3.11}$$

where $diff_{ij}$ is the difference in AMRV between cluster i and cluster j, $cluster_{nums}$ is the total number of clusters by partitioning, and $AMRV_{opt}$ is the pre-defined optimal value for AMRV.

Eq. (3.11) illustrates the expected effect of the observability index in the partitioning method. The expectation is that each cluster would have approximately the same level of observability as each other after the partitioning, which is reflected by the value of f_i.

According to this expectation that we defined above, the ideal situation would be that each cluster $AMRV_i$ value would hover around a pre-defined value K after applying the observability index. Under this constraint, the overall system $AMRV_{sys}$ value would approach the pre-defined value K as well, which can be expressed as:

$$AMRV_{sys} = \frac{\sum AMRV_i}{N_c} = K \tag{3.12}$$

Therefore, the value of K can be defined as an optimal value of AMRV, i.e. $AMRV_{opt}$. The optimal value K would be the constraint of the partitioning method from the model of the observability index, which is the expected observability threshold of each cluster. By the redefined

description of observability, the framework of the index can be expressed as:

$$\begin{aligned} \min \;\; & diff_{ij},\ i \neq j \in cluster_{nums} \\ S.t \;\; & diff_{ij} = AMRV_i - AMRV_j \\ & AMRV_a = \frac{\sum_{b \in C_a} f_b}{n_{c_a}} \\ & AMRV_{opt} = K \end{aligned} \tag{3.13}$$

Generally speaking, the ideal case is that where each cluster will concentrate around K measurement redundancy value after applying the observability index, shown as Eq. (3.12), to the partitioning method. However, an index function needs to be presented at this point. As the proposed index will be applied to the method proposed in [19], the index function should act as a fitness score function having the value ranging from 0 to 1 $[0,1]$. Its fitness score reflects the wellness of the balance distribution of the measurement, which is highly related to the $AMRV$.

Borrowing ideas from [19] and the definition of log-normal distribution, the index fitness function will be constructed based on the property of the log-normal distribution function. A log-normal process is the statistical realization of the multiplicative product of many independent random variables, each of which is positive. The log-normal distribution is a skewed distribution based on the normal distribution. It occurs at the scenario that the logarithm of a variable has a normal distribution[78, 86]. The probability distribution function of the log-normal distribution can be expressed as:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{x-\mu}{2\sigma^2}} \tag{3.14}$$

where μ is the mean and σ is the standard deviation. Figure 3.5 shows the standard plot of the normal distribution, and the log-normal distribution.

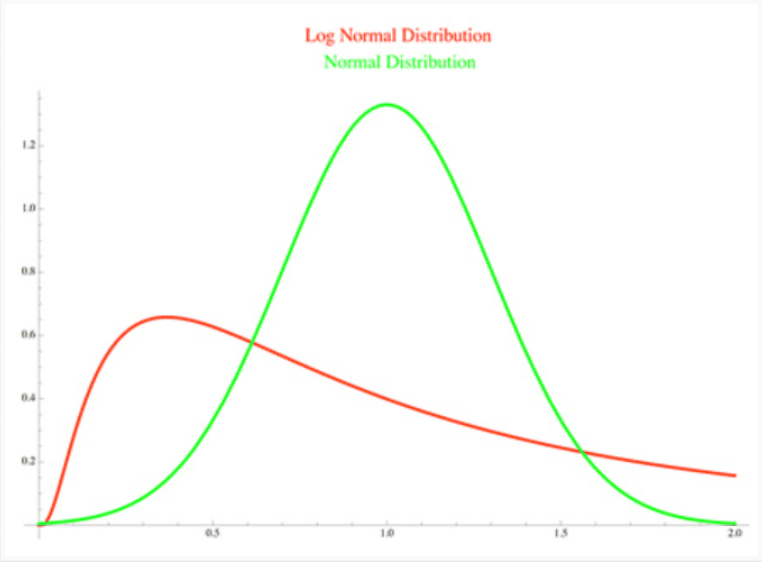


Figure 3.6: Normal Distribution and Log-Normal Distribution

According to the definition of the log-normal distribution in [86], there are two arguments for the construction of the log-normal distribution. The first argument is the location parameter and it corresponds to the mean of the associated normal distribution. The second argument is the scale parameter and corresponds to the standard deviation of the associated normal distribution [86]. The mean is the expected center of a series of data and the scale can be constructed by the standard deviation. Given the description of the log-normal distribution, it has the property that we would like to incorporate into the construction of the observability index.

Assuming the total number of the clusters is a random number ranging from 1 to the total number of buses in a system $Cluster_{num} \in [1, n_{bus}]$, each AMRV could be approximately regarded as a log-normal random variable according to the observation of its distribution property. According to the Eq. (3.11), the distribution of $AMRV_{sys}$ could be regarded as a log-normal sum distribution [13, 78]. By directly using the proposed results from [68, 78], the log-normal sum cumulative distribution function(CDF) can be expressed in a closed-form after the approximation, denoted as:

$$F_L(x) = \Phi\left(\frac{\log x - \mu_L}{\sigma}\right) \tag{3.15}$$

where $\Phi(\cdot)$ is the zero-mean, unit variance Gaussian CDF described:

$$\Phi(g) = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{g^2}{2}} \tag{3.16}$$

Substituting Eq. (3.16) into Eq. (3.15), the log-normal sum CDF can be expressed as:

$$F_L(x) = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{(\log x - \mu_L)^2}{2\sigma^2}} \tag{3.17}$$

We will use this mathematical model to construct the expression of the observability index. x in this model is the *AMRV* evaluated by Eq. (3.10). The mean value μ and the standard variance value σ are defined by the properties of the system, which are mainly used to define the tolerance range of the constraint of the observability index.

The first argument that needs to be defined is the definition of the mean value μ. Per previous discussion, we would like to use *AMRV* to constrain the cluster observability value so that they could equally hover around K. [17] suggests that N-1 contingency need to be considered for observability analysis. The main idea of the N-1 contingency is the natural property of losing at least one measurement of the analyzed system. The measurement could either be the PMU measurement or be the conventional measurement. As for the initial stage of the research, the PMU measurement is the only consideration factor. At this point, the N-1 contingency means there would be one or more PMUs with functionality loss or the communication between the PMU and the control center. Combining the N-1 contingency concept and the observation result from the measurement redundancy value, the value K is set as 3 in our research. Thus, the mean value μ is denoted as $\ln(3)$.

The second argument is the value of the standard deviation. The value of the standard deviation is $w\ln(n_{num})$, where w is a penalty factor defining the tolerance width of the index function. If w is large, the penalty for the $AMRV_{sys}$ will be increased when it is far away from the optimal value. The value of w is set based on experimental observations, which will be presented in the next chapter. n_{num} is the number of buses of the system. Its setting relates the width of the index tolerance to the system size. By this setting, this shape is expected to be 1 when the $AMRV_{sys}$ is the optimal value K, and to be 0 when $AMRV_{sys}$ has the value far outside of the tolerance width range.

The index is called the measurement redundancy index (MRI) for the partitioning method. Using the log-normal likelihood model, it can be ensured that the MRI is normalized to fit within

the range $[0,1]$ to keep the consistency. Hence, the index function can be expressed as:

$$MRI = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{(\log(AMRV)-\log 3)^2}{2(w\cdot\log(n_{bus}))^2}} \tag{3.18}$$

where $AMRV$ varies for each cluster and n_{bus} is the total number of buses in the system.

3.3 The Fitness Score Function of the Partitioning Method

This thesis borrows the concept from the proposed method presented in [19]. The observability index MRI will be the extra index incorporated in our method. The overall fitness score f will be the evaluation of the partitioning method. It is structured by aggregating all the constraints, the indices in other words, with various weights assignments. This thesis focuses on the measurement distribution during the partitioning method, which is denoted by MRI described by Eq. (3.18).

Besides that, [19] presents 4 other constraint indices, which are Electrical Cohesiveness Index (ECI), Between-Cluster Connectness Index (BCCI), Cluster Count Index (CCI), Cluster Size Index (CSI). In power systems applications, the most important objectives are the inner-cluster and inter-cluster distance in terms of the partitioning method [19]. Besides, the constraints conditions are all constructed based on the electrical distance. At this point, the ECI and BCCI are the two essential indices needed in this research, which are directly evaluated based on the electrical distance value.

In terms of the CSI, it can be seen that the measurement redundancy value would be affected by the number of buses contained in a cluster. With the increasing number of buses in a cluster, it has the possibility that the measurement redundancy value can be increased. Moreover, the AMRV will be different by containing a different number of buses in a system even if the measurement redundancy value of the system remains unchanged. Thus, the CSI will be considered in the partitioning method presented in this research.

Since a connection exists between CSI and CCI according to the demonstration in [19], CCI will be considered in our partitioning method as well. Furthermore, the fitness score function should have an index that can produce the result of 0 when there is any kind of disconnection. Finally, the binary coefficient called Cluster Connectedness (CC) will also be included [19]. Thus,

the fitness score f in this research can be expressed as:

$$f = ECI^{\alpha} \cdot BCCI^{\beta} \cdot CCI^{\gamma} \cdot CSI^{\zeta} \cdot MRI^{\lambda} \cdot CC \tag{3.19}$$

where $\alpha, \beta, \gamma, \zeta$ and λ are the weight coefficients. These coefficients values are user-defined scalars ranging from 0 to 1, $\{\alpha, \beta, \gamma, \zeta, \lambda\} \in [0, 1]$. They represent how much weight the partitioning method would have on the corresponding constraint index. The value of each index will be discussed in the next chapter.

Three advantages are presented by using the fitness score equation. First of all, the fitness score equation will denote how well the partitioning performs by representing a number ranging from 0 to 1. Secondly, the product of the five individual indices satisfies the preferential independence, whose definition is presented in [19]. The preferential independence means that the result of the fitness score equation would not produce clustering solution with high total fitness, but low scores for one or more indices. Finally, the fitness score would ensure that there would not be only one bus cluster.

3.4 Multi-area State Estimation Based on CKF With Estimated Noise

According to the dynamic model shown as Eqn. (2.51) and (2.52) in Chapter 2, there are two functions that determine the state estimation implementation algorithm, namely, the dynamic vector function $\mathbf{f}(\cdot)$ and measurement vector function $\mathbf{h}(\cdot)$. The state transition from time k to $k+1$ is realized by the dynamic vector function $\mathbf{f}(\cdot)$. [50, 31, 3] define the model of $\mathbf{f}(\cdot)$ as a non-linear function. However, the state vector in our research only contains the voltage magnitude and voltage angle, which are assumed to change slowly, since no generator failure is considered in our research. In order to simplify the implementation process, three assumptions are made about $\mathbf{f}(\cdot)$ [43]:

- No drastic change is experienced by the system dynamics.
- Implementation time frame is small enough, which allows a linear model to describe the state transition between consecutive instants of time.
- The system process noise is assumed to be zero-mean white Gaussian noise, and it is enough to describe the uncertainties.

By considering these three assumptions, a generic linear model can be obtained for the dynamic state function, i.e. [43, 37]:

$$\mathbf{x}_{k+1} = F_k \mathbf{x}_k + \mathbf{g}_k + \mathbf{w}_k \tag{3.20}$$

where F_k is the constant matrix representing the state transition, $\mathbf{g}_k$ is the vector associated with the trend behavior of the state trajectory, and $\mathbf{w}_k$ is the process Gaussian noise with covariance matrix Q. Moreover, $\mathbf{g}_k$ will be the state vector from previous step in the implementation [37].

In this work, we only use PMU measurements for state estimation. Based on the properties of PMUs, the typical module for the PMU measurement function is defined as [15, 14]:

$$I_{ij} = Y_{ij} \cdot (V_i - V_j) \tag{3.21}$$

where I_{ij} is the branch current phasor, V_i and V_j are the voltage phasors, and Y_{ij} is the admittance between bus i and bus j. Typically, this module is described by [15]:

$$\begin{aligned} I_{real} &= V_{ij}^{real} G_{ij} - V_{ij}^{imag} B_{ij} \\ I_{imag} &= V_{ij}^{real} B_{ij} + V_{ij}^{imag} G_{ij} \end{aligned} \tag{3.22}$$

where G_{ij}, B_{ij} are the real and the imaginary parts of the admittance Y_{ij}, $V_{ij}^{real}, V_{ij}^{imag}$ are the real and the imaginary part of the voltage phasor. In order to implement the linearized measurement functions, the measurement from the PMU needs to be converted from raw data to complex number as phasor, and the complex number needs to be split into the real and imaginary parts. This conversion potentially increases the computational complexity for large scale systems. For simplification purposes, the complex form is implemented by deriving the number from raw data directly, i.e.

$$\mathbf{I_{ab}} = \frac{\mathbf{v_a} - \mathbf{v_b}}{R_{ab} + jX_{ab}} \tag{3.23}$$

where $\mathbf{I_{ab}}$ is the measured branch current phasor, $\mathbf{v_a}$ and $\mathbf{v_b}$ are the bus voltage phasors, R_{ab} and X_{ab} form the given branch impedance. Since $\mathbf{h}(\cdot)$ functions is non-linear function, the state estimation algorithm needs to use a suitable non-linear dynamic state estimation algorithm.

As previously discussed, the CKF has better performance than WLS and other Kalman filter estimators. The CKF state estimator will be chosen for SE in the scenario where only PMU measurements are considered. Global system SE will be implemented at the beginning in order to verify that the CKF can produce the expected estimation result. Based on the description of

the CKF, the system and measurement noise variance needs to be initialized according to the knowledge of the system. The system noise and measurement noise have fixed mean and variance after the initial guess on the noise. the CKF experiment using this assumption does not have good convergence for the SE with the inaccurate initial noise variance. The estimated result and the convergence performance can be explained by the statement in [31] where the CKF estimator needs a precise initial value on the statistical characteristic of system noise. In other words, the estimated result using the CKF will be biased or even diverge if the statistical characteristics of the system noise are not exactly known [3]. In order to improve the performance of the CKF, we will introduce an on-line estimate and update noise statistics to leverage the information obtained in the filtering process [3, 30]. The estimated noise method is introduced to mitigate disturbances due to unknown or inaccurate system noise on system state estimation.

[3, 30] employ the maximum posterior (MP) technique for the online estimate and update noise process. MP is a technique based on deep-rooted Bayesian formalism to estimate parameters by maximizing the posterior probability densities [30], which is a conditional density function as well. The MP estimation of variable θ can be constructed as [30]:

$$\hat{\theta} = \arg\max_{\theta} p(\theta|z) \tag{3.24}$$

where z is the measurement related to the unknown variable θ, and $p(\theta|z)$ represents the conditional density function of θ with respect to z. Based on the property of MP, system state vector and measurement vector obey the Gaussian distribution to ensure the accuracy [3, 30]. The noise factors after MP will be denoted as $\hat{Q}$ for the system variance, $\hat{\mathbf{v}}$ and $\hat{R}$ for measurement noise mean and variance. They are defined as [3]:

$$\hat{Q}_{k+1} = \frac{1}{k+1}\sum_{j=0}^{k}\left\{\left[\hat{\mathbf{x}}_{j+1} - \hat{\mathbf{x}}_{j+1|j}\right]\left[\hat{\mathbf{x}}_{j+1} - \hat{\mathbf{x}}_{j+1|j}\right]^{T}\right\} \tag{3.25}$$

where $\hat{Q}_{k+1}$ stands for the variance of the system noise at time epoch $k+1$, $\hat{\mathbf{x}}_{j+1|j}$ stands for the prediction result from second step after time nonlinear transform function $\mathbf{f}(\cdot)$ of Eq. (2.59), $\hat{\mathbf{x}}_{j+1}$ is the final estimation result for time epoch j.

$$\hat{\mathbf{v}}_{k+1} = \frac{1}{k+1}\sum_{j=0}^{k}\left\{\mathbf{z}_{j+1} - \hat{E}\left[\mathbf{h}_{j+1}(\mathbf{x}_{j+1}|\mathbf{z}_k)\right]\right\} \tag{3.26}$$

where $\mathbf{z}_{j+1}$ is the measurement assumed to be given, $\hat{E}\left[\mathbf{h}_{j+1}(\mathbf{x}_{j+1}|\mathbf{z}_k)\right] = \mathbf{h}_{j+1}(\hat{\mathbf{x}}_{j+1})$ is the posterior mean of the final state estimation propagating by the nonlinear measurement function $\mathbf{h}_k(\cdot)$.

$$\hat{R}_{k+1} = \frac{1}{k+1}\sum_{j}^{k}\left\{\left[\mathbf{z}_{j+1} - \hat{\mathbf{z}}_{j+1}\right]\left[\mathbf{z}_{j+1} - \hat{\mathbf{z}}_{j+1}\right]^T\right\} \tag{3.27}$$

where $\hat{\mathbf{z}}_{j+1}$ is the predicted measurement after the update of cubature points in step 3 shown as Eq. (2.64).

After the initial guess of the system noise and measurement noise, the on-line noise update and estimation is implemented. The definition of $\hat{Q}_{k+1}$ will be substituted into the stage of updating the mean and covariance of state variable in step 2, specifically using Eq. (2.60). $\hat{\mathbf{v}}_{k+1}$ will be substituted into the stage of estimating measurement of the cubature point, shown as Eq. (2.63). $\hat{R}_{k+1}$ will be substituted into the stage of updating the auto-covariance value of measurement $\mathbf{z}_{k+1}$ shown as Eq. (2.65).

After this algorithm is applied to the state estimation of the global system, the modified two-level MASE will be implemented in this research. First of all, the global system needs to be decomposed into several sub-systems. The partitioning method introduced previously will be implemented to obtain the system decomposition. The system will be partitioned into several sub-systems without overlapping because of the use of the index BCCI. Thus, the MASE will be the non-overlapping areas level. Similar to the typical two-level MASE, the first level will have the local state estimation independently implemented on each sub-system. Unlike the typical two-level MASE, the voltage angle will be the only factor that needs to be coordinated at the second level in the modified two-level MASE. It means that the information needs to be coordinated will be reduced by using the modified two-level MASE. Two main reasons cause the reduction of coordination. First of all, the connection between sub-systems will be interrupted during the MASE as the non-overlapping property of our decomposition. Because of the independence of the sub-system, all buses within a sub-system can be directly or indirectly evaluated by using PMU measurements. Thus, the boundary measurement will not be regarded as a pseudo-measurement. Hence, the boundary bus measurement will not be re-evaluated at the central coordination level.

The overall process of the modified two-level MASE can be summarized by the following steps:

1. Partition the global system into sub-systems using the GA employing the indices ECI,

BCCI, CCI, CSI, and MRI. Measurements are all evaluated by PMU.

2. First level MASE is implemented with the SE applied to individual sub-systems independently by using modified CKF algorithm. For each sub-system, the slack bus will be selected according to the application. Since PMU measurements will be used, the reference bus will be the bus that has the PMU installed, and it has the smallest bus index. The first level estimate is denoted as: $\hat{\mathbf{x}}_k = \left[\hat{v}_1, \hat{v}_2, \ldots, \hat{v}_n, \hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_n\right]^T$.

3. All the estimated state vectors will be transformed by the central processor for the coordination. As the boundary buses measurements are evaluated from PMUs, only the angle adjustment will be needed at the coordination level. Since each slack bus has a PMU installed, the difference between the local slack bus and the global slack bus can be evaluated according to the data from PMU. For example, if cluster 2 slack bus θ_0^2 has different angle than the global system slack bus angle θ_0^G. The difference can be evaluated based on the PMU data $\Delta\theta_0^2 = \theta_0^G - \theta_0^2$. The angle estimated result for cluster 2 can be re-evaluated by $\theta_i^2 = \hat{\theta}_i^2 + \Delta\theta_0^2$.

Chapter 4: Performance Evaluation

In this chapter, we evaluate state estimation performance of 3 scenarios, namely, the evaluation of having extra PMU placed in terms of the robustness of observability using electrical adjacent matrix B, and the performance of adding the proposed measurement redundancy index MRI, and the performance of the application of the partitioning method to MASE. Such performances are obtained via simulation using Matlab. The test systems only include IEEE 14-bus system, IEEE 30-bus system, IEEE 57-bus system, RTS 96 system, IEEE 118-bus system.

4.1 PMU Placement Performance Evaluation

As we proposed in Ch. 3, the problem that PMU placement algorithms try to solve can be initially stated as:

$$\begin{aligned} &\min \sum_{i=1}^{N} n_i^{pmu} \\ &Subject\ to\ A \cdot \mathbf{x} \geq \mathbf{b} \end{aligned} \tag{4.1}$$

where the matrix A is the adjacency matrix. The matrix A can be evaluated for a given Y_{bus}, which can be evaluated from MatPower. MatPower is employed in this simulation process to obtain the quality matrices to derive the bus and branch data. The `makeYbus` function from MatPower is employed, which is used to build the bus admittance matrix and branch admittance matrices from equations [99]:

$$\begin{aligned} I_f &= Y_f \cdot V \\ I_t &= Y_t \cdot V \\ Y_f &= Y_{ff} \cdot C_f + Y_{ft} \cdot C_t \\ Y_t &= Y_{tf} \cdot C_f + Y_{tt} \cdot C_t \\ Y_{bus} &= C_f^T \cdot Y_f + C_t^T \cdot Y_t \end{aligned} \tag{4.2}$$

where V is the voltage matrix, I_f, I_t are the branch current matrices at the `from` and `to` ends of all branches relating to the branch admittance matrices Y_f, Y_t, C_f, C_t are the built in connectivity constants. From the evaluated Y_{bus}, the matrix A is formed according to the Y_{bus} value. The

entries will be 1 for those buses which have values on the Y_{bus}, and its values will be 0 otherwise, which can be expressed as:

$$a_{ij} = \begin{cases} 1, Y_{ij}^{bus} \neq 0 \\ 0, Y_{ij}^{bus} = 0 \end{cases} \tag{4.3}$$

in which a_{ij} stands the (i, j) entry of matrix A, i, j are the bus indexes $i, j \in [1, N_{bus}]$, and Y_{ij}^{bus} is the Y_{bus} value.

The value of **b** is defined by the expectation of the user according to the measurement types being involved. Since the first step is to find the minimum PMU number and only PMU measurement will be obtain to realize the system observability, all the value of **b** are set to be 1 at the initial stage in Eq. (4.1). It means that the object bus would be considered as observed even if there is one measurement.

4.1.1 The Extra PMU Location By Using Gamma Value

As we proposed in Ch.3, matrix B needs to be constructed to evaluate the gamma values. The matrix B can be modeled by using the proposed method introduced in the Ch. 3. It will involve the evaluation of the electrical distance. The evaluation process directly uses the proposed method in [19], whose framework is:

$$e_{a,b} = (J_{P\theta}^{+})_{a,a} - (J_{P\theta}^{+})_{a,b} - (J_{P\theta}^{+})_{b,a} + (J_{P\theta}^{+})_{b,b} \tag{4.4}$$

where $J_{P\theta}$ is the Jacobian matrix $\frac{\Delta P}{\Delta \theta}$. The evaluation algorithm of the Eq. (4.4) uses the method introduced in [19], which is based on the matrix pseudo-inverse method proposed in [36].

Testing Case Name	Number of Links
IEEE 14-bus system	20
IEEE 30-bus system	41
IEEE 57-bus system	80
RTS 96 system	120
IEEE 118-bus system	186

Table 4.1: Number of Links of test systems.

The proposed threshold algorithm can be implemented after the evaluation of the electrical distance matrix, which starts searching from $\tau = 0.1$. Table 4.1 shows the number of links for 6 different test systems, which is the basic constraint for the threshold algorithm. The threshold value for each bus will be employed to construct the matrix B based on the equation:

$$b_{ij} = \begin{cases} 1, \forall e(i,j) < \tau \\ 0, \forall e(i,j) \geq \tau \end{cases} \tag{4.5}$$

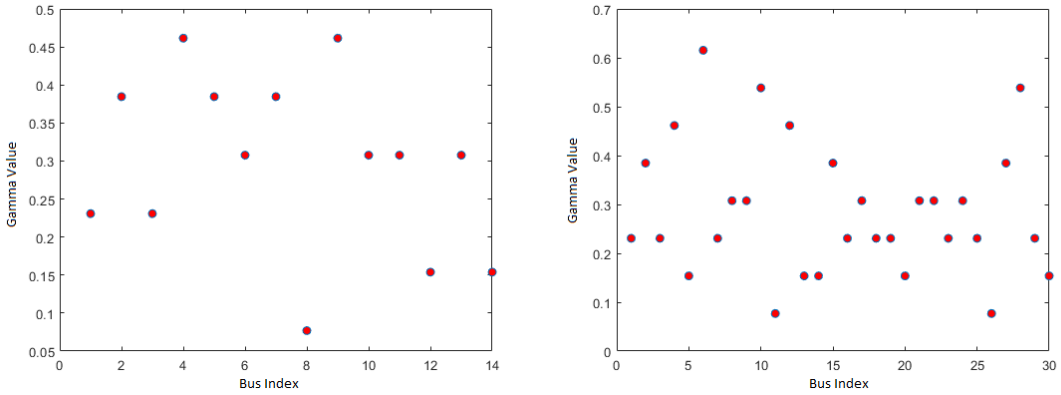
The entries b_{ij} will be used to evaluate the *Gamma* value based on the equation:

$$Gamma_i^B = \sum_{j=1}^{N} \frac{b_{ij}}{N-1} \tag{4.6}$$

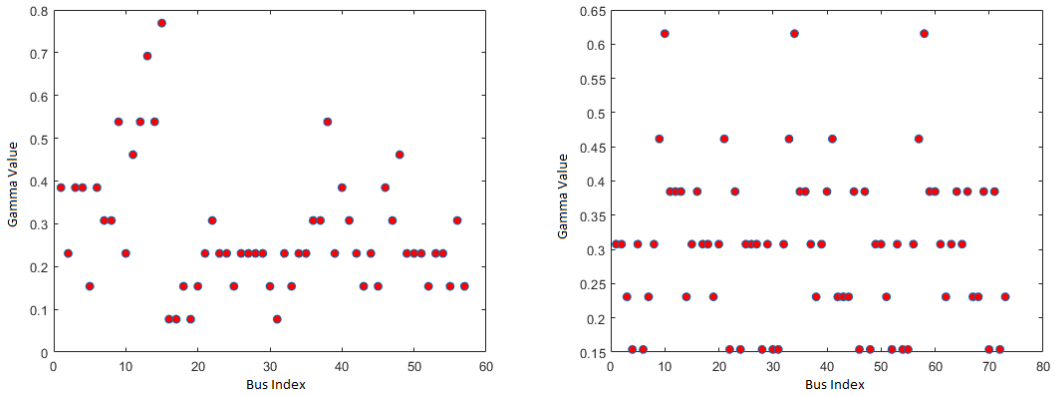
where the value of N in our simulation will be $N \in [14, 30, 57, 73, 118]$. Fig. 4.1 presents the gamma values of 5 systems by using Eq. (4.6), in which the x-axis is the bus index and y-axis is the gamma value. With the aim of maximizing the observability, the locations of the last two rows in Fig. 4.1 are selected as the candidate locations for the extra PMU based on the observation of experiment. Table 4.2 shows the candidate locations of each system for the extra PMUs.

Testing Systems	PMU Located Bus Number
IEEE 14-bus system	[8, 12, 14]
IEEE 30-bus system	[5, 11, 13, 14, 20, 26, 30]
IEEE 57-bus system	[5, 16, 17, 18, 19, 20, 25, 31, 33, 43, 45, 52, 55, 57]
RTS 96 system	[3, 4, 6, 7, 14, 19, 22, 24, 28, 30, 31, 38, 42, 43, 44, 46, 48, 51, 52, 54, 55, 62, 67, 68, 70, 72, 73]
IEEE 118-bus system	[6, 10, 13, 16, 20, 28, 33, 43, 48, 50, 53, 57, 67, 72, 84, 86, 87, 97, 98, 99, 107, 111, 112, 117]

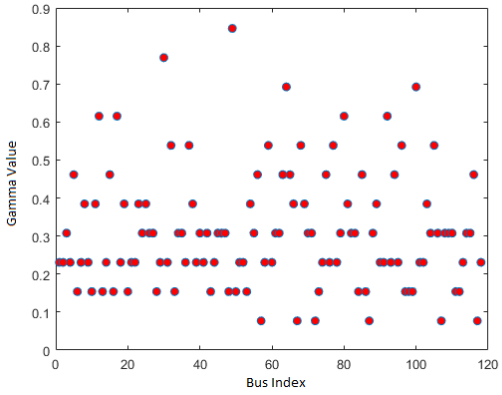
Table 4.2: The Candidate Locations of Extra PMUs.



(a) Left: IEEE 14-bus System
Right: IEEE 30-bus System



(b) Left: IEEE 57-bus System
Right: RTS 96 System



(c) IEEE 118-bus System

Figure 4.1: Gamma values For Different Systems

4.1.2 Performance Evaluation of Extra PMU Placement

Recently, various methods have been introduced for the PMU placement problem to maintain complete system observability by considering different contingencies in the power system [2]. The contingencies might cause the abnormal conditions for PMUs, such as loss of communication or fail on the operation. [82, 84, 83] propose a method to have minimum number of PMU while ensuring system observability when the system loses a single or multiple PMUs. Our research will use the idea to evaluate the performance of systems with and without extra PMU installed. Specifically, the system observability will be presented for two kinds of systems while one or more PMUs fail.

Following subsections present the performance for 5 systems using different kinds of PMU placement techniques. The performance for two kinds of systems, with and without extra PMU, includes the number of unobserved buses and the percentage loss in terms of system observability value. The failure scenarios are losing at random one PMU on each period and losing a random combination of two PMUs on each period.

Specifically speaking, the simulation aims to present the number of unobservable buses N_{unob} as well as the percentage loss PCT_{loss} in terms of the system observability by considering the loss of each of the single assigned PMU. In the simulation, the unobserved bus will be evaluated based on whether its measurement redundancy value equal to zero. The percentage loss is evaluated based on the overall system observability values before and after loss of the PMU, which aims to denote the impact on the system observability caused by the PMU unavailability. The percentage loss is described by:

$$PCT_{loss} = 1 - \frac{f_l}{f_{fo}} \tag{4.7}$$

where f_l and f_{fo} are the measurement redundancy values. The scalar values are expressed as $f_j = \sum_j a_{ij} \cdot x_j$, for the scenarios of losing one PMU and all PMUs available.

The simulation results for the IEEE 14-bus system through IEEE 118-bus system will be presented. In the results, “PMU Loc” stands for the location of the unavailable PMU, and its corresponding unobserved bus number and losing measurement redundancy value will be represented specifically.

4.1.2.1 The Evaluation of DFS with Heuristic Algorithm Having Extra PMU

Firstly, the PMU placement problem is solved by using DFS with Heuristic Algorithm without inserting the extra PMU. The result is shown as Table 4.3. Table 4.4 shows the PMU locations after inserting the extra PMU based on the candidate extra PMU locations shown as Table 4.2.

Testing Systems	PMU Located Bus Number
IEEE 14-bus system	[1, 4, 6, 8, 10, 14]
IEEE 30-bus system	[1, 5, 6, 10, 11, 12, 18, 24, 26, 27]
IEEE 57-bus system	[1, 4, 7, 9, 15, 19, 21, 24, 27, 30, 32, 36, 38, 39, 41, 46, 50, 52, 54]
RTS 96 system	[1, 7, 9, 10, 11, 16, 21, 23, 24, 25, 31, 33, 34, 38, 39, 42, 43, 46, 47, 49, 55, 57, 58, 64, 69, 71, 72]
IEEE 118-bus system	[1, 5, 9, 12, 13, 17, 19, 21, 25, 28, 30, 32, 36, 37, 41, 43, 46, 49, 52, 57, 58, 59, 62, 65, 70, 72, 73, 77, 80, 83, 86, 89, 91, 93, 95, 100, 102, 105, 110, 115, 116, 118]

Table 4.3: Minimum Number of PMUs Using DFS Heuristic Algorithm.

Testing Systems	PMU Located Bus Number
IEEE 14-bus system	[1, 4, 6, 8, 10, 12, 14]
IEEE 30-bus system	[1, 5, 6, 10, 11, 12, 13, 14, 18, 20, 24, 26, 27, 30]
IEEE 57-bus system	[1, 4, 7, 5, 9, 15, 16, 17, 18, 19, 20, 21, 24, 25, 27, 30, 31, 32, 33, 36, 38, 39, 41, 43, 45, 46, 50, 52, 54, 55, 57]
RTS 96 system	[1, 7, 9, 10, 11, 16, 21, 23, 24, 25, 31, 33, 34, 38, 39, 42, 43, 46, 47, 3, 4, 6, 14, 19, 22, 28, 30, 44, 48, 51, 52, 54, 62, 67, 68, 70, 73, 49, 55, 57, 58, 64, 69, 71, 72]
IEEE 118-bus system	[1, 5, 9, 12, 13, 17, 19, 21, 25, 28, 30, 32, 36, 37, 41, 43, 46, 49, 52, 6, 10, 16, 20, 33, 48, 50, 53, 67, 84, 87, 97, 98, 99, 107, 111, 112, 117, 57, 58, 59, 62, 65, 70, 72, 73, 77, 80, 83, 86, 89, 91, 93, 95, 100, 102, 105, 110, 115, 116, 118]

Table 4.4: PMU Locations For the Modified PMU Placement Technique.

PMU Locs	1	4	6	8	10	14
N_{unob}	1	2	2	1	1	1
PCT_{loss}	14%	28%	23%	9%	14%	14%

Table 4.5: DFS 14-bus PMU placement case

PMU Locs	1	4	6	8	10	12	14
N_{unob}	1	2	0	1	1	0	1
PCT_{loss}	12%	24%	20%	8%	12%	12%	12%

Table 4.6: Modified 14-bus PMU placement case

PMU Locs	1	5	6	10	11	12	18	24	26	27
N_{unob}	2	1	1	3	1	4	2	2	1	3
PCT_{loss}	7%	7%	19%	17%	5%	14%	7%	9%	5%	12%

Table 4.7: DFS 30-bus PMU placement case

PMU Locs	1	5	6	10	11	12	13	14	18	20	24	26	27	30
N_{unob}	2	1	1	2	1	1	0	0	1	0	2	1	0	0
PCT_{loss}	6%	6%	15%	13%	4%	12%	4%	6%	6%	6%	7%	4%	9%	6%

Table 4.8: Modified 30-bus PMU placement case

PMU Locs	1	4	7	9	15	19	21	24	27	30	32	36	38	39
N_{unob}	3	2	1	3	1	1	1	2	2	1	3	3	3	2
PCT_{loss}	7%	8%	5%	9%	8%	4%	4%	6%	4%	4%	5%	5%	8%	4%
PMU Locs	41	46	50	52	54									
N_{unob}	4	2	2	1	1									
PCT_{loss}	6%	4%	4%	4%	4%									

Table 4.9: DFS 57-bus PMU placement case

PMU Locs	1	4	5	7	9	15	16	17	18	19	20	21	24	25	27	30
N_{unob}	1	0	0	1	1	0	0	0	0	0	0	0	1	0	2	0
PCT_{loss}	4%	5%	3%	3%	6%	5%	3%	3%	3%	3%	3%	3%	4%	3%	3%	3%
PMU Locs	31	32	33	36	38	39	41	43	45	46	50	52	54	55	57	
N_{unob}	0	1	0	3	2	0	1	0	0	2	2	1	0	0	0	
PCT_{loss}	3%	3%	2%	3%	5%	3%	4%	3%	3%	3%	3%	3%	3%	3%	3%	

Table 4.10: Modified 57-bus PMU placement case

PMU Locs	1	7	9	10	11	16	21	23	24	25	31	33	34	38	39	42	43
N_{unob}	2	1	1	1	0	3	3	2	1	2	1	2	2	1	2	1	1
$PCT_{loss}(\%)$	3	2	5	5	4	4	6	5	2	3	2	5	5	2	5	3	3
PMU Locs	46	47	49	55	57	58	64	69	71	72							
N_{unob}	1	2	2	1	2	2	4	1	3	1							
$PCT_{loss}(\%)$	2	5	3	2	5	5	4	5	5	2							

Table 4.11: DFS RTS96 PMU placement case

PMU Locs	1	3	4	6	7	9	10	11	14	16	19	21	22	23	24	25	28
N_{unob}	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	1	0
$PCT_{loss}(\%)$	2	2	2	2	2	3	3	3	2	3	2	4	2	3	2	2	2
PMU Locs	30	31	33	34	38	39	42	43	44	46	47	48	49	51	52	54	55
N_{unob}	0	1	0	0	1	0	1	0	0	1	1	0	0	0	0	0	1
$PCT_{loss}(\%)$	2	1	3	3	2	3	2	2	3	2	3	2	2	2	2	2	1
PMU Locs	57	58	62	64	67	68	69	70	71	72	73						
N_{unob}	0	0	0	0	0	0	0	0	1	0	0						
$PCT_{loss}(\%)$	3	3	2	3	2	3	3	2	3	2	2						

Table 4.12: Modified RTS96 PMU placement case

PMU Locs	1	5	9	12	13	17	19	21	25	28	30	32	36	37	41	43	46
N_{unob}	1	3	2	4	1	0	1	2	1	2	0	1	1	3	1	2	1
$PCT_{loss}(\%)$	2	3	2	4	2	4	3	2	2	2	3	3	2	4	2	2	2
PMU Locs	49	52	57	58	59	62	65	70	72	73	77	80	83	86	89	91	93
N_{unob}	7	2	2	2	4	3	3	3	2	1	4	5	2	2	4	2	2
$PCT_{loss}(\%)$	1	3	1	1	2	1	5	1	1	1	1	1	3	2	2	1	3
PMU Locs	95	100	102	105	110	115	116	118									
N_{unob}	1	1	1	3	4	1	1	1									
$PCT_{loss}(\%)$	2	5	2	3	3	2	1	2									

Table 4.13: DFS 118-bus PMU placement case

PMU Locs	1	5	6	9	10	12	13	16	17	19	20	21	25	28	30
N_{unob}	1	1	0	0	0	1	1	0	0	0	0	1	1	2	0
$PCT_{loss}(\%)$	1	2	1	1	1	3	1	1	3	2	1	1	2	1	2
PMU Locs	32	33	36	37	41	43	46	48	49	50	52	53	57	58	59
N_{unob}	1	0	1	1	1	2	0	0	0	0	0	0	0	1	3
$PCT_{loss}(\%)$	3	1	1	3	1	1	2	1	5	1	1	1	1	1	3
PMU Locs	62	65	67	70	72	73	77	80	83	84	86	87	89	91	93
N_{unob}	0	2	0	2	1	1	1	2	0	0	0	0	2	1	1
$PCT_{loss}(\%)$	2	2	1	3	1	1	3	4	2	1	1	1	3	1	1
PMU Locs	95	97	98	99	100	102	105	107	110	111	112	115	116	117	118
N_{unob}	1	0	0	0	0	1	1	0	1	0	0	1	1	0	1
$PCT_{loss}(\%)$	1	1	1	1	4	1	3	1	2	1	1	1	1	1	1

Table 4.14: Modified 118-bus PMU placement

The above tables show the comparison of the change on the unobserved bus numbers, and loss percentage for each test system when different single PMU is unavailable. As the tables results show, the effect on the system observability has been minimized by having all extra PMU installed based on the electrical distance. In other words, the robustness has been realized in terms of the system observability when we consider the possibility of single PMU unavailable.

The following is the summary of the average number of unobserved bus, and the average

losing percentage caused by the single unavailable PMU in five test system:

- IEEE 14-bus system:
  - **DFS**: average unobserved bus number: 1.3, average loss percentage: 17%
  - **Modified**: average unobserved bus number: 0.86, average loss percentage: 14%
- IEEE 30-bus system:
  - **DFS**: average unobserved bus number: 2, average loss percentage: 10%
  - **Modified**: average unobserved bus number: 0.86, average loss percentage: 7%
- IEEE 57-bus system:
  - **DFS**: average unobserved bus number: 2, average loss percentage: 5%
  - **Modified**: average unobserved bus number: 0.58, average loss percentage: 3%
- IEEE 73-bus system:
  - **DFS**: average unobserved bus number: 1.7, average loss percentage: 4%
  - **Modified**: average unobserved bus number: 0.24, average loss percentage: 2%
- IEEE 118-bus system:
  - **DFS**: average unobserved bus number: 1.6, average loss percentage: 2%
  - **Modified**: average unobserved bus number: 0.63, average loss percentage: 1%

After presenting the robustness achieved on the system observability when inserting an extra PMU, the observability performance will be presented after losing two PMUs simultaneously in a period. The locations of the two PMU need to be decided by certain criteria since there would be many different combination sets for a large scale system. Inspired by [75], the PMU installed on the critical bus will be chosen. If the loss of PMU on the bus will cause the largest damage on the system observability, the bus can be defined as a critical bus. It implies that, among the given PMU placements, that the system observability would have the maximized observability value by putting the PMU on the critical bus. At this point, we assume that any two PMU would have higher probability to be attacked if they have larger system observability than others. As the

massive number of possible combinations, the simulation here will only present no more than 20 sets of combinations.

Combination set	{4,6}	{1,4}
Unobserved bus Number	4	4

Table 4.15: DFS IEEE 14-bus results

Combination set	{10,12}	{12,18}	{12,27}	{1,12}	{6,10}
Unobserved bus Number	7	7	7	6	6

Table 4.16: DFS IEEE 30-bus results

Combo	{9,41}	{1,41}	{32,41}	{36,41}	{38,41}	{1,9}	{1,15}	{1,32}	1,36}
Unobserved	8	7	7	7	7	6	6	6	6
Combo	{1,38}	{4,41}	{9,32}	{9,36}	{9,38}	{24,41}	{27,41}	{32,36}	
Unobserved	6	6	6	6	6	6	6	6	

Table 4.17: DFS IEEE 57-bus results

Combo	{16,64}	{21,64}	{21,71}	{64,71}	{1,64}	{16,21}	{16,71}	{23,64}	{25,64}	{33,64}
Unobserved	7	7	7	7	6	6	6	6	6	6
Combo	{34,64}	{39,64}	{47,64}	{49,64}	{57,64}	{58,64}	{64,69}	{1,16}	{1,21}	{1,71}
Unobserved	6	6	6	6	6	6	6	5	5	5

Table 4.18: DFS RTS 96 results

Combo	{12,110}	{5,12}	{5,110}	{12,37}	{12,59}	{12,80}	{12,105}	{37,110}	{59,62}	{59,110}
Unobserved	8	7	7	7	7	7	7	7	7	7
Combo	{80,110}	{105,110}	{1,12}	{5,37}	{5,59}	{5,80}	{5,105}	{9,12}	{9,110}	{12,21}
Unobserved	7	7	6	6	6	6	6	6	6	6

Table 4.19: DFS IEEE-118 bus results

The following table shows the performance with the extra PMU inserted. It can be seen that less unobserved number of buses will be caused for the system with extra PMU inserted when consider the possibility of losing two PMUs simultaneously in a period.

Combination set	{4,8}	{1,4}
Unobserved bus Number	4	4

Table 4.20: DFS IEEE 14-bus With Extra PMUs

Combination set	{10,24}	{1,10}	{1,24}	{6,10}	{1,5}
Unobserved bus Number	5	4	4	4	3

Table 4.21: DFS IEEE 30-bus With Extra PMUs

Combo	{27,36}	{36,38}	{36,46}	{36,50}	{38,50}	{1,36}	{7,36}	{9,36}	{24,27}
Unobserved	5	5	5	5	5	4	4	4	4
Combo	{24,36}	{27,38}	{27,46}	{27,50}	{32,36}	{36,41}	{36,52}	{38,46}	{46,50}
Unobserved	4	4	4	4	4	4	4	4	4

Table 4.22: DFS IEEE 57-bus With Extra PMUs

Combo	{7,21}	{7,23}	{7,25}	{7,31}	{7,38}	{7,42}	{7,46}	{7,47}	{7,55}	{7,71}
Unobserved	2	2	2	2	2	2	2	2	2	2
Combo	{21,23}	{21,25}	{21,31}	{21,38}	{21,42}	{21,46}	{21,47}	{21,22}	{21,55}	{21,71}
Unobserved	2	2	2	2	2	2	2	2	2	2

Table 4.23: DFS RTS 96 With Extra PMUs

Combo	{28,59}	{43,59}	{59,62}	{59,65}	{59,70}	{59,80}	{59,89}	{1,59}	{5,59}	{12,59}
Unobserved	5	5	5	5	5	5	5	4	4	4
Combo	{13,59}	{21,59}	{25,59}	{28,43}	{28,65}	{28,70}	{28,80}	{28,89}	{32,59}	{36,59}
Unobserved	4	4	4	4	4	4	4	4	4	4

Table 4.24: DFS IEEE-118 bus With Extra PMUs

4.1.2.2 The Evaluation of BILP Algorithm Having Extra PMU

Integer Linear Programming (ILP) is a mathematical optimization or feasibility strategy. "Integer" means that all the variables in the model are required to be integers, and the optimization strategy is called linear program when the objective function and constraints are linear. Recent years, it has been gaining a practical interest in a variety of decision making applications such as planning, scheduling, and telecommunication networks [63, 88, 2]. Binary integer linear programming (BILP) is a special integer linear programming, which searches for a binary integer feasible solution and updates the best binary integer feasible point found so far as the search tree grows[9]. BILP is commonly employed for finding the optimal PMU placement solution [63, 88, 9, 26] since it has an identical math expression as the PMU placement problems, whose canonical form can be expressed as:

$$\begin{aligned} &\max \mathbf{c^T x} \\ &\textit{Subject to } \; A \cdot \mathbf{x} \leq \mathbf{b} \\ &x_i = \begin{cases} 1, \mathbf{x} \in \lambda \\ 0, \text{otherwise} \end{cases} \end{aligned} \tag{4.8}$$

where λ represents certain criteria for the value of $\mathbf{x}$. By considering the properties of its linearity, the maximizing of a straight line is the same as minimizing the negative of the line, which means multiplying by -1 on both side of the inequality in Eq. (2.46). Thus, it could be expressed as:

$$\begin{aligned} &\min \mathbf{c^T x} \\ &\textit{Subject to } \; A \cdot \mathbf{x} \geq \mathbf{b} \\ &x_i = \begin{cases} 1, \mathbf{x} \in \lambda \\ 0, \textit{otherwise}. \end{cases} \end{aligned} \tag{4.9}$$

Eq. (2.47) can be easily merged to the PMU placement problem by defining the constraint λ as $x_i \xrightarrow{connect} x_j$, and constant vector $\mathbf{b} = \mathbf{1}$.

In the simulation, we use *cplexbilp* function in CPLEX Toolbox to solve the BILP problem in terms of optimal PMU placement to ensure the system observability. The result is shown in Table 4.25. Compared with the result of DFS with heuristic method presented in Table 4.3, BILP can provide fewer PMU number than DFS.

Testing Systems	PMU Located Bus Number
IEEE 14-bus system	[2,6,7,9]
IEEE 30-bus system	[1,2,6,10,11,12,15,19,25,29]
IEEE 57-bus system	[2,6,12,19,22,25,27,32,36,39,41,45,46,49,51,52,55]
RTS 96 system	[2,3,10,16,21,23,26,27,32,34,40,45,47,50,56,58,64,70,71,72]
IEEE 118-bus system	[1,5,9,12,15,17,21,25,28,34,37,40,45,49,52,56,62, 64,68,70,71,76,77,80,85,87,91,94,101,105,110,114]

Table 4.25: Minimum Number of PMUs Using CPLEX

Combined with extra PMU presented in Table 4.2, the PMU placement with extra PMU can be shown as Table 4.26. The total number of the BILP based with extra PMU will be compared with that for the DFS using heuristic algorithm, which is shown as Table 4.27. As can be seen in the table, these two methods would generate approximately the same number of PMU after inserting all extra PMUs.

Testing Systems	PMU Located Bus Number
IEEE 14-bus system	[2,6,7,8,9,12,14]
IEEE 30-bus system	[1,2,5,6,10,11,12,13,14,15,19,20,25,26,29,30]
IEEE 57-bus system	[2,5,6,12,16,17,18,19,20,22,25,27,31,32,33,36, 39,41,43,45,46,49,51,52,55,57]
RTS 96 system	[2,3,4,6,7,10,14,16,19,21,22,23,24,26,27,28,30,31,32,34,38,40,42, 43,44,45,46,47,48,50,51,52,54,55,56,58,62,64,67, 68,70,71,72,73]
IEEE 118-bus system	[1,5,6,9,10,12,13,15,16,17,20,21,25,28,33,34,37,40,43,45, 48,49,50,52,53,56,57,62,64,67,68,70,71,72,76,77,80,84,85, 86,87,91,94,97,98,99,101,105,107,110,111,112,114,117]

Table 4.26: PMU Placements For BILP with Extra PMUs

Testing Systems	Total PMU Numbers of DFS	**Total PMU Numbers of BILP**
IEEE 14-bus system	7	7
IEEE 30-bus system	14	16
IEEE 57-bus system	31	26
RTS 96 system	45	45
IEEE 118-bus system	60	55

Table 4.27: Comparison of Total Number of PMUs between BILP- and DFS-based Methods

In order to evaluate the robustness due to the extra PMU, the unobserved total number of buses and the loss percentage will be presented for the cases when a single PMU and two PMUs are lost. The results are presented on Table 4.28 to Table 4.37.

PMU Locs	2	6	7	9
N_{unob}	3	4	1	2
PCT_{loss}	23%	23%	21%	23%

Table 4.28: IEEE 14-bus PMU Placement

PMU Locs	2	6	7	8	9	12	14
N_{unob}	3	1	0	0	1	0	0
PCT_{loss}	19%	19%	15%	7%	19%	11%	11%

Table 4.29: IEEE 14-bus PMU Placement With Extra PMUs

PMU Locs	1	2	6	10	11	12	15	19	25	29
N_{unob}	1	1	3	3	1	2	1	1	3	2
PCT_{loss}	7%	11%	18%	16%	5%	13%	11%	7%	9%	7%

Table 4.30: IEEE 30-bus PMU Placement

PMU Locs	1	2	5	6	10	11	12	13	14	15	19	20	25	26	29	30
N_{unob}	1	0	0	2	3	1	1	0	0	1	0	0	1	0	0	0
PCT_{loss}	5%	8%	5%	13%	12%	3%	10%	3%	5%	8%	5%	5%	6%	3%	5%	5%

Table 4.31: IEEE 30-bus PMU Placement With Extra PMUs

PMU Locs	2	6	12	19	22	25	27	32	36	39	41	45	46	49	51	52	55
N_{unob}	3	5	3	3	3	3	3	3	4	3	2	5	3	3	2	1	3
PCT_{loss}	5%	8%	10%	5%	6%	6%	5%	6%	6%	5%	8%	5%	5%	8%	5%	5%	5%

Table 4.32: IEEE 57-bus PMU Placement

PMU Loc	2	5	6	12	16	17	18	19	20	22	25	27	31	32
N_{unob}	2	0	2	0	0	0	0	0	0	2	2	3	0	1
PCT_{loss}	3%	3%	5%	7%	3%	3%	4%	3%	3%	4%	4%	3%	3%	4%
PMU Locs	33	36	39	41	43	45	46	49	51	52	55	57		
N_{unob}	0	3	0	1	0	3	3	2	1	3	2	0		
PCT_{loss}	2%	4%	3%	5%	3%	3%	3%	5%	3%	3%	3%	3%		

Table 4.33: IEEE 57-bus PMU Placement With Extra PMUs

PMU Locs	2	3	10	16	21	23	26	27	32	34	40	45	47	50	56	58	64
N_{unob}	2	3	4	4	3	3	2	3	1	2	3	3	4	3	2	4	3
$PCT_{loss}(\%)$	4	4	6	5	7	6	4	5	4	6	5	6	6	4	4	6	5
PMU Locs	70	71	72														
N_{unob}	2	3	2														
$PCT_{loss}(\%)$	3	6	3														

Table 4.34: RTS96 PMU Placement

PMU Locs	2	3	4	6	7	10	14	16	19	21	22	23	24	26	27	28	30
N_{unob}	0	0	0	0	0	1	0	0	0	1	0	2	0	0	0	0	0
$PCT_{loss}(\%)$	2	2	2	2	2	3	2	3	2	4	2	3	2	2	2	3	2
PMU Locs	31	32	34	38	40	42	43	44	45	46	47	48	50	51	52	54	55
N_{unob}	0	0	1	0	0	0	0	0	0	0	2	0	0	0	0	0	0
$PCT_{loss}(\%)$	2	1	2	3	2	3	2	2	3	3	2	3	2	2	2	2	1
PMU Locs	56	58	62	64	67	68	70	71	72	73							
N_{unob}	0	1	0	0	0	0	2	1	0	0							
$PCT_{loss}(\%)$	2	3	2	3	2	3	2	3	2	2							

Table 4.35: RTS96 PMU Placement With Extra PMUs

PMU Locs	1	5	9	12	15	17	21	25	28	34	37	40	45	49	52	56
N_{unob}	1	3	2	3	1	4	3	3	2	2	2	1	2	3	2	5
$PCT_{loss}(\%)$	2	4	2	5	4	4	2	2	2	3	4	3	2	8	2	4
PMU Locs	62	64	68	70	71	76	77	80	85	87	91	94	101	105	110	114
N_{unob}	7	2	2	2	4	3	3	3	2	1	4	5	2	2	4	2
$PCT_{loss}(\%)$	3	2	3	4	2	2	5	5	4	1	2	4	2	4	3	2

Table 4.36: IEEE 118-bus PMU Placement

PMU Locs	1	5	6	9	10	12	13	15	16	17	20	21	25	28	33
N_{unob}	1	1	0	0	0	0	0	0	0	4	0	1	3	2	0
$PCT_{loss}(\%)$	1	3	1	1	1	4	1	3	1	3	1	1	2	1	1
PMU Locs	34	37	40	43	45	48	49	50	52	53	56	57	62	64	67
N_{unob}	1	2	1	0	0	0	1	0	0	0	3	0	1	2	0
$PCT_{loss}(\%)$	2	3	2	1	2	1	6	1	1	1	3	1	2	2	1
PMU Locs	68	70	71	72	76	77	80	84	85	86	87	91	94	97	98
N_{unob}	2	1	1	0	1	2	0	2	0	0	2	3	0	0	0
$PCT_{loss}(\%)$	2	3	2	1	1	4	4	1	3	1	1	1	3	1	1
PMU Locs	99	101	105	107	110	111	112	114	117						
N_{unob}	2	2	0	1	0	0	3	0	0						
$PCT_{loss}(\%)$	1	3	1	2	1	1	1	1	1						

Table 4.37: IEEE 118-bus PMU Placement With Extra PMUs

The following itemized list describes the performance of each system with and without extra PMU. The notation 'BILP' means the performance without extra PMU, and 'Modified' means the performance with extra PMU.

- IEEE 14-bus system:
  - **BILP**: average unobserved bus number: 2.5, average percentage loss: 25%
  - **Modified**: average unobserved bus number: 0.71, average percentage loss: 14%
- IEEE 30-bus system:
  - **BILP**: average unobserved bus number: 1.8, average percentage loss: 10%
  - **Modified**: average unobserved bus number: 0.63, average losing percentage: 6%
- IEEE 57-bus system:
  - **BILP**: average unobserved bus number: 3, average percentage loss: 6%
  - **Modified**: average unobserved bus number: 1.15, average percentage loss: 4%
- IEEE 73-bus system:

  - **BILP**: average unobserved bus number: 2.7, average percentage loss: 5%
  - **Modified**: average unobserved bus number: 0.25, average percentage loss: 2%

- IEEE 118-bus system:

  - **BILP**: average unobserved bus number: 2.6, average percentage loss: 3%
  - **Modified**: average unobserved bus number: 0.85, average percentage loss: 2%

From the summary above, it can be seen that the extra PMU can minimize the damage on the system observability when we consider the loss of a single PMU in a observation period. Similar to the DFS, the experiment of losing PMUs will be implemented to evaluate the performance of the extra PMU for the system observability. The results are presented in the following tables:

Combination set	{2,6}	{6,9}
Unobserved bus Number	8	6

Table 4.38: BILP IEEE 14-bus results

Combination set	{2,10}	{2,7}
Unobserved bus Number	6	5

Table 4.39: BILP With Extra PMUs For IEEE 14-bus results

Combination set	{6,10}	{6,25}	{10,25}	{12,15}	{25,29}
Unobserved bus Number	7	6	6	6	6

Table 4.40: BILP IEEE 30-bus results

Combination	{6,10}	{1,10}	{10,11}	{10,12}	{10,15}	{10,25}	{1,6}	{2,10}	{5,6}	{5,10}	{6,11}	{6,12}
Unobserved	5	4	4	4	4	4	3	3	3	3	3	3

Table 4.41: BILP With Extra PMUs For IEEE 30-bus results

Combination set	{6,41}	{6,32}	{32,41}	{2,6}	{2,41}	{6,12}	{6,19}	{6,22}	{6,25}
Unobserved bus Number	10	9	9	8	8	8	8	8	8
Combination set	{6,27}	{6,36}	{6,45}	{6,46}	{6,52}				
Unobserved bus Number	8	8	8	8	8				

Table 4.42: BILP IEEE 57-bus results

Combination	{27,36}	{27,45}	{27,46}	{27,52}	{36,45}	{36,46}	{36,52}	{45,46}	{45,52}	{46,52}
Unobserved Num	6	6	6	6	6	6	6	6	6	6
Combination	{2,27}	{2,36}	{2,45}	{2,46}	{2,52}	{6,27}	{6,36}	{6,45}	{6,46}	{6,52}
Unobserved Num	5	5	5	5	5	5	5	5	5	5

Table 4.43: BILP With Extra PMUs For IEEE 57-bus results

Combination	{10,16}	{10,23}	{10,47}	{16,21}	{16,47}	{2,10}	{3,10}	{3,16}	{3,47}	{10,21}
Unobserved Num	8	8	8	8	8	7	7	7	7	7
Combination	{10,27}	{10,40}	{10,45}	{10,50}	{10,64}	{10,71}	{16,23}	{16,27}	{16,40}	{16,45}
Unobserved Num	7	7	7	7	7	7	7	7	7	7

Table 4.44: BILP RTS 96 results

Combination	{10,23}	{23,47}	{23,70}	{34,47}	{47,70}	{10,47}	{10,70}	{19,23}	{21,23}	{21,47}
Unobserved Num	4	4	4	4	4	3	3	3	3	3
Combination	{21,70}	{23,34}	{23,58}	{23,71}	{34,70}	{44,47}	{47,58}	{47,71}	{58,70}	{58,71}
Unobserved Num	3	3	3	3	3	3	3	3	3	3

Table 4.45: BILP With Extra PMUs For RTS 96 results

Combination	{56,85}	{56,105}	{85,105}	{105,110}	{17,56}	{17,85}	{17,105}	{49,56}	{56,80}	{56,110}
Unobserved	10	10	10	10	9	9	9	9	9	9
Combination	{80,85}	{80,105}	{85,110}	{5,56}	{5,85}	{5,105}	{12,17}	{12,56}	{12,85}	{12,105}
Unobserved	9	9	9	8	8	8	8	8	8	8

Table 4.46: BILP IEEE-118 bus results

Combination	{17,25}	{17,56}	{17,94}	{17,114}	{17,28}	{17,37}	{17,64}	{17,68}	{17,77}	{17,85}
Unobserved	7	7	7	7	6	6	6	6	6	6
Combination	{17,91}	{17,101}	{17,105}	{25,28}	{25,56}	{25,94}	{25,114}	{56,94}	{56,114}	{91,94}
Unobserved	6	6	6	6	6	6	6	6	6	6

Table 4.47: BILP With Extra PMUs For IEEE-118 bus results

By comparing two different system scenarios after losing 2 PMUs in the same period, it can be seen that the total unobserved bus number is less for the systems with extra PMU than that for the systems without extra PMUs. Combined with the experiment results from DFS in the above subsection, it can be concluded that the extra PMUs increases the robustness of the system observability. It can minimize the damage of the system observability when N-1/N-2 contingency occurs on the system PMUs. Moreover, it can be seen that the total number of PMUs when using either DFS with heuristic method or CPLEX method would not change significantly after adding extra PMU. In other words, the cost for adding all extra PMUs is approximately the same for each method, and the system observability can be improved significantly on each method after inserting all extra PMUs.

4.1.3 The Performance of Modified GA With All Extra PMUs

We will employ the *ga solver* in the Global optimization toolbox in Matlab to solve the PMU placement problem. This toolbox solves mixed integer linear programming problems using the genetic algorithm. The genetic algorithm attempts to minimize a penalty function, if the problem is feasible, instead of maximizing a fitness function. The penalty function is combined with binary tournament selection to select individuals for subsequent generation [7, 87].

As introduced in chapter two, three major steps will be implemented for the modified GA

technique for the PMU placement. In the first step, the total number of PMU from CPLEX will be the linear constraint set for the GA solver. However, the linear constraints can not be set directly in the Matlab GA toolbox. Instead of direct setting, the linear constraint can be realized by using two inequalities. For example, if the equality constraint needs to defined as $ax = b$, it can be realized by defining two inequalities: $ax \geq b$ and $ax \leq b$. The linear equality *ga solver* is default as $A \cdot \mathbf{x} \leq \mathbf{b}$, which does not match the linear inequality for the PMU placement problem. The mismatch problem can be solved by multiplying the negative sign on both side, expressed as: $-A \cdot \mathbf{x} \geq -\mathbf{b}$. Following tables represent the results for different GA results and its corresponding observability values.

Test Systems	**PMU Placements**	**Observability Values**
IEEE 14-bus system	[2,6,7,9];	19;
	[2,8,10,13];	14;
	[2,6,8,9];	17;
	[2,7,10,13];	16;
IEEE 30-bus system	[2,3,6,10,11,12,19,23,25,27];	46;
	[3,5,8,10,11,12,19,23,25,30];	44;
	[1,6,7,10,11,12,18,24,25,29];	43;
	[1,7,10,11,12,18,23,25,28,30];	38;
	[1,5,9,10,12,15,19,25,28,29];	42;
	[1,5,8,9,10,12,18,23,25,29];	39;
	[1,7,8,9,10,12,15,18,25,29];	41;
	[2,4,6,9,10,12,15,18,25,27];	52;
IEEE 57-bus system	[1,6,9,15,18,20,24,25,28,32,36,38,41,47,51,53,57];	64;
	[1,4,9,19,22,25,27,29,32,36,38,41,44,46,50,54,57];	68;
	[1,4,7,9,20,23,27,30,32,36,39,41,44,46,47,50,53];	67;
	[1,4,9,10,20,22,25,27,29,32,36,39,41,45,46,49,54];	68;
	[1,4,6,9,15,20,24,28,31,32,36,38,41,46,50,53,57];	72;
	[1,4,9,20,22,26,29,30,32,36,41,44,46,49,50,53,57];	69;
	[1,4,7,9,13,19,22,25,27,32,36,39,41,44,47,50,53];	67;
	[1,4,9,20,24,25,28,29,32,36,38,39,41,44,46,50,54]	65;

Table 4.48: GA PMU locations for IEEE 14-, 30-, 57-bus system

Test Systems	PMU Placements	Observability Values
RTS 96 system	[2,3,10,16,21,23,26,27,32,34,40,45,47,50,51,55,58,64,70,71];	98;
	[2,3,10,16,21,23,26,27,32,34,40,45,47,50,51,56,58,64,70,71];	100;
	[2,3,10,16,21,23,26,27,32,34,40,45,47,50,56,58,64,70,71,72];	99;
	[2,3,10,16,21,23,26,27,32,34,40,45,47,50,51,55,58,64,69,71];	101;
	[2,3,10,16,21,23,26,27,32,34,40,45,47,50,51,56,58,64,69,71];	103;
	[2,3,10,16,21,23,26,27,32,34,40,45,47,50,56,58,64,69,71,72];	102;
	[1,9,10,19,21,27,32,33,34,40,41,47,50,53,56,59,60,65,67,72];	96;
	[1,9,10,15,18,20,21,26,27,34,37,40,45,50,51,53,55,59,65,69];	95;
IEEE 118-bus system	[2,5,9,11,12,17,21,25,29,34,37,42,45,49,53,56,62,64,68,70,71, 76,79,85,86,89,92,96,100,105,110,114];	161;
	[3,5,9,11,12,17,20,23,29,30,34,37,40,45,49,52,56,62,64,68,71 ,75,77,80,85,86,90,94,102,105,110,115];	163;
	[2,5,9,12,15,17,21,23,28,30,34,35,40,45,49,52,56,62,64,68,71, 75,77,80,85,86,90,94,101,105,110,115];	159;
	[3,5,9,12,15,17,21,25,29,34,37,40,45,49,52,56,62,64,68,70,71, 75,77,80,85,86,90,94,101,105,110,114];	164;
	[2,5,10,12,13,17,20,23,26,28,34,37,40,45,49,52,56,62,63,68,71, 75,77,80,85,86,90,94,102,105,110,114];	156;
	[3,5,10,11,12,17,20,23,29,30,34,37,40,45,49,52,56,62,64,68,71, 75,77,80,85,86,91,94,102,105,110,115];	162;
	[3,5,9,11,12,17,21,23,26,28,34,37,40,45,49,52,56,62,64,68,71, 75,77,80,85,87,91,94,101,105,110,114];	160;
	[3,5,10,11,12,17,21,24,25,29,34,37,41,45,49,53,56,62,64,68,71, 75,77,80,85,87,91,94,101,105,110,114]	157;

Table 4.49: GA PMU locations for RTS-96 and IEEE-118 system

It can be seen that the GA-based PMU placement algorithm produces more than one optimal solution at the same cost, which can potentially apply to multiple applications with various constraints. In our research, the additional constraint for the optimal solution is the system observability value, which is the value shown on the rightmost column in the following tables. The solution having the largest observability value will be selected as the optimal solution in

our scenario. The following table shows the solutions and its corresponding largest observability value within limit times for different systems

Test System	**PMU placement results**	**Observability**
IEEE 14-bus system	[2,6,7,9];	19
IEEE 30-bus system	[2,4,6,9,10,12,15,18,25,27];	52
IEEE 57-bus system	[1,4,6,9,15,20,24,28,31,32,36,38,41,46,50,53,57];	74
RTS 96 system	[2,3,10,16,21,23,26,27,32,34,40,45,47,50,51,56,58,64,69,71];	103
IEEE 118-bus system	[3,5,9,12,15,17,21,25,29,34,37,40,45,49,52,56,62,64,68,70, 71,75,77,80,85,86,90,94,101,105,110,114];	164

Table 4.50: The PMU placements having largest system observability

Table 4.25 provides the result by using the BILP generated from CPLEX toolbox. The observability values for that results can be evaluated shown as the following list:

- IEEE 14-bus: 19
- IEEE 30-bus: 46
- IEEE 57-bus: 64
- RTS 96: 99
- IEEE 118-bus: 163

By comparing with the observability values from modified GA method shown in Table 4.50, it can be seen that the modified GA will have better performance without extra PMU for the 5 systems in terms of the observability values.

According to the extra PMU location provided in Table 4.2, the extra PMU will be added onto the placement results generated by modified GA PMU placement method. The result will be presented in the following table. From the observability values, it can be seen that the robustness on the modified GA algorithm remaining true as well.

Test System	PMU placement results	Observability
IEEE 14-bus system	[2,6,7,8,9,12,14];	27
IEEE 30-bus system	[2,4,5,6,9,10,11,12,13,14,19,20,24,25,26,27,30];	69
IEEE 57-bus system	[1,4,5,6,9,15,16,17,18,19,20,24,25,28,31,32,33, 36,38,41,43,45,46,50,52,53,55,57];	104
RTS 96 system	[2,3,4,6,7,10,14,16,19,21,22,23,24,26,27,28,30,31,32, 34,38,40,42,43,44,45,46,47,48,50,51,52,54,55,56,58, 62,64,67,68,69,70,71,72,73];	184
IEEE 118-bus system	[3,5,6,9,10,12,13,15,16,17,20,21,25,28,29,33,34,37,40,43, 45,48,49,50,52,53,56,57,62,64,67,68,70,71,72,75,77,80,84, 85,86,87,90,94,97,98,99,101,105,107,110,111,112,114,117];	228

Table 4.51: Modified GA-Based PMU Placement Algorithm With Extra PMUs.

4.1.4 Performance of Gamma Values

[69] introduces a similar idea in terms of defining the locations for the extra PMU, as our *Gamma* value. Instead of using the electrical distance as the measurement, [69] uses the resistance distance to evaluate the average nodal distance. Based on the resistance distance, [69] proposes to select the bus having the minimum average resistance value to be the bus where the PMU is installed. The test systems of [69] include IEEE 14-bus system, 30-bus system, and 57-bus system only. Fig. 4.2 shows the average resistance distance from [69]. In the figure, the red dots represent the average resistance values, and the blue dots represent the buses where the PMU should be installed.

In order to evaluate the performance of our extra PMU installation method, two comparisons are made between the results of [69] PMU placement algorithm and the proposed PMU placement algorithm. Firstly, extra PMUs will be placed based on the results from the modified GA PMU placement algorithm, which has the best performance among three PMU placement methods introduced in our research. The total number of PMUs is shown in Table 4.51, which is the total with the extra PMU installed. According to the results provided by [69], Table 4.52 provides the total PMU numbers of both algorithms for the common used testing systems, which are IEEE 14-bus system, IEEE 30-bus system, IEEE 57-bus system, and IEEE 118-bus system. It can be seen that our total number of PMUs with extra PMU is less than the numbers in [69]

for IEEE 57-bus system and IEEE 118-bus system.

Testing Systems	Number of PMU [69]	**Number of PMU Using Our Method**
IEEE 14-bus system	7	**7**
IEEE 30-bus system	17	**17**
IEEE 57-bus system	35	**28**
IEEE 118-bus system	93	**55**

Table 4.52: Minimum number of PMUs needed by the method in [69] and by our proposed method

The second comparison is at the aspect of the measurement redundancy of each bus and the total measurement redundancy of the system. The measurement redundancy means the number of times the bus is being observed by the measurements being collected [74]. Since the PMU measurement is the only measurement used by the system, the measurement redundancy values are the essential coefficients to evaluate the system observability via the PMU placement algorithm at the initial stage. Since the measurement redundancy values cannot be evaluated without the actual locations of the PMUs, only three systems are included according to the information provided by [69]. The three systems are IEEE 14-bus system, IEEE 30-bus system, and IEEE 57-bus system, which are shown in Fig 4.2. In the figure, blue dots denotes the locations of the PMUs evaluated from the method in [69].

Starting with the IEEE 14-bus test system, Fig. 4.3 provides a comparison of the results between the two different methods. Shown as Fig. 4.3, the orange X markers denote the measurement redundancy values of each bus for the proposed PMU placement algorithm, which is labeled ED results, i.e. electrical distance results. The blue $+$ markers denote the values of each bus for the algorithm proposed in [69], which is called RD result standing for the resistance distance results. It can be seen that the proposed algorithm has better performance on bus 4, 5, 7, 8, 9 and the method from [69] has better performance on bus 6 and 14. In addition to that, the overall measurement redundancy value is 27 for the proposed method, and 21 for the other. Generally speaking, the proposed method has better performance.

The performances of the other two test systems are slightly different than that of the IEEE 14-bus system, which are shown in Fig. 4.4. We use the same denotation of the measurement redundancy value in two different scenarios. We can see in Fig. 4.4 that the number of buses

having larger measurement redundancy values are approximately the same in either IEEE 30-bus test system or IEEE 57-bus test system. The total system measurement redundancy value leads to the same conclusion. For IEEE 30-bus case, the system value is 69 when using the electrical distance measurement and the system value is 55 when using the resistance distance measurement value. For IEEE 57-bus case, the system value is 104 for the electrical distance measurement while that for resistance distance measurement is 114.

By analyzing the above performance results, it could be concluded that using electrical distance as measurement requires fewer PMUs than using resistance distance for the purpose of adding extra PMU to robust the system observability.

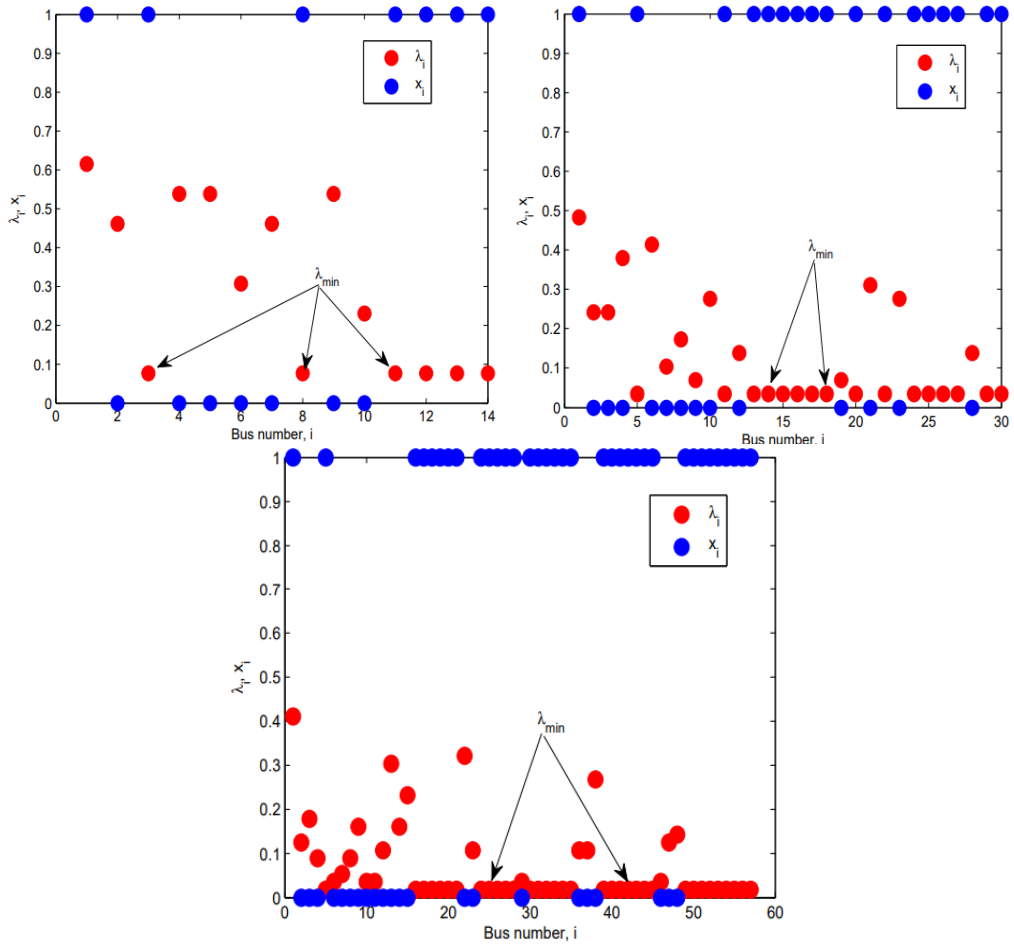


Figure 4.2: IEEE 14-bus, IEEE 30-bus, IEEE 57-bus Average Resistance Distance [69]

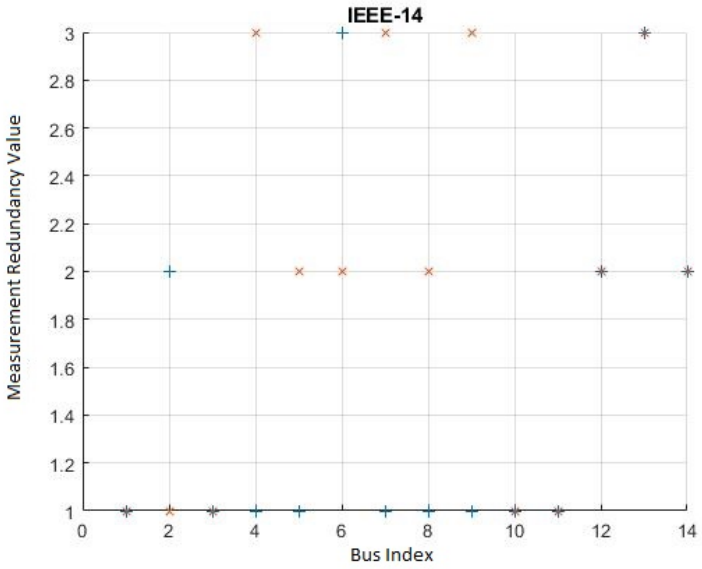
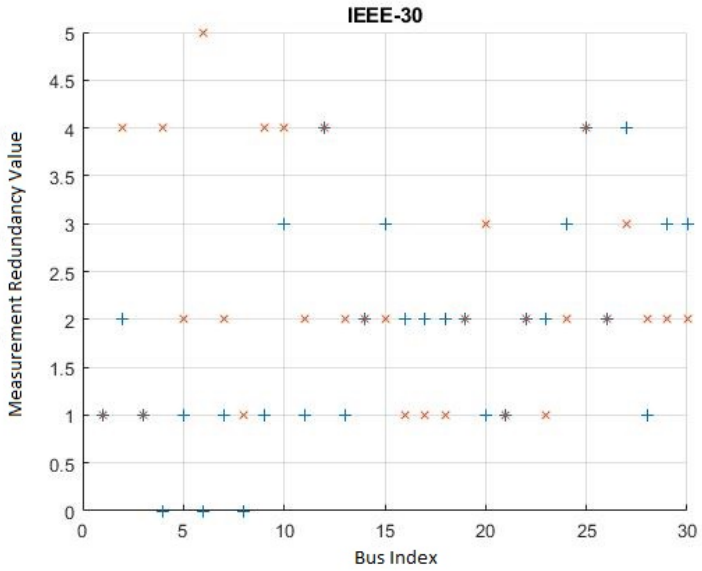
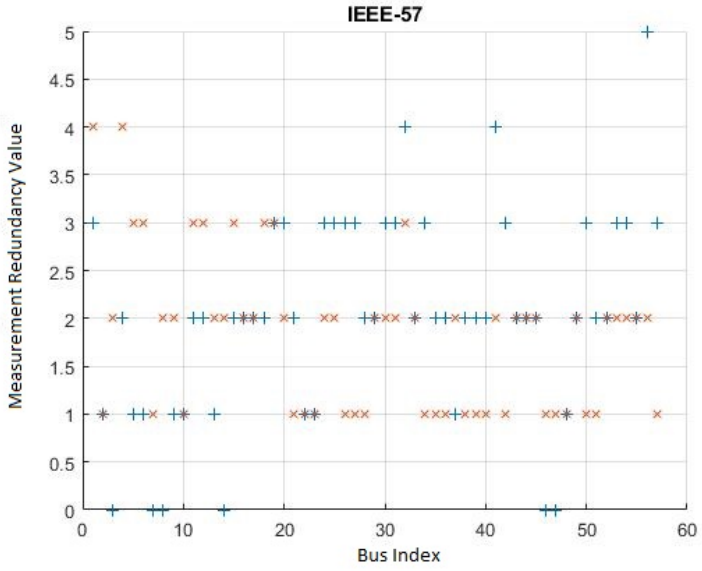


Figure 4.3: The Measurement Redundancy Values Comparison



(a) IEEE 30-bus Test System



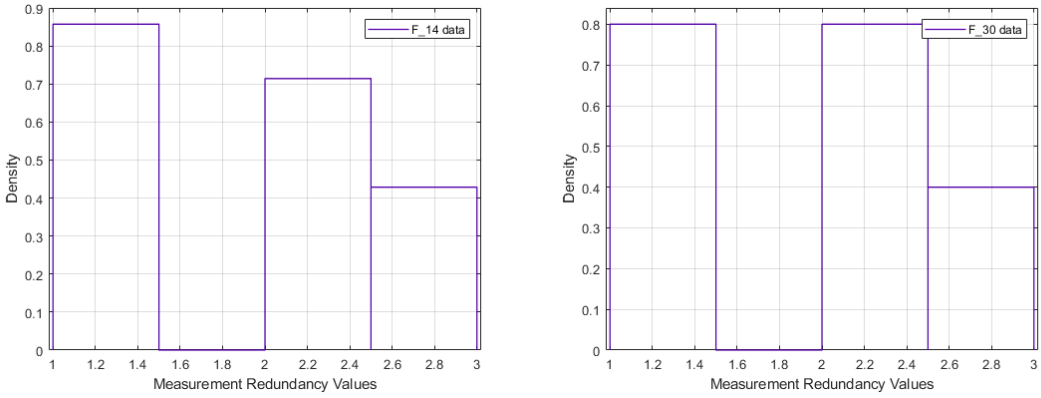
(b) IEEE 57-bus Test System

Figure 4.4: Measurement Redundancy Values

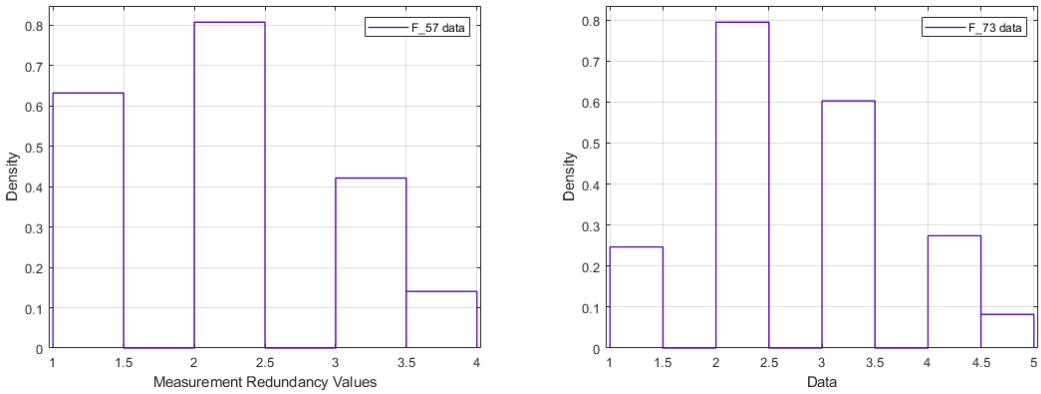
4.2 Performance Evaluation of Proposed Partitioning Method

4.2.1 Performance Evaluation of Measurement Redundancy Index

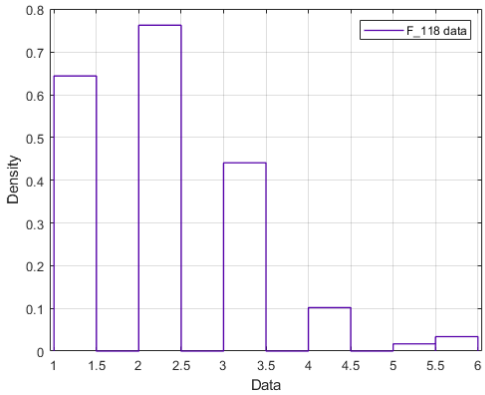
As we proposed in the Ch. 3, the mean value of the proposed index is set as $\mu = \log 3$ as the optimal value is set as $K = 3$. A probability density function is employed to verify this setting. By using the `distribution fitter` function in Matlab, the density properties are generated based on the measurement redundancy value of each bus. Fig 4.5 presents the density properties of 5 test systems. The figure presents all possible measurement redundancy values for 5 systems with their corresponding density. For IEEE 14-bus and IEEE 30-bus, they have fewer possibilities because they have relatively low total number of buses. It leads to the wide bins histograms as shown in the figure. It means that the possible measurement redundancy values of these two systems are relatively concentrated. The width of the bins would become narrower when the system becomes larger, which means that it has more possible values than that for smaller system.



(a) Left: IEEE 14-bus System
Right: IEEE 30-bus System



(b) Left: IEEE 57-bus System
Right: RTS 96 System



(c) IEEE 118-bus System

Figure 4.5: Test Systems Measurement Redundancy Values Density

Using the density property of each system in Fig. 4.5, it can be seen that both systems have a concentration around the measurement redundancy value of 2 except the smallest test system IEEE 14-bus case. One can conclude that the neutral value of the measurement redundancy would be 2 for the most cases, with or without applying the proposed index to the partitioning method. The expectation of having one extra measurement of each bus would giving the extra constraint for the partitioning method. In order to record the measurement redundancy distribution after applying the partitioning method, we implement the proposed method and setting used in [19] without any kinds of modification to form the base case for performance evaluation of the proposed index.

In terms of the base-line simulation setting, the indexes are ECI, BCCI, CCI, CSI, and the user-defined relative importance for each index is equally important denoting as $scalar = [1,1,1,1]$. Thus, the fitness score function can be denoted as:

$$f = ECI \cdot BCCI \cdot CCI \cdot CSI \cdot CC \tag{4.10}$$

where CC is connectivity coefficient with constant setting as 1. Besides this setting, values of $AMRV$ are needed as well. The $AMRV$ value of cluster i can be evaluated by using the following expression:

$$AMRV_k = \frac{\sum_{j \in C_i} f_j}{n_{c_i}} \tag{4.11}$$

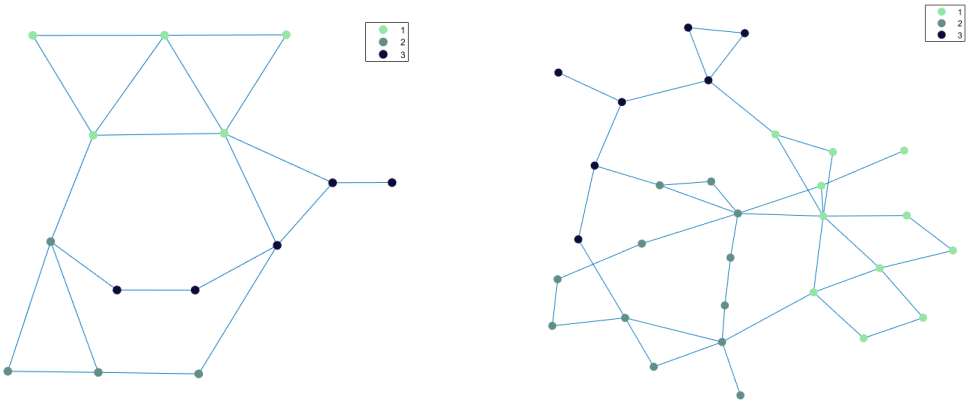
where n_{c_k} is the number of buses of cluster k, j is one of the bus within the cluster a and the scalar value f_j can be evaluated as [66]:

$$f_j = \sum_i a_{ij} \cdot x_j, \; i,j = [1, num_{buses}] \tag{4.12}$$

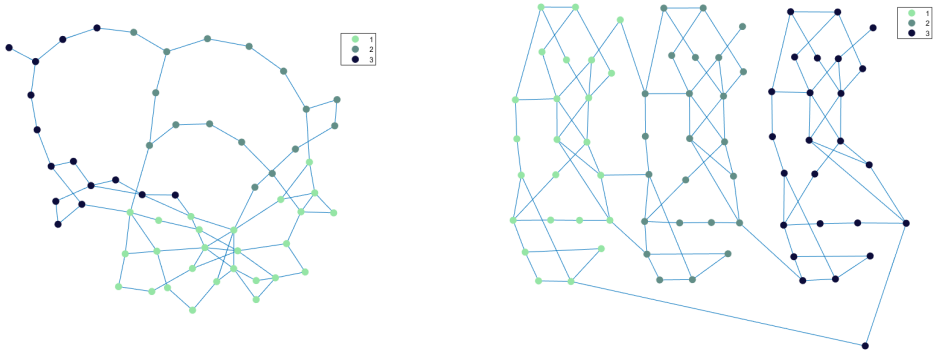
where scalar value x_j either equal to 1 or equal to 0, a_{ij} is the (i,j) entry of matrix A, and num_{bus} is the largest bus index in cluster k. As we proposed in Ch. 3, the value of x_j would not be counted if the bus is outside of the cluster. The base-line simulation result without having the proposed index will be presented in two kinds of form. Table 4.5 records the number of clusters, the number of buses in each cluster, and the $AMRV$ value for each cluster after applying the partitioning method. Fig. 4.6 shows the geographic results after the partitioning method proposed in [19].

Testing Systems	Partitioning Results	$AMRV_i$
IEEE 14-bus system	Clusters: 3, [5, 4, 5]	C_1: 1.8, C_2: 2.75, C_3: 1
IEEE 30-bus system	Clusters: 3, [11, 12, 7]	C_1: 1.45, C_2: 2, C_3: 2
IEEE 57-bus system	Clusters: 3, [26, 16, 15]	C_1: 2, C_2: 2.125, C_3: 2
RTS 96 system	Clusters: 3, [24, 24, 25]	C_1: 2.67, C_2: 2.42, C_3: 2.64
IEEE 118-bus system	Clusters: 3, [45, 43, 30]	C_1: 1.95, C_2: 2.21, C_3: 2.13

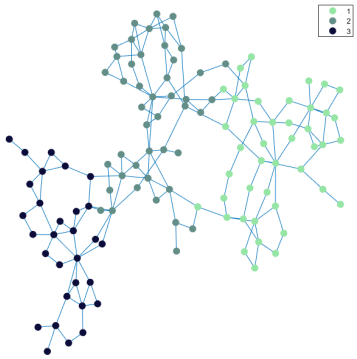
Table 4.53: Records of $AMRV_i$ for The Method with Indices ECI, BCCI, CCI, CSI, CC.



(a) Left: IEEE 14-bus System
Right: IEEE 30-bus System



(b) Left: IEEE 57-bus System
RTS 96 System



(c) IEEE 118-bus System

Figure 4.6: Clustering Results with Indices ECI, BCCI, CCI, CSI, CC

From Table 4.53, the *AMRV* value of two clusters having the same buses included would differ from each other. Besides, cluster having fewer buses included would have larger *AMRV* value than the cluster having more buses included. It means that the measurements for the observability are not balanced distributed while using the partitioning method without the proposed observability related index.

The MRI is inserted after the verification of the necessity, according to the discussion illustrated in Ch. 3, the MRI can be expressed as:

$$MRI = \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{(\log(AMRV)-\log 3)^2}{2(w\cdot\log(n_{bus}))^2}} \tag{4.13}$$

The fitness score function at this stage can be expressed as:

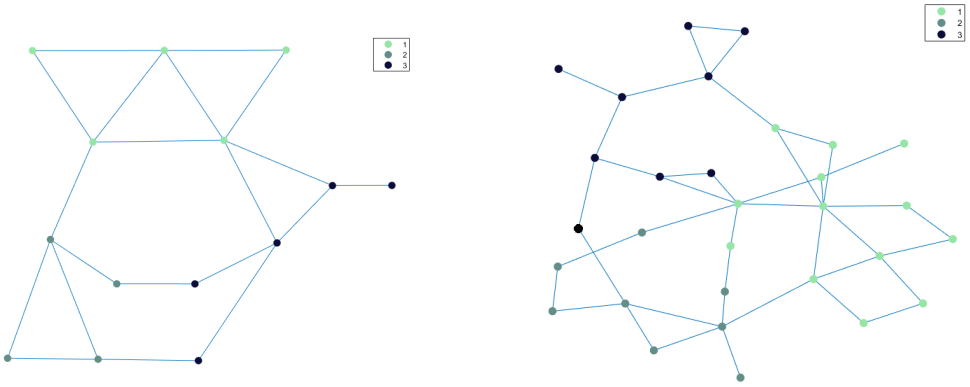
$$f = ECI^{\alpha} \cdot BCCI^{\beta} \cdot CCI^{\gamma} \cdot CSI^{\zeta} \cdot MRI^{\lambda} \cdot CC \tag{4.14}$$

Few coefficients could affects the simulation result, which are the w value of the standard deviation in Eq. 4.10, and the relative importance factors $scalars = [\alpha, \beta, \gamma, \zeta, \lambda]$. The search of the optimal values starts from $w = 0.05$ and $scalars = [1, 1, 1, 1, 1]$.

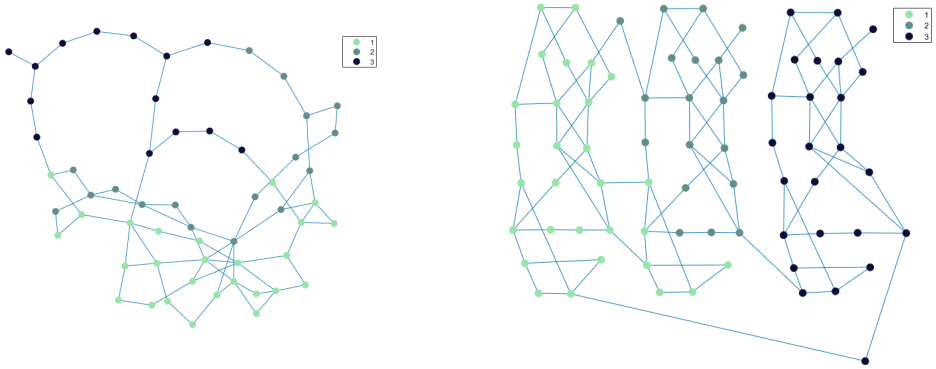
Firstly, the presented performance will be the improvement on AMRV values after inserting MRI into the partitioning method. In order to present the baseline performance, the partitioning method is implemented after applying the DFS heuristic PMU placement technique. Table 4.54 presents the performance of the first attempt after inserting the MRI. Fig. 4.7 presents the clustering geographic results after inserting the MRI.

Testing Systems	Partitioning Results	$AMRV_i$
IEEE 14-bus system	Clusters: 3, [5, 4, 5]	C_1: 1.8, C_2: 2, C_3: 1.6
IEEE 30-bus system	Clusters: 3, [13, 9, 8]	C_1: 1.77, C_2: 1.89, C_3: 1.75
IEEE 57-bus system	Clusters: 3, [25, 18, 14]	C_1: 2, C_2: 2.11, C_3: 2
RTS 96 system	Clusters: 3, [29, 19, 25]	C_1: 2.55 , C_2: 2.52, C_3: 2.64
IEEE 118-bus system	Clusters: 3, [42, 47, 29]	C_1: 2.10, C_2: 2.09, C_3: 2.10

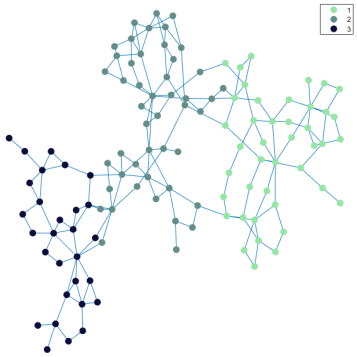
Table 4.54: Records of $AMRV_i$ for Proposed Method with Equal Importance.



(a) Left: IEEE 14-bus System
Right: IEEE 30-bus System



(b) Left: IEEE 57-bus
Right: RTS 96 System



(c) IEEE 118-bus System

Figure 4.7: Clustering Results After Inserting MRI and Using Equal Importance

A comparison is made in Table 4.55, which shows the $AMRV_i$ value of each cluster for 5 different test systems after inserting the proposed MRI into the partitioning method. Table 4.55 employs the $diff$ value, which is the value between the cluster having the largest $AMRV$ value and the cluster having the smallest $AMRV$ value. It is obvious that the $diff$ value has been minimized as we proposed in Ch. 3 for both test systems. The most significant case is IEEE 118-bus system. The $AMRV$ values for 3 clusters are nearly equal to each other, which is 2.10. The $diff$ value is only 0.01, which has 96% improvement compared with the results before inserting the proposed MRI. Through the improvement in terms of $AMRV$ values, it can be seen that the observability of each cluster can be balanced distributed by inserting the proposed MRI index into the partitioning method.

Testing Systems	$AMRV_i$ Without MRI	$AMRV_i$ With MRI
IEEE 14-bus system	[C_1: 1.8, C_2: 2.75, C_3: 1] $diff = 1.75$	[C_1: 1.8, C_2: 2, C_3: 1.6] $diff = 0.4$
IEEE 30-bus system	[C_1: 1.75, C_2: 2, C_3: 2] $diff = 0.25$	[C_1: 1.77, C_2: 1.89, C_3: 1.75] $diff = 0.24$
IEEE 57-bus system	[C_1: 2, C_2: 2.125, C_3: 2] $diff = 0.125$	[C_1: 2, C_2: 2.11, C_3: 2] $diff = 0.11$
RTS 96 system	[C_1: 2.67, C_2: 2.42, C_3: 2.64] $diff = 0.25$	[C_1: 2.55 , C_2: 2.52, C_3: 2.64] $diff = 0.12$
IEEE 118-bus system	[C_1: 1.95, C_2: 2.21, C_3: 2.13] $diff = 0.26$	[C_1: 2.10, C_2: 2.09, C_3: 2.10] $diff = 0.01$

Table 4.55: Records of $AMRV_i$ for Proposed Method With Equal Importance.

In order to have a detailed visual on the results after inserting the MRI, a system drawing tool is employed to present the performance, which is developed by Monash University named `Steady-state AC Network Visualization Tool`. By using this tool, the partitioning results before and after inserting the MRI index are present, which are shown in Figs. 4.8 to 4.12. These figures obtain the same partitioning results as Fig 4.6 and Fig. 4.7, in which the blue lines represent the clustering boundaries by using the partitioning method before inserting the MRI index, and the green lines represent the clustering boundaries by using the partitioning method after inserting the MRI index. Besides the boundaries, these figures also shows

the PMUs locations specifically, which are marked as red circles. The following is an itemized summary of the information presented in each figure:

- **IEEE 14-bus System:**
  - *Original Partitioning Method [19]:* There are **3** clusters in total. Cluster 1 has buses 1-5. It has 5 buses in total with 2 PMUs. Cluster 2 has bus 6 and buses 12-14. It has 4 buses in total with 3 PMUs. Cluster 3 has buses 7, 8 and buses 9-11. It has 5 buses with 2 PMUs.
  - *Proposed Partitioning Method:* There are **3** clusters in total. Cluster 1 has buses 1-5. It has 5 buses in total with 2 PMUs. Cluster 2 has bus 6 and buses 11-13. It has 4 buses in total with 2 PMUs. Cluster 3 has buses 7-10 and bus 14. It has 5 buses in total with 3 PMUs.
  - *Comparison between 2 methods:* Bus 14 has been moved from original cluster 2 to proposed cluster 3. Bus 11 has been moved from original cluster 3 to proposed cluster 2. Cluster 1 has the identical results between two methods. The PMUs number of proposed cluster 2 has been decreased, and the PMUs number of proposed cluster 3 has been increased.
- **IEEE 30-bus System:**
  - *Original Partitioning Method [19]:* There are **3** clusters in total. Cluster 1 has buses 1-9, bus 11, and bus 28. It has 11 buses in total with 4 PMUs. Cluster 2 has bus 10 and buses 12-22. It has 12 buses in total with 5 PMUs. Cluster 3 has buses 23-27, and buses 29, 30. It has 7 buses with 4 PMUs.
  - *Proposed Partitioning Method:* There are **3** clusters in total. Cluster 1 has buses 1-11, bus 17 and bus 28. It has 13 buses in total with 5 PMUs. Cluster 2 has buses 12-16, buses 18-20 and bus 23. It has 9 buses in total with 5 PMUs. Cluster 3 has buses 21, 22, buses 24-27 and buses 29, 30. It has 8 buses in total with 4 PMUs.
  - *Comparison between 2 methods:* Buses 10, 17 have been moved from original cluster 2 to proposed cluster 1. Buses 21, 22 have been moved from original cluster 2 to proposed cluster 3. The number of PMUs of cluster 1 has been increased, and the number of PMUs of cluster 3 has been decreased.
- **IEEE 57-bus System:**

- *Original Partitioning Method [19]:* There are **3** clusters in total. Cluster 1 has buses 1-17, bus 38, buses 44-51. It has 26 buses with 11 PMUs. Cluster 2 has buses 18-29, buses 52-55. It has 16 buses with 10 PMUs. Cluster 3 has buses 30-37, buses 39-43, and buses 56, 57. It has 15 buses with 10 PMUs.
- *Proposed Partitioning Method:* There are **3** clusters in total. Cluster 1 has buses 1-6, bus 10, buses 12-18, buses 37-39, buses 44-51. It has 26 buses with 11 PMUs. Cluster 2 has buses 7-9, bus 11, buses 27-29, bus 40-43, and buses 52-57. It has 17 buses with 10 PMUs. Cluster 3 has buses 19-26, buses 30-36. It has 14 buses with 10 PMUs.
- *Comparison between 2 methods:* Buses 7-9 and bus 11 have been moved from original cluster 1 to proposed cluster 2. Bus 18 has been moved from original cluster 2 to proposed cluster 1. Buses 19-26 have been original cluster 1 to proposed cluster 3. Buses 37, 39 have been moved from original cluster 1 to proposed cluster 1. Buses 40-43 have been moved from original cluster 3 to proposed cluster 2. Buses 56, 57 have been moved from original cluster 1 to proposed cluster 3. The number of PMUs of all clusters have not been changed, but the partitioning results has changed much.

- **RTS 96 System:**
  - *Original Partitioning Method [19]:* There are **3** clusters in total. Cluster 1 has buses 1-24. It has 24 buses with 15 PMUs. Cluster 2 has buses 25-48. It has 24 buses with 14 PMUs. Cluster 3 has buses 49-73. It has 25 buses with 16 PMUs.
  - *Proposed Partitioning Method:* There are **3** clusters in total. Cluster 1 has buses 1-6, buses 8-24, buses 39-42, and buses 45, 46. It has 29 buses with 18 PMUs. Cluster 2 has bus 7, buses 25-38, buses 43, 44, and buses 47, 48. It has 19 buses with 11 PMUs. Cluster 3 has buses 49-73. It has 25 buses with 16 PMUs.
  - *Comparison between 2 methods:* Buses 7 from original cluster 1 has been moved to proposed cluster 2. Buses 39-42, and buses 43, 44, 47, 48 have been moved from original cluster 2 to proposed cluster 1. Cluster 3 in two methods is identical. The number of PMUs has been increased in cluster 1, and that number of cluster 2 has been decreased.

- **IEEE 118-bus System:**

- *Original Partitioning Method [19]:* There are **3** clusters in total. Cluster 1 has buses 1-41, buses 113-115, and bus 117. It has 45 buses with 21 PMUs. Cluster 2 has buses 42-81, bus 97, bus 116 and bus 118. It has 43 buses with 20 PMUs. Cluster 3 has buses 82-96, and buses 98-112. It has 30 buses with 18 PMUs.
- *Proposed Partitioning Method:* There are **3** clusters in total. Cluster 1 has buses 1-23, buses 25-38, bus 43, buses 113-115, bus 117. It has 42 buses with 22 PMUs. Cluster 2 has bus 24, buses 39-42, buses 44-81, buses 97, 98 and buses 116, 118. It has 44 buses with 21 PMUs. Cluster 3 has buses 82-96 and buses 99-112. It has 28 buses with 16 PMUs.
- *Comparison between 2 methods:* Buses 24, 39, 40, 41 of original cluster have been moved to proposed cluster 2. Bus 43 of original cluster 2 has been moved to proposed cluster 1. Bus 98 of original cluster 3 has been moved to proposed cluster 1. The numbers of PMUs of cluster 1 and cluster 2 have been increased, and that of cluster 3 has been decreased.

As the detailed demonstrations above shows, there are many movements on the buses between two different results which are caused by the two partitioning methods. The total number of PMUs remains the same for two methods, and the topology of each system for different method remains as well. It means that the global connectivity and observability stays the same for two partitioning methods. These movements are important in terms of sub-system observability, which is represented by the measurement redundancy value evaluated by Eq. 3.7 and Eq. 3.8. The measurement redundancy value depends on the PMU measurements in our first stage simulation. At this point, the distribution of PMU needs to be balanced on each sub-system. The movements focus on removing the ”useless” PMU measurements from one sub-system to relocate them into the other sub-system that needs them.

We can take the IEEE 30-bus system for example. As can be seen in Table 4.55, cluster 1 has lower measurement redundancy value than cluster 2 and cluster 3 in original partitioning method. With the purpose of obtaining a balance, the initial idea would be increase the AMRV of cluster 1 and decrease the AMRVs of cluster 2 and cluster 3 simultaneously so that the balance of the sub-system observability can be achieved. As seen in Fig. 4.9, the original cluster 1 has 11 buses with only 4 PMUs, but cluster 2 has 6 PMUs for 12 buses and cluster 3 has 4 PMUs for 7 buses. At this point, the hypothesis of the implementation would decrease the number of PMUs in the clusters having high AMRV or increase the number of unobserved bused in those clusters.

By inserting the MRI index, the partitioning method realizes the averaging measurements purpose by moving the buses between clusters. The method moves two buses, bus 10 and bus 17 from old cluster 2 to the new cluster 1 in order to increase the AMRV. Meanwhile, 2 unobserved buses 21, 22 have been moved from the old cluster 2 to new cluster 3 in order to decrease the old cluster 3 AMRV. The new AMRVs are shown in Table 4.7, which shows that the measurement has been obviously averaged.

Another good example would be the RTS 96 system. The original partitioning method almost gives average cluster size for each cluster on this system. However, it can be seen from Table 4.55 that cluster 1 and cluster 2 have 0.25 AMRV difference, who have same cluster size and approximately same PMUs. The difference is caused by the PMUs locations and the topology characteristics, which can be seen from Fig 4.6, 4.7 or Fig. 4.11. The proposed method helps on the averaging implementation by contributing more unobserved buses from old cluster 2 to the old cluster 1 under ensuring sub-system observability condition. The movement results in an improvement of the AMRV as well, which can be seen in Table 4.55.

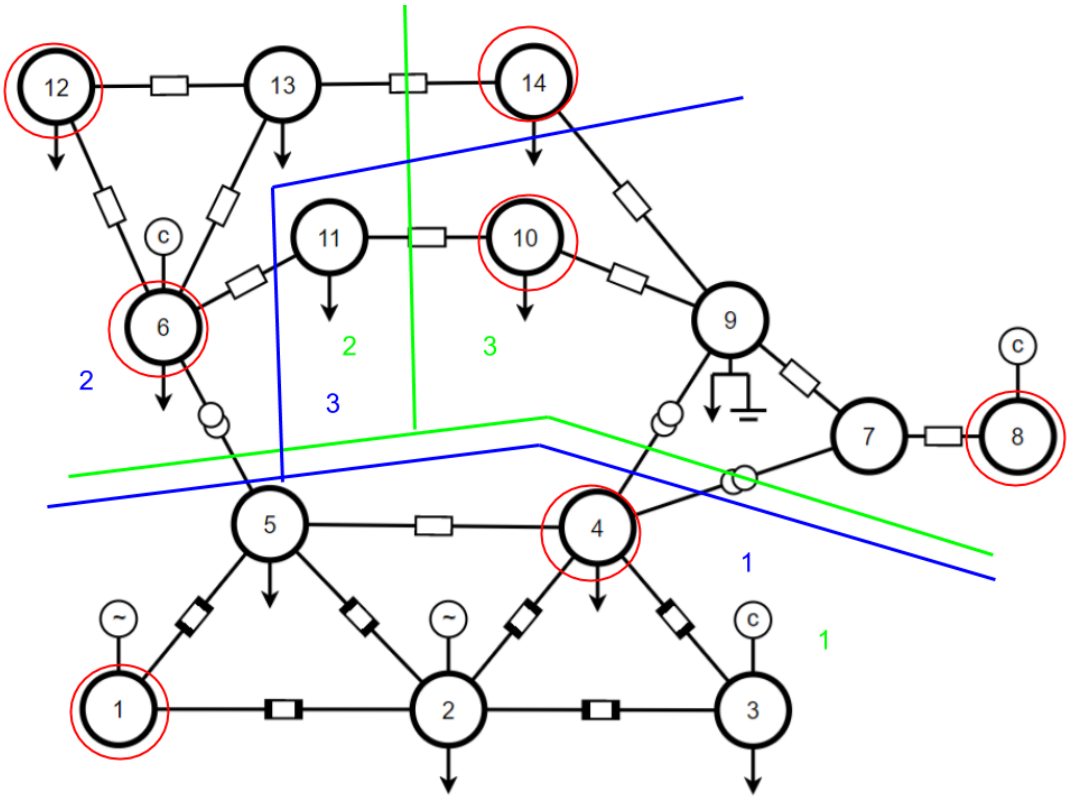


Figure 4.8: IEEE 14-bus System

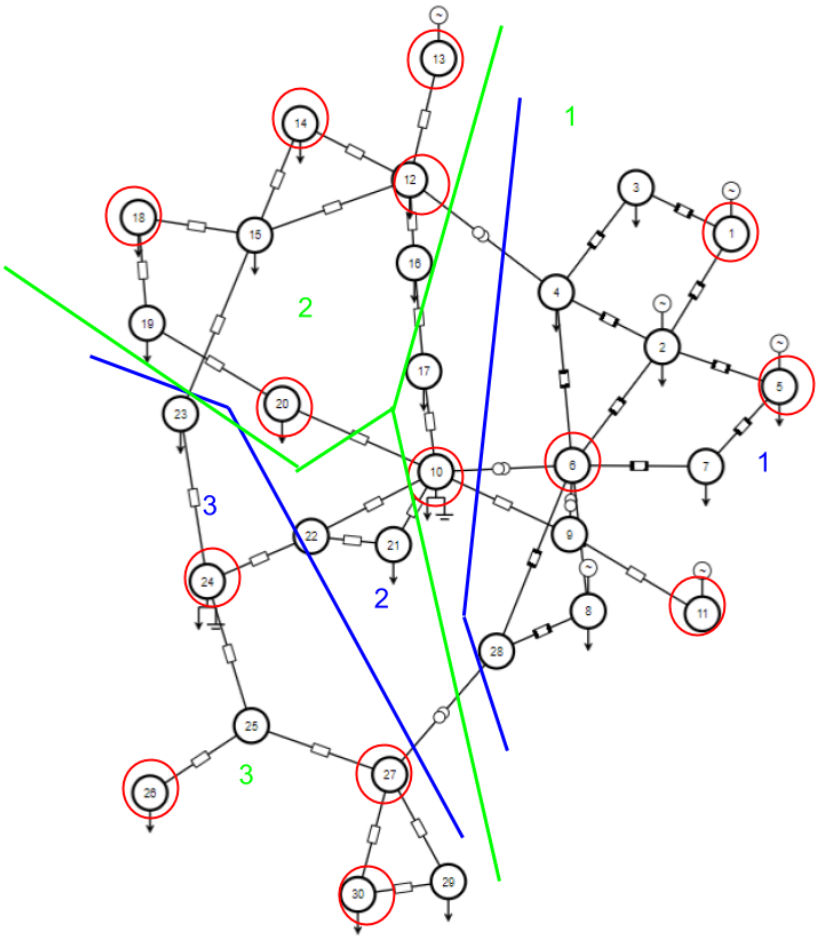


Figure 4.9: IEEE 30-bus System

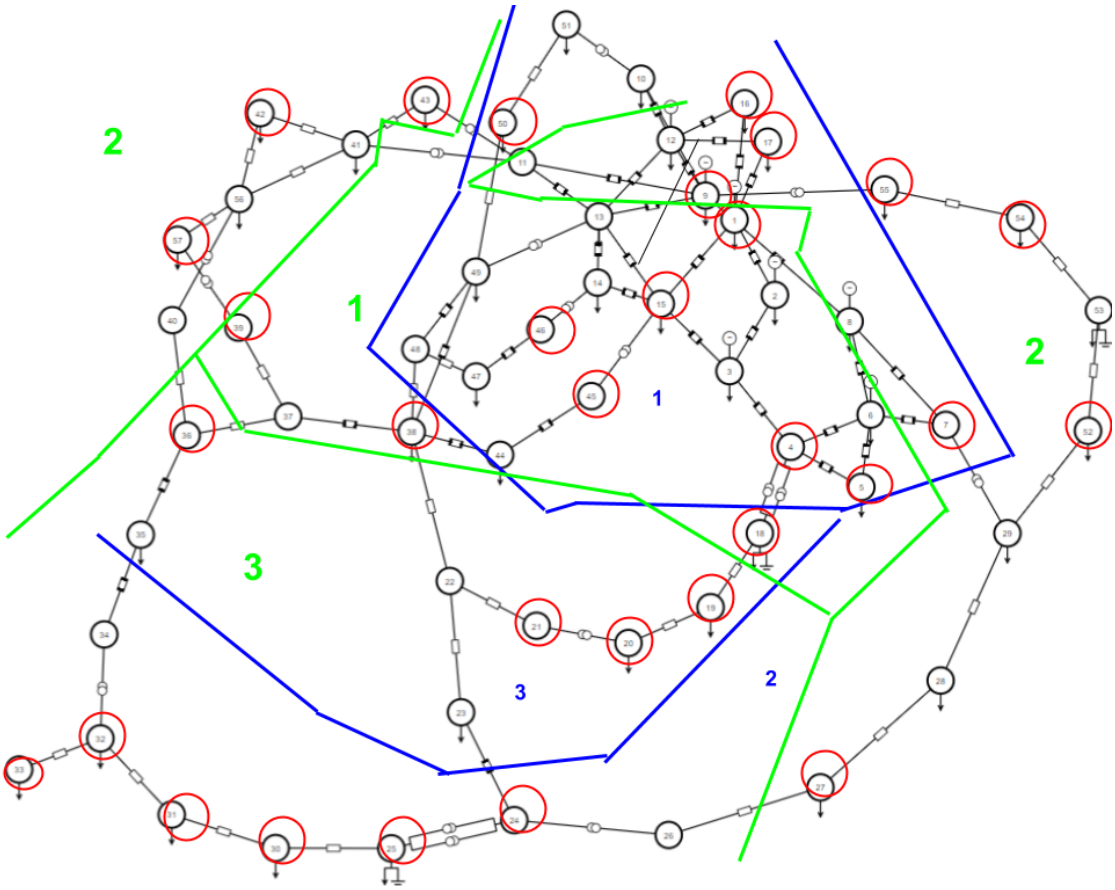


Figure 4.10: IEEE 57-bus System

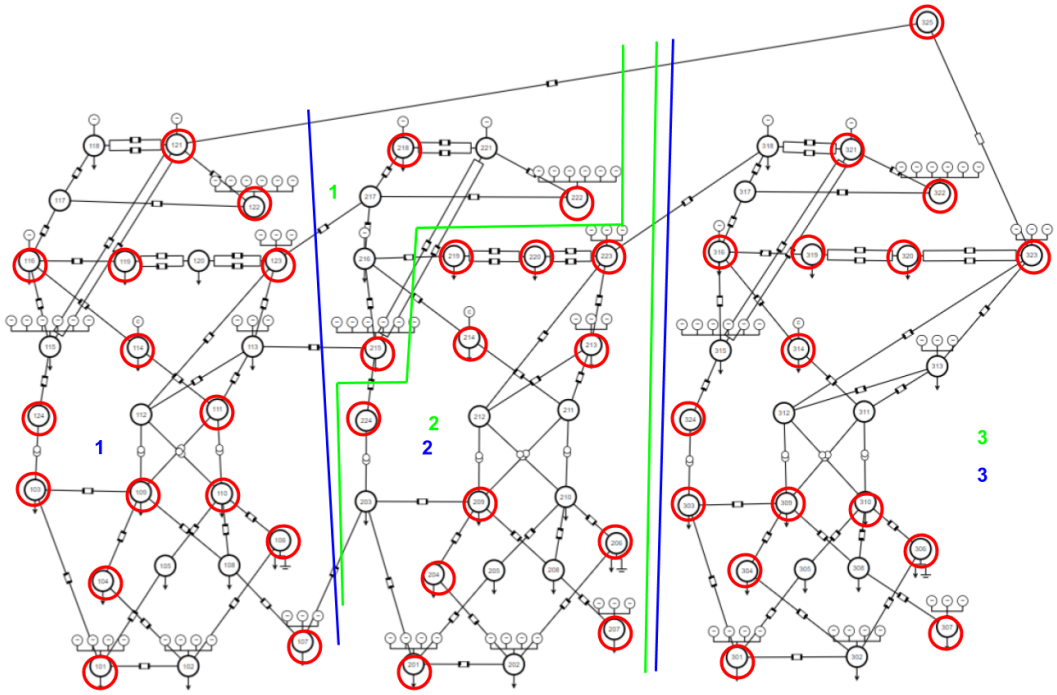


Figure 4.11: RTS 96 System

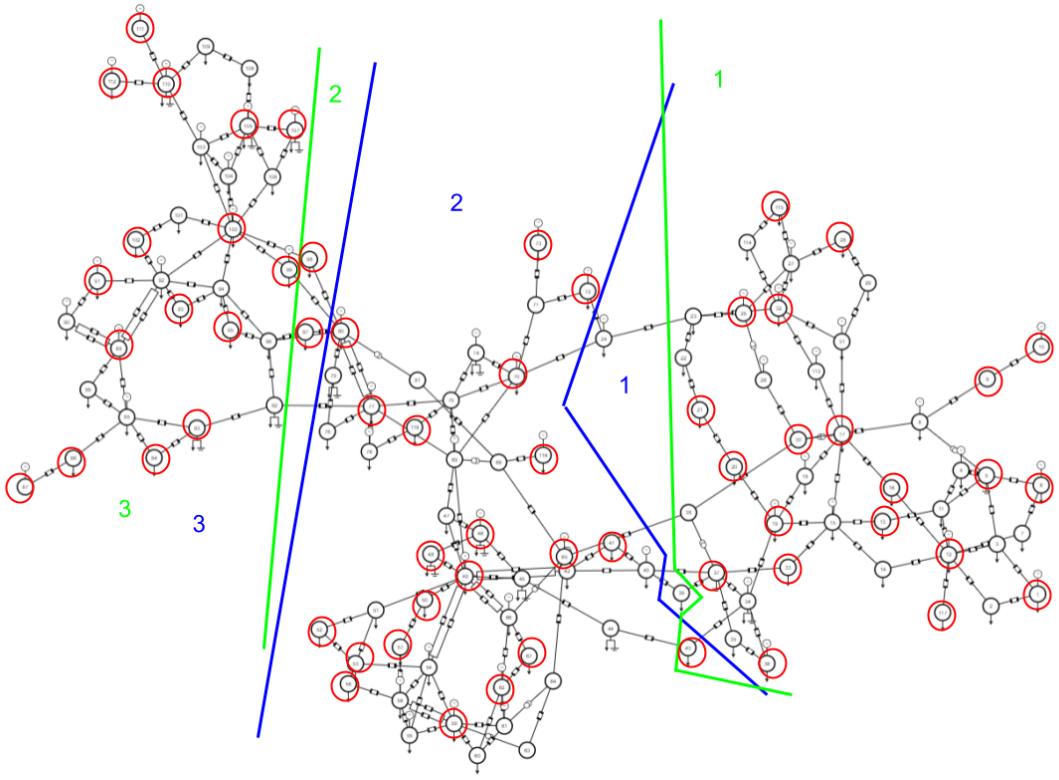


Figure 4.12: IEEE 118-bus System

In addition to the data shown in Table 4.7, we would like to compare quantitavily the $diff$, which represents the property of the observability, with the overall fitness score f from the partitioning method. As IEEE 118-bus system has the best performance result based on the Table 4.7, this system will be used as the example to illustrate how the use of our partitioning method could improve the performance of observability distribution onto each cluster. For data comparison purposes, we randomly extract samples during the process of the partitioning method. The samples we extracted includes the $diff$ value representing the observability and overall fitness score evaluating from the partitioning method. The process clusters the IEEE 118-bus system into 3 clusters at the end of 3000 iterations. Fig. 4.13 shows the relationship between the $diff$ value and the fitness scores.

In Fig. 4.13, the x-axis represents the fitness score value during the process of partitioning and the y-axis represents the difference between the largest *AMRV* and smallest *AMRV* among 3 clusters. The star marks in the figure represent the sampled data. The red dash line is a virtual

line used to demonstrate the samples trend. The point located at the right bottom represents the best score inside 3000 iteration, which has 0.2208 overall fitness score. The results shown in Fig. 4.13 show a strong and statistically significant indication between the fitness score f and the $diff$ value from $AMRV$. It proves that they have negative correlation between each other, which is an important evidence in support of inserting our proposed index into the partitioning method. It defines that the proper scalar on the MRI can improve the observability measurement distribution.

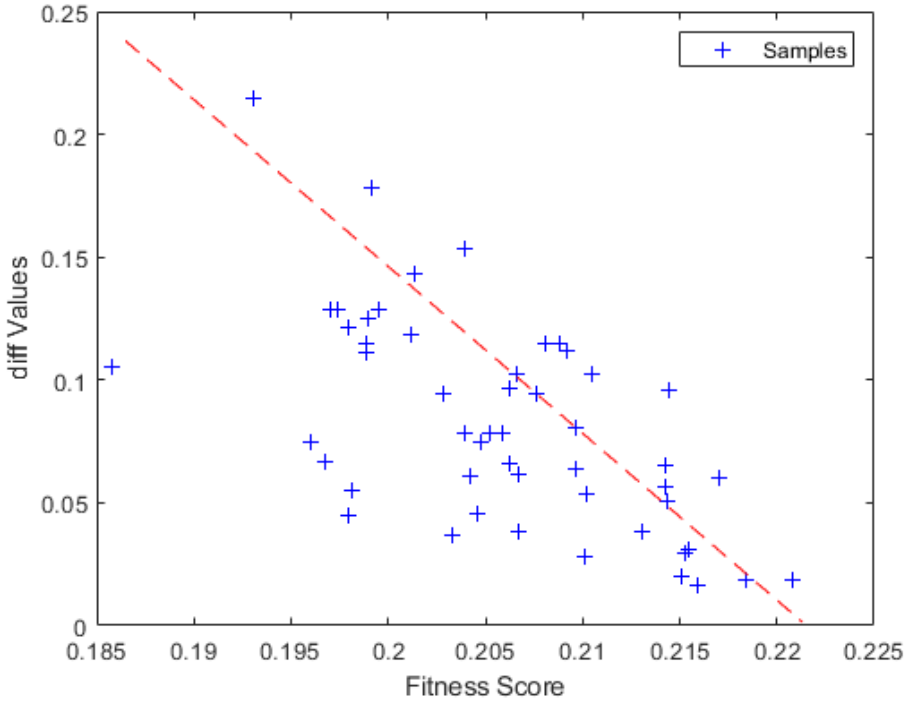


Figure 4.13: Plot for Clustering Quality and The Amount of $AMRV$ Difference

Based on the performance evaluation presented above, it can be concluded that MRI index is necessary and is reasonable to be inserted in terms of the observability measurements distribution during the partitioning process.

4.2.2 Performance of Proposed Partitioning With Extra PMUs

In this subsection we evaluate performance differences based on the observability of two kinds of systems, namely, the system with extra PMUs and the system without extra PMUs. The results presented in the previous sections suggest that the partitioning method with MRI can improve the system observability by balancing the observability on each cluster. The tested systems include IEEE-14, IEEE-30, IEEE-57, RTS 96 and IEEE-118 systems. Besides that, the partitioning method in this section is built upon the modified-GA PMU placement technique.

1. **IEEE 14-bus Partitioning Results**: Optimal AMRV value is set as 2, and optimal cluster number is 3

   - *Without Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [5,4,5]. The total observability values for each cluster are [5,4,7]. The AMRV values for each clusters are: [1, 1, 1.4]. The PMU installed number is [1,1,2].
   - *With Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [6,4,4]. The total observability values for each cluster are [9,6,6]. The AMRV values for each clusters are: [1.5, 1.5, 1.5]. The PMU installed number is [3,2,2].

   By comparing the results for the system with extra PMUs and without extra PMU, it can be seen that the total number of clusters is the same. The number of buses for each cluster changes the observability value for each cluster is improved, and the AMRV difference between each cluster decreases.

2. **IEEE 30-bus Partitioning Results**: Optimal AMRV value is set as 2, and optimal cluster number is 3

   - *Without Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [9,11,10]. The total observability values for each cluster are [13,17,14]. The AMRV values for each clusters are: [1.4, 1.6, 1.4]. The PMU installed number is [3,3,4].
   - *With Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [11,13,6]. The total observability values for each cluster are [24,26,13].

The AMRV values for each clusters are: [2.2, 2, 2.2]. The PMU installed number is [6,7,4].

By comparing the results for the system with extra PMUs and without extra PMUs, it can be seen that the total number of clusters is the same. The number of buses for each cluster changes. Although the difference of AMRV between each cluster remains the same for the two scenarios, the extra PMUs can increase the overall system observability and individual cluster observability values.

3. **IEEE 57-bus Partitioning Results**: Optimal AMRV value is set as 3, and optimal cluster number is 3

   - *Without Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [25,17,15]. The total observability values for each cluster are [34,20,17]. The AMRV values for each clusters are: [1.4, 1.2, 1.1]. The PMU installed number is [3,3,4].
   - *With Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [19,24,14]. The total observability values for each cluster are [33,41,24]. The AMRV values for each clusters are: [1.7, 1.7, 1.7]. The PMU installed number is [9,11,8].

   By comparing the results for the system with extra PMUs and without extra PMUs, it can be seen that the total number of clusters is the same. Although the number of buses for each cluster changes, the overall distribution remain the same. In terms of the observability, extra PMUs scenario improves each cluster's individual observability value as well as the AMRV. Also, the extra PMUs scenario is not different in terms of the AMRV, but no extra PMUs has largest difference as 0.3.

4. **RTS 96 Partitioning Results**: Optimal AMRV value is set as 3, and optimal cluster number is 3

   - *Without Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [27,21,25]. The total observability values for each cluster are [36,28,35]. The AMRV values for each clusters are: [1.3, 1.3, 1.4]. The PMU installed number is [7,6,7].

- *With Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [30,19,24]. The total observability values for each cluster are [71,45,60]. The AMRV values for each clusters are: [2.4,2.4,2.5]. The PMU installed number is [18,12,15].

By comparing the results for the system with extra PMUs and without extra PMUs, it can be seen that the total number of clusters is the same. As for the topology property of the RTS 96 system, the number of buses in each cluster changes in order to balance the AMRV of each cluster. Although the extra PMUs scenario has not improved the AMRV difference, it increases the AMRV value while maintaining the smallest difference.

5. **IEEE 118 Partitioning Results**: Optimal AMRV value is set as 3, and optimal cluster number is 3

   - *Without Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [15,57,46]. The total observability values for each cluster are [21,75,63]. The AMRV values for each clusters are: [1.4, 1.3, 1.4]. The PMU installed number is [4,16,12].
   - *With Extra PMUs*: 3 clusters result after the partitioning with the bus number for each cluster as: [47,48,23]. The total observability values for each cluster are [89,91,44]. The AMRV values for each clusters are: [1.9,1.9,1.9]. The PMU installed number is [22,22,11].

   By comparing the results for the system with extra PMUs and without extra PMUs, the total number of clusters stay the same. The number of buses containing in each cluster changes between the system with an extra PMU and without an extra PMU. It leads to the change of the AMRV of each cluster. It can be seen that the observability of each cluster has been improved. Meanwhile, the AMRV difference has been minimized.

From the above data, it can be observed that the system with extra PMUs has improved observability on each sub-cluster. Besides, the balance of the observability values has been retained, which can be seen by the AMRV for each sub-cluster. In the mean time, the performance of the partitioning method does not change, which can be seen from the total number of cluster and the bus number of each cluster.

4.3 Application on Multi-area State Estimation Based on CKF

4.3.1 Global CKF State Estimation Using PMU Only

The discretized power system state estimation model is given by [81]:

$$\mathbf{x_{k+1}} = \mathbf{f}(\mathbf{x_k}) + \mathbf{w_k} \tag{4.15}$$

$$\mathbf{z_k} = \mathbf{h}(\mathbf{x_k}) + \mathbf{v_k} \tag{4.16}$$

where $\mathbf{w_k}$ is the system process noise vector, $\mathbf{z_k}$ is the measurement vector, $\mathbf{v_k}$ is the measurement noise vector. The vector $\mathbf{x_k}$ is the state vector at time epoch k and is described by:

$$\begin{aligned} \mathbf{x_k} &= \begin{bmatrix} \mathbf{v_k} \\ \theta_\mathbf{k} \end{bmatrix} \\ &= \begin{bmatrix} v_1^k \\ v_2^k \\ \vdots \\ v_n^k \\ \theta_1^k \\ \theta_2^k \\ \vdots \\ \theta_n^k \end{bmatrix} \end{aligned} \tag{4.17}$$

where v_n is the voltage magnitude of bus n, and θ_n is the voltage angle of bus n. Flat start is the typical setting for the state vector in the state estimation process, i.e. all the bus voltage magnitudes are 1p.u.and bus voltage angles are 0 degree. As for the system dynamics vector function $\mathbf{f}(\cdot)$, it is defined as:

$$\mathbf{f}(\mathbf{x}_k) = \alpha\mathbf{x}_k + (1-\alpha)\mathbf{u}_k \tag{4.18}$$

where $\mathbf{x}_k$ is the state vector at time k, $\mathbf{u}_k$ is the system input vector of time k, α is a system parameter used to define the variance of the state vector.

The vector function $\mathbf{h}(\cdot)$ is the measurement function used to express the relation between the state vector and the measurement vector. Typically, the power flow equation is used as the measurement function $\mathbf{h}(\cdot)$. In this research, PMU is assumed to be the only measurement to

derive raw data from the system. The typical power flow measurement function would not fit for this scenario. A new measurement function is employed to match the PMU property. Per PMU property, it can measure the voltage phasor of the bus having a PMU installed and the current phasor of the attached branch. The attached bus voltage phasor can be evaluated based on the measured PMU data. Assuming that the branch resistance and reactance data are given, the PMU basic measurement function can be written as:

$$\mathbf{I_{ab}} = \frac{\mathbf{v_a} - \mathbf{v_b}}{R_{ab} + jX_{ab}} \tag{4.19}$$

where $\mathbf{I_{ab}}$ is the measured branch current phasor, $\mathbf{v_a}$ is the bus voltage phasor, R_{ab} and X_{ab} are the given branch impedance data. In the above equation, the $\mathbf{v_b}$ will be the state variable $\mathbf{x_k}$ in Eq. 4.16, and $\mathbf{I_{ab}}$ will be the measurement $\mathbf{z_k}$ in Eq. 4.16. Thus, assuming $\mathbf{v_b}$ stands for the measured voltage phasor, the measurement transfer function can be expressed as:

$$h(\mathbf{x_k}) = \frac{\mathbf{v_k}\angle\theta_\mathbf{k} - \mathbf{v_b}\angle\theta_\mathbf{b}}{R_{ab} + jX_{ab}} \tag{4.20}$$

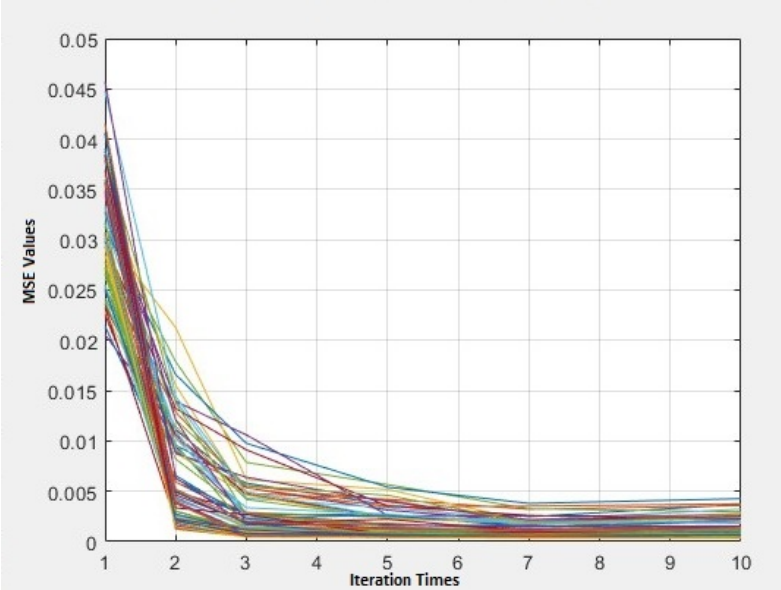
Based on the CKF definition discussed in Ch. 2, prior knowledge of the system noise and the measurement noise probability distribution is required for the CKF algorithm implementation. According to the research papers [97, 49, 56], the typical assumption for the system noise is that it is a Gaussian distributed with zero mean. As for the measurement noise, it will be assumed to be non-zero mean Gaussian distributed noise based on the results presented in [96, 12]. The standard deviation values for system noise and measurement noise, as well as the mean value for the measurement noise, need to be defined based on the observation of various experiments. [12] provides the range of mean values and standard deviation value for the measurement noise. By using the values from [12], the main purpose of this experiment is to obtain an estimate of the standard deviation for the system noise so that it would have good performance on both convergence and accuracy. The standard deviation is a value ranging from 0 to 1 $\sigma \in (0,1]$. In order to evaluate the performance of the CKF in the state estimation, a matrix is defined to represent that, which is the empirical mean square error (MSE). The evaluation of MSE will be implemented based on the *Monte-Carlo* method. The MSE is denoted as:

$$MSE_i(k) = \frac{1}{Run}\sum_{j=1}^{Run}\left[x_i(k) - \hat{x}_i^j(k)\right]^2 \tag{4.21}$$

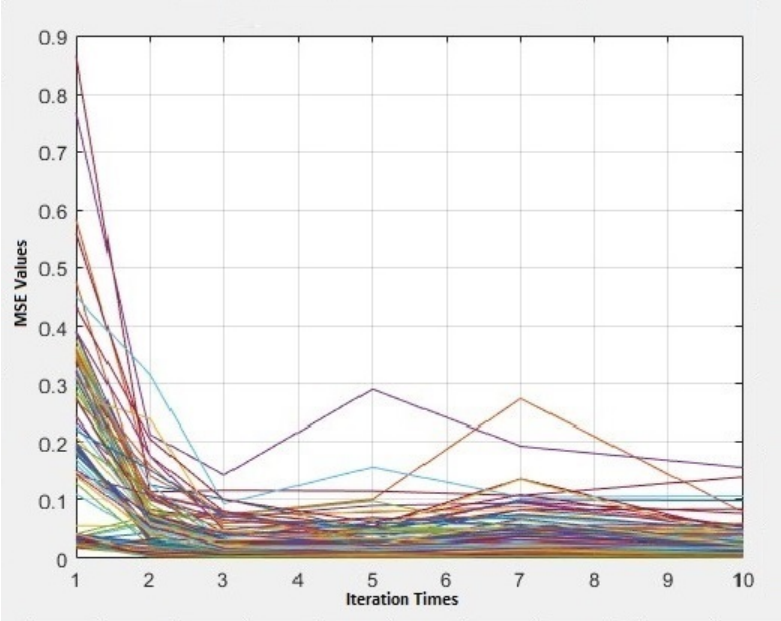
where i is the bus index, j is the Monte-Carlo index, k is the time epoch, $x_i(k)$ is the true state variable value and $\hat{x}_i^j(k)$ is the estimated state variable value after *Monte-Carlo* run j. The Monte-Carlo run time is set to be 100 in our simulation. The MSE will represent the convergence and the accuracy of the CKF state estimation.

The experiment is carried out by implementing the CKF state estimation on the RTS 96 system and the IEEE 118-bus system without partitioning. The purpose is to find the proper noise setting prior to the MASE. During the experiment for finding the proper noise σ values, a phenomenon is observed that the voltage angle will always converged regardless of the σ value. On the other hand, the voltage magnitude has inadequate convergence if the σ values are not properly selected. Furthermore, the MSE performance of voltage magnitude does not converged well for $\sigma_{sys} > 0.06$, either for RTS 96 and IEEE 118, and some even have divergence. Based on the observation, the convergence becomes the first evaluation priority for the performance of the state estimation algorithm.

Figure 4.14 through Figure 4.16 shows the performance result of different values of standard deviation of system noise σ_{sys}, in which the y-axis is the voltage magnitude MSE value and the x-axis is the iteration times setting $iter$. The evaluation systems are RTS96 and IEEE-118 system. For all simulations, the initial posterior error covariance $P_{0|0}$ is set to be the identity matrix, i.e. $P_{0|0} = I$.

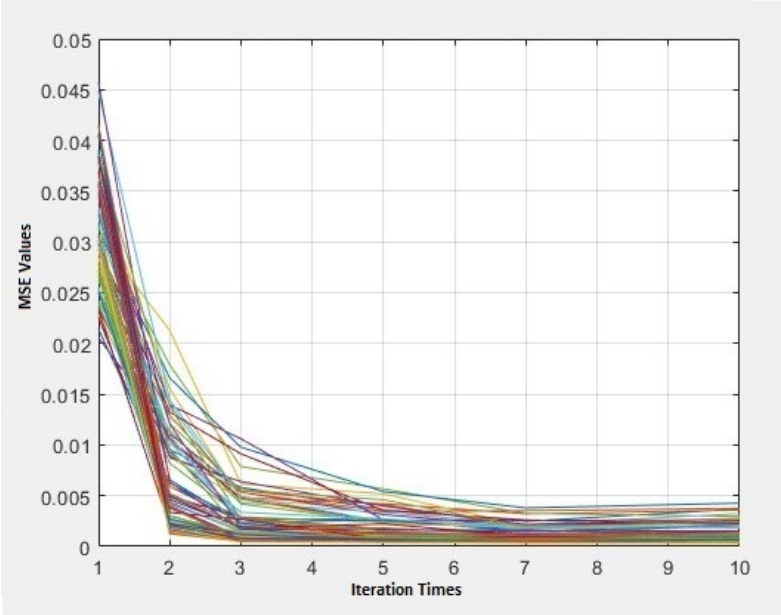


(a) RTS 96 System

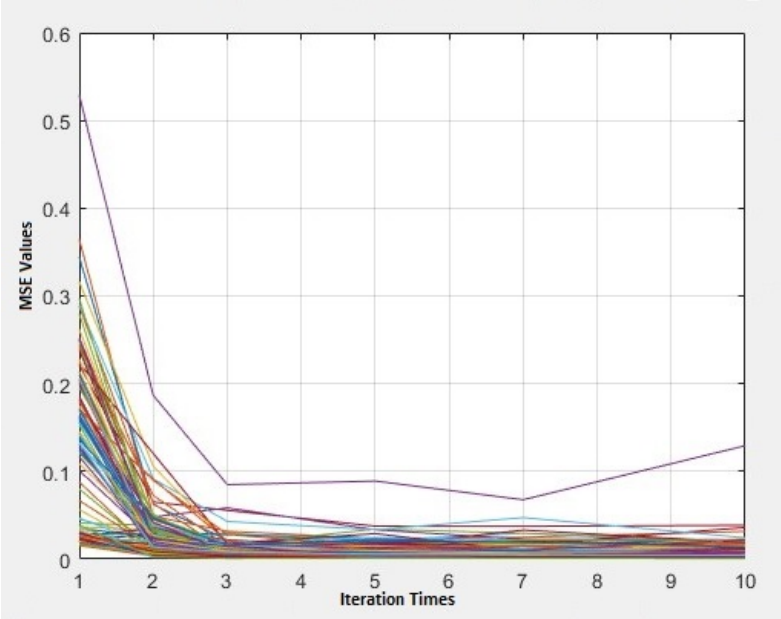


(b) IEEE 118-bus System

Figure 4.14: MSE Performance For $\sigma_{sys} = 0.04$

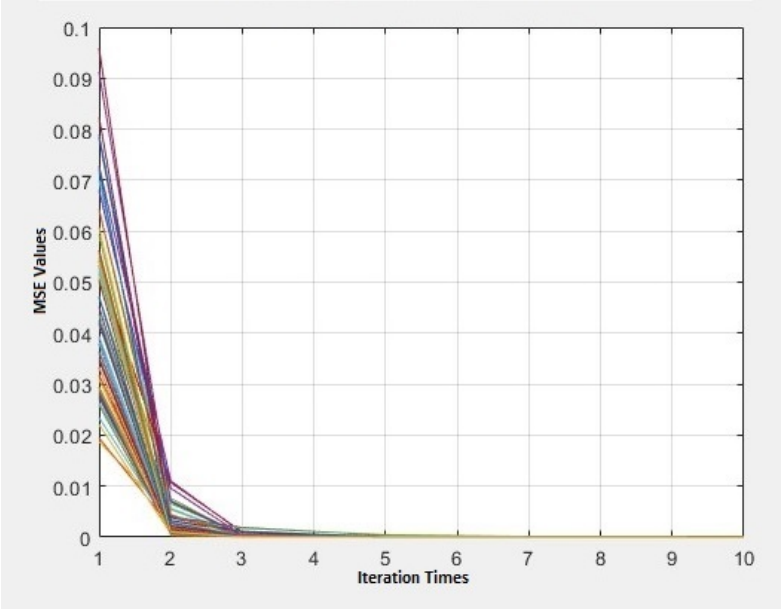


(a) RTS 96 System

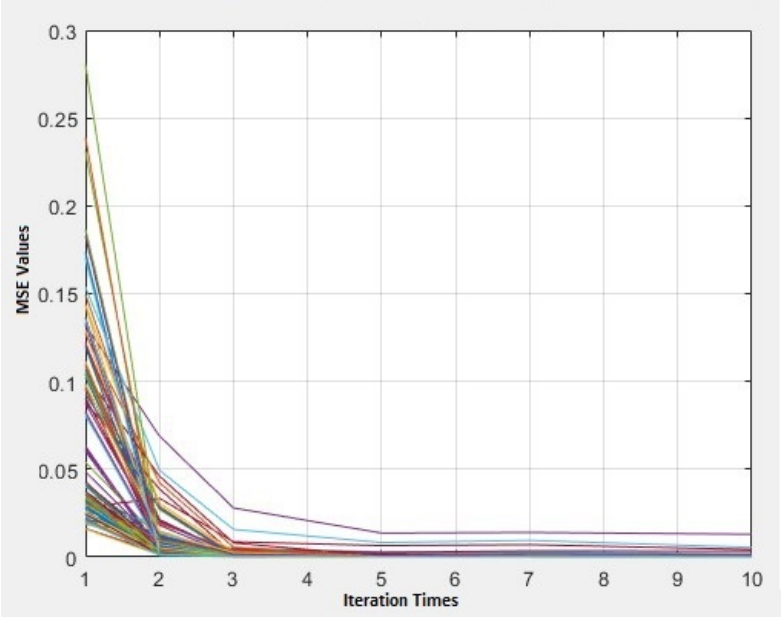


(b) IEEE 118-bus System

Figure 4.15: MSE Performance For $\sigma_{sys} = 0.02$



(a) RTS 96 System



(b) IEEE 118-bus System

Figure 4.16: MSE Performance For $\sigma_{sys} = 0.01$

By observing previously figures, it can be concluded that the closer to 0 the variance of the system noise is, the more the system model is trusted. However, it can be seen that even with high confidence system noise setting it does not converge very well (see Figure 4.15 and Figure 4.16). The fluctuation is always present as far as system noise is concerned, because we never know the system model perfectly. Moreover, the correlation between the system noise and the measurement noise would affect the performance as well.

As discussed in the previous chapter, the estimated noise technique will be employed to decrease the impact on the CKF performance due to the inaccurate knowledge of model parameters. For the system noise w, only the covariance function needs to be estimated since the mean value is set to zero. Let us consider the following estimate of the system noise variance:

$$\hat{Q}_{k+1} = \frac{1}{k+1}\sum_{j}^{k}\left\{\left[\hat{\mathbf{x}}_{j+1} - \hat{\mathbf{x}}_{j+1|j}\right]\cdot\left[\hat{\mathbf{x}}_{j+1} - \hat{\mathbf{x}}_{j+1|j}\right]^{T}\right\} \tag{4.22}$$

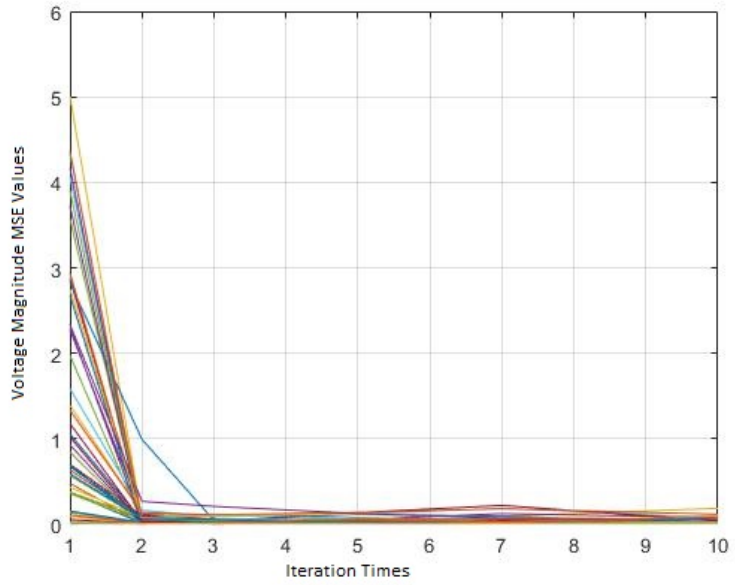
where $\hat{\mathbf{x}}_{j+1}$ is the unconditional estimate at time epoch j, and $\hat{\mathbf{x}}_{j+1|j}$ is the prediction estimate after updating the cubature quadrature points. In terms of the measurement noise estimation model, the mean value needs to be considered as well. Consider now the following estimates of mean and variance of measurement noise:

$$\hat{\mathbf{v}}_{k+1} = \frac{1}{k+1}\sum_{j=0}^{k}\left[\mathbf{z}_{j+1} - \mathbf{h}(\hat{\mathbf{x}}_{j+1})\right] \tag{4.23}$$

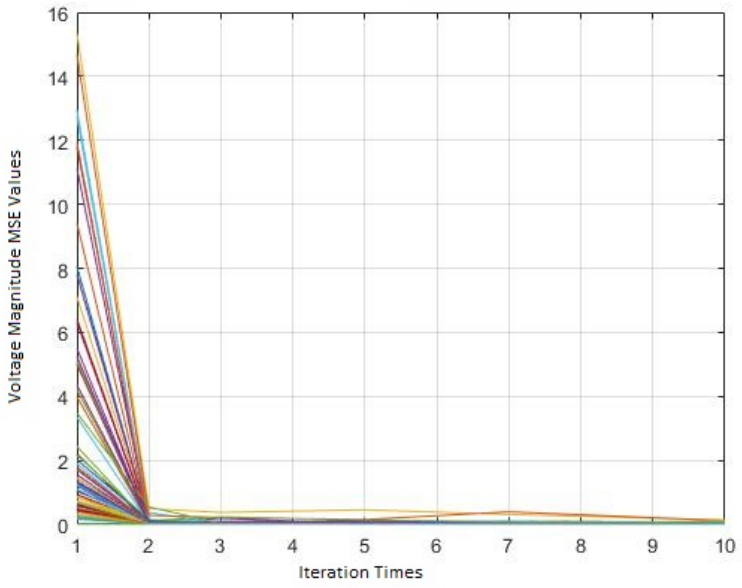
$$\hat{R}_{k+1} = \frac{1}{k+1}\sum_{j}^{k}\left\{\left[\mathbf{z}_{j+1} - \hat{\mathbf{z}}_{j+1|j}\right]\cdot\left[\mathbf{z}_{j+1} - \hat{\mathbf{z}}_{j+1|j}\right]^{T}\right\} \tag{4.24}$$

where $\mathbf{h}(\cdot)$ is the measurement transform function, $\mathbf{z}_{j+1}$ is the true measurement which is assumed to be given, $\hat{\mathbf{z}}_{j+1|j}$ is the prediction of the measurement. In this simulation, the iteration times k is set to be $[1,2,3,5,10]$ for every 100 Monte-Carlo runs. It is implemented on RTS 96 system and IEEE 118-bus systems without partitioning.

By using the estimated noise technique, the convergence has been improved without accurate system setting in advance. In the simulation, the initial mean value of the measurement noise μ_{mea} is randomly generated from $(0,1)$, and the initial σ_{mea} is set to 0.08. It is derived from the maximum measurement noise decibel value presented in [12] by using equation [12] $dB_{noise} = 20\log\frac{1}{\sigma_{mea}}$. The performance for different σ_{sys} values are presented in the following figures:

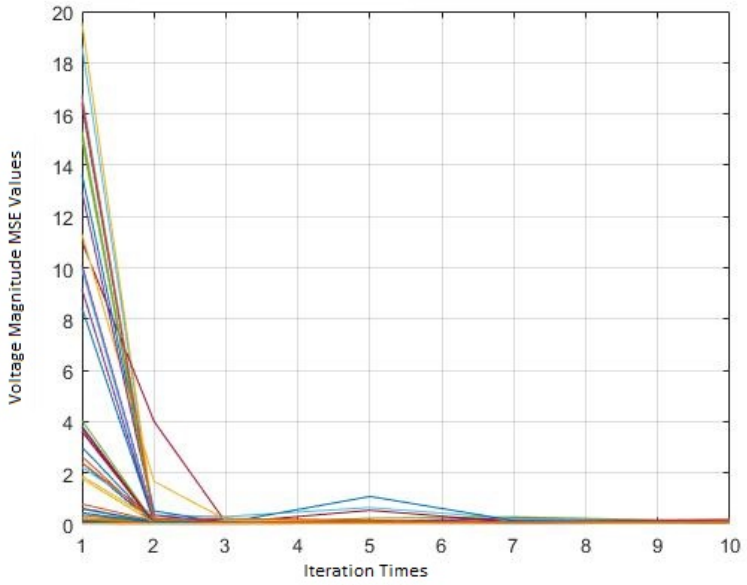


(a) RTS 96 System

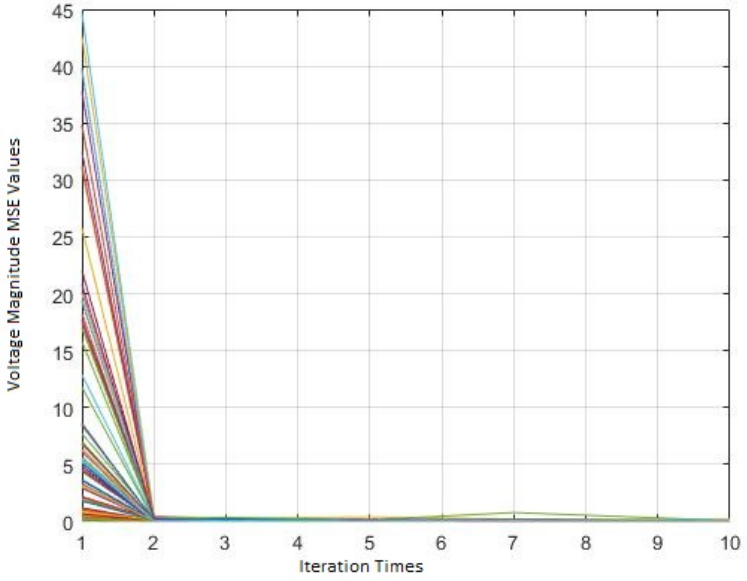


(b) IEEE 118-bus

Figure 4.17: MSE Performance For $\sigma_{sys} = 0.1$ Using Estimated Noise

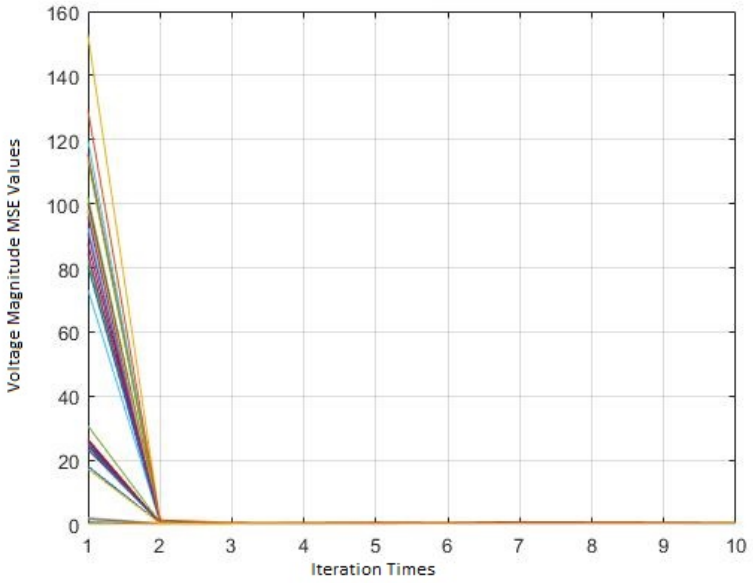


(a) RTS 96 System

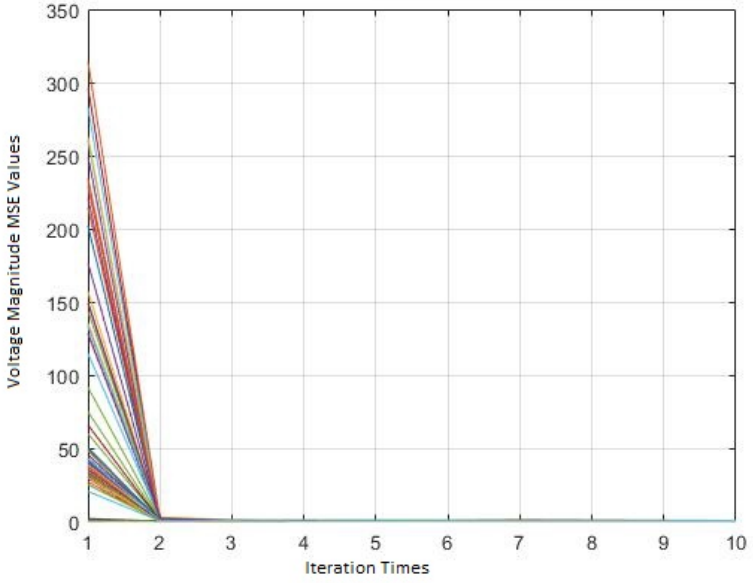


(b) IEEE 118-bus System

Figure 4.18: MSE Performance For Different $\sigma_{sys} = 0.2$ Using Estimated Noise



(a) RTS 96 System



(b) IEEE 118-bus System

Figure 4.19: MSE Performance For Different $\sigma_{sys} = 0.5$ Using Estimated Noise

Compared with the convergence performance with a fixed noise value presented previously, it can be seen that the convergence is improved with guessed initial value by using the estimated noise method. In addition to the convergence, the performance of accuracy needs to be evaluated as well. The comparison will be carried out between the system with a fixed noise value and the system with estimated noise method. By inspecting the experiment result, it can be seen that both scenarios achieve the minimum mean value at the last iteration. The following itemized list summarizes the performance of two scenarios:

1. **RTS 96**:

   - $\sigma_{\mathbf{sys}} = \mathbf{0.1}$:
     Estimated noise method has 0.0147 as the mean MSE value for voltage magnitude, and the worst case MSE is less than 0.06.
     The fixed noise method has 0.03 as the mean MSE value, and the worst case is larger than 0.225.
   - $\sigma_{\mathbf{sys}} = \mathbf{0.2}$:
     Estimated noise method has 0.0488 as the mean MSE value for voltage magnitude, and the worst case MSE is less than 0.15.
     The fixed noise method has 0.2439 as the mean MSE value, and the MSE worst case has value larger than 1.
   - $\sigma_{\mathbf{sys}} = \mathbf{0.5}$:
     Estimated noise method has 0.2861 as the mean MSE value for voltage magnitude, and the worst case MSE is less than 0.45.
     The fixed noise method has 5.0337 as the mean MSE value, and the MSE does not converge.

2. **IEEE 118-bus**:

   - $\sigma_{\mathbf{sys}} = \mathbf{0.1}$:
     Estimated noise method has 0.0176 as the mean MSE value for voltage magnitude, and the worst case MSE is less than 0.08.
     The fixed noise method has 0.2858 as the mean MSE value, and the worst case is larger than 1.
   - $\sigma_{\mathbf{sys}} = \mathbf{0.2}$:
     Estimated noise method has 0.0530 as the mean MSE value for voltage magnitude,

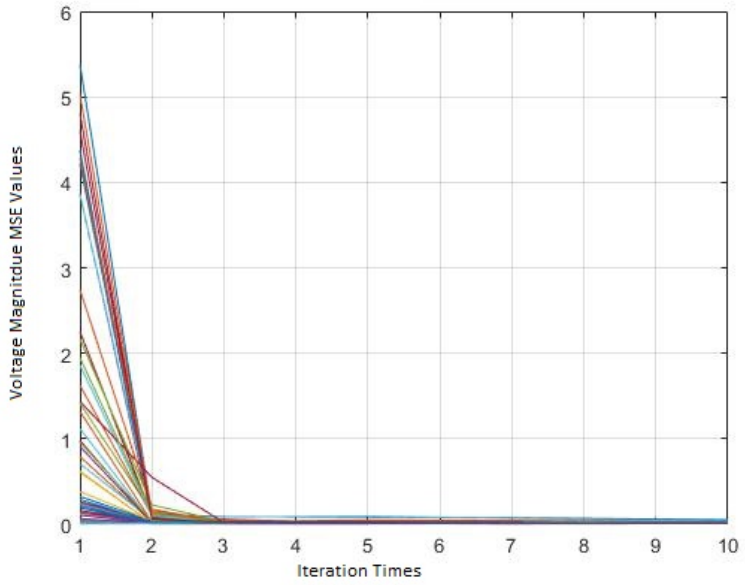
and the worst case MSE is less than 0.3.
The fixed noise method has 1.9421 as the mean MSE value, and the MSE diverged.

- $\sigma_{\mathbf{sys}} = \mathbf{0.5}$:
  Estimated noise method has 0.2980 as the mean MSE value for voltage magnitude, and the worst case MSE is less than 0.63.
  The fixed noise method has 26.9832 as the mean MSE value, and the MSE diverged.

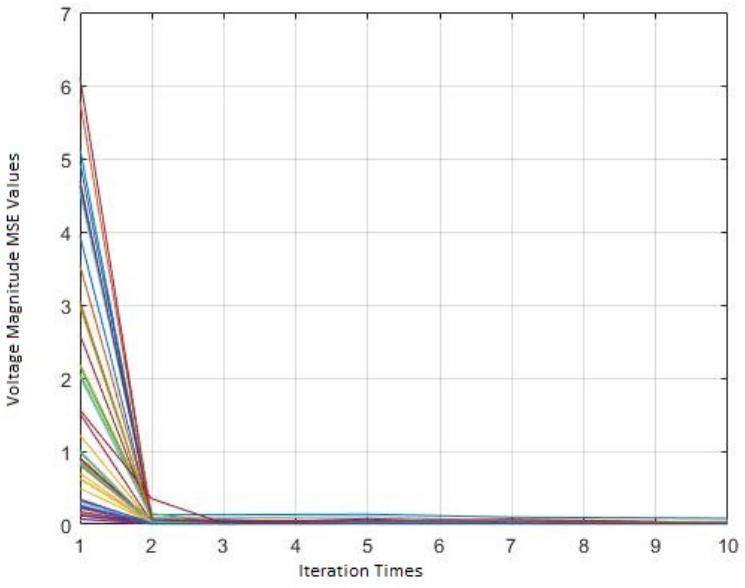
We can therefore conclude that the estimated noise method has higher accuracy than the fixed noise method under the same σ_{sys} value setting.

4.3.1.1 Performance With Extra PMUs Using CKF State Estimation

Based on the results obtained in the previous experiment, the estimated noise method proved to have better convergence and accuracy than using fixed noise method. However, the robustness from extra PMU on the accuracy needs to be verified. The comparison will be made between two scenarios, which is system with extra PMU and system without extra PMU. The comparison will focus on evaluating the MSE converged value by using the CKF with estimated noise method. In order to have the optimum performance of both scenarios, the noise initial value is set to 0.1 including σ_{sys}, μ_{mea} and σ_{mea}. Three systems will be considered in the comparison, which are IEEE 57-bus, RTS 96, and IEEE 118-bus systems. Their MSE performance is presented in the following figure:

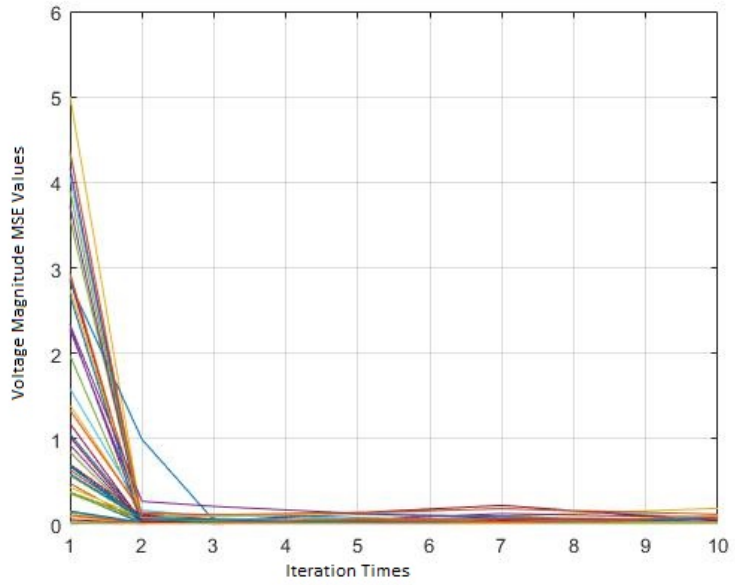


(a) IEEE 57-bus System: Adding All Extra PMUs

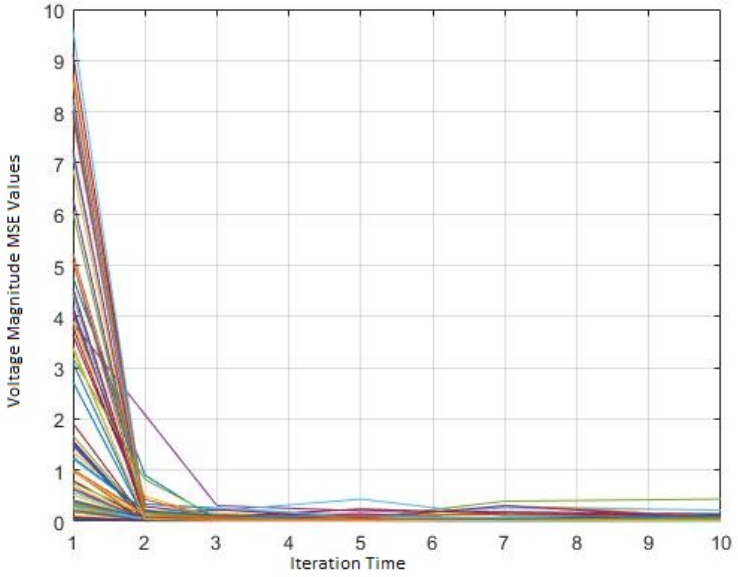


(b) IEEE 57-bus System No Extra PMU

Figure 4.20: Performance Of IEEE 57-bus With and Without PMUs

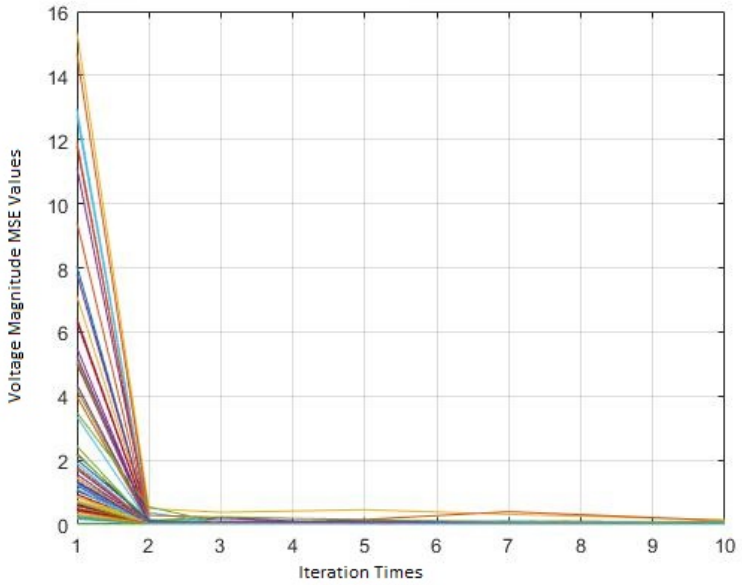


(a) RTS 96 System: Adding All Extra PMUs

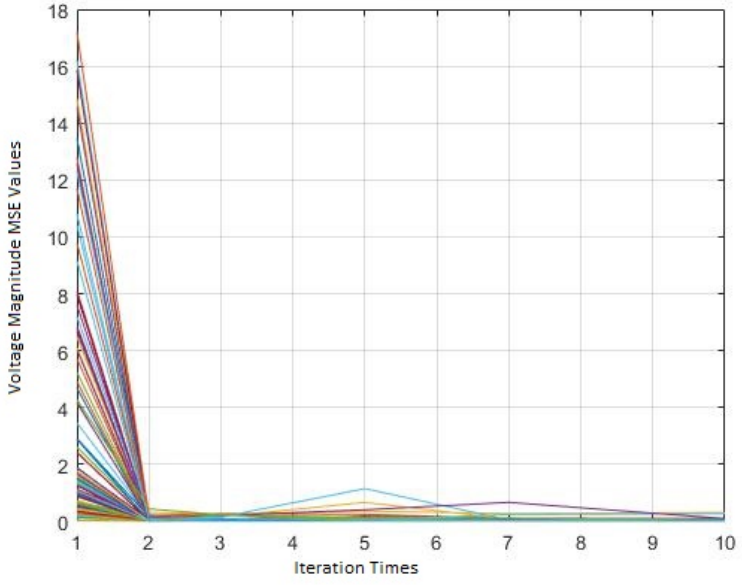


(b) RTS 96 System: No Extra PMU

Figure 4.21: Performance Of RTS 96 With and Without PMUs



(a) IEEE 118-bus: Adding All Extra PMUs



(b) IEEE 118-bus System: No Extra PMU

Figure 4.22: Performance Of IEEE 118-bus With and Without PMUs

From previous figures, the left hand side plots are the performances of the systems with extra PMUs installed, and the right hand side figures are the performance of the systems without extra PMUs. In the figures, the x-axis denotes the iteration times and y-axis denotes the values of MSE evaluated based on Eq. 4.21. The performance results are evaluated based on iteration times $iter = [1,2,3,5,10]$, and 100 Monte-Carlo runs for each iteration times setting. Based on the figures, few observation data will be presented for the comparison. The observation data is presented below. All the observation data are derived from the data at $iter = 10$. The reason for choosing $iter = 10$ is based on the observation of the performance that all of scenarios will converged at $iter = 10$, and the best performance under either scenario is occur at $iter = 10$.

1. **IEEE 57-bus**:

   - **With Extra PMUs**:
     The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.052. The overall average MSE magnitude value is 0.0135 at the last iteration.
   - **Without Extra Pmus**:
     The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.082. The overall average MSE magnitude value is 0.0153 at the last iteration.

2. **RTS 96**:

   - **With Extra PMUs**:
     The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.06. The overall average MSE magnitude value is 0.0147 at the last iteration.
   - **Without Extra PMUs**:
     The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.42. The overall average MSE magnitude value is 0.0422 at the last iteration.

3. **IEEE 118-bus**:

   - **With Extra PMUs**:
     The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.076. The overall average MSE magnitude value is 0.0176 at the last iteration.
   - **Without Extra PMUs**:
     The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.31. The overall average MSE magnitude value is 0.0276 at the last iteration.

It can be observed that the accuracy at the last iteration has better performance, worse case value and the average MSE value, for with extra PMUs than system without extra PMUs. It means that the extra PMU can help improve the state estimation accuracy when only PMU measurements are employed. Besides this observation, another observation is that the MSE value from estimated noise method will eventually converge at the last iteration times setting $iter = 10$ either with or without extra PMU scenario.

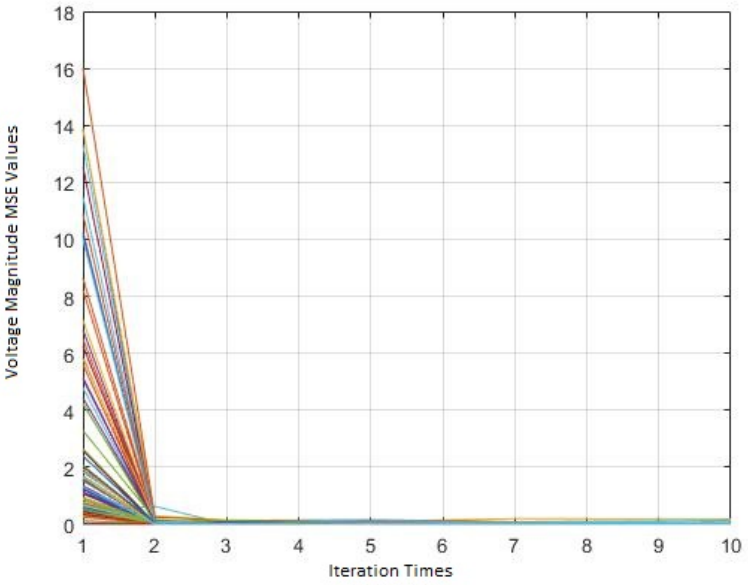
Additional performance evaluation will be provided below on IEEE 118-bus system to prove the affect on the state estimation accuracy by inserting extra PMUs. It is the performance results for the scenario that losing partial extra PMUs (5 PMUs off whole extra PMUs) per period from total number of 55 PMUs (with extra PMUs) to 32 PMUs (without extra PMUs). Recap from previous section, the without extra PMUs for IEEE 118-bus are:
$[3,5,9,12,15,17,21,25,29,34,37,40,45,49,52,56,62,64,68,70,71,75,77,80,85,86,90,94,101,105,110,114]$.
The extra PMU installed are:
$[6,10,13,16,20,28,33,43,48,50,53,57,67,72,84,87,97,98,99,107,111,112,117]$.
The performance includes the MSE data at the last iteration and the corresponding MSE performance figure.

- **No PMUs on 99,107,111,112,117**:
  The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.195. The overall average MSE magnitude value is 0.0192.

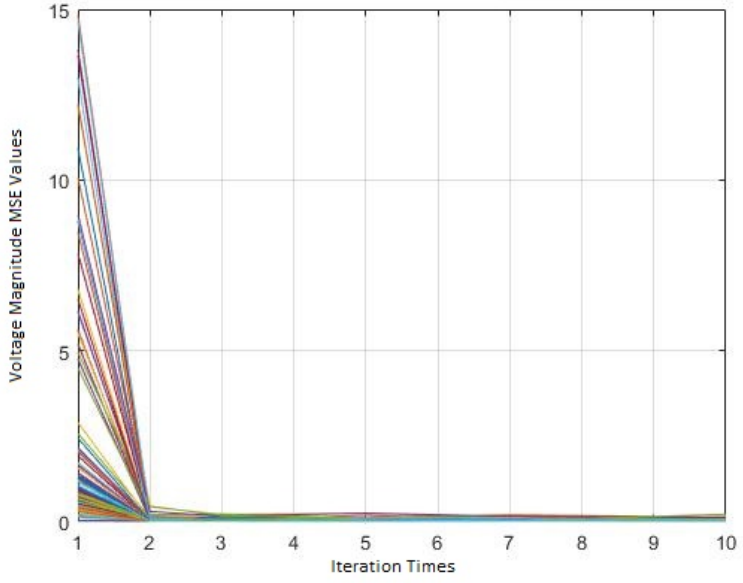
- **No PMUs on 72,84,87,97,98,99,107,111,112,117**:
  The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.20. The overall average MSE magnitude value is 0.0229.

- **No PMUs on 48,50,53,57,67,72,84,87,97,98,99,107,111,112,117**:
  The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.20. The overall average MSE magnitude value is 0.0247.

- **No PMUs on 16,20,28,33,43,48,50,53,57,67,72,84,87,97,98,99,107,111,112,117**:
  The MSE converge at $iter = 10$. The worst MSE magnitude performance at $iter = 10$ is 0.27. The overall average MSE magnitude value is 0.0269.

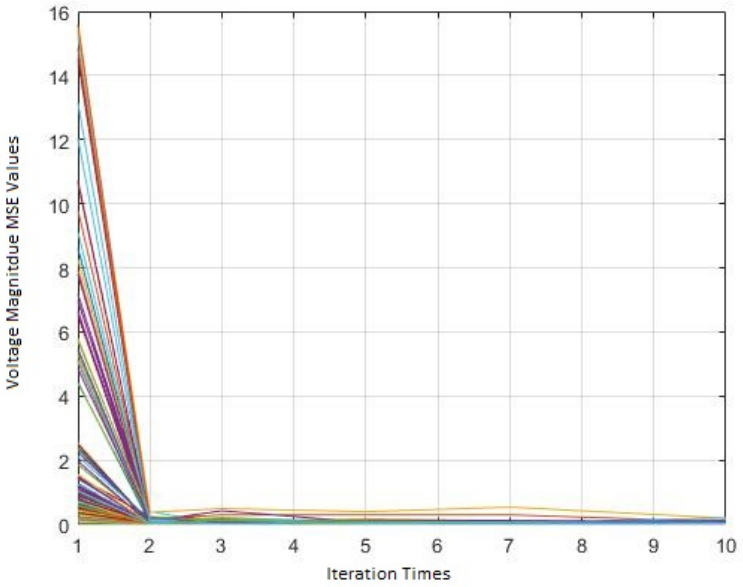


(a) Losing 5 Extra PMUs

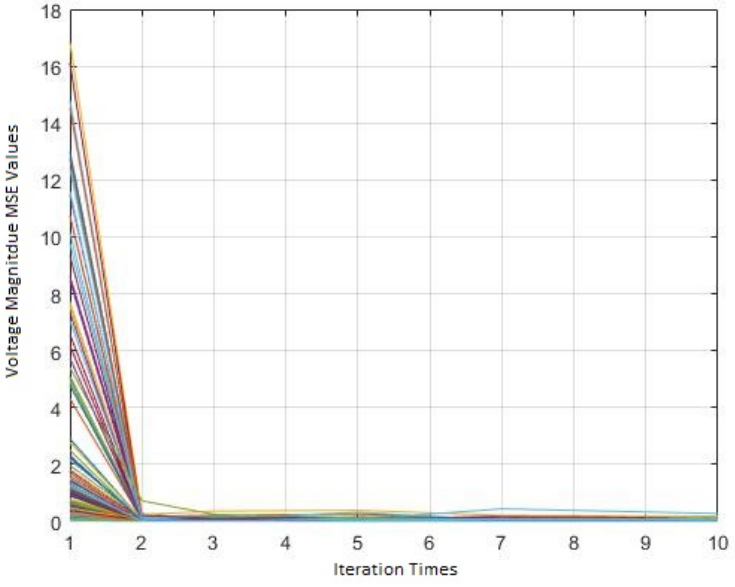


(b) Losing 10 Extra PMUs

Figure 4.23: Performance Of Losing Partial Extra PMUs on IEEE 118-bus



(a) Losing 15 Extra PMUs



(b) Losing 20 Extra PMUs

Figure 4.24: Performance Of Losing Partial Extra PMUs on IEEE 118-bus

Based on the results presented above, it can be seen that it has the same worst case value for *No PMUs on 72,84,87,97,98,99,107,111,112,117* case as that for *No PMUs on 48,50,53,57,67,72,84, 87,97,98,99,107,111,112,117*. However, the average MSE value increases from case with *No PMUs on 72,84,87,97,98,99,107,11,112,117* to case with *No PMUs on 48,50,53,57,67,72,84,87,97,98,99,107, 111,112,117*. At this point, it can be observed that it is still true that the accuracy will decrease with the loss of the extra PMUs number increase. Besides, it can be seen that the convergence is still the same, as seen in Fig. 4.17.

4.3.2 Multi-Area State Estimation

We now evaluate state estimation performance using a distributed approach after applying our proposed partitioning algorithm with extra PMUs and estimated noise method on CKF. The two level estimation process will be implemented in our research for MASE, which is discussed in previous chapter. As introduced previously, our proposed partitioning algorithm can generate different number of clusters according to the user application. The MASE performance evaluation include the results for three different number of clusters by using our proposed partitioning algorithm.

4.3.2.1 First Level MASE

The first level is the sub-system level, which will implement the state estimation on each cluster locally. Thus, the two levels state estimation will be implemented after the partitioning method. After partitioning, the measuring of the boundary buses has various methods defined by the user or the applications. In our research, the boundary buses will "disconnect" with the buses outside of the same cluster. In other words, all buses can only be measured by the buses within same cluster. In order to realize this definition, the connectivity matrix A will be modified according to the partitioning results. In the second level state estimation, the angle for the state vector will be adjusted. The estimated voltage angle will be modified based on the difference value between the subsystem reference bus and the global system reference bus.

The PMU locations will be evaluated by using the GA-based PMU placement algorithm. All extra PMUs will be added onto the results after the GA-based PMU placement. The proposed partitioning method with MRI will be implemented to have the sub-systems decomposition. By using our proposed partitioning method, three cluster numbers are generated, which are 3

clusters, 4 cluster, and 5 clusters.

The simulation is implemented on two systems, i.e. RTS 96 and IEEE 118-bus systems. The performance results include the clustering results, the MSE performance of magnitude and angle of each cluster, and the worst MSE value. The simulation will starts from 3 cluster scenario.

RTS 96 System:

- **Clustering Results**:
  Cluster 1: Total number of buses is 30. The AMRV value is 2.4.
  Cluster 2: Total number of buses is 19. The AMRV value is 2.4.
  Cluster 3: Total number of buses is 24. The AMRV value is 2.5.

- **MSE Performance**:
  Cluster 1: Reference bus is bus 1. The MSE performance can be seen on Fig 4.25. The average MSE of the voltage magnitude is 0.0015, and the worst case MSE value of the magnitude is 0.0035. The average MSE of the voltage angle is 0.0023, and the worst case MSE value of the angle is 0.0103.
  Cluster 2: Reference bus is bus 7. The MSE performance can be seen on Fig 4.26. The average MSE of the voltage magnitude is 0.0018, and the worst case MSE value of the magnitude is 0.0062. The average MSE of the voltage angle is 0.0022, and the worst case MSE value of the angle is 0.0063.
  Cluster 3: Reference bus is bus 14. The MSE performance can be seen on Fig 4.27. The average MSE of the voltage magnitude is 0.0010, and the worst case MSE value of the magnitude is 0.0020. The average MSE of the voltage angle is 0.0017, and the worst case MSE value of the angle is 0.0095.

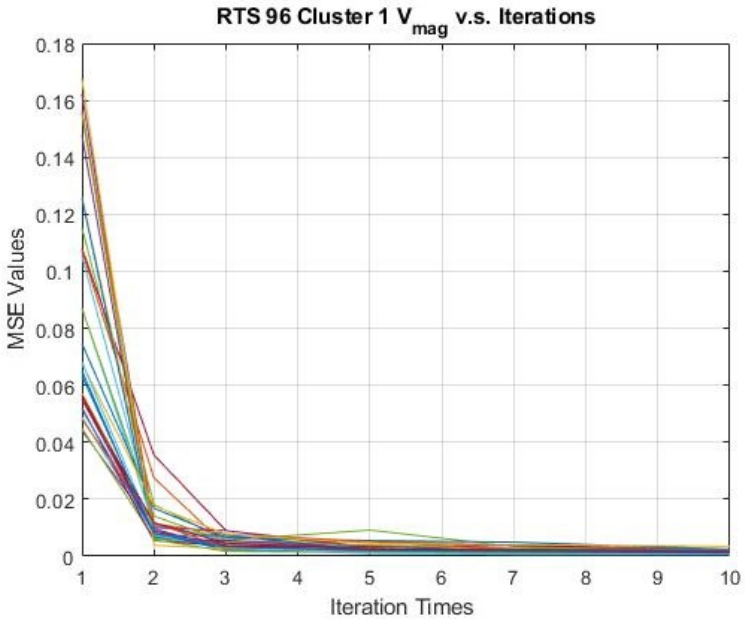
From above presented data, it can be seen that the average MSE maximum value is less than 0.4% for the voltage magnitude, and the average voltage angle MSE maximum value is less than 0.3% at the first level state estimation. Recalling that the centralized overall system having 0.0147 as the best performance for the voltage magnitude MSE, it can be seen that the first level local state estimation has better performance than the overall system one in terms of the voltage magnitude.

IEEE 118-bus System:

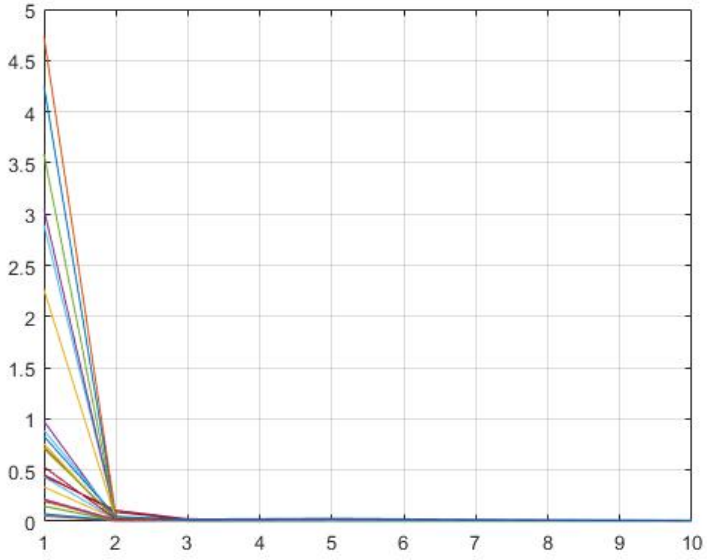
- **Clustering Results**:
  Cluster 1: Total number of buses is 47. The AMRV value is 1.9.

Cluster 2: Total number of buses is 48. The AMRV value is 1.9.
Cluster 3: Total number of buses is 24. The AMRV value is 2.5.

- **MSE Performance**:
  Cluster 1: Reference bus is bus 1. The MSE performance can be seen on Fig 4.28. The average MSE of the voltage magnitude is 0.0016, and the worst case MSE value of the magnitude is 0.005. The average MSE of the voltage angle is 0.0017, and the worst case MSE value of the angle is 0.0104.
  Cluster 2: Reference bus is bus 24. The MSE performance can be seen on Fig 4.29. The average MSE of the voltage magnitude is 0.009, and the worst case MSE value of the magnitude is 0.0298. The average MSE of the voltage angle is 0.0085, and the worst case MSE value of the angle is 0.0269.
  Cluster 3: Reference bus is bus 45. The MSE performance can be seen on Fig 4.30. The average MSE of the voltage magnitude is 0.0036, and the worst case MSE value of the magnitude is 0.0102. The average MSE of the voltage angle is 0.0024, and the worst case MSE value of the angle is 0.0062.

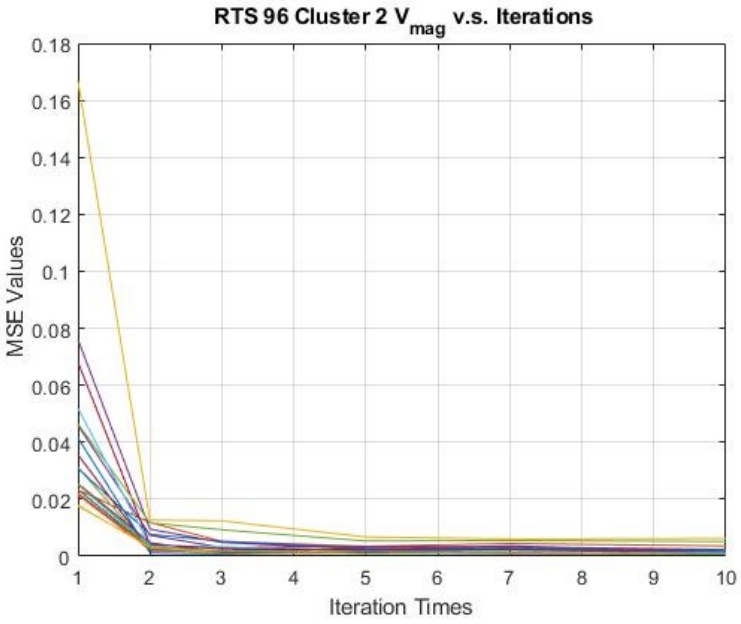


(a) RTS 96 Area 1 Voltage Magnitude

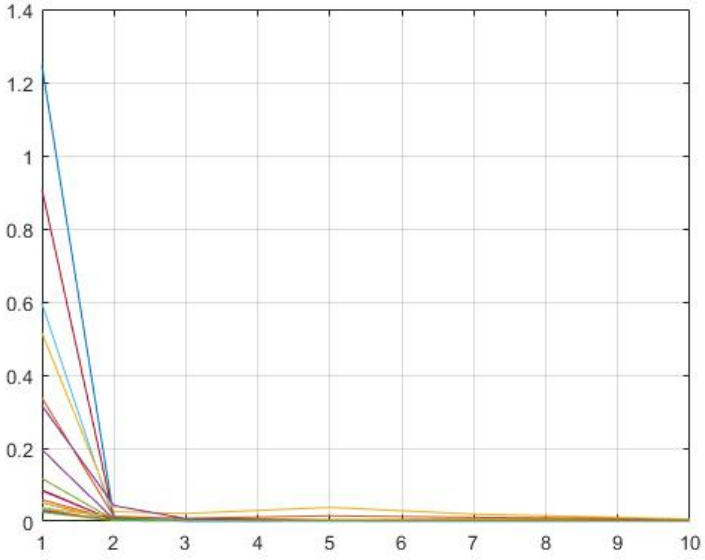


(b) RTS 96 Area 1 Voltage Angle

Figure 4.25: RTS 96 Multi-Area First Level State Estimation MSE for Cluster 1

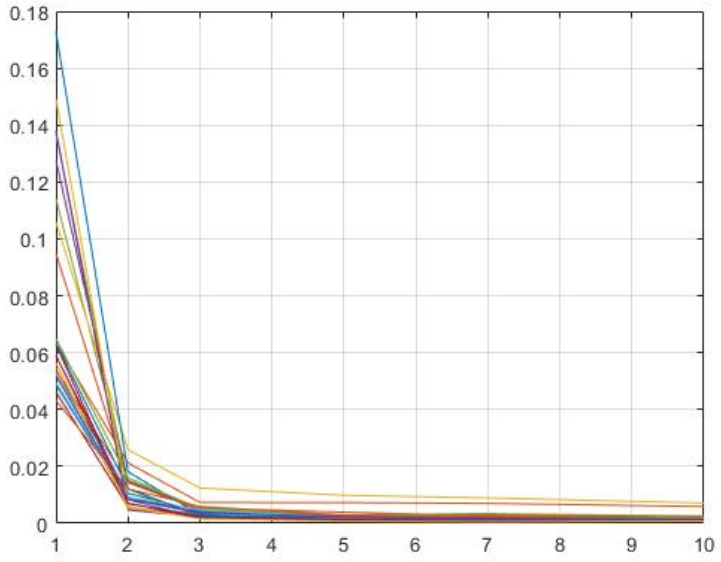


(a) RTS 96 Area 2 Voltage Magnitude

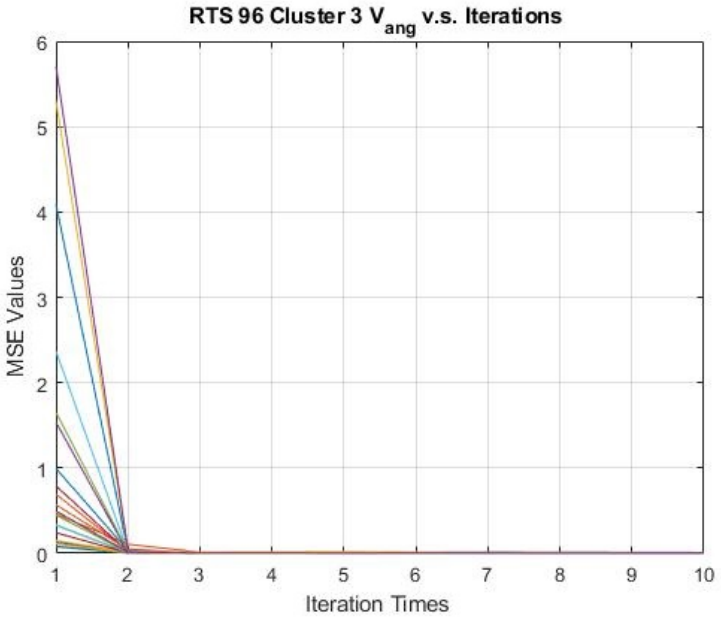


(b) RTS 96 Area 2 Voltage Angle

Figure 4.26: RTS 96 Multi-Area First Level State Estimation MSE for Cluster 2

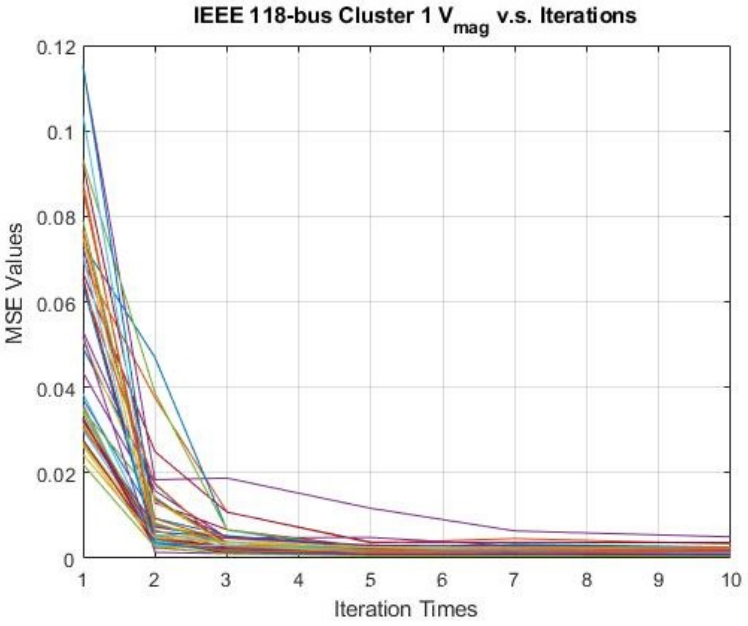


(a) RTS 96 Area 3 Voltage Magnitude

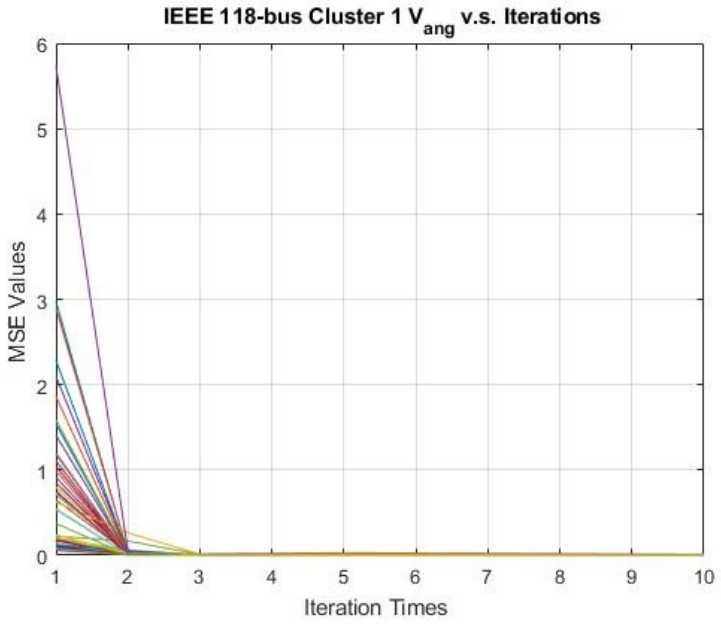


(b) RTS 96 Area 3 Voltage Angle

Figure 4.27: RTS 96 Multi-Area First Level State Estimation MSE for Cluster 3

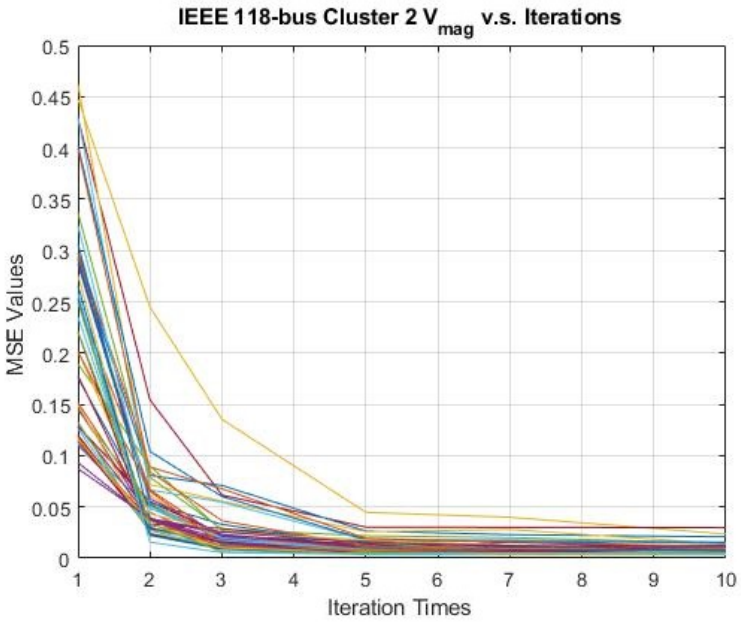


(a) IEEE 118 Area 1 Voltage Magnitude

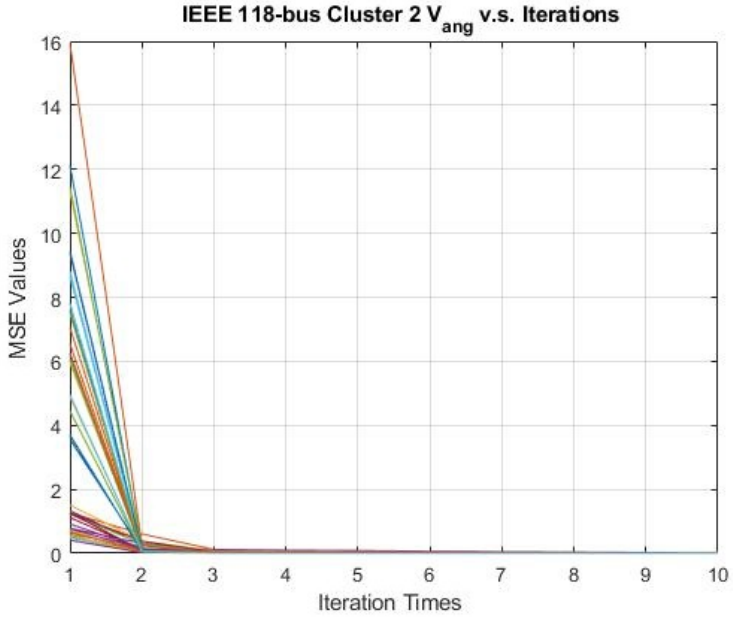


(b) IEEE 118 Area 1 Voltage Angle

Figure 4.28: IEEE 118-bus Multi-Area First Level State Estimation MSE for Cluster 1

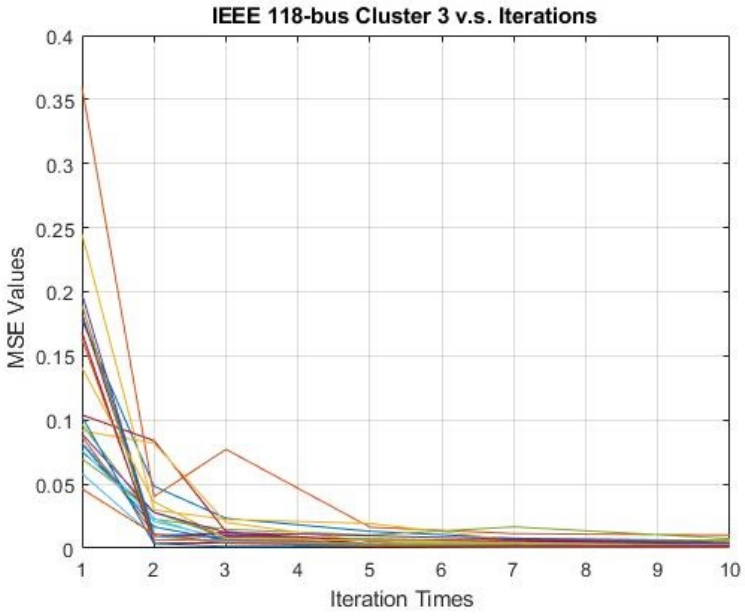


(a) IEEE 118 Area 2 Voltage Magnitude

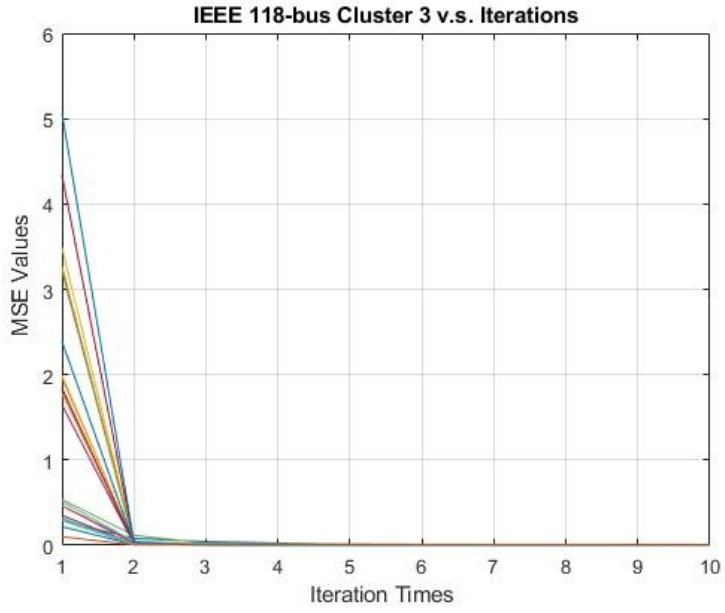


(b) IEEE 118 Area 2 Voltage Angle

Figure 4.29: IEEE 118-bus Multi-Area First Level State Estimation MSE for Cluster 2



(a) IEEE 118 Area 3 Voltage Magnitude



(b) IEEE 118 Area 3 Voltage Angle

Figure 4.30: IEEE 118-bus Multi-Area First Level State Estimation MSE for Cluster 3

From the 3 clusters results, we conclude that both RTS 96 and IEEE 118-bus systems have good convergence performance at the last iteration. Each cluster has good performance on the estimation accuracy as well, whose MSE value is around 0.002 except cluster 2 of IEEE 118-bus, which was a MSE of 0.008. Partition with 4 clusters:

RTS 96 System:

- **Clustering Results**:
  Cluster 1: Total number of buses is 27. The AMRV value is 2.33.
  Cluster 2: Total number of buses is 20. The AMRV value is 2.35.
  Cluster 3: Total number of buses is 15. The AMRV value is 2.33.
  Cluster 4: Total number of buses is 11. The AMRV value is 2.36.

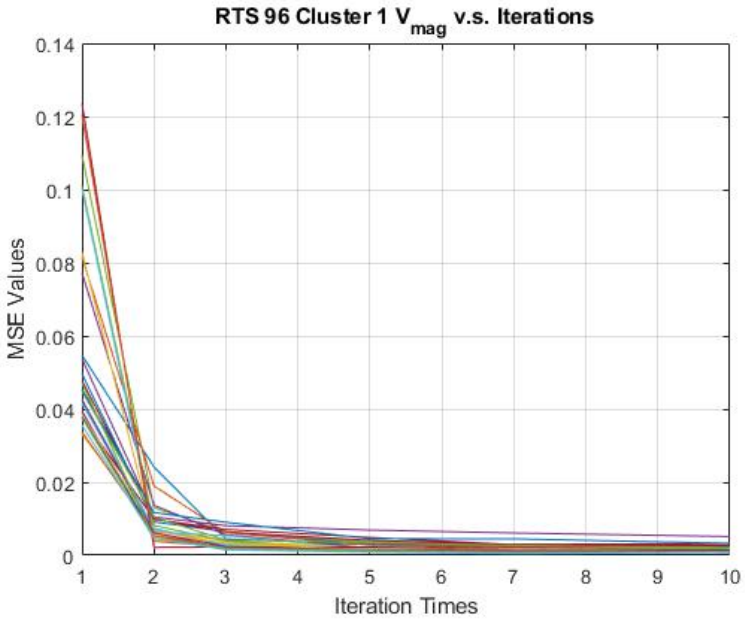
- **MSE Performance**:
  Cluster 1: Reference bus is bus 1. The MSE performance can be seen on Fig 4.31. The average MSE of the voltage magnitude is 0.0017, and the worst case MSE value of the magnitude is 0.0051. The average MSE of the voltage angle is 0.0019, and the worst case MSE value of the angle is 0.0084.
  Cluster 2: Reference bus is bus 21. The MSE performance can be seen on Fig 4.32. The average MSE of the voltage magnitude is 0.0017, and the worst case MSE value of the magnitude is 0.0052. The average MSE of the voltage angle is 0.0029, and the worst case MSE value of the angle is 0.0091.
  Cluster 3: Reference bus is bus 26. The MSE performance can be seen on Fig 4.33. The average MSE of the voltage magnitude is 0.0021, and the worst case MSE value of the magnitude is 0.007. The average MSE of the voltage angle is 0.0025, and the worst case MSE value of the angle is 0.0079.
  Cluster 4: Reference bus is bus 50. The MSE performance can be seen on Fig 4.34. The average MSE of the voltage magnitude is 0.0016, and the worst case MSE value of the magnitude is 0.0047. The average MSE of the voltage angle is 0.0020, and the worst case MSE value of the angle is 0.0058.

IEEE 118-bus System:

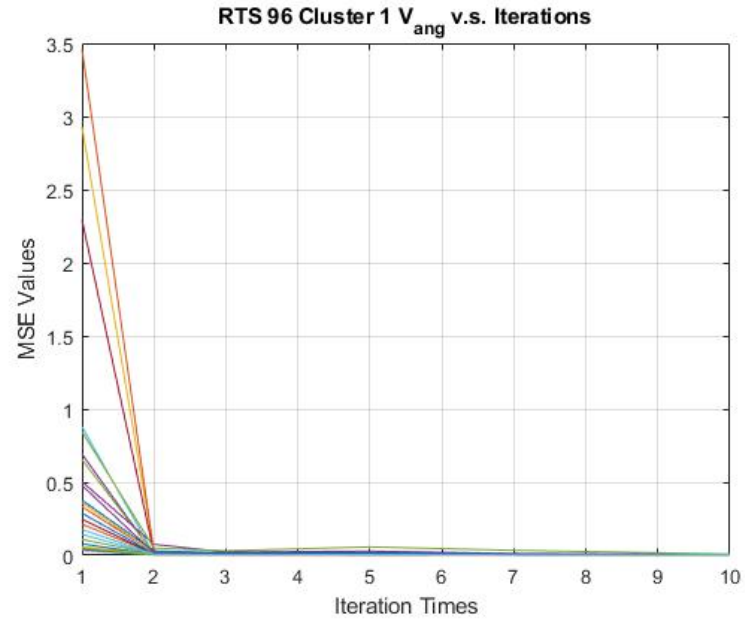
- **Clustering Results**:
  Cluster 1: Total number of buses is 35. The AMRV value is 1.86.

Cluster 2: Total number of buses is 34. The AMRV value is 1.85.
Cluster 3: Total number of buses is 11. The AMRV value is 1.82.
Cluster 4: Total number of buses is 38. The AMRV value is 1.82.

- **MSE Performance**:
  Cluster 1: Reference bus is bus 1. The MSE performance can be seen on Fig 4.35. The average MSE of the voltage magnitude is 0.0020, and the worst case MSE value of the magnitude is 0.005. The average MSE of the voltage angle is 0.0023, and the worst case MSE value of the angle is 0.0143.
  Cluster 2: Reference bus is bus 45. The MSE performance can be seen on Fig 4.36. The average MSE of the voltage magnitude is 0.0024, and the worst case MSE value of the magnitude is 0.0054. The average MSE of the voltage angle is 0.0021, and the worst case MSE value of the angle is 0.0074.
  Cluster 3: Reference bus is bus 33. The MSE performance can be seen on Fig 4.37. The average MSE of the voltage magnitude is 0.0025, and the worst case MSE value of the magnitude is 0.0048. The average MSE of the voltage angle is 0.0016, and the worst case MSE value of the angle is 0.0026.
  Cluster 4: Reference bus is bus 75. The MSE performance can be seen on Fig 4.38. The average MSE of the voltage magnitude is 0.0018, and the worst case MSE value of the magnitude is 0.005. The average MSE of the voltage angle is 0.0019, and the worst case MSE value of the angle is 0.0073.

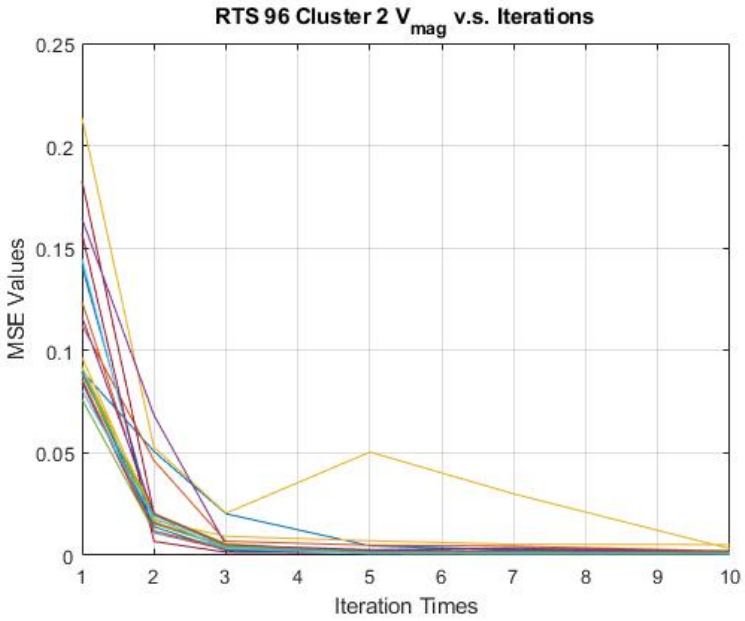


(a) RTS 96 Area 1 Voltage Magnitude

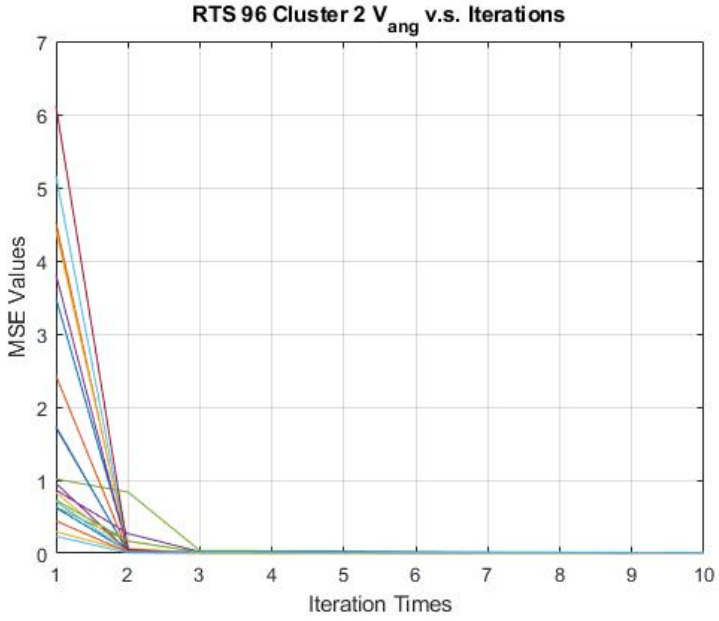


(b) RTS 96 Area 1 Voltage Angle

Figure 4.31: RTS 96 Multi-Area First Level State Estimation MSE for Cluster 1

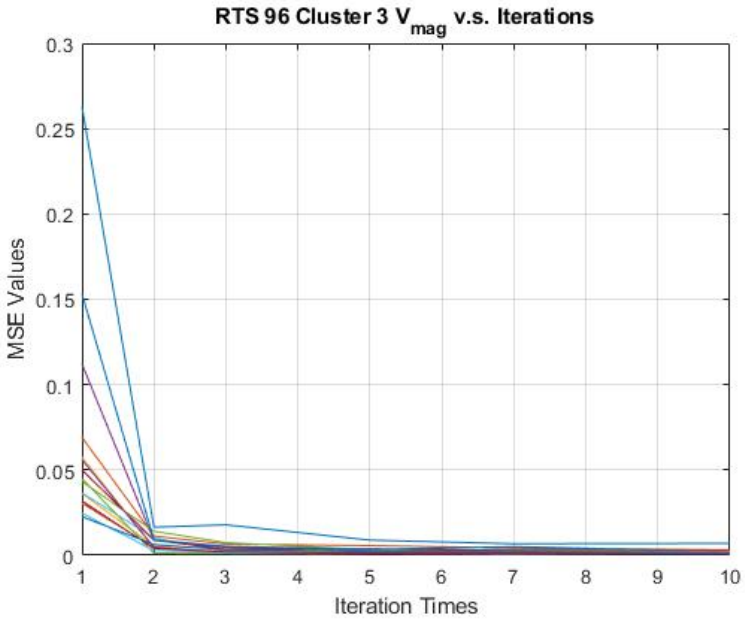


(a) RTS 96 Area 2 Voltage Magnitude

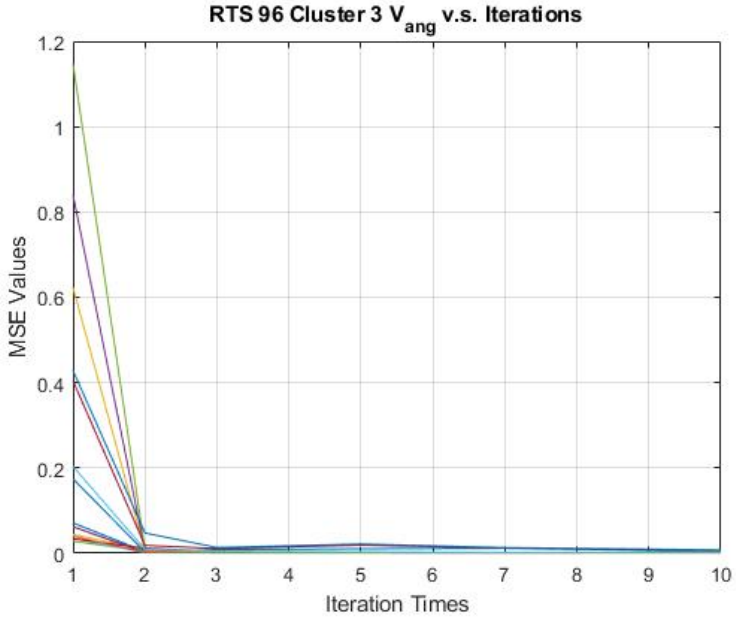


(b) RTS 96 Area 2 Voltage Angle

Figure 4.32: RTS 96 Multi-Area First Level State Estimation MSE for Cluster 2

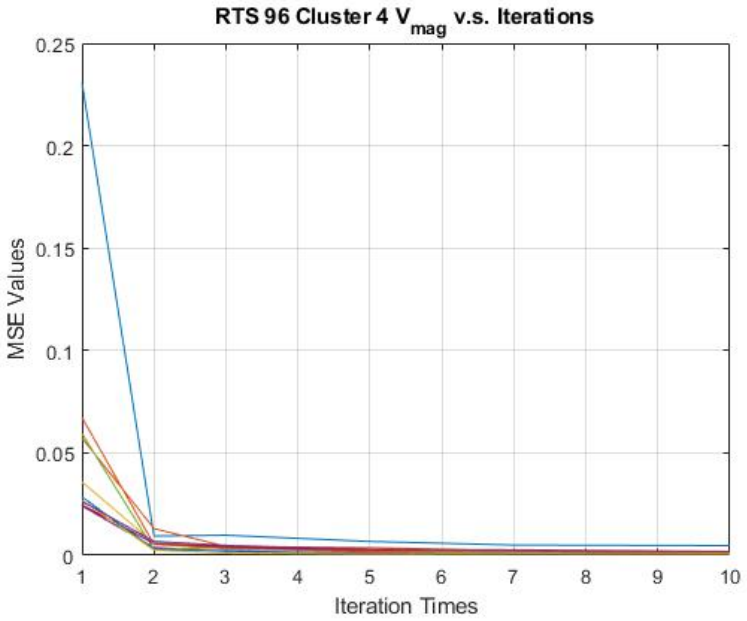


(a) RTS 96 Area 3 Voltage Magnitude

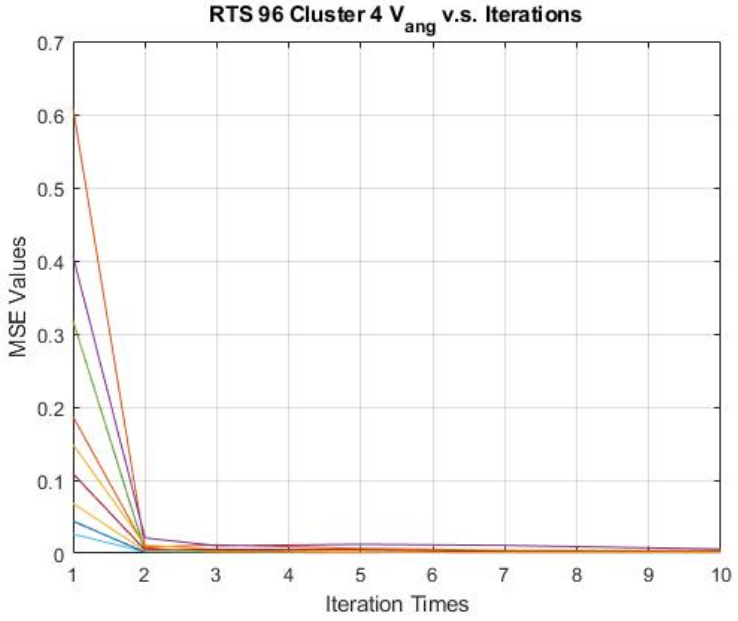


(b) RTS 96 Area 3 Voltage Angle

Figure 4.33: RTS 96 Multi-Area First Level State Estimation MSE for Cluster 3

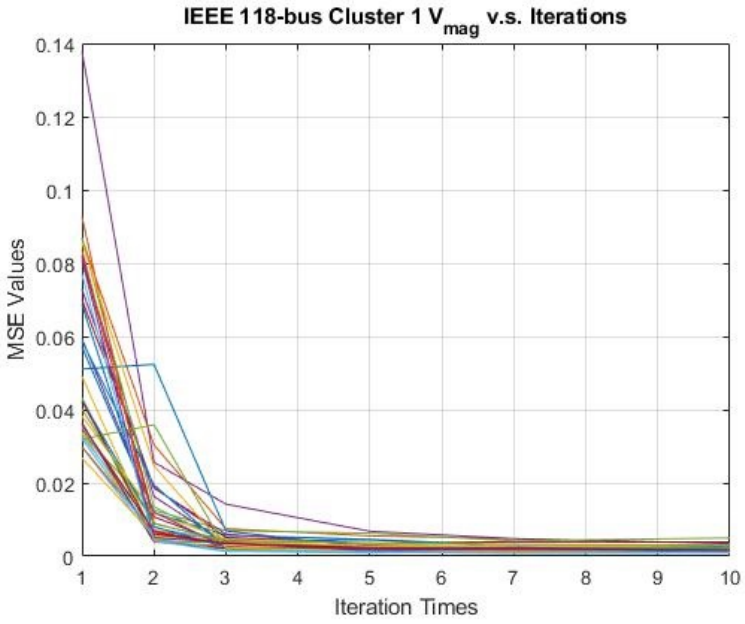


(a) RTS 96 Area 4 Voltage Magnitude

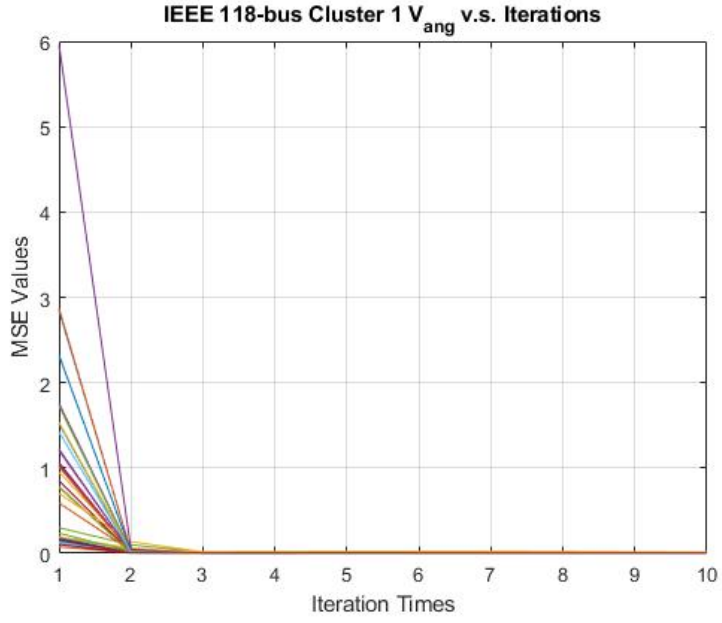


(b) RTS 96 Area 4 Voltage Angle

Figure 4.34: RTS 96 Multi-Area First Level State Estimation MSE for Cluster 4

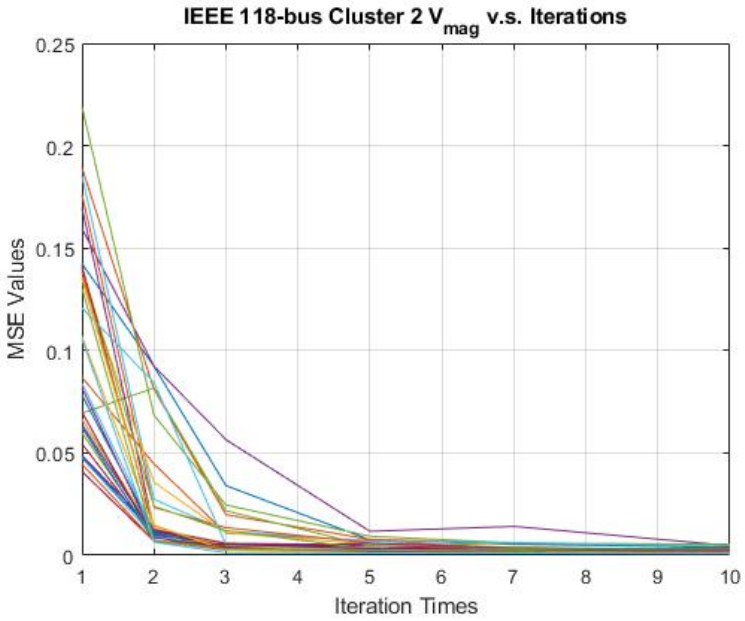


(a) IEEE 118 Area 1 Voltage Magnitude

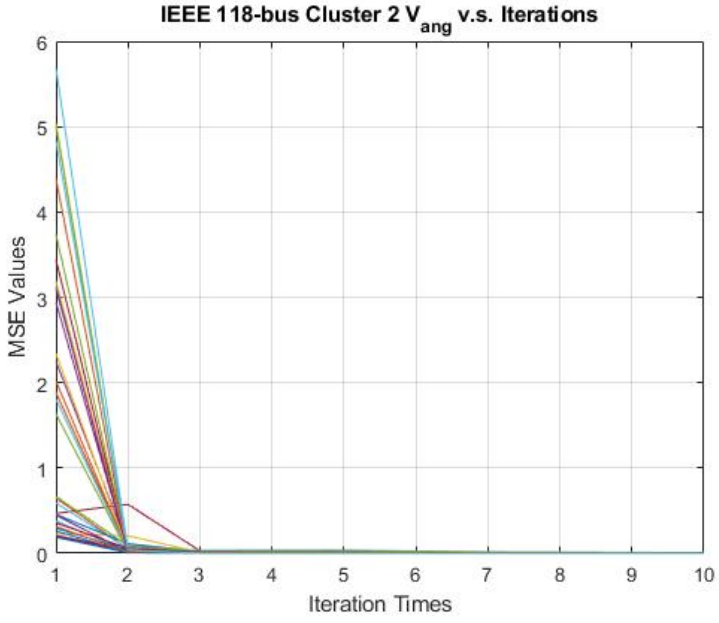


(b) IEEE 118 Area 1 Voltage Angle

Figure 4.35: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster 1

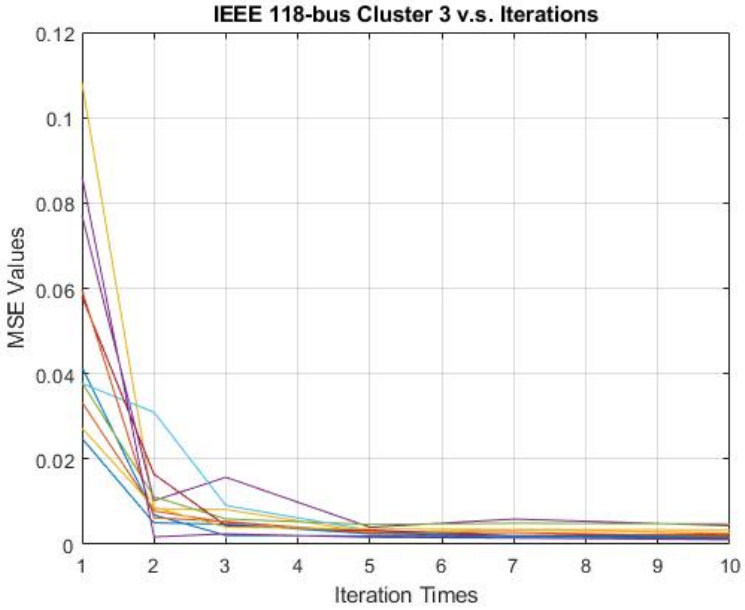


(a) IEEE 118 Area 2 Voltage Magnitude

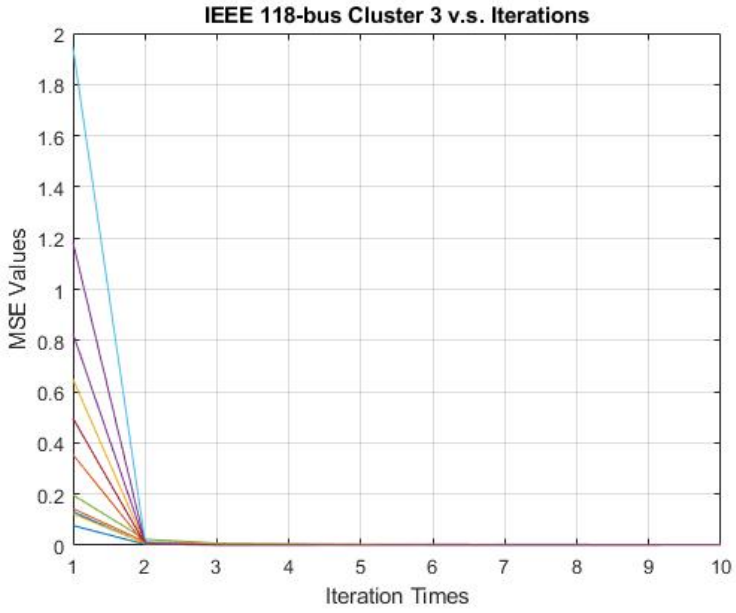


(b) IEEE 118 Area 2 Voltage Angle

Figure 4.36: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster 2

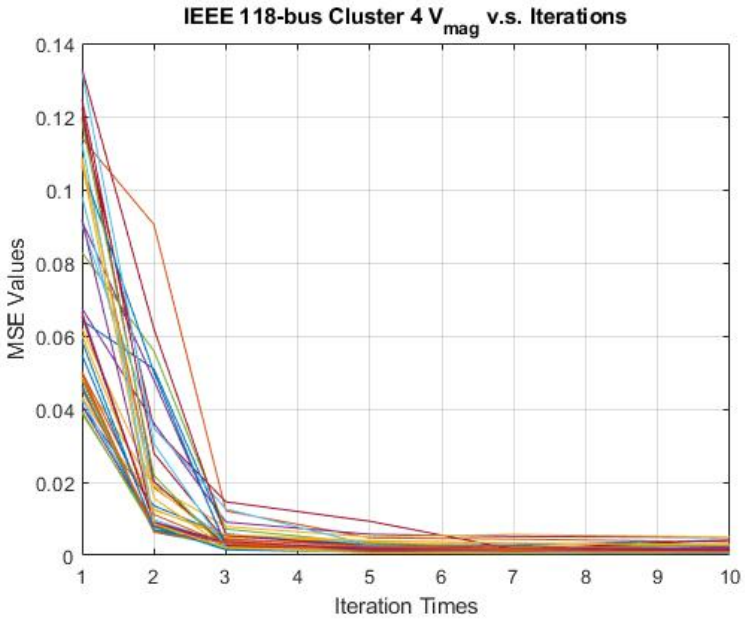


(a) IEEE 118 Area 3 Voltage Magnitude

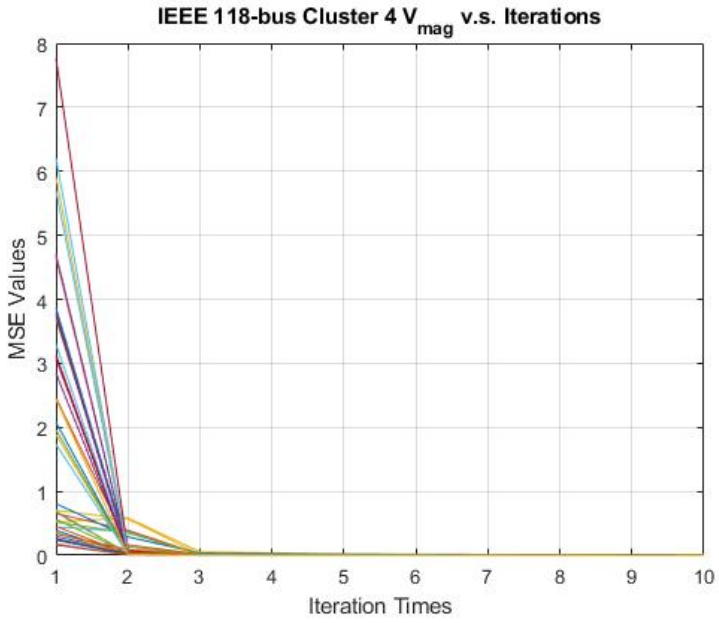


(b) IEEE 118 Area 3 Voltage Angle

Figure 4.37: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster 3



(a) IEEE 118 Area 4 Voltage Magnitude



(b) IEEE 118 Area 4 Voltage Angle

Figure 4.38: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster 4

The following results for the state estimation for 5 sub-systems.

RTS 96 System:

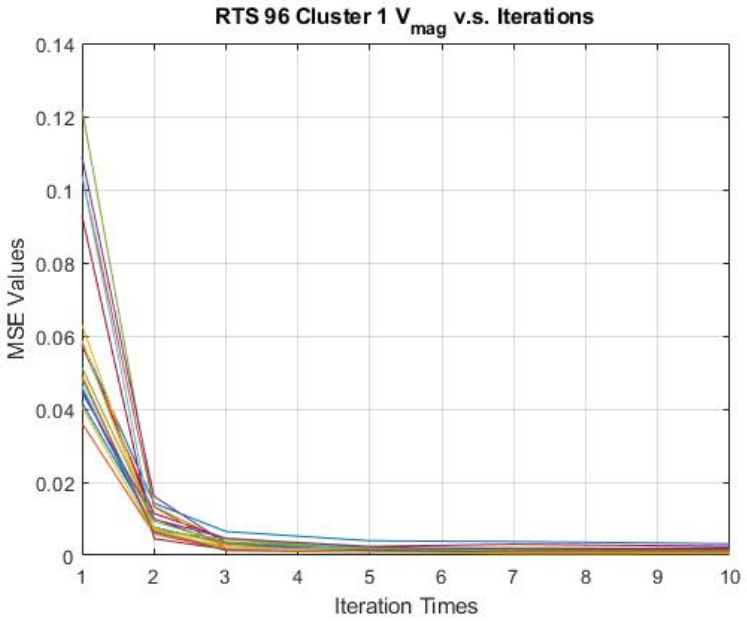
- **Clustering Results**:
  Cluster 1: Total number of buses is 19. The AMRV value is 2.26.
  Cluster 2: Total number of buses is 13. The AMRV value is 2.31.
  Cluster 3: Total number of buses is 17. The AMRV value is 2.41.
  Cluster 4: Total number of buses is 10. The AMRV value is 2.30.
  Cluster 5: Total number of buses is 14. The AMRV value is 2.36.

- **The MSE Performance**:
  Cluster 1: Reference bus is bus 1. The MSE performance can be seen on Fig 4.39. The average MSE of the voltage magnitude is 0.0014, and the worst case MSE value of the magnitude is 0.0032. The average MSE of the voltage angle is 0.0021, and the worst case MSE value of the angle is 0.0085.
  Cluster 2: Reference bus is bus 7. The MSE performance can be seen on Fig 4.40. The average MSE of the voltage magnitude is 0.0016, and the worst case MSE value of the magnitude is 0.0044. The average MSE of the voltage angle is 0.0012, and the worst case MSE value of the angle is 0.0023.
  Cluster 3: Reference bus is bus 23. The MSE performance can be seen on Fig 4.41. The average MSE of the voltage magnitude is 0.0027, and the worst case MSE value of the magnitude is 0.0099. The average MSE of the voltage angle is 0.0034, and the worst case MSE value of the angle is 0.0095.
  Cluster 4: Reference bus is bus 50. The MSE performance can be seen on Fig 4.42. The average MSE of the voltage magnitude is 0.0017, and the worst case MSE value of the magnitude is 0.0047. The average MSE of the voltage angle is 0.0015, and the worst case MSE value of the angle is 0.0027.
  Cluster 4: Reference bus is bus 62. The MSE performance can be seen on Fig 4.43. The average MSE of the voltage magnitude is 0.0017, and the worst case MSE value of the magnitude is 0.0072. The average MSE of the voltage angle is 0.0022, and the worst case MSE value of the angle is 0.0053.

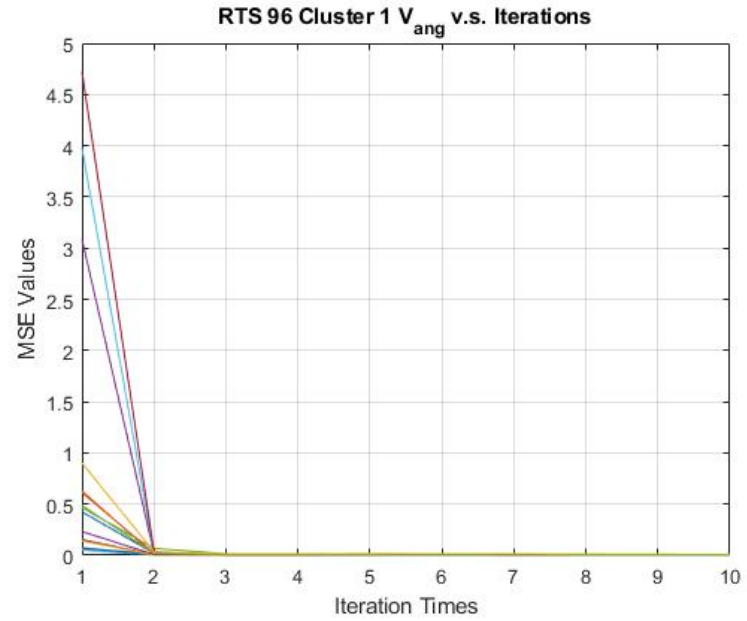
IEEE 118-bus System:

- **Clustering Results**:
  Cluster 1: Total number of buses is 16. The AMRV value is 2.00.
  Cluster 2: Total number of buses is 26. The AMRV value is 1.69.
  Cluster 3: Total number of buses is 21. The AMRV value is 1.90.
  Cluster 4: Total number of buses is 27. The AMRV value is 1.81.
  Cluster 5: Total number of buses is 28. The AMRV value is 1.68.

- **The MSE Performance**:
  Cluster 1: Reference bus is bus 1. The MSE performance can be seen on Fig. 4.44. The average MSE of the voltage magnitude is 0.0018, and the worst case MSE value of the magnitude is 0.0037. The average MSE of the voltage angle is 0.0032, and the worst case MSE value of the angle is 0.0147.
  Cluster 2: Reference bus is bus 15. The MSE performance can be seen on Fig. 4.45. The average MSE of the voltage magnitude is 0.0020, and the worst case MSE value of the magnitude is 0.0045. The average MSE of the voltage angle is 0.0013, and the worst case MSE value of the angle is 0.0031.
  Cluster 3: Reference bus is bus 68. The MSE performance can be seen on Fig. 4.46. The average MSE of the voltage magnitude is 0.0036, and the worst case MSE value of the magnitude is 0.0078. The average MSE of the voltage angle is 0.0026, and the worst case MSE value of the angle is 0.0076.
  Cluster 4: Reference bus is bus 40. The MSE performance can be seen on Fig. 4.47. The average MSE of the voltage magnitude is 0.0034, and the worst case MSE value of the magnitude is 0.0109. The average MSE of the voltage angle is 0.0024, and the worst case MSE value of the angle is 0.0068.
  Cluster 5: Reference bus is bus 83. The MSE performance can be seen on Fig. 4.48. The average MSE of the voltage magnitude is 0.0010, and the worst case MSE value of the magnitude is 0.0019. The average MSE of the voltage angle is 0.0020, and the worst case MSE value of the angle is 0.0085.

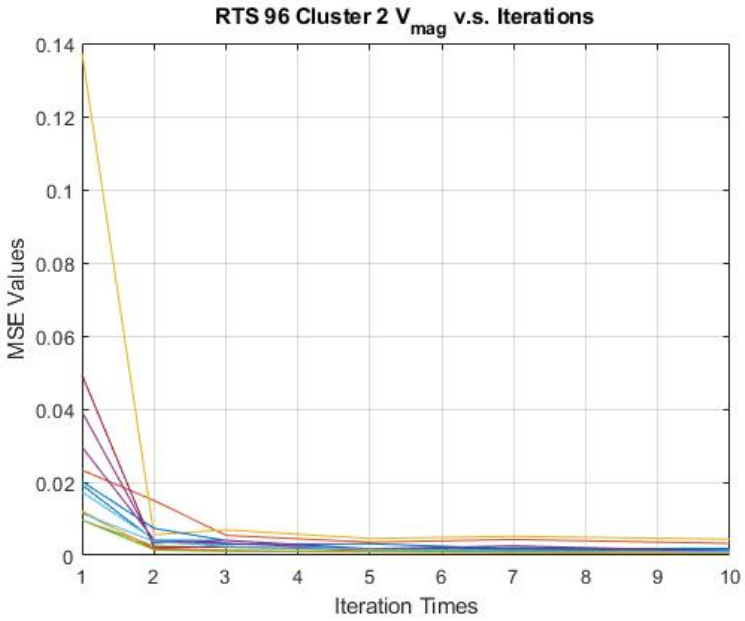


(a) RTS 96 Area 1 Voltage Magnitude

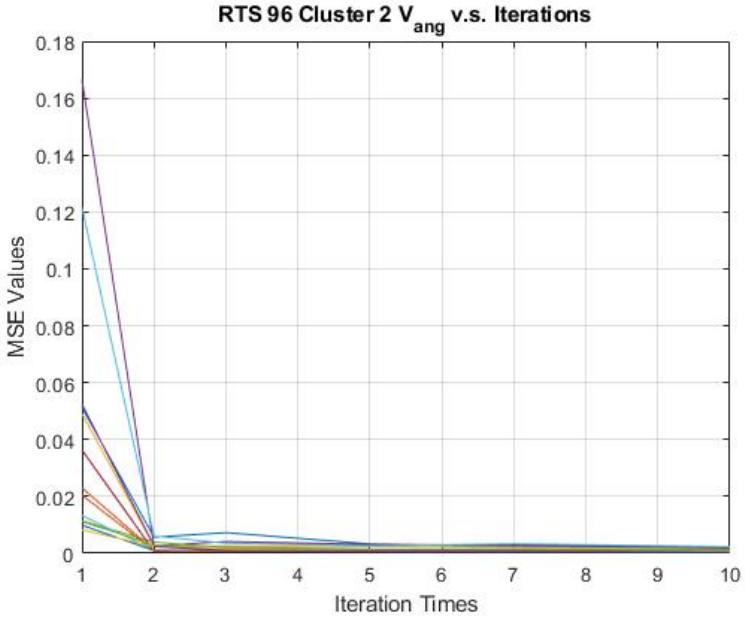


(b) RTS 96 Area 1 Voltage Angle

Figure 4.39: RTS 96 Multi-Area First Level State Estimation MSE for Cluster No.1

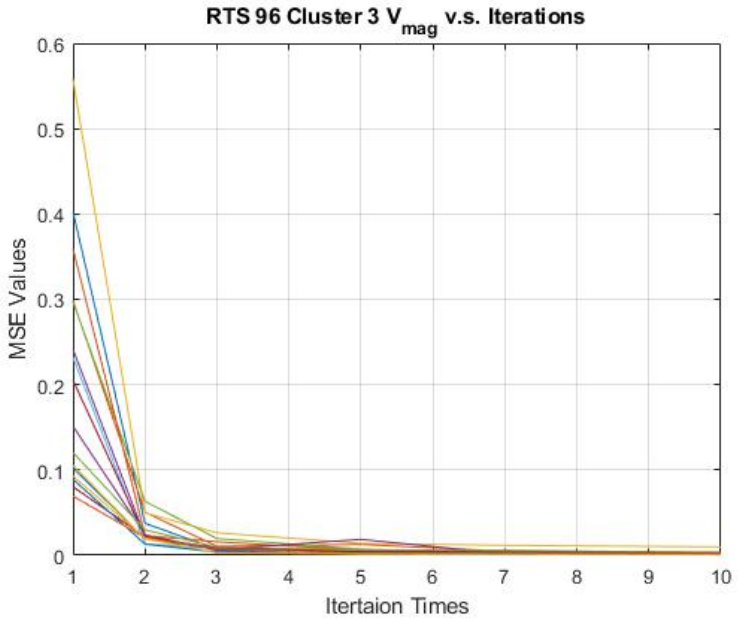


(a) RTS 96 Area 2 Voltage Magnitude

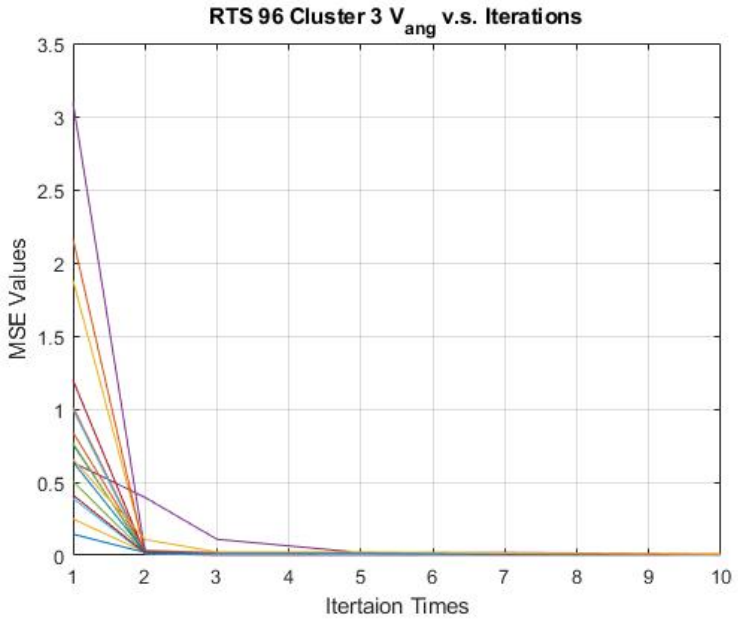


(b) RTS 96 Area 2 Voltage Angle

Figure 4.40: RTS 96 Multi-Area First Level State Estimation MSE for Cluster No.2

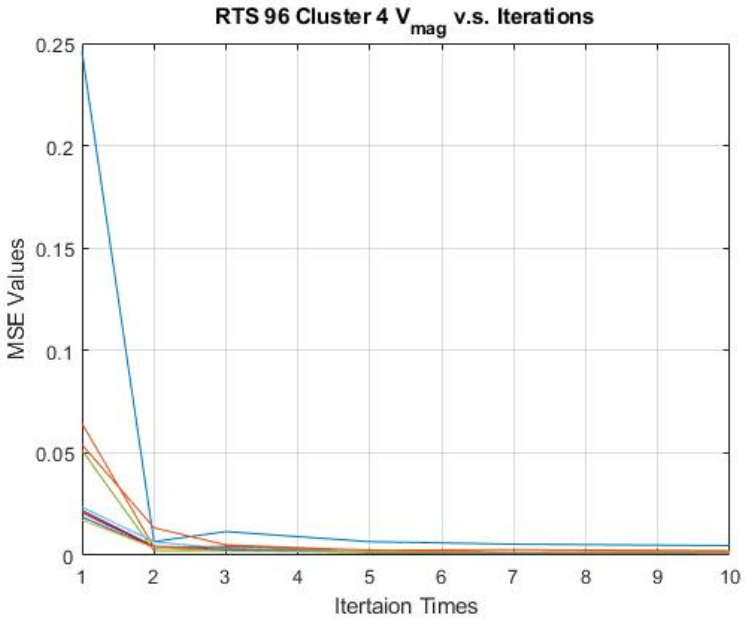


(a) RTS 96 Area 3 Voltage Magnitude

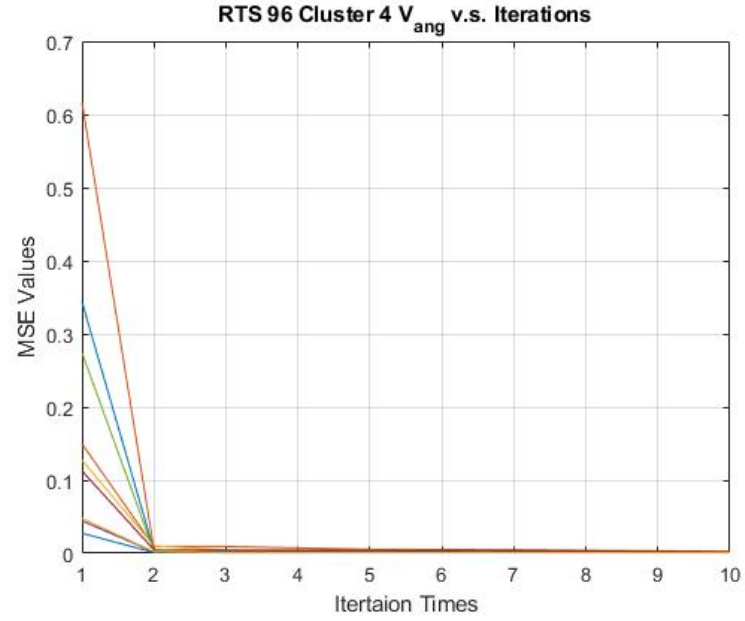


(b) RTS 96 Area 3 Voltage Angle

Figure 4.41: RTS 96 Multi-Area First Level State Estimation MSE for Cluster No.3

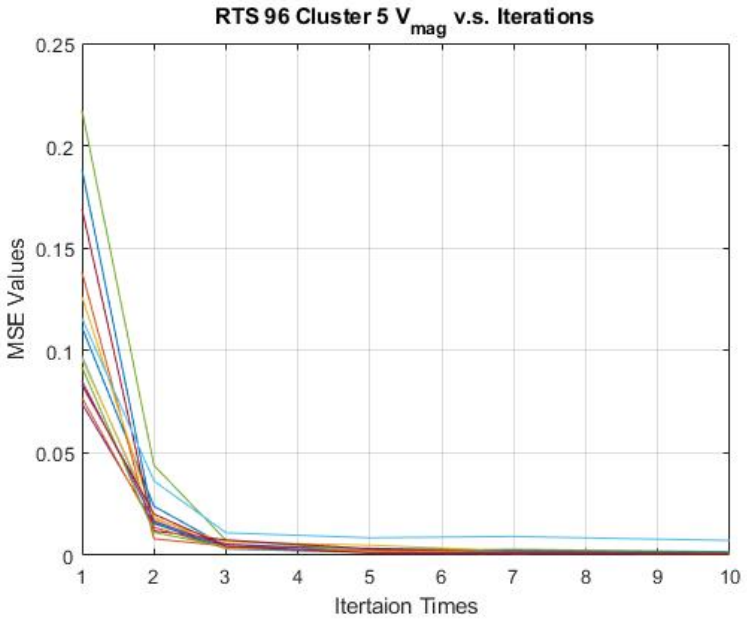


(a) RTS 96 Area 4 Voltage Magnitude

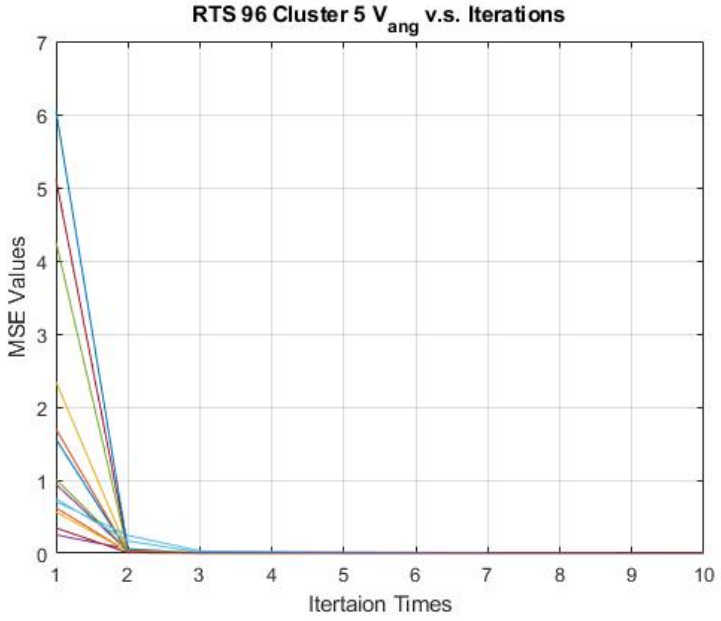


(b) RTS 96 Area 4 Voltage Angle

Figure 4.42: RTS 96 Multi-Area First Level State Estimation MSE for Cluster No.4

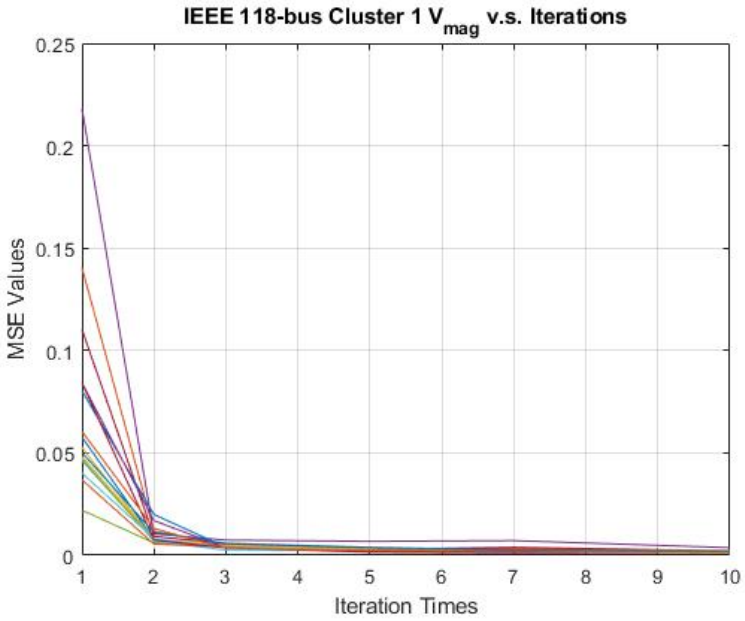


(a) RTS 96 Area 5 Voltage Magnitude

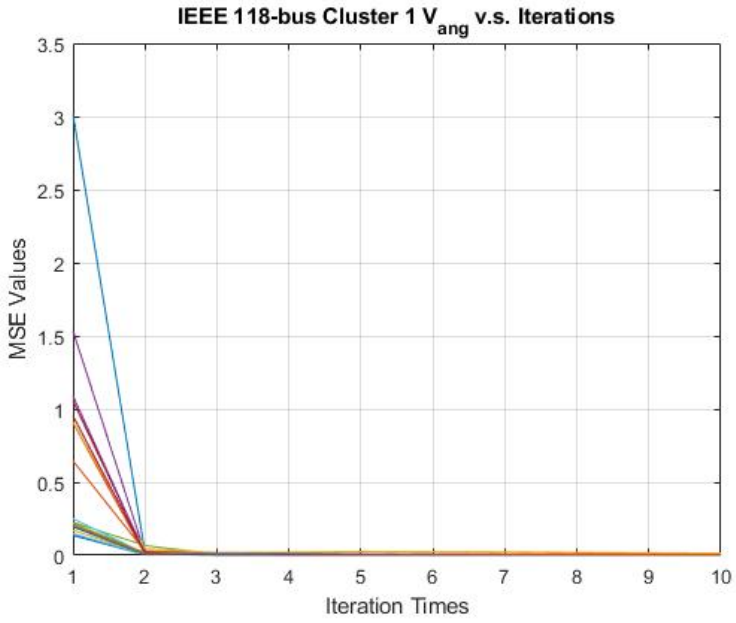


(b) RTS 96 Area 5 Voltage Angle

Figure 4.43: RTS 96 Multi-Area First Level State Estimation MSE for Cluster No.5

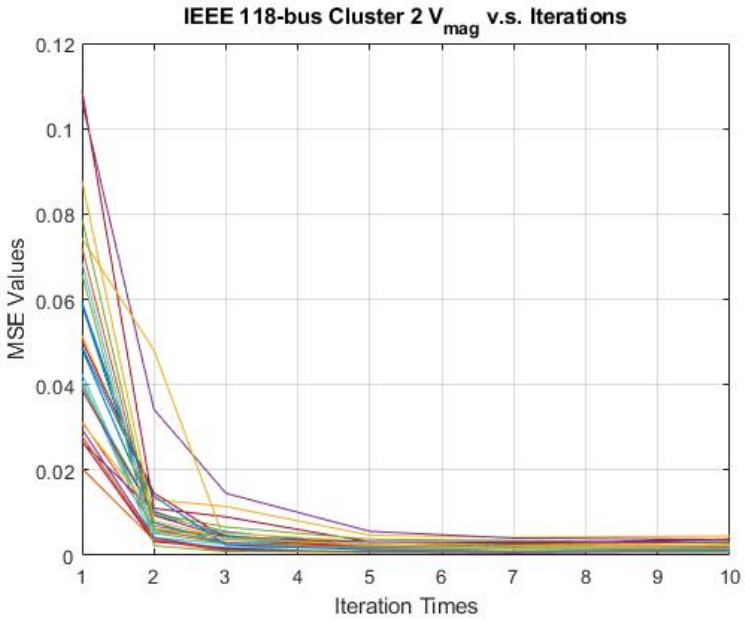


(a) IEEE 118 Area 1 Voltage Magnitude

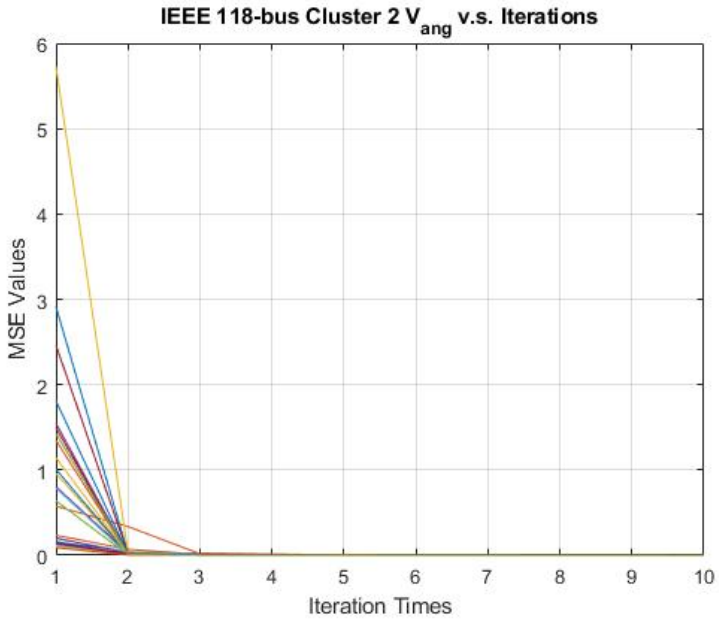


(b) IEEE 118 Area 1 Voltage Angle

Figure 4.44: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster No.1

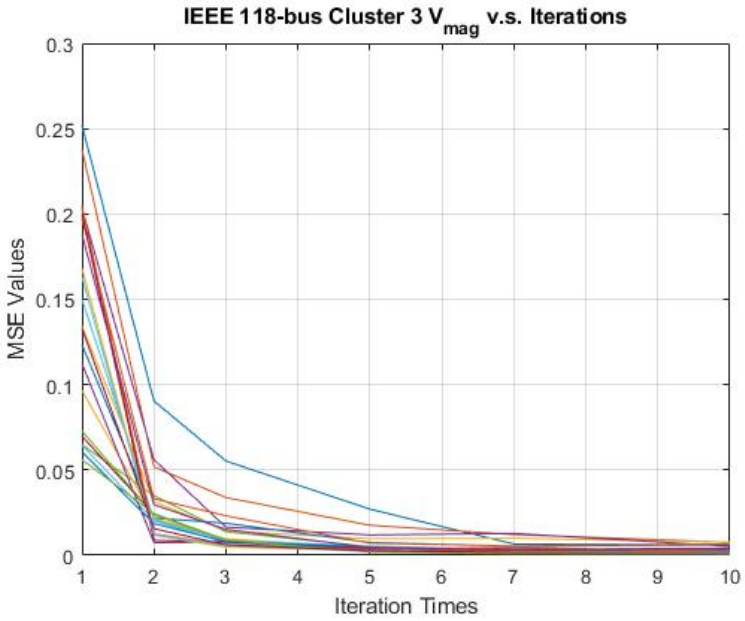


(a) IEEE 118 Area 2 Voltage Magnitude

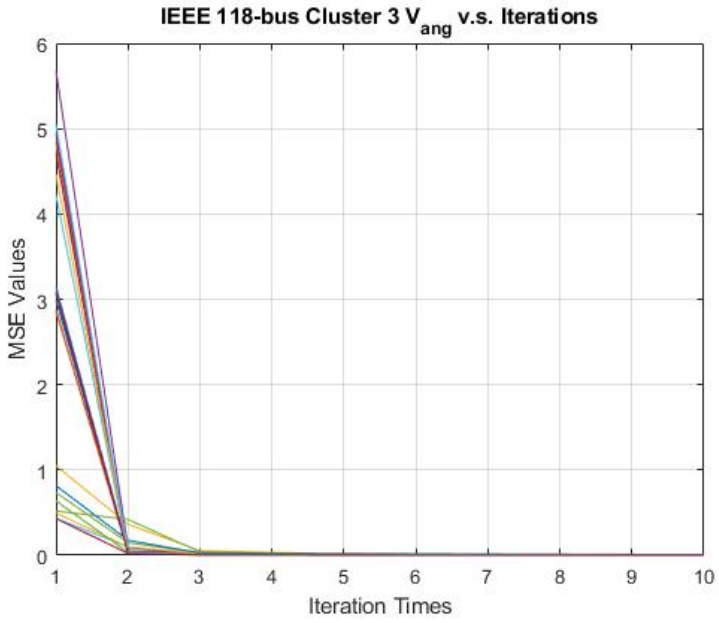


(b) IEEE 118 Area 2 Voltage Angle

Figure 4.45: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster No.2

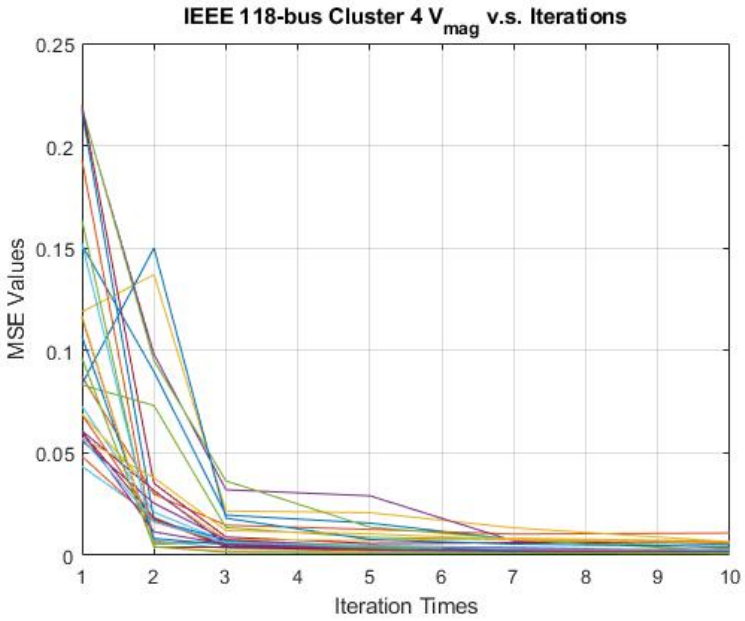


(a) IEEE 118 Area 3 Voltage Magnitude

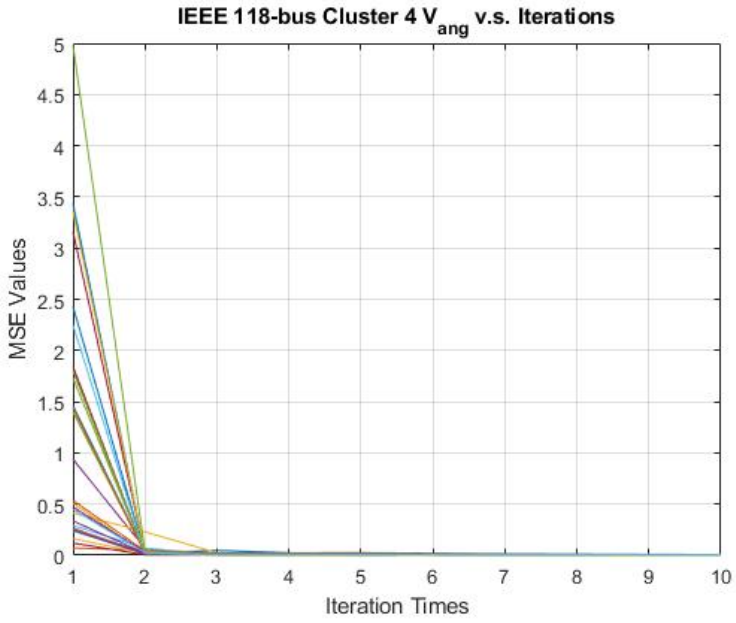


(b) IEEE 118 Area 3 Voltage Angle

Figure 4.46: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster No.3

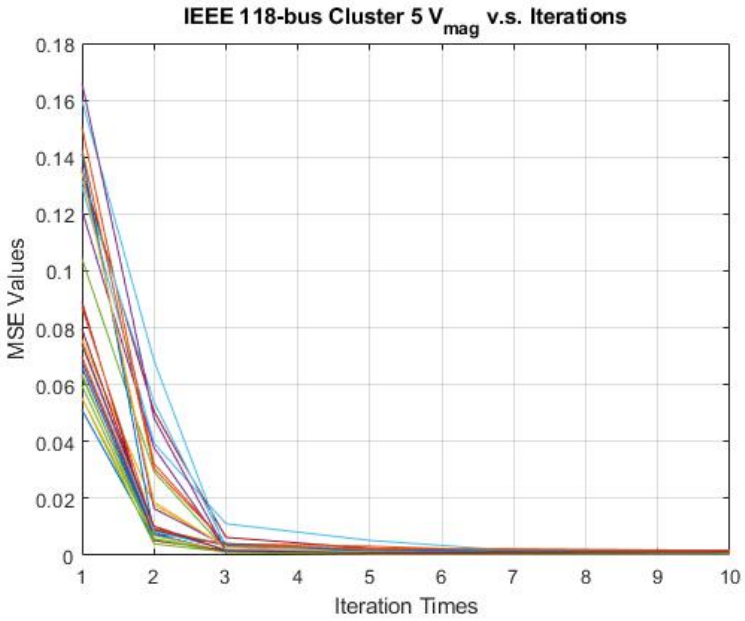


(a) IEEE 118 Area 4 Voltage Magnitude

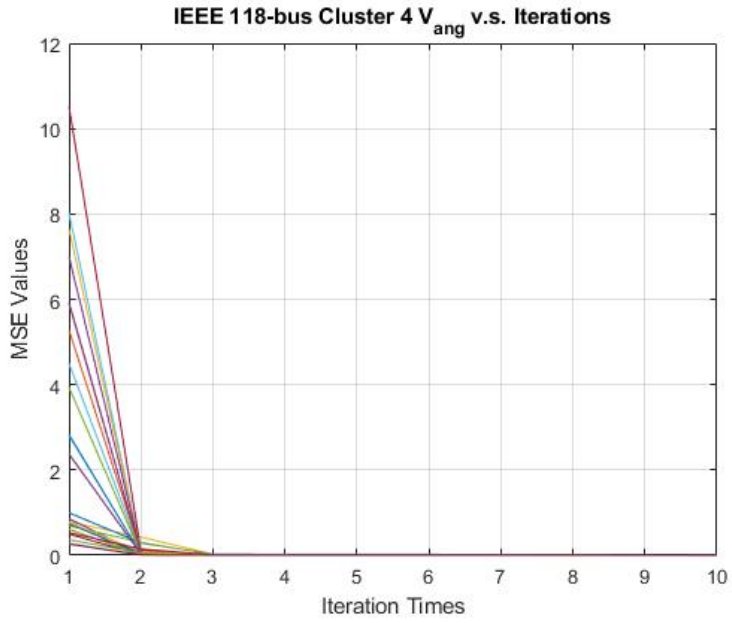


(b) IEEE 118 Area 4 Voltage Angle

Figure 4.47: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster No.4



(a) IEEE 118 Area 5 Voltage Magnitude



(b) IEEE 118 Area 5 Voltage Angle

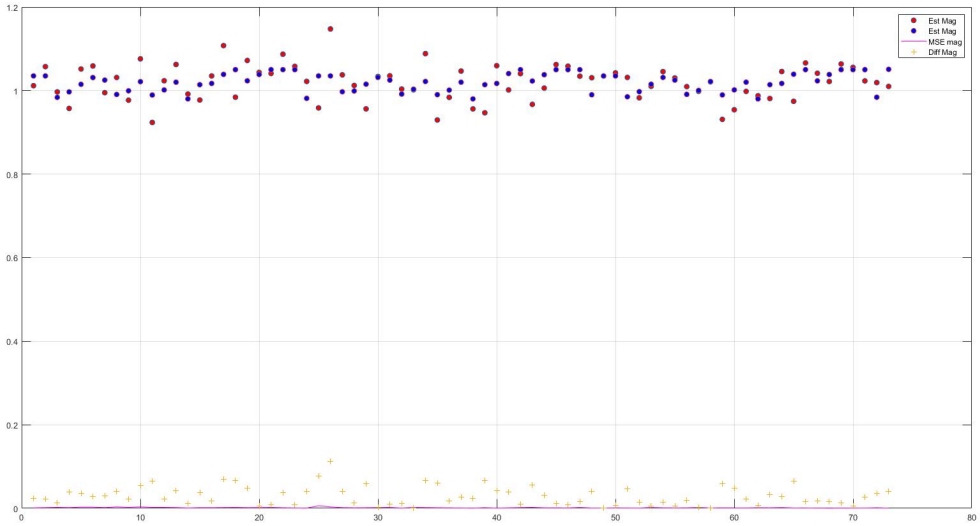
Figure 4.48: IEEE 118 Multi-Area First Level State Estimation MSE for Cluster No.5

It can be seen from the results presented above that good convergence and estimation accuracy is achieved for for each cluster. The MSE values are different between the 4 clusters and 5 clusters cases. For RTS 96 system, the voltage magnitude MSE values concentrate on the value around 0.002 for both cases. The voltage angle MSE values have the range from 0.0015 to 0.0034 for these two cases. For IEEE 118-bus system, the voltage magnitude MSE values concentrate on the value around 0.002 as well for the 4 clusters case, and it has the range from 0.0010 to 0.0036. The voltage angle MSE values are approximately 0.002 for each cluster for 4 sub-systems case, and it has range from 0.0013 to 0.0032. Moreover, it can be concluded that the highest average MSE values occur on the 4 clusters case for both RTS 96 and IEEE 118-bus system. Meanwhile, the smallest *AMRV* difference occurs on the 4 clusters case as well.

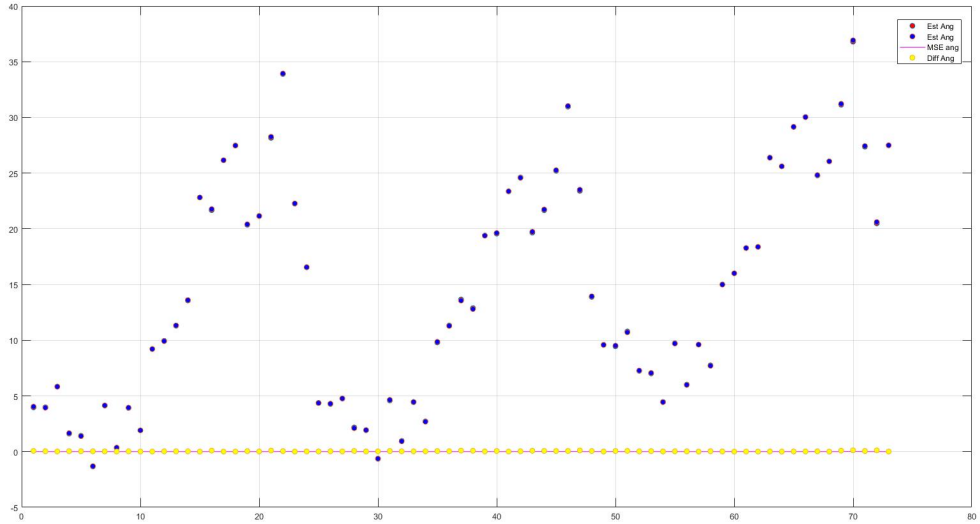
4.3.2.2 Central Coordinate Level MASE

After the evaluation of the first level sub-system state estimation, we now evaluate the second level state estimation performance, which is also called the central coordinate level. As indicated in the previous chapter, the angle adjustment on the local slack bus will be the only action in the second level. According to the results presented in [19, 18, 20], 3 clusters after partitioning would be the typical case to be implemented evaluation. At this point, the 3 clusters first level state estimation results will be used. The evaluation will be implemented on the RTS 96 system and the IEEE 118-bus systems.

Figure 4.49 presents the second level estimation result for RTS 96. The figure presents the actual state variable values, estimated values, the difference between the actual and the estimated value, and the corresponding MSE value. As can be seen in these figures, the maximum difference is 0.1 for both voltage magnitude and angle. Although the maximum difference for the magnitude and the angle are the same, the angle has good performance at the last iteration. It can be seen that they almost match the actual value. For the magnitude estimation, it has good estimation results for the most part except buses from number 10 to number 30, who have the largest difference values around 0.1.



(a) RTS 96 Second Level Voltage Magnitude



(b) RTS 96 Second Level Voltage Angle

Figure 4.49: RTS 96 Multi-Area Second Level State Estimation Performance

For the reason that the voltage angle does not have the obvious bias either by MASE or centralized SE, the voltage magnitude is the only state variable being compared. Figure 4.50 presents the result of the centralized state estimation. The centralized state estimation employs the proposed noise statistic estimation technique based on CKF method. As seen from the results, the largest estimation bias is 0.259, which is obviously larger than that the results by using modified multi-area state estimation. Besides, it can be seen from the results that the majority of the bias values are located above 0.1. It leads to the observation that there are more places having estimation bias for centralized state estimation than that for the multi-area state estimation.

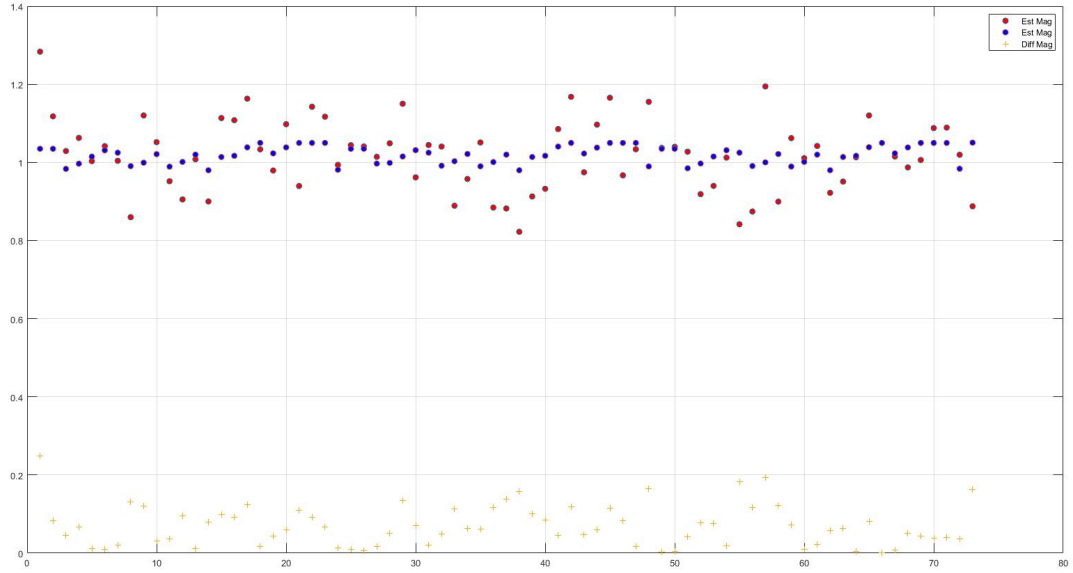
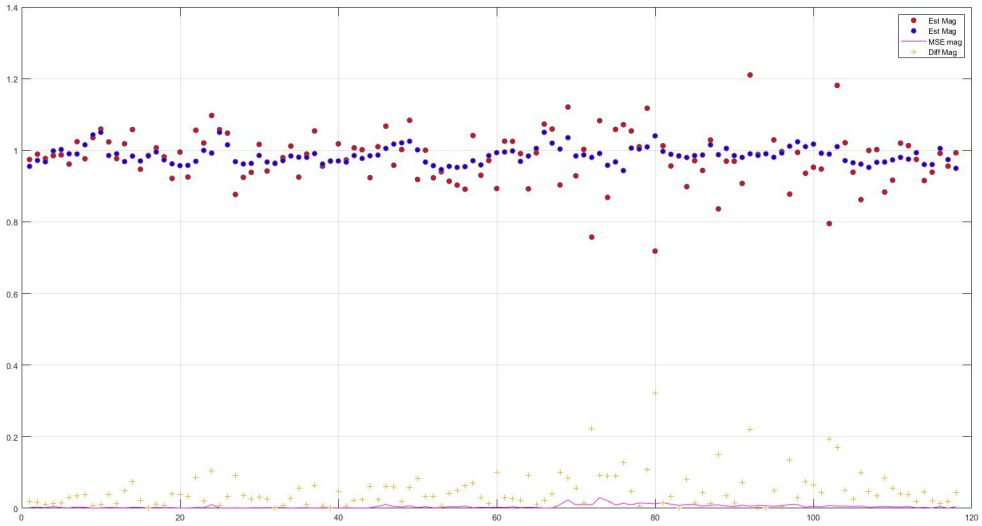
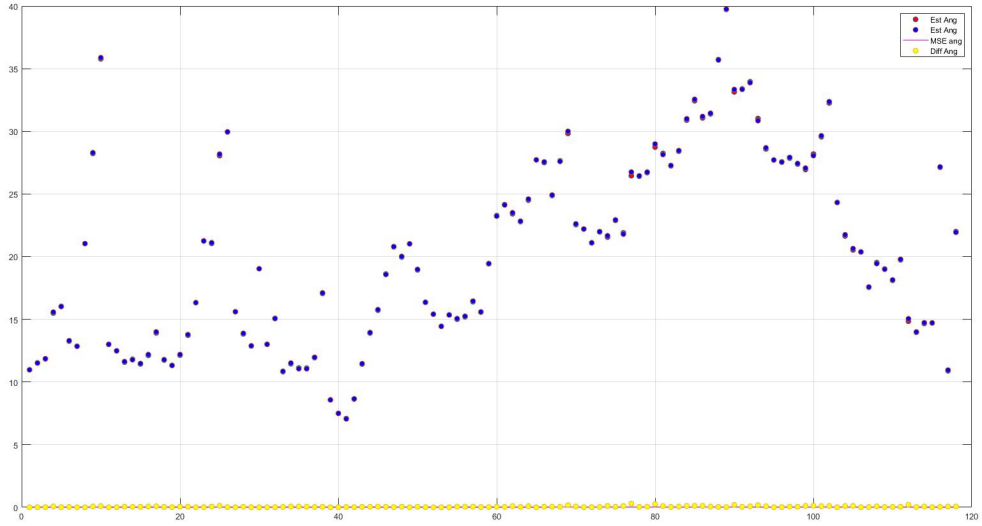


Figure 4.50: Centralized State Estimation Performance For RTS 96 System

Figure 4.51 presents the second level estimation result for IEEE 118-bus. In the figure, it presents the actual state variable values, estimated values, the difference between the actual value and the estimated value, and the corresponding value. As can be seen in this figure, the maximum difference is 0.32 for the voltage magnitude and 0.29 for the angle. Similar to the RTS 96, angle estimation performs well. For the magnitude estimation, the worst estimation performance occurs at the parts from bus 70 to bus 90, which happens to be located in area 2.



(a) IEEE 118 Second Level Voltage Magnitude



(b) IEEE 118 Second Level Voltage Angle

Figure 4.51: IEEE 118 Multi-Area Second Level State Estimation MSE

The centralized SE result is presented as Fig. 4.52. Similar as the presentation of Fig. 4.50, Fig 4.52 presents the centralized SE results of voltage magnitude, the actual state variable results of voltage magnitude, and the SE bias values voltage magnitude. By comparing the centralized SE and the SE results from central coordination level of MASE, it can be seen that the centralized SE has larger estimation bias than that for the MASE second level result, and the estimation difference is relatively significant for all buses. Although the largest estimation bias for the centralized SE is 0.36, which has only 0.04 difference than that for MASE result, there are less bus having estimation bias less than 0.1.

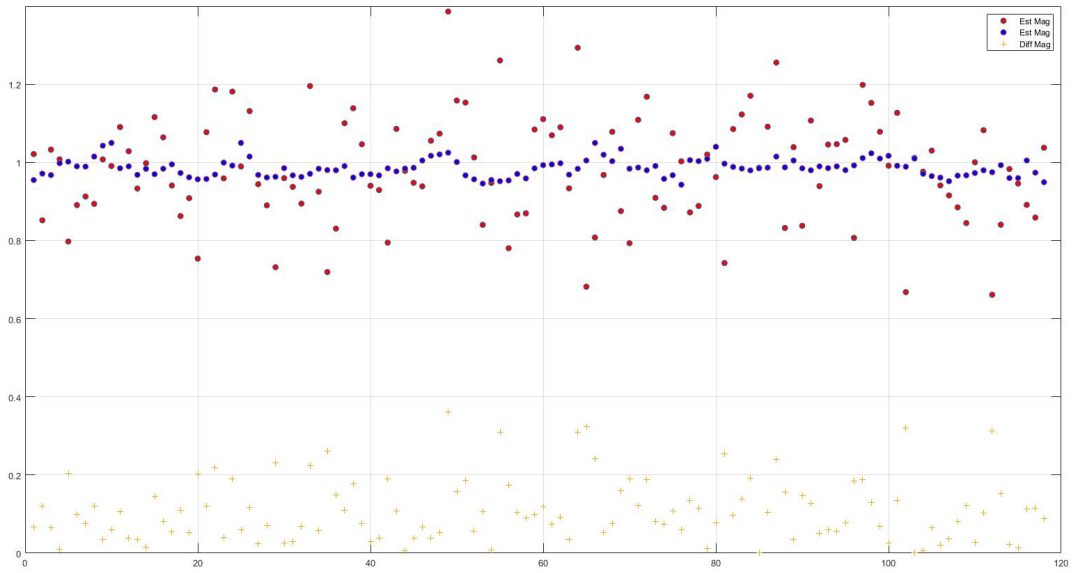


Figure 4.52: Centralized State Estimation Performance For IEEE 118-bus System

By analyzing the estimation performance on RTS96 and IEEE 118-bus systems, we conclude that the MASE of our research meets the feature of the expected MASE, which are higher accuracy and less required exchanging of data.

Chapter 5: Conclusion

The main purpose of this research was to improve the performance of MASE by developing a partitioning method with consideration of the impact on the state estimation. System observability was an important factor for the partitioning method which employed the GA algorithm for the partitioning of the power system. System observability was also used to convert the measurement redundancy index (MRI) in conjunction with four other indices to apply the GA. In order to use MRI, PMU measurements were assumed to be the only available measurements of the system. Observability robustness was realized by adding extra PMUs based on electrical distance criterion. The MASE approach was used to evaluate performance of the partitioning method along with a modified CKF in order to improve the convergence of the estimates. The CKF was further modified by estimating both both system and measurement noise. The modified CKF performed very well for the local state estimation of the two-level MASE. After collecting estimates from the first level, the central coordination level (the second level) only needed to adjust the voltage angle to obtain the global state estimate.

Various performance evaluations were carried out in the research. For the PMU placement, the observability robustness was verified by evaluating the system observability performance for two scenarios. The performance was evaluated in terms of the unobserved bus numbers and the loss of observability as a percentage. It was observed that the system with extra PMUs would have less adverse effects on the observability when one PMU or two PMUs failures occurred.

The partitioning performance was evaluated using the $AMRV$ value as the measurement of each cluster. Specifically, the difference in $AMRV$ values were compared before and after inserting the MRI. The geographical results for the two scenarios were presented as well. A comparison was carried out between the partitioning method with additional PMU and the partitioning method without additional PMU. This lead to the conclusion that the MRI can be a constraint that has a positive impact on the partitioning method and it can improve the sub-system observability. Moreover, the additional PMU produces a higher observability value and a smaller difference in the $AMRV$ value.

Lastly, we implemented the power system state estimation using the CKF with an online noise updating technique. The state estimation results were presented to show convergence with

and without the online noise updating technique, with and without additional PMUs. The simulations showed that the estimation with online noise estimation had better convergence and does not require accurate prior knowledge on the noise model.

Our state estimation was applied to MASE very successfully and it can be concluded that MASE along with our partitioning algorithm can achieve more accurate estimation within a shorter period of time by exchanging less information than those for the centralized state estimation.

Bibliography

[1] Ali Abur and Antonio Gomez Exposito. *Power system state estimation: theory and implementation*. CRC press, 2004.

[2] M. M. Ahmed and K. Imran. An optimal pmu placement against n-1 contingency of pmu using integer linear programming approach. In *2019 International Conference on Applied and Engineering Mathematics (ICAEM)*, pages 127–132, Aug 2019.

[3] I. Arasaratnam and S. Haykin. Cubature kalman filters. *IEEE Transactions on Automatic Control*, 54(6):1254–1269, June 2009.

[4] Fatihcan M Atay, Tuerker Biyikoglu, and Jürgen Jost. Synchronization of networks with prescribed degree distributions. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 53(1):92–98, 2006.

[5] T. L. Baldwin, L. Mili, M. B. Boisen, and R. Adapa. Power system observability with minimal phasor measurement placement. *IEEE Transactions on Power Systems*, 8(2):707–715, May 1993.

[6] Ravindra B Bapat, Ivan Gutmana, and Wenjun Xiao. A simple method for computing resistance distance. *Zeitschrift für Naturforschung A*, 58(9-10):494–498, 2003.

[7] P. P. Bedekar, S. R. Bhide, and V. S. Kale. Optimum pmu placement considering one line/ one pmu outage and maximum redundancy using genetic algorithm. In *The 8th Electrical Engineering/ Electronics, Computer, Telecommunications and Information Technology (ECTI) Association of Thailand - Conference 2011*, pages 688–691, May 2011.

[8] J. Bialek. Topological generation and load distribution factors for supplement charge allocation in transmission open access. *IEEE Transactions on Power Systems*, 12(3):1185–1193, Aug 1997.

[9] S. Billakanti and C. Venkaiah. An effective binary integer linear programmed approach for optimal placement of pmus in power systems. In *2014 International Conference on Smart Electric Grid (ISEG)*, pages 1–8, Sep. 2014.

[10] S. Blumsack, P. Hines, M. Patel, C. Barrows, and E. C. Sanchez. Defining power network zones from measures of electrical distance. In *2009 IEEE Power Energy Society General Meeting*, pages 1–8, July 2009.

[11] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martin Chavez, and D-U Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4-5):175–308, 2006.

[12] M. Brown, M. Biswal, S. Brahma, S. J. Ranade, and H. Cao. Characterizing and quantifying noise in pmu data. In *2016 IEEE Power and Energy Society General Meeting (PESGM)*, pages 1–5, July 2016.

[13] NB Chakrabarti. A note on a sum of lognormals. *arXiv preprint arXiv:1606.07300*, 2016.

[14] L. Chen, T. S. Bi, A. C. Xue, and Q. X. Yang. Approach for linear state estimation considering zero-injected currents. In *2nd IET Renewable Power Generation Conference (RPG 2013)*, pages 1–5, Sep. 2013.

[15] Y. Cheng, C. Lu, K. Men, and L. Tu. Application of the complex algorithm in pmu-only state estimaiton. In *2014 International Conference on Power System Technology*, pages 349–354, Oct 2014.

[16] G. C. Contaxis and G. N. Korres. A reduced model for power system observability: analysis and restoration. *IEEE Transactions on Power Systems*, 3(4):1411–1417, Nov 1988.

[17] Napoleon Costilla-Enriquez, Claudio R Fuerte-Esquivel, and Victor J Gutierrez-Martinez. A sensitivity-based approach for the detection of multiple-line outages using phasor measurements. *IEEE Transactions on Power Systems*, 2019.

[18] E. Cotilla-Sanchez, P. D. H. Hines, C. Barrows, and S. Blumsack. Comparing the topological and electrical structure of the north american electric power infrastructure. *IEEE Systems Journal*, 6(4):616–626, Dec 2012.

[19] E. Cotilla-Sanchez, P. D. H. Hines, C. Barrows, S. Blumsack, and M. Patel. Multi-attribute partitioning of power networks based on electrical distance. *IEEE Transactions on Power Systems*, 28(4):4979–4987, Nov 2013.

[20] E. Cotilla-Sanchez, P. D. H. Hines, and C. M. Danforth. Predicting critical transitions from time series synchrophasor data. *IEEE Transactions on Smart Grid*, 3(4):1832–1840, Dec 2012.

[21] Paolo Crucitti, Vito Latora, and Massimo Marchiori. A topological analysis of the italian electric power grid. *Physica A: Statistical mechanics and its applications*, 338(1-2):92–97, 2004.

[22] P. Cuffe and A. Keane. Visualizing the electrical structure of power systems. *IEEE Systems Journal*, 11(3):1810–1821, Sep. 2017.

[23] Himansu Das, Sanjay Kumar Mishra, and Diptendu Sinha Roy. The topological structure of the odisha power grid: a complex network analysis. *IJMCA*, 1(1):012–016, 2013.

[24] Richard C Dorf and Robert H Bishop. *Modern control systems*. Pearson, 2011.

[25] Agoston E Eiben, James E Smith, et al. *Introduction to evolutionary computing*, volume 53. Springer, 2003.

[26] B. Ertürk and M. Göl. Binary integer programming based pmu placement in the presence of conventional measurements. In *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, pages 1–6, Oct 2016.

[27] X. Fan and D. Duan. Decentralized multi-area state estimation with hybrid measurements. In *2017 IEEE Power Energy Society General Meeting*, pages 1–5, July 2017.

[28] M. Farsadi, H. Golahmadi, and H. Shojaei. Phasor measurement unit (pmu) allocation in power system with different algorithms. In *2009 International Conference on Electrical and Electronics Engineering - ELECO 2009*, pages I–396–I–400, Nov 2009.

[29] E. E. Fetzer and P. M. Anderson. Observability in the state estimation of power systems. *IEEE Transactions on Power Apparatus and Systems*, 94(6):1981–1988, Nov 1975.

[30] Zhaohui Gao, Dejun Mu, Shesheng Gao, Yongmin Zhong, and Chengfan Gu. Adaptive unscented kalman filter based on maximum posterior and random weighting. *Aerospace Science and Technology*, 71:12–24, 2017.

[31] Zhaohui Gao, Dejun Mu, Yongmin Zhong, Chengfan Gu, and Chengcai Ren. Adaptively random weighted cubature kalman filter for nonlinear systems. *Mathematical Problems in Engineering*, 2019:1–13, 01 2019.

[32] R. Gelagaev, P. Vermeyen, J. Vandewalle, and J. Driesen. Numerical observability analysis of distribution systems. In *Proceedings of 14th International Conference on Harmonics and Quality of Power - ICHQP 2010*, pages 1–6, Sep. 2010.

[33] J Duncan Glover, Mulukutla S Sarma, and Thomas Overbye. *Power System Analysis & Design, SI Version*. Cengage Learning, 2012.

[34] David E Goldberg et al. A note on boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing. *Complex Systems*, 4(4):445–460, 1990.

[35] Antonio Gómez-Expósito, Antonio de la Villa Jaén, Catalina Gómez-Quiles, Patricia Rousseaux, and Thierry Van Cutsem. A taxonomy of multi-area state estimation methods. *Electric Power Systems Research*, 81(4):1060–1069, 2011.

[36] TNE Greville. Some applications of the pseudoinverse of a matrix. *SIAM review*, 2(1):15–22, 1960.

[37] Jia Guo. Distributed dynamic multi-area state estimation with fusion technique. 2019.

[38] H Happ. The operation and control of large interconnected power systems. *IEEE Transactions on Circuit Theory*, 20(3):212–222, 1973.

[39] N Hashim, N Hamzah, MF Abdul Latip, and AA Sallehhudin. Transient stability analysis of the ieee 14-bus test system using dynamic computation for power systems (dcps). In *2012 Third International Conference on Intelligent Systems Modelling and Simulation*, pages 481–486. IEEE, 2012.

[40] Paul Hines, Eduardo Cotilla-Sanchez, and Seth Blumsack. Do topological models provide good information about vulnerability in electric power networks? *Chaos (Woodbury, N.Y.)*, 20:033122, 09 2010.

[41] Åke J Holmgren. Using graph models to analyze the vulnerability of electric power networks. *Risk analysis*, 26(4):955–969, 2006.

[42] Chen Huo. Partitioning techniques to improve resilience and cyber security in electric grid infrastructure. 2018.

[43] A. Jain and N. R. Shivakumar. Power system tracking and dynamic state estimation. In *2009 IEEE/PES Power Systems Conference and Exposition*, pages 1–8, March 2009.

[44] Kumar Jitender, JN Rai, Arora BB Vipin, and CK Singh. Improvement in power system state estimation by use of phasor measurement unit. *International Journal of Engineering RESEARCH and Technology (IJERT). ISSN*, pages 2778–0181.

[45] S. J. Julier and J. K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, March 2004.

[46] I. Kamwa, A. K. Pradhan, and G. Joos. Automatic segmentation of large power systems into fuzzy coherent areas for dynamic vulnerability assessment. *IEEE Transactions on Power Systems*, 22(4):1974–1985, Nov 2007.

[47] I. Kamwa, A. K. Pradhan, G. Joos, and S. R. Samantaray. Fuzzy partitioning of a real power system for dynamic vulnerability assessment. *IEEE Transactions on Power Systems*, 24(3):1356–1365, Aug 2009.

[48] V. Kekatos and G. B. Giannakis. Distributed robust power system state estimation. *IEEE Transactions on Power Systems*, 28(2):1617–1626, May 2013.

[49] H. Khazraj, F. Faria da Silva, and C. L. Bak. A performance comparison between extended kalman filter and unscented kalman filter in power system dynamic state estimation. In *2016 51st International Universities Power Engineering Conference (UPEC)*, pages 1–6, Sep. 2016.

[50] H. Khazraj, F. Faria da Silva, and C. L. Bak. A performance comparison between extended kalman filter and unscented kalman filter in power system dynamic state estimation. In *2016 51st International Universities Power Engineering Conference (UPEC)*, pages 1–6, Sep. 2016.

[51] Douglas Klein and Milan Randic. Resistance distance. *Journal of Mathematical Chemistry*, 12:81–95, 12 1993.

[52] G. N. Korres, P. J. Katsikas, K. A. Clements, and P. W. Davis. Numerical observability analysis based on network graph theory. *IEEE Transactions on Power Systems*, 18(3):1035–1045, Aug 2003.

[53] Gabriel Kron. A set of principles to interconnect the solutions of physical systems. *Journal of Applied Physics*, 24(8):965–980, 1953.

[54] G. R. Krumpholz, K. A. Clements, and P. W. Davis. Power system observability: A practical algorithm using network topology. *IEEE Transactions on Power Apparatus and Systems*, PAS-99(4):1534–1542, July 1980.

[55] L. U. Kumar and A. R. Devi. Power system dynamic state estimation using kalman filtering technique. In *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 1173–1178, July 2018.

[56] L. U. Kumar and A. R. Devi. Power system dynamic state estimation using kalman filtering technique. In *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 1173–1178, July 2018.

[57] Alexis Kyriacou, Panayiotis Demetriou, Christos Panayiotou, and Elias Kyriakides. Controlled islanding solution for large-scale power systems. *IEEE Transactions on Power Systems*, 33(2):1591–1602, 2017.

[58] P. Lagonotte, J. C. Sabonnadiere, J. . Leost, and J. . Paul. Structural analysis of the electrical system: application to secondary voltage control in france. *IEEE Transactions on Power Systems*, 4(2):479–486, May 1989.

[59] S. Lakshminarasimhan and A. A. Girgis. Hierarchical state estimation applied to wide area power system. In *2006 Power Systems Conference: Advanced Metering, Protection, Control, Communication, and Distributed Resources*, pages 523–527, March 2006.

[60] J. Li, C. Liu, and K. P. Schneider. Controlled partitioning of a power network considering real and reactive power balance. *IEEE Transactions on Smart Grid*, 1(3):261–269, Dec 2010.

[61] J. B. A. London, L. Mili, and N. G. Bretas. An observability analysis method for a combined parameter and state estimation of a power system. In *2004 International Conference on Probabilistic Methods Applied to Power Systems*, pages 594–599, Sep. 2004.

[62] P. M. Mahadev and R. D. Christie. Envisioning power system data: concepts and a prototype system state representation. *IEEE Transactions on Power Systems*, 8(3):1084–1090, Aug 1993.

[63] S. Manda, K. Balakrishna, and A. Sriharibabu. Linear programming method to find the minimal placements of phasor measuring units for power system state estimation. In *2017 International Conference on Power and Embedded Drive Control (ICPEDC)*, pages 360–364, March 2017.

[64] M. Meriem, C. Bouchra, B. Abdelaziz, S. O. B. Jamal, E. M. Faissal, and C. Nazha. Study of state estimation using weighted-least-squares method (wls). In *2016 International Conference on Electrical Sciences and Technologies in Maghreb (CISTEM)*, pages 1–5, Oct 2016.

[65] Brad L Miller, David E Goldberg, et al. Genetic algorithms, tournament selection, and the effects of noise. *Complex systems*, 9(3):193–212, 1995.

[66] H. Mori and S. Tsuzuki. A fast method for topological observability analysis using a minimum spanning tree technique. *IEEE Transactions on Power Systems*, 6(2):491–500, May 1991.

[67] H. Mosbah and M. El-Hawary. A distributed multiarea state estimation. In *2018 IEEE Canadian Conference on Electrical Computer Engineering (CCECE)*, pages 1–5, May 2018.

[68] Saralees Nadarajah. Explicit expressions for moments of log normal order statistics. *Economic Quality Control*, 23(2):267–279, 2008.

[69] Kyatsandra G Nagananda. Electrical structure-based pmu placement in electric power systems. *arXiv preprint arXiv:1309.1300*, 2013.

[70] M Nazari-Heris and B Mohammadi-Ivatloo. Application of heuristic algorithms to optimal pmu placement in electric power systems: an updated review. *Renewable and Sustainable Energy Reviews*, 50:214–228, 2015.

[71] Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[72] R. F. Nuqui and A. G. Phadke. Phasor measurement unit placement techniques for complete and incomplete observability. *IEEE Transactions on Power Delivery*, 20(4):2381–2388, Oct 2005.

[73] M. Pau, F. Ponci, A. Monti, S. Sulis, C. Muscas, and P. A. Pegoraro. An efficient and accurate solution for distribution system state estimation with multiarea architecture. *IEEE Transactions on Instrumentation and Measurement*, 66(5):910–919, May 2017.

[74] Jyoti Paudel. Phasor measurement unit deployment approach for maximum observability considering vulnerability analysis. 2015.

[75] Jyoti Paudel, Xufeng Xu, Karthikeyan Balasubramaniam, and Elham B Makram. A strategy for pmu placement considering the resiliency of measurement system. *Journal of Power and Energy Engineering*, 3(11):29, 2015.

[76] A. Primadianto and C. Lu. A review on distribution system state estimation. *IEEE Transactions on Power Systems*, 32(5):3875–3883, Sep. 2017.

[77] VH Quintana and N Müller. Partitioning of power networks and applications to security control. In *IEE Proceedings C (Generation, Transmission and Distribution)*, volume 138, pages 535–545. IET, 1991.

[78] F. Rajwani and N. C. Beaulieu. Accurate simple closed-form approximations to distributions and densities of lognormal sum random variables. In *IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall. 2004*, volume 1, pages 111–114 Vol. 1, Sep. 2004.

[79] A. O. M. Saleh and M. A. Laughton. Cluster analysis of power-system networks for array processing solutions. *IEE Proceedings C - Generation, Transmission and Distribution*, 132(4):172–178, July 1985.

[80] M. Shahraeini and M. H. Javidi. A survey on topological observability of power systems. In *2011 IEEE Power Engineering and Automation Conference*, volume 3, pages 373–376, Sep. 2011.

[81] A. Sharma, S. C. Srivastava, and S. Chakrabarti. A cubature kalman filter based power system dynamic state estimator. *IEEE Transactions on Instrumentation and Measurement*, 66(8):2036–2045, Aug 2017.

[82] D. R. Shrivastava, S. A. Siddiqui, and K. Verma. Optimal pmu placement for complete power system observability under (p–1) contingency. In *2017 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, pages 129–132, Dec 2017.

[83] D. R. Shrivastava, S. A. Siddiqui, and K. Verma. Optimal pmu placement for coordinated observability of power system under contingencies. In *2017 IEEE International Conference on Circuits and Systems (ICCS)*, pages 334–339, Dec 2017.

[84] Satyendra P. Singh, Amit K. Thakur, and S.P. Singh. Pmu placement for maximum observability of power system under different contingencies. *Energy Procedia*, 117:893 – 900, 2017. First International Conference on Power Engineering Computing and CONtrol (PECCON-2017) 2nd -4th March .2017." Organized by School of Electrical Engineering, VIT University, Chennai, Tamil Nadu, India.

[85] H. Tebianian and B. Jeyasurya. Dynamic state estimation in power systems using kalman filters. In *2013 IEEE Electrical Power Energy Conference*, pages 1–5, Aug 2013.

[86] ON THECHARMSOFS, VA RIABILITYANDP RO BA BILITY—NORMAL, and OR LOG. Log-normal distributions across the sciences: Keys and clues. *BioScience*, 51(5), 2001.

[87] N. P. Theodorakatos, N. M. Manousakis, and G. N. Korres. Optimal placement of pmus in power systems using binary integer programming and genetic algorithm. In *MedPower 2014*, pages 1–6, Nov 2014.

[88] N. P. Theodorakatos, N. M. Manousakis, and G. N. Korres. Optimal placement of pmus in power systems using binary integer programming and genetic algorithm. In *MedPower 2014*, pages 1–6, Nov 2014.

[89] T. Van Cutsem, J. L. Horward, and M. Ribbens-Pavella. A two-level static state estimator for electric power systems. *IEEE Transactions on Power Apparatus and Systems*, PAS-100(8):3722–3732, Aug 1981.

[90] Z. Wang, A. Scaglione, and R. J. Thomas. Generating statistically correct random topologies for testing smart grid communication and control networks. *IEEE Transactions on Smart Grid*, 1(1):28–39, June 2010.

[91] Douglas Brent West et al. *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River, NJ, 1996.

[92] Chai Wah Wu. Synchronization in networks of nonlinear dynamical systems coupled via a directed graph. *Nonlinearity*, 18(3):1057, 2005.

[93] Wenjun Xiao and Ivan Gutman. Resistance distance and laplacian spectrum. *Theoretical Chemistry Accounts*, 110(4):284–289, 2003.

[94] Bei Xu. *Optimal monitoring and visualization of steady state power system operation*, volume 68. 2006.

[95] P. Yuan, W. Zhang, and J. Chen. A network partition approach for distributed three-phase state estimation of active distribution networks. In *2018 IEEE Power Energy Society General Meeting (PESGM)*, pages 1–5, Aug 2018.

[96] J. Zhao and L. Mili. A framework for robust hybrid state estimation with unknown measurement noise statistics. *IEEE Transactions on Industrial Informatics*, 14(5):1866–1875, May 2018.

[97] J. Zhao, M. Netto, and L. Mili. A robust iterated extended kalman filter for power system dynamic state estimation. *IEEE Transactions on Power Systems*, 32(4):3205–3216, July 2017.

[98] Zhenyu Huang, K. Schneider, and J. Nieplocha. Feasibility studies of applying kalman filter techniques to power system dynamic state estimation. In *2007 International Power Engineering Conference (IPEC 2007)*, pages 376–382, Dec 2007.

[99] Ray D Zimmerman and Deqiang Gan. Matpower-a matlab power system simulation package: User's manual. *Power Systems Engineering Research Center*, 1997.