Phasor Measurement Unit (PMU) Placement Optimisation in Power Transmission Network based on Hybrid Approach

Jiangxia Zhong

Master of Engineering

2012

RMIT

Phasor Measurement Unit (PMU) Placement Optimisation in Power Transmission Network based on Hybrid Approach

A thesis submitted in fulfilment of the requirements for
the degree of Master of Engineering

Jiangxia Zhong

School of Electrical and Computer Engineering
RMIT University
August 2012

Declaration:

I certify that except where due acknowledgement has been made, the work is that of the candidate alone. This thesis is a presentation of my original research work and has not been submitted previously, in whole or in part, to qualify for any other academic award. Furthermore, the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program.

The work was done under the guidance of Associate Professor Alan Wong, at RMIT University, Melbourne.

Candidate: Jiangxia Zhong

30/05/2012

Acknowledgements:

I would like to express my deep and sincere gratitude to my major supervisor, Associate Professor Alan Wong in RMIT University. His wide knowledge in power distribution network and his logical way of thinking have been of great value for me. His understanding, encouraging and personal guidance have provided a good basis for the present thesis.

My sincere thank also goes to my second supervisor, Dr. Peter Graszliewicz, who promoted me the ability of critical thinking.

I am grateful to my colleagues in University Dr. Chen Zhang, Dr. Jerry Yu and Mr. Zhe Zhang, for advising me in the right way of research career. Thanks to Ruth Fluhr for my thesis editing.

Special thanks to my father and mother for their supports and encouragement.

Abstract:

This thesis presents novel optimal placement approaches of phasor measurement unit (PMU) for applications such as state estimation and fault detection. In this thesis, the PMU placement is realised based on two hybrid algorithms namely Approximation Algorithm and Global Optimization Algorithm. The proposed algorithms will ensure optimum PMU placement with full network observability under different contingency conditions. The IEEE 14, 24, 30, 57 and the New England 39 standard test systems will be used to exam the proposed algorithm adequately and the result will be compared to existing methods. In this thesis, we demonstrated that the proposed methods are very effective in determining the minimum number of PMU and the results are comparable to the best methods presented in the past literature. In addition, the comparison between the proposed methods to the existing methods show that the proposed hybrid approaches achieve higher System Observability Redundancy Index (SORI) which will in turn improve the reliability and stability of power transmission.

Key words: Phase measurement, optimization methods, power transmission, observability

Table of contents

List of Figures

List of Tables

Chapter 1 Introduction

In contemporary society, many countries around the world are affected by power failures, which are caused by factors such as lack of investment into power system infrastructure, inadequate asset maintenance, and continuous increase in electricity consumption that overstresses the power transmission and distribution system. Consequently, power companies suffer from losses of billons of dollars, and inconvenience to private and business customers. In order to prevent the prevalent blackouts, implementation of state-of-the-art technologies, such as a state estimation of the transmission network, is required to achieve better controllability, higher reliability and stability of the power system [1]. The Phasor Measurement Unit (PMU) is a device that is employed to detect the voltage and current waveform that is synchronised with a clocking signal obtained continuously from the global positioning system (GPS). Integrating with the GPS receiver [2], the base station is able to receive the synchronous data from each PMU in real time. The location of malfunction circuits or transmission lines can be immediately identified if phase differences between different PMUs are detected.

1.1 Historical Overview

Phase angle [3] between the voltage phase and current phase as the basic measuring function of PMU has been utilised to monitor the condition of power networks. Theoretically, the active (real) power flow in a distribution line is proportional to the sine of the angle difference between voltages at the two terminals of the line. In which case, the angle difference was deemed as a special consideration to manage and operate the power network.

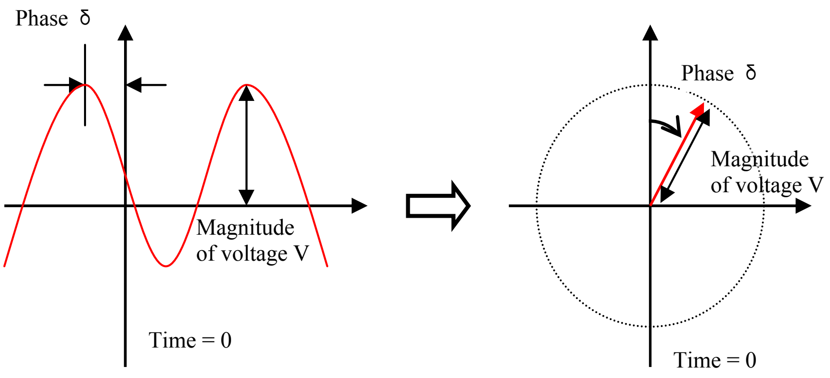
In the early 1980s, novel phase angle measurement equipment was introduced [4]. The communication channel, which was based on LORAN-C, GOES satellite transmissions and the HBG radio transmissions in Europe was utilised to maintain the reference signal in synchronisation. Researchers established the local phase angle with respect to the time reference for resolving zero crossing of the phase voltage. The phase voltage was referred to the common reference signal and the phase angle difference between two sets of phasor measurements was computed. However, the best-achieved time reference from the communication channels mentioned above only provided measurement accuracy in the order of 40 microseconds. As a consequence of this, these devices could not offer high precision to realize power network measurements in time synchronization.

The global positioning system (GPS), which was invented and deployed by the U.S. Department of Defense in 1993 [5], was introduced to the next generation of phase measurement devices called the Phasor Measurement Unit (PMU). The GPS consists of space satellites, control stations and user equipment. A total of twenty-four satellites orbit at a height of a million miles from the earth, and transmit the high frequency signals to the control stations to provide the precise message of the time and orbital information. As a result, the users can achieve 24-hour continuous real-time information-processing that is synchronized to the international standard time. In addition, the GPS provides high precision timing, ranging from 1 nanosecond to 10 nanoseconds [6]. At the same time, the GPS receiver can supply a unique pulse signal in one-second intervals, which is known as 1 pulse per second (PPS). Therefore, the issue of unsynchronized standard time in the power grid was solved by installing or embedding the GPS receivers into various devices in arbitrary positions, such as in the high voltage sub-station and transmission towers.

Apparently, the implementation of the GPS technology into the power network is a perfect idea as it allows for accuracy and reliability of clock synchronization. For instance, the accuracy of the GPS timing pulse is better than 1 microsecond, which for a 50 Hz system corresponds to about 0.02 degrees. The accurate measurement obtained by such a clock is a priority for the purposes of state estimation and fault analysis of power systems. In comparison with the previous methods, it offers several advantages, such as clock synchronization with high accuracy in the nanoseconds range, extensional usage range, and no special demand for a communication channel; there is also less chance of the device being influenced by weather conditions and/or geography.

1.2 Phasor Measurements

Phasor is a fundamental concept in Electrical Engineering that represents a sinusoidal signal represented by the quantity of its magnitude and phase with respect to a reference. In the figure of sinusoidal waveform as depicted in Fig. 1.1, the distance between the sinusoidal peak of signal and the time reference (E.g. time = 0) is defined as a phase angle and it is transferred to an angular measurement in the phasor representation.

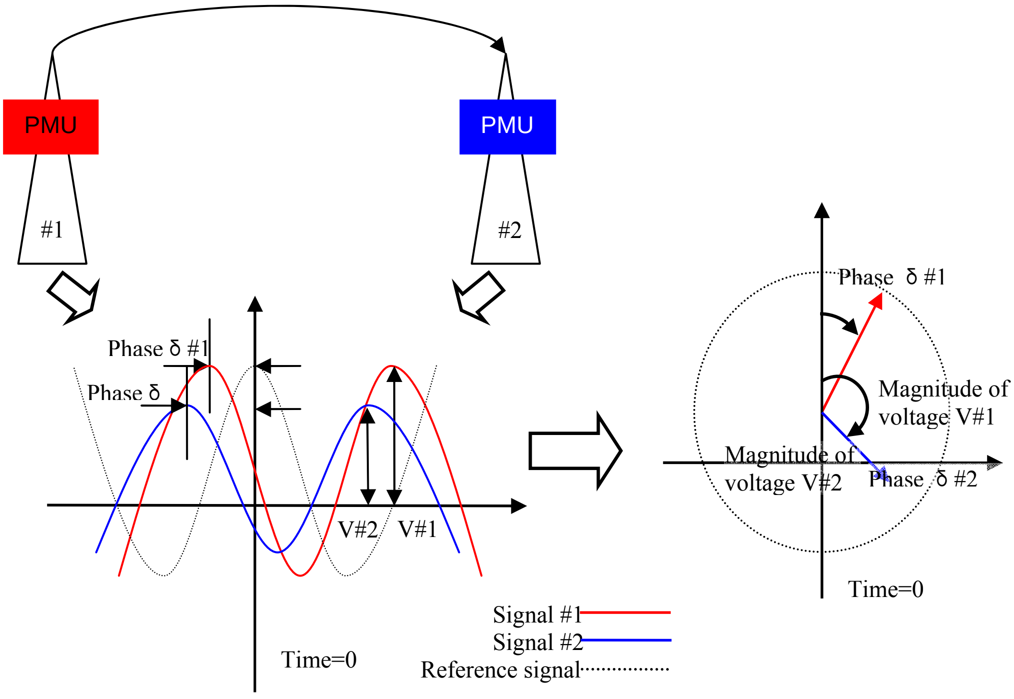


Sinusoidal waveform

Phasor representation

Figure 1.1 Sinusoidal waveform and its phasor representation

Phasor technology including the Phasor Measurement Unit (PMU) [7] is a valuable measurement technology in the power system for monitoring the condition of transmission and distribution networks. As shown in Figure 1.2, the phasor of the 50Hz component is obtained based on the digitally-sampled analog voltage waveform that is synchronized with the clocking signal from the GPS receiver in distributed locations (#1 and #2). The time reference is titled as a ‘common reference’ signal and it helps to synchronise the different waveforms at all different sites. The amplitude difference between Signal #1 and Signal #2 in Fig. 1.2 is due to the signal attenuation on the overhead transmission line.



Sinusoidal waveform

Phasor

Figure 1.2 Signals received by PMUs

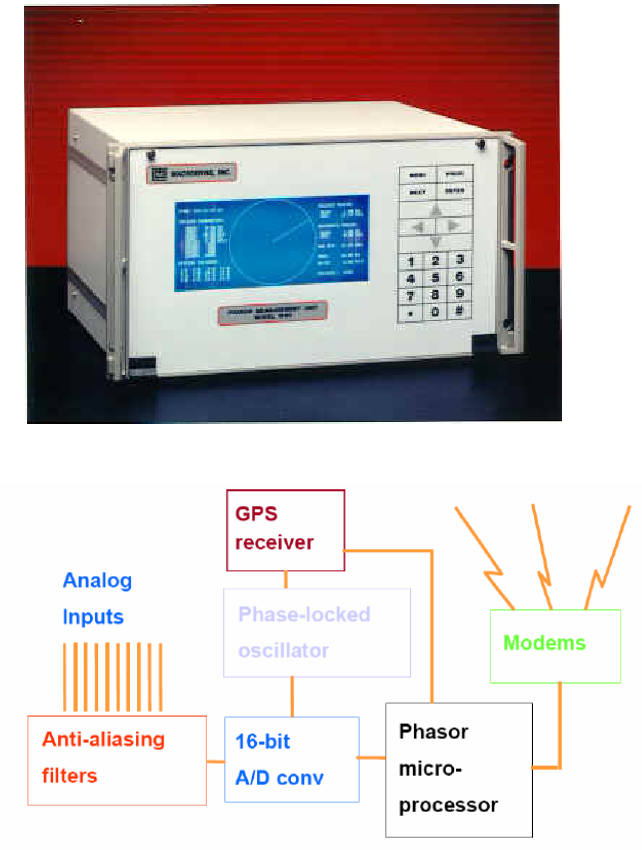


Figure 1.3 Phasor Measurement Unit (PMU) [8] and its function block diagram [9]

The Phasor Measurement Unit embeds the Global Positioning System (GPS) receiver clocks to achieve the synchronising of sampled signals at nominated locations of the entire power network. In the real-life system, the PMU receives the voltage and current waveforms as inputs, which are derived from standard Current Transformer (CT) and Potential Transformer (PT). The input signals are isolated, filtered and sampled at an effective rate of 48 samples per cycle of the fundamental frequency.

The phasor microprocessor, as shown in Fig. 1.3, uses the recursive Discrete Fourier Transform (DFT) algorithm to calculate the local positive sequence, fundamental frequency and voltage and current phasors from the sampled data. The resultant time-tagged phasors are immediately available for local or remote applications via the standard communications ports.

By operating interrelated software of the PMU, the users are capable of monitoring the phasors across the whole transmission network for any abnormal events. The phasor data provides information of pre-fault or post-fault conditions. Therefore, the operators in the central control room can sequentially and continuously acquire and calculate the values of phasor.

1.3 Example of PMU Implementation

Since phasor technology and the PMU device were brought into modern electric power networks, performances such as the reliability, stability and controllability of power networks have reached a higher level. In the following sections, various examples and the main benefits of implementing PMUs in power networks will be represented.

1.3.1 State Estimation

In the power network, the classification of state estimation in real-time control functions includes: scheduling generation and interchange; monitoring outages and scheduling alternatives; supervising scheduled outages; scheduling frequency and time corrections; coordinating bias settings; and emergency restoration of system [4]. Basically, all of the state estimation considerations above are operated by running a large number of load flow analyses and results are calculated over a long execution period. By employing the Phasor Measurement Units that achieve high performance in time synchronization with excellent precision, the state estimation measurement is able to be implemented based on complex bus voltages [10]; this approach is used in preference to one based on early state estimation algorithms [11], that utilize the measurement of line flows, involving both real and reactive power, to estimate the bus voltage magnitudes and angles.

By using the state estimator, measurement errors can be detected, identified and corrected using the bad data processing technique [12][13]. This procedure can be treated as part of the state estimation process and could also be a post-estimation procedure.

In a given power system, a measurement is classified as either critical or redundant. Redundant measurements can be removed from the measurement system without causing the system to become unobservable. When a redundant measurement is erroneous, this can be detected by statistical tests based on measurement residuals. Removal of critical measurements, however, will lead to an unobservable system, and errors in these types of measurements cannot be detected. In a well-designed measurement system, the bad data processing can be accomplished if any critical measurement will be observed by multiple measurement devices. In principle, adding any type of measurement will improve redundancy. It is simple to incorporate synchronized phasor measurements into state estimators along with conventional measurements, and this can be shown to improve state estimation performance [14]. Given a power system that is fully observable by existing measurements, a few extra PMUs can be used to convert any existing critical measurements into redundant ones, thereby making all bad data in the system detectable.

1.3.2 Static state estimation

After suffering huge losses in the Northeast blackout in 1965 [4], the US power utilities started to utilise analog measuring technology and communication systems, such as the supervisory control and data acquisition (SCADA) system which consists of remote terminal units (RTUs) to measure phasors. The conventional method, such as elaborate error models [15], integrates the system dynamic equation by using the magnitude and angle of bus voltages as the states, and subsequently measures real and reactive power flows. This traditional method could identify problems which were directly or indirectly affecting the operation of the power network. However, in terms of the large dimension of the matrix, the computational time and memory were impossible to execute in real-time. To overcome the challenges, researchers and network operators introduced a new idea called the ‘static’ state estimation. Static-state estimation is an approximation method that is used to predict the future behaviour of a power network based on its present condition. For instance, the consideration of oscillation in a power network can be analysed using information from the generators and loads which are measured by the PMUs. In order to predict an oscillation that might occur in the future (e.g. one minute period), the data collected by PMUs are transferred to a network model and the static and dynamic behaviours are calculated. Assume that the system does not change during the monitoring period; once the network operator receives a report on the possibility of losing the static stability, the operator could follow the predetermined procedures to deal with the issue.

1.3.3 Fault Detection

In recent years, major efforts have been dedicated to the exploration and development of new methodologies that detect faults that occur in the overhead transmission line [16, 17]. In power systems, the type of fault can be categorised as either a permanent fault or a temporary fault. A permanent fault caused by events, such as a broken transmission line or the malfunction of a power generator, can be located easily, as the detection devices receive huge differences in signal characteristics during the pre-fault and post-fault moment. In contrast, the temporary fault that is normally caused by insulator flashover would not cause the supply of the overhead transmission to collapse immediately. However, flashover on insulators has the potential to lead to a full breakdown of the insulator when those transient phenomena occur frequently. Therefore, it is vital to protect and analyse the whole network and localize the fault in advance

The PMU-based fault location technique [18-20] is able to determine the fault location through synchronized fault voltages which are monitored by neighbouring PMU installed nodes. Based on these fault node voltages, which are measured by PMUs, line currents between these nodes can be calculated. Then, node injection currents at two terminals of the faulted line are formed from the line currents. Based on the calculated fault node injection currents, fault nodes can be deduced or fault locations in transmission lines can be calculated accurately.

1.3.4 Wide Area Monitoring System (WAMS)

The wide area monitoring system (WAMS) [21] is a new concept for maintaining dynamic stability in the overhead transmission line network, based on the PMU. Recently, many countries have been proactively implementing WAMS in their power system networks. Unlike previous monitoring systems, WAMS is constructed based on time-synchronized measurement, novel computing technology, and communication technology to achieve the synchronization of data acquisition and real-time recording from equipment and systems in distributed locations, as shown in Fig. 1.4. The real-time data will be delivered to the central control station where the network operator will be able to measure and analyse the data at any point of the power network. WAMS has more functional advantages over conventional systems and would eventually replace the traditional supervisory control and data acquisition (SCADA) system for steady-state monitoring. Furthermore, WAMS has the capability to analyse the network oscillation, monitor and estimate the network static stability, perform the time-stamp for fault localization and detect the network voltage instability. It is so far deemed the most advanced method to detect and avoid pervasive blackout. By optimally placing PMUs in the power network, the controllers can recognize unusual activities within the power network through the interface in the power control centre.

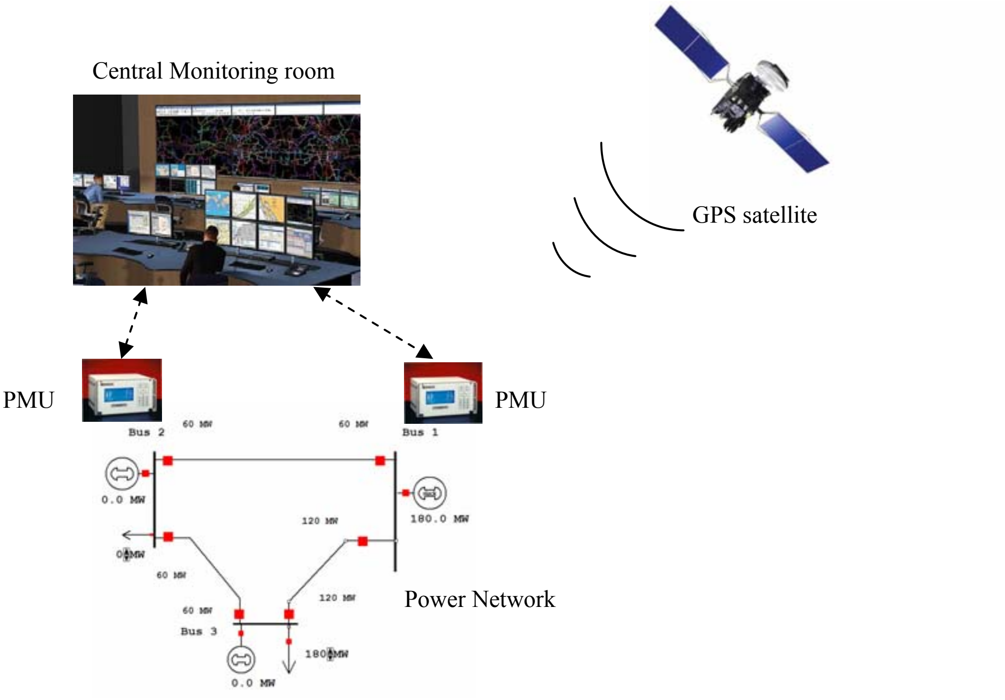


Figure 1.4 Wide area monitoring system

1.4 Thesis Objectives and Description

The optimal PMU placement is a complex optimization task for power system networks. In this thesis, the main objectives are to design novel PMU placement optimisation algorithms based on several hybrid approaches. The proposed algorithms will ensure optimum PMU placement with full network observability under different contingency conditions. The IEEE 14, 24, 30, 57 and the New England 39 standard test systems will be used to examine the proposed algorithm adequately and the result will be compared to existing methods.

The outlines of this thesis are as follows:

Chapter 1: A historical overview of PMU Technology is introduced, and the application of the PMU, for both state estimation and fault detection, is discussed in detail. The outlines of the thesis are also presented in this Chapter.

Chapter 2: This chapter reviews the existing optimised PMU Placement (OPP) methods. The existing methods can be presented in three categories, namely: Heuristic Method, Meta-Heuristic Method; and Deterministic Method. In this chapter, the concept of the System Observability

Redundancy Index (SORI) will be introduced. A network with higher SORI value indicates that the monitoring system is more reliable.

Chapter 3: This chapter presents the first proposed OPP method based on Breadth First Algorithm and Greedy Algorithm. This innovative method analyses the network using the hierarchical layer approach and later determines the optimum number of PMUs using Greedy Algorithm.

Chapter 4: This chapter demonstrates the second proposed OPP method, mainly constructed using Exhaustive Search Algorithm. The idea behind this novel method is to eliminate a run of unnecessary possibilities by applying a circular sequential approach consisting of Domination Set and Binary Search Algorithm to quickly and accurately locate the minimal number of PMUs needed.

Chapter 5: In this chapter, the results obtained using the proposed approaches are tabulated, and a comparison of the total number of PMUs needed is presented. The system observability redundancy index (SORI) to select the best performance method among the new and existing methods is reported, followed by a comprehensive discussion of the two proposed methods. The summary of the thesis, incorporating the main achievement of optimal PMU placement, as well as anticipated future works, concludes this chapter.

Chapter 2 Literature review

The ever-growing global population, as well as the consumption of electricity, has triggered the increasing demand for reliable electricity. The transmission and distribution network plays an essential role in power systems to transmit power from the generators to the customers. To prevent events such as loss of electricity, power network providers must recognise the quality and stability of various parts of the power transmission network through monitoring and measuring equipment. Based on the GPS synchronized clock, the phasor measurement unit can measure a vast amount of critical power network information, which includes bus voltage, bus current, generator speed and power angle.

By receiving the real-time PMU measurement information over wide locations, the operators in the central control room can monitor and analyse the quality of the distribution network under static and dynamic operating conditions. Phadke A. G. [1] suggested that the installation of PMUs in all substations can significantly improve the power network reliability. Nonetheless, the disposal investment of PMU device with the single price of $19,000 USD [22] in all locations is unaffordable. To reduce the maintenance fee and unit costs, Optimal PMU Placement (OPP) is implemented to minimize the amount of PMUs installed and to achieve the entire degree of observability. As a result, the problem of optimal PMU placement has been focused on as a new angle of research interest in the power network in recent years.

In general, the OPP algorithms can be categorised into three groups namely: Heuristic Method, Meta-Heuristic Method and Deterministic Method [23]. Examples of algorithms that fall under each group are tabulated in Table 2.1.

Table 2.1 Examples of Optimal PMU Placement Methods [23]

Heuristic Method	**Meta-Heuristic Method**	**Deterministic Methods**
Depth-First Algorithm (DFS) [24,25] Domination Set [26,27] Greedy Algorithm [28]	Genetic Algorithms [31,32] Particle Swarm Optimization (PSO) [33,34,37]	Integer Programming [39,40,41] Binary Search Method [42]

2.1 Heuristic Method

Heuristic Method, which is generally known as Approximation Algorithm, is a type of mathematical optimization technique to locate the optimal solution using optimum computational time and memory space. Heuristic Methods are often used to accelerate the process of finding a reasonable solution when an exhaustive search is impractical. The final optimal outcome from the Heuristic Method cannot be guaranteed. Examples of Heuristic Methods, which include Depth-First Algorithm, Domination Set and Greedy Algorithm, will be discussed in the following paragraphs.

2.1.1 Depth-First Algorithm (DFS)

Because of the similarity in the structure of power networks, several graph theory approaches have been introduced to the OPP problem to determine the optimization solution. In the work presented by Farsadi et. al. [24], the optimization methods based on Depth-First Algorithm were applied to analyse the minimal PMU needed in IEEE 14-bus and 57-bus system. The first PMU was placed at the bus with the largest number of connected branches and if there was more than one bus with this characteristic, one was randomly chosen. Subsequent PMUs were placed, based on the same criterion, until the complete network visibility was obtained. This approach is described in Fig. 2.1.

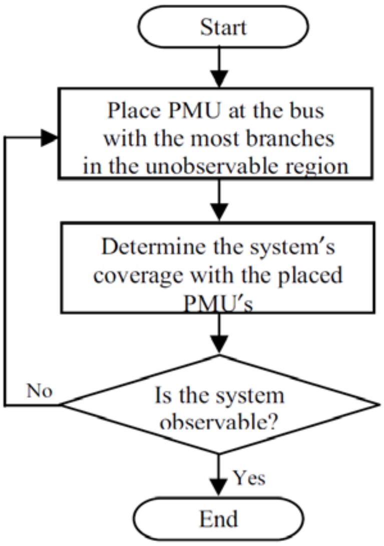


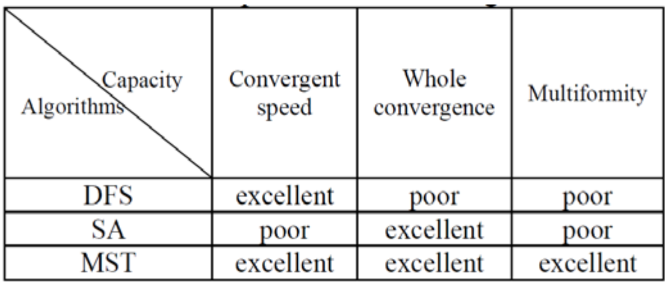
Figure 2.1 Flowchart of Depth First Algorithm [24]

.

By testing the algorithm using PSAT software and comparing DFS algorithm against other OPP methods, such as Graph Theoretic Procedure and Annealing Method, the DFS did not provide optimal results for the IEEE 14-bus and IEEE-57 bus systems.

In the paper presented by Cai [25], the author analysed and explained the advantages and disadvantages in three optimization methods, including Depth-First Search (DFS), Simulated Annealing Method (SA) and Minimum Spanning Tree Method (MST). Out of the three methods, DFS was computationally more efficient and achieved excellent results in convergent speed, as shown in Table 2.2. On the other hand, this algorithm failed to consider the whole convergence and multiformity, which indicates that the final solution was not optimum.

Table 2.2 Comparison of DFS, SA and MST methods [25]



Capacity Algorithms	Convergent speed	Whole convergence	Multiformity
DFS	excellent	poor	poor
SA	poor	excellent	poor
MST	excellent	excellent	excellent

2.1.2 Domination Set

In Graph Theory, Domination Set locates the smallest number of vertices for observing the full graph. Haynes et al [26] introduced the conception of domination set into the power network. By applying the topological theory into the power network, the equation G= (V, E) represents an electric power system. According to Domination Set Theory, a vertex is abbreviated to ‘V’, which represents an electrical node or a substation bus to which transmission lines, loads, and generators are connected, and ‘E’ stands for an edge or link in the power system network which represents a transmission line connecting two electrical nodes. According to the voltage and phase angle measured by the phasor measurement unit, the author mathematically proved several optimization rules as well as investigated the theoretical properties of $\gamma_P(T)$ in trees T with restrictions. An example of a 6-bus system with domination set has been simplified in Fig. 2.2. The nodes with red colour represent the optimal PMU location (domination set point), leading to the network being fully observed.

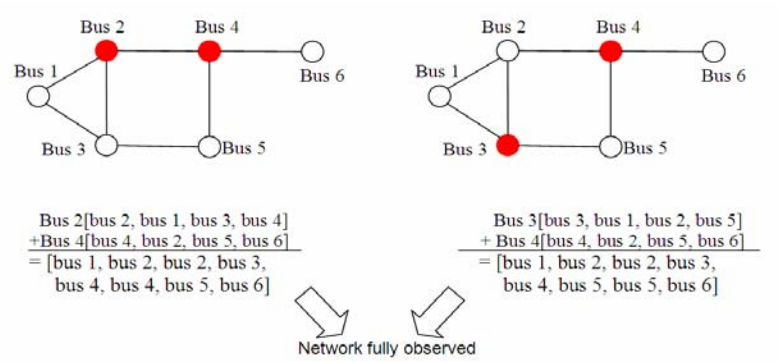


Figure 2.2 6-bus system with Domination Set

Table 2.3 Connectivity of 6-bus system

Bus	Degrees	Connectivity
1	2	Bus 2, Bus 3
2	3	Bus 1, Bus 3, Bus 4
3	3	Bus 1,Bus 2, Bus 5
4	3	Bus 2,Bus 5,Bus 6
5	2	Bus 4,Bus 3
6	1	Bus 4

Yuan [27], drawing on the work of Haynes [26], implemented the rule of the power dominating number into the linear algorithm to find out the optimal PMU placement. The procedure of the algorithm was clearly shown in a flowchart. The minimum number of PMUs needed was presented for the IEEE 14-bus system and the New England 39-bus system. However, the exact locations of the PMUs were not detailed in the paper.

2.1.3 Greedy Algorithm

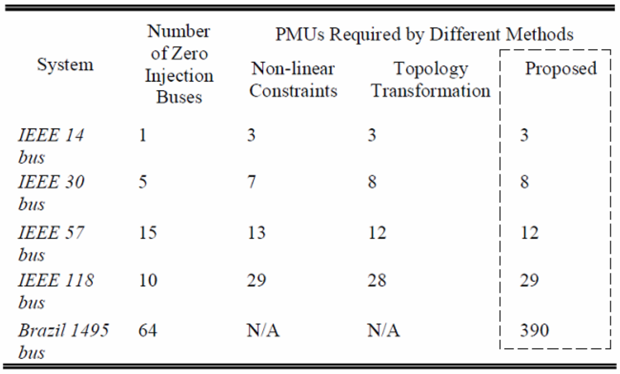
M. Zhou [28] introduced the matrix reduction rules and algorithm to effectively reduce the computational efforts. In this research, Greedy Algorithm was applied to solve the minimum number of PMUs necessary for achieving full observability.

Generally, Greedy Algorithm generates decisions according to one rule: at each stage, choose and install a PMU at the bus that contains the largest number of uncovered buses. In his investigation, Zhou [28] assumed that there are n buses F=$S_1, S_2, ...S_n$ in the system, serving as a finite PMU placement candidate set, of which m buses $U=S_{j1},S_{j2},...S_{jm}$ were installed with PMUs for the time being. The uncovered bus set in place by current PMUs was represented as

$X=S_{i1}, S_{i2}\ldots S_{ik}$. While X was not empty, an S* from F-U was chosen, such that the maximum number of $S_{*\,i}$ from X would be covered. In order to balance the computation efforts and feasibility of optimization performance, an approximated optimum would be achieved instead of the exact optimum. In reference [29], a theorem was presented that Greedy Algorithm is a performance-guaranteed method for placement problems. An instance of the set covering problem given by F=$S_1, S_2,\ldots S_n$, ,where each $S_m \subseteq X$, and $\cup s_M=X$, was assumed. The supposition was made that J_{opt} is an optimal solution, and J_{greedy} is a solution found by greedy algorithm. $|J_{greedy}| \le H\left(\max_{m=1,\ldots,n}|S_m|\right) \bullet |J_{opt}|$ where $H(d) = \log_e(d)$. The equality was held only if $|S_k| = \max|S_m|$, for $k=1,\ldots,n$, which meant that every single bus throughout the system connected with the same number of neighbour buses. Generally, it was not the case for a large real system, so, "<" was always established.

As shown in Table 2.4, the performance of Greedy Algorithm approach marked by dotted line was acceptable for IEEE 14-bus system and IEEE 57-bus system. Nevertheless, in IEEE-30 and IEEE-118 bus system, this method proposed to include an extra PMU to observe the full network in comparison with the reference method.

Table 2.4 optimal number of PMUs by Greedy Algorithm [28]



System	Number of Zero Injection Buses	PMUs Required by Different Methods		
		Non-linear Constraints	Topology Transformation	Proposed
IEEE 14 bus	1	3	3	3
IEEE 30 bus	5	7	8	8
IEEE 57 bus	15	13	12	12
IEEE 118 bus	10	29	28	29
Brazil 1495 bus	64	N/A	N/A	390

2.2 Meta-Heuristic Methods

Meta-Heuristic Method, which is an improvement on the Heuristic Method, involves intelligent search processes that can deal with discrete variables and non-continuous cost functions [29]. Basically, this method combines randomized algorithm and local optimization algorithm to solve the optimal problem. Two types of Meta-Heuristic Methods that were applied in the OPP problem were Genetic Algorithm and Particle Swarm Optimisation Method.

2.2.1 Genetic Algorithm

Genetic algorithm (GA) is one of a number of evolutionary algorithms, which generates solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection and crossover.

Affenzeller [30] stated that genetic algorithm is basically executed in the following way: An initial population of individuals (also called "chromosomes") is generated randomly or heuristically. During each iteration step, also called "generation", the individuals of the current population are evaluated and assigned a certain fitness value. In order to form a new population, individuals are first selected, and then produce offspring candidates which, in turn, form the next generation of parents. This ensures that an 'individual' is chosen the expected number of times and is approximately proportional to its relative performance in population. For producing new solution candidates, genetic algorithms use two operators, namely crossover and mutation:

- Crossover is the primary genetic operator. It takes two individuals, called parents, and produces one or two new individuals, called offspring, by combining parts of the parents. In its simplest form, the operator works by swapping (exchanging) substrings before and after a randomly selected crossover point.

- The second genetic operator, mutation, is essentially an arbitrary modification which helps to prevent premature convergence by randomly sampling new points in the search space. In the case of bit strings, mutation is applied by simply flipping bits randomly in a string with a certain probability called 'mutation rate'.

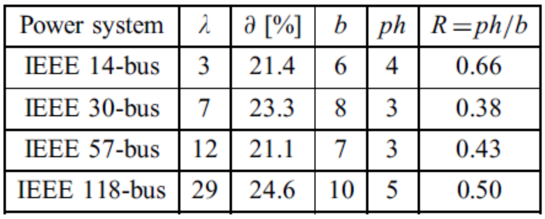
Marin et al [31] employed Genetic Algorithm to determine the optimum number of phasor measurement units in a power system network. The authors summarised the rules of observability analysis and the fitness function based on the parameters N_H (the number of the buses that are not observables) and N_{PMU} (the number of PMUs in the network); this is demonstrated in Equation 2.1 below. The best results were obtained using a=1, b=2 and c=1.

$$f = aN_{PMU} + bN_H + cN_{PMU}N_H \qquad (2.1)$$

where a, b and c are constants.

Step 1: Given the power network, build its corresponding chromosome
Step 2: Create the initial population
Step 3: For each individual, calculate its fitness function by (4)
Step 4: Apply selection operator
Step 5: Apply crossover operator
Step 6: Apply mutation operator
Step 7: Apply elitist strategy
Step 8: Go to step 3 until gene generations are completed

Table 2.5 optimal number of PMUs by Genetic Algorithm [31]



Power system	λ	∂ [%]	b	ph	$R = ph/b$
IEEE 14-bus	3	21.4	6	4	0.66
IEEE 30-bus	7	23.3	8	3	0.38
IEEE 57-bus	12	21.1	7	3	0.43
IEEE 118-bus	29	24.6	10	5	0.50

In this analysis, Genetic Algorithm was adapted to solve the problem of optimal PMU placement and the number of PMUs in the power system acceptable in optimization under IEEE 14-bus, 30-bus, 57-bus and 118-bus system. Another achievement was that new characteristics (Table 2.5) were created to indicate the condition of the power network, which included the percentage of PMUs needed, where *b* is the largest number of incident branches of each network, *ph* is the necessary number of current phasors that must be measured in each PMU to give an optimal solution, and R is a quality factor that relates *ph* and *b*.

In another analysis [32], the author introduced a non-dominated sorting genetic algorithm to carry out, simultaneously, a consideration of minimization of the number of PMUs and maximization of the measurement redundancy. The optimization was carried out without any preferential information given with respect to the objectives. The result of the search process was a set of (ideally Pareto-optimal) candidate solutions, from which the decision-maker would be able to choose the most desirable one. The important advantage of the algorithm is that it provides the entire Pareto-optimal front, instead of a single point solution, and can lend itself to application in an entire class of problems, where multi-objective optimization on a prohibitively large enumerative search space is required.

2.2.2 Particle Swarm Optimization (PSO)

Particle Swarm Optimisation (PSO) Method was originally proposed by Eberhart and Kennedy in 1995 [33] for researching social behaviour, especially for simulating the movement of organisms in a bird flock. It is a population-based search algorithm that exploits a population of individuals to probe promising regions of the search space. The population here is called a swarm and the individuals are called particles.

As discussed by Sadu, Kumar and Kavasseri [34], Particle Swarm Optimization is able to solve the optimal placement problem of PMUs. The PSO method achieved higher performance by using less number of iterations compared to the algorithms, such as Original Clonal Algorithm (CLONALG) [35] and Adaptive Clonal Algorithm (CLONALG) [36].

Ahmadi [37] presented a binary particle swarm optimization (BPSO) based methodology for the optimal placement of phasor measurement units. The author successfully achieved results that required the minimal number of PMUs and yet provided maximum measurement redundancy in the power network. The simulation results were verified under a condition of normal operation with or without zero-injection for IEEE 14-bus, 30-bus, 57-bus and 118-bus respectively.

2.3 Deterministic Methods

In computer science, a Deterministic Algorithm can be defined as an algorithm that is able to predict behaviour. In other words, given a particular parameter as an input, the system will generate the predicted output. Deterministic algorithms are by far the most studied and familiar kind of algorithm, as well as being able to be measured on TOMLAB Optimization Toolbox [38] efficiently.

2.3.1 Integer Programming

Aminifar et al [39] formulated a unique model for the optimal placement of contingency-constrained phasor measurement units (PMUs) in electric power networks. By applying Integer Linear Programming, the conventional complete observability of power networks first obtained and different contingency conditions in power networks including measurement losses and line outages were added to the simulation model. As a result, the required number of PMUs in each scenario was comprehensively studied and compared.

Chakrabarti [40] presented an Integer Quadratic Programming approach to minimize the total number of PMUs required and to maximize the measurement redundancy at the power system buses. Existing conventional measurements were studied in the proposed PMU placement method. In this analysis, complete observability of the system was ensured under normal operating conditions as well as under the outage of a single transmission line or a single PMU. Simulation results on the IEEE 14-bus, 30-bus, 57-bus, and 118-bus test systems as well as on a 298-bus test system were presented. Dua [41] introduced the Integer Linear Programming (ILP) framework to explain the zero injection constraints. In addition, the System Observability Redundancy Index (SORI) was also introduced to expand the bus coverage in the minimal solution.

2.3.2 Binary Search

A Binary Search Algorithm, which was proposed, by Chakrabarti and Kyriakides [42] was used to determine the minimum number of PMUs required. An Exhaustive Binary Search was employed to calculate the minimum number of PMUs needed to make system fully observable. In general, a total number of combinations were able to be measured as

$$N_{solution} = \frac{P!}{N_{PMU}!(P - N_{PMU})!} \tag{2.2}$$

where P represents the total number of buses in the power system and where N_{PMU} represents the number of PMUs needed in the power system.

It is clear that the total number of buses in the power system will affect the total computational time in a non-linear relationship. For the purpose of reducing the execution period, the author introduced a theoretical upper bound of the minimum number of PMUs needed to make the system observable; this was derived from [39]

$$N_{PMU}^{ub} = [(N + s/2)/3] \tag{2.3}$$

where N is the total number of candidate buses in the system and s is the number of unknown power injections.

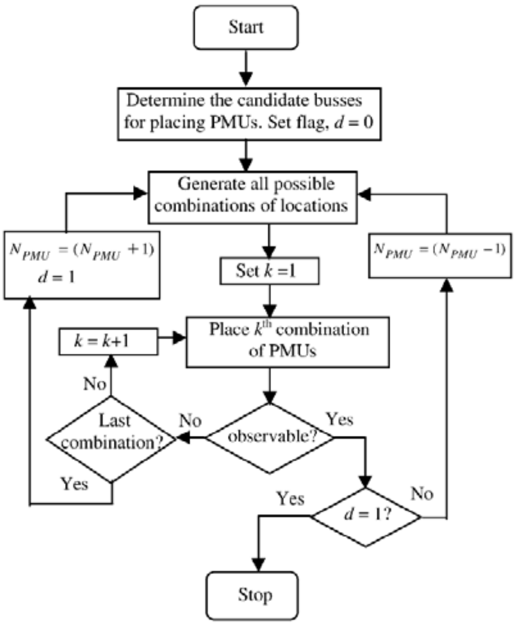


Figure 2.3 Binary search algorithm to determine the minimum number of PMUs required to make the system observable [42]

According to [42], if the system is found to be unobservable for all of the combinations of PMU locations, then the minimum number of PMUs is increased by one, i.e., N_{PMU}= (N_{PMU}+1). If the system is found observable for any of the combinations of locations, the minimum number of PMUs is reduced by one, i.e., N_{PMU}= (N_{PMU}-1). The search is repeated until the minimum number of PMUs is obtained. The search process ensures that, if N_{PMU} is the minimum number of PMUs, none of the solutions of length (N_{PMU}-1) can make the system observable. An exhaustive set of combinations of size (N_{PMU}-1) was examined for observability before concluding that N_{PMU} is the minimum number of PMUs.

Through the proposed procedure, complete observability of the system was ensured under normal operating conditions as well as under the outage of a single transmission line or a single PMU. Moreover, the conception of measurement redundancy had been added to filter out the best adequate PMU placement. The simulation results were represented for the IEEE Standard 14, 24, 30 bus system and the New England 39 bus system.

2.4 The Rules of Network Observability

Based on the fundamental laws of branch current and node voltage in circuit theory, several rules are applied to analyse the network to ensure that the network is fully observable.

Case 1: For PMU installed buses, voltage phasor and current phasor of all its incident branches are known. These are called ‘direct measurements’. According to the function of the phasor measurement unit, a PMU located in the Bus D, as shown in Fig. 2.4, indicates that the voltage in this bus can be directly measured. Meanwhile, the branch currents attached to the node are also measured by the PMU. In Case 1, the known parameters measured by PMU are V_D, I_{AD}, I_{BD} and I_{CD} and the characteristic of transmission line are $R_{AD} + jX_{AD}$, $R_{BD} + jX_{BD}$, $R_{CD} + jX_{CD}$.

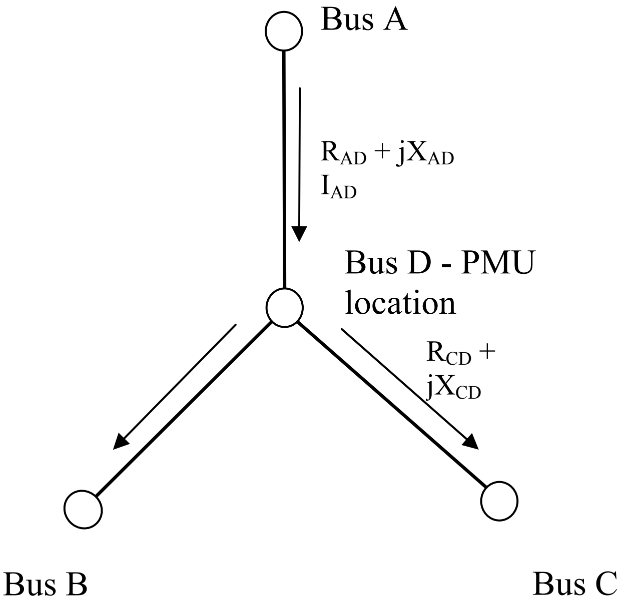


Figure 2.4 The first observability rule

Case 2: If the voltage and current phasors at one end of a branch are known, the voltage phasor at the other end of the branch can be obtained using equation 2.4 to 2.6. These are called ‘pseudo measurements’.

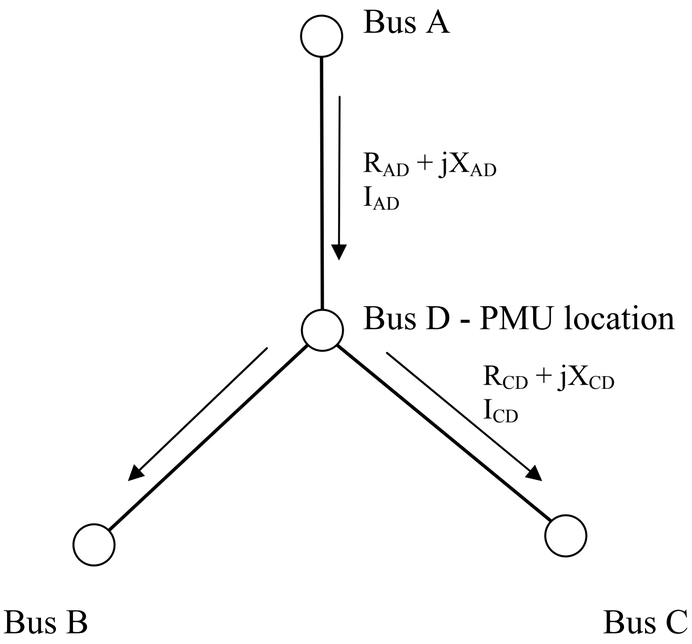


Figure 2.5 The second observability rule

Based on the known parameter such as line impedance and branch currents, the magnitude of voltage will be resolved using the following equation:

$$V_A = V_D + I_{AD}\,(R_{AD} + jX_{AD}) \tag{2.4}$$

$$V_B = V_D - I_{BD}\,(R_{BD} + jX_{BD}) \tag{2.5}$$

$$V_C = V_D - I_{CD}\,(R_{CD} + jX_{CD}) \tag{2.6}$$

Case 3: If voltage phasors of both ends of a branch are known, the current phasor of this branch can be obtained directly. These measurements are also known as 'pseudo measurements'.

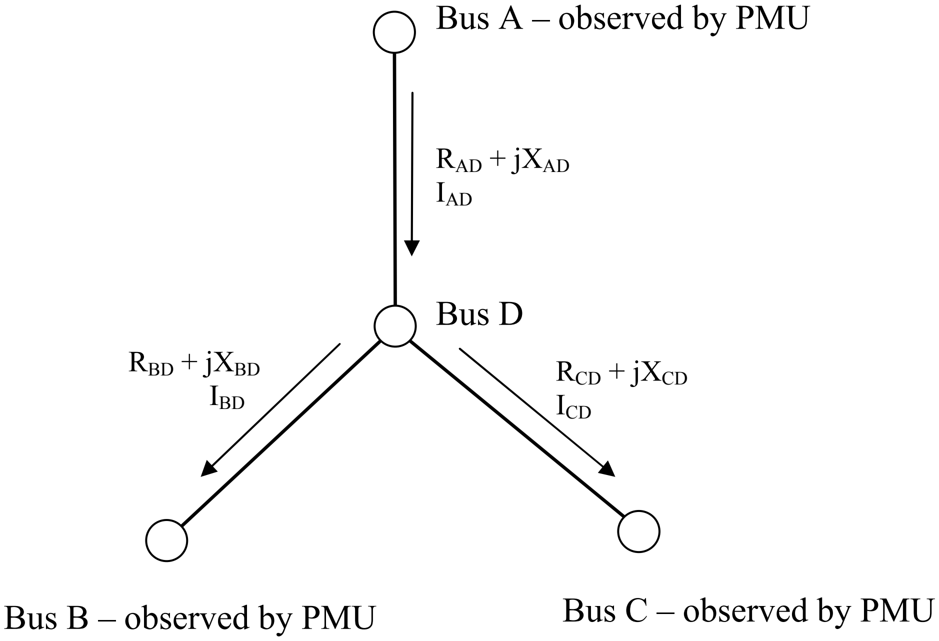


Figure 2.6 The third observability rule

Under this circumstance, assuming that the magnitudes of voltage in Bus A, Bus B and Bus C are observed and measured by the PMUs, the line current in the branch of BD, AD and CD as well as the voltage in Bus D can be calculated. The equations in solving the unknown information are as follows:

$$V_D = V_A - I_{AD}(R_{AD} + jX_{AD}) \tag{2.7}$$

$$V_D = V_B + I_{BD}(R_{BD} + jX_{BD}) \tag{2.8}$$

$$V_D = V_C + I_{CD}(R_{CD} + jX_{CD}) \tag{2.9}$$

$$I_{AD} = I_{BD} + I_{CD} \tag{2.10}$$

2.5 System Observability Redundancy Index (SORI)

In optimal PMU placement, the redundancy index is an important factor for representing the stability of the power network. Due to a multiple number of optimum solutions being available after applying the optimization algorithm, Bus Observability Index (BOI) [41] will be implemented to indicate the performance on quality of optimization. In BOI, bus-i (β_i) will be defined as the number of PMUs which are able to observe a given bus. Consequently, the maximum bus observability index is limited to maximum connectivity (η_i) of a bus plus one:

$$\beta_i \leqslant \eta_i + 1 \tag{2.11}$$

In order to select the most favorable outcomes among a number of optimal solutions obtained using different optimization methods, the System Observability Redundancy Index (SORI) is, in principle, a measurement of the sum of bus coverage for all the implemented buses (i=1 to n) in an active system. Higher SORI value indicates that the PMU-based monitoring system is more reliable. Therefore, the best optimization method will be selected based on the SORI. The SORI can be calculated using Equation 2.11, where γ represents System Observability Redundancy Index

$$\gamma = \sum_{i=1}^{n} \beta_i \tag{2.12}$$

Chapter 3 Hybrid approach based on Approximation Algorithm

3.1 Introduction

In this chapter, a novel hybrid approach will be introduced [43]. As the name implies, the proposed method organises the connectivity of the electrical network in terms of layers. Combining the unique characteristics of Breadth-First Algorithm and Greedy Algorithm, this method is able to achieve optimum placement using less computation time. Similar to the Depth-First Search Method, as mentioned in Chapter 2, the Breadth-First Search has been used to solve many problems in Graph Theory, and one of the main advantages of this method is that it requires less computational power. In the proposed method, Breadth-First Algorithm plays the role of restructuring the power network into a hierarchical format. The information of every single bus is saved in a database that can be accessed conveniently at a later stage. Subsequently, Greedy Algorithm that produces the best choice at each stage is applied to achieve optimum condition across the network.

In the case of a multilevel system constructed using Breadth-First Search Method, Greedy Algorithm can be interpreted as a procedure which is implemented at the highest coverage bus(es) in each layer (structured by Breadth-First Algorithm) until the power system is fully observed. Once a bus has been selected, the interconnections will be eliminated. However, one execution loop starting from the highest incident node is not sufficient to acquire the best solution. On the other hand, if Greedy Algorithm is applied multiple times using every single node as the starting points; the total execution time will be greatly extended. A solution has been proposed in this chapter which utilizes the buses with the highest degree, in each layer, as the initial bus when Greedy Algorithm is applied. Consequently, this unique combination of Breadth-First Search Method and Greedy Algorithm is applied successfully to solve OPP in the normal operation condition with and without zero-injection effect.

The procedure of the hybrid approach for optimizing PMU placement can be defined in the flow chart shown in Figure 3.1

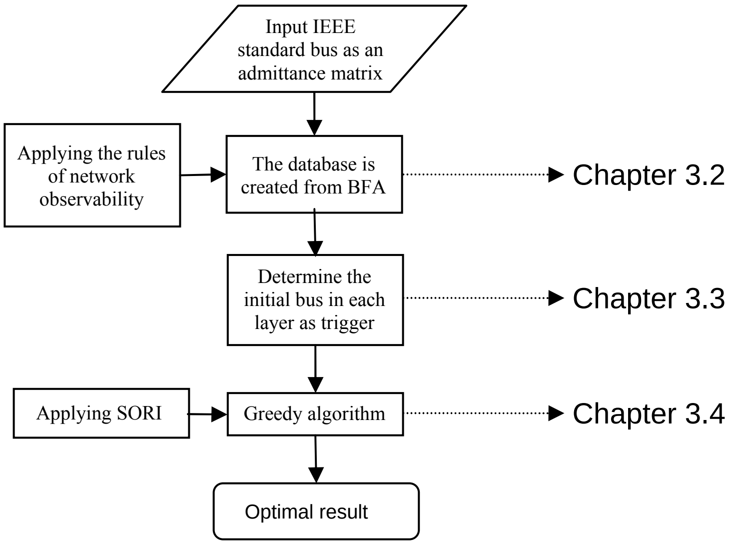
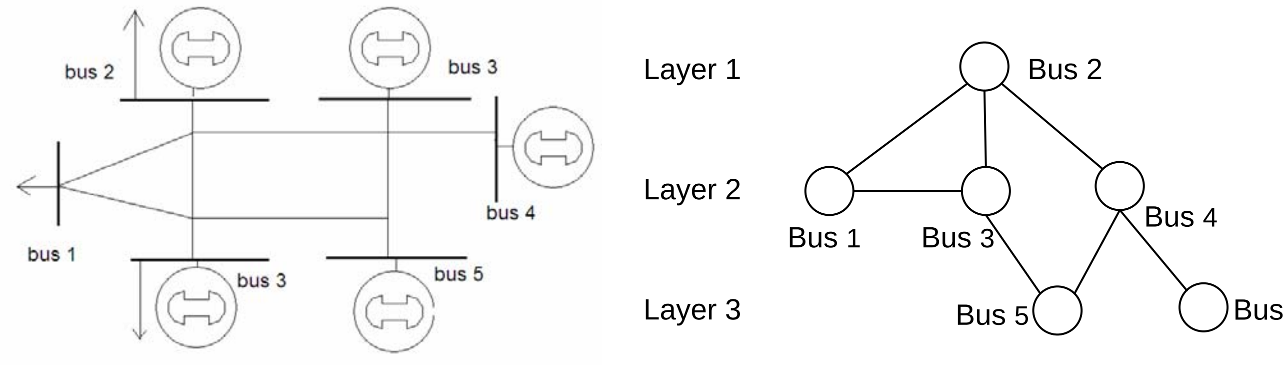


Figure 3.1 Flowchart of PMU optimal placement by combined algorithm

3.2 The application of Breadth-First Search Algorithm

The Breadth-First Search Algorithm is one of the simplest algorithms for searching a graph and an archetype for many important graph algorithms [30]. According to the theory of the Breadth-First Search Algorithm, the graph G= (V, E) represents an electric power system., a vertex is abbreviated to 'V', which represents an electrical node or a substation bus to which transmission lines, loads, and generators are connected, and "E" represents an edge or link in the power system network where a transmission line connects two electrical nodes. A source vertex *s* is defined in a known and breadth-first search; it is a traditional and ordered mode of searching the path from the vertex *s* to another vertex through the edges (E) of G. In the meantime, the graph tree, this starts at the root (vertex s) and extends to any other nodes (or vertex), is built and connected by the shortest link between the root and nodes.



a) b)

Figure 3.2 a) Single line diagram of a 6-bus b) Spanning tree diagram

The graph tree as shown in Fig. 3.2 is an example of a breadth-first tree, which is generated from the root node to all the neighboring nodes. For each of those nearest nodes, the breadth-first algorithm explores their undiscovered neighbour nodes, layer by layer, until all the interconnection information has been ascertained. In the system, the description of ***parent, child*** and ***brother*** can be used to describe the relationship of connectivity in the layer diagram. In table 3.1, Bus 3 will be identified as the ***parent*** of bus 5. On the contrary, bus 5 is the ***child*** of bus 3. Furthermore, the parameter of ***brother*** indicates the nodes are presented in the same layer (e.g. in layer 2, bus 1 is the ***brother*** of bus 3 and vice versa). Consequently, the interconnection database will be set up and stored.

Table 3.1 connectivity of 6-bus system

	Bus	**Degrees**	**Parent**	**Brother**	**Child**
Layer 1	2	3	NULL[a]	NULL	1,3,4
Layer 2	1	2	2	3	NULL
	3	3	2	1	5
	4	3	2	NULL	5,6
Layer 3	5	2	3,4	NULL	NULL
	6	1	4	NULL	NULL

NULL: no connection between the buses.

3.3 Selection of initial bus

In the proposed method, *initial bus* is defined as the bus with the highest incidence in each layer. It is important to note that multiple initial buses in any layer can exist if these buses have the same degree or number of buses connected to them. The benefit of introducing the concept of initial bus before implementing the Greedy Algorithm is that these initial buses in each layer allow us to explore more than one possible transverse through the spanning tree. In addition, our results show that the application of the Greedy Algorithm takes into consideration that initial buses can achieve a better outcome than the method presented by M.Zhou [28]. The comparison between the proposed method and M. Zhou’s method is presented in Chapter 5.

3.4 The application of Greedy Algorithm

Greedy Algorithm is a type of approximation algorithm used to solve the optimal question in a simple and effective way [30]. In the proposed hybrid approach, the algorithm is implemented to determine the minimal number of PMUs, based on the spanning tree created using Breadth-First Search Method. Generally, Greedy Algorithm makes a locally optimal choice to lead to a globally optimal solution. Basically, this algorithm makes decisions according to a single rule which states that, at each stage, a PMU should be installed at the bus that covers the largest number of uncovered buses and this process should be repeated until the system is fully observed.

To illustrate this, Greedy Algorithm is applied to the 6-bus system, as shown in Figure 3.2. In the initial step, PMU is placed on Bus 2, which is one of the highest connected buses with a connectivity degree of 3 in the network, and Bus 2 can observe all adjoining buses, namely bus 1, 2, 3 and 4. In the second step, the algorithm searches for the bus that covers the largest number of uncovered buses (E.g. bus 5 and bus 6 are uncovered). Since bus 4 covers both bus 5 and bus 6, it is clear that a PMU should be placed in bus 4 and, at this stage, the network is fully observed and the Greedy Algorithm will be terminated.

Table 3.2 Processing of Greedy Algorithm in 6-bus system

	Location of PMU
First step	**bus 2** covers bus 1,2,3,4
Second step	**bus 4** covers bus 5,6

3.2 The Placement Algorithm

In this section, the way of solving an optimal PMU placement problem by combining the Greedy Algorithm and the Breadth-First Algorithm will be presented. The IEEE standard 14-bus system, as shown in Fig. 3.3, will first be considered and studied under normal operation with and without zero-injection effect.

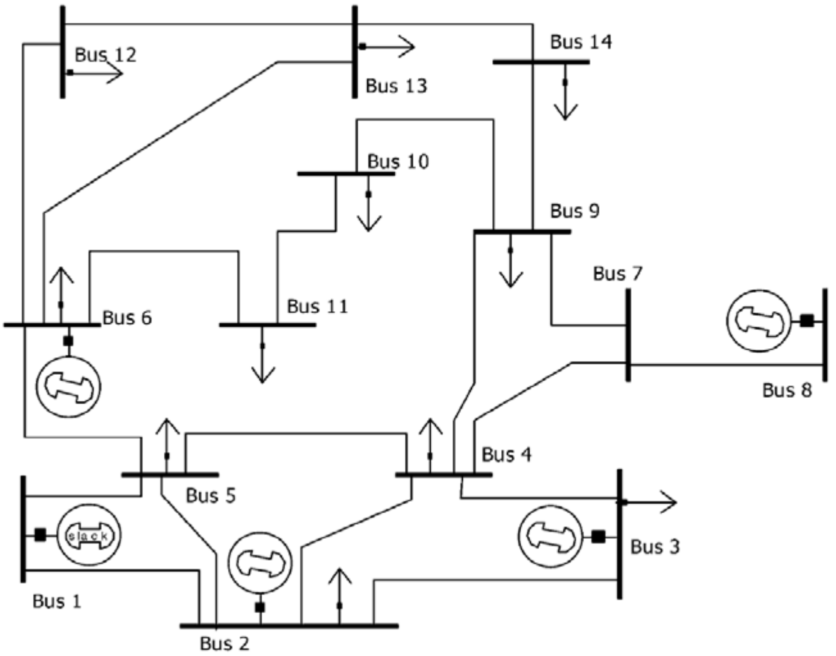


Figure 3.3 IEEE standard 14-bus test system

3.5.1 Normal Operation Condition without Zero-injection Effect

Firstly, the equation 3.1 shows the way of generating the connectivity matrix A, as follows:

$$A_{k,m} = \begin{cases} 1 & \text{if} \quad k = m \text{ or } k \text{ and } m \text{ are connected} \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \tag{3.1}$$

Meanwhile, for the purpose of programming, the connectivity matrix is converted into binary format. Matrix A presents the interconnectivity of the IEEE standard 14-bus system.

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{3.2}$$

According to the IEEE standard 14-bus system, as shown in Figure 3.3, it can be organized using Breadth-First Algorithm (BFA) into a spanning tree, which represents a top-to-bottom hierarchical structure. Any of the highest incident buses can be used as the top node in the hierarchical structure as illustrated in Figure 3.4. Subsequently, the connectivity of all nodes, such as the interconnections and the degrees of connectivity, are recorded as parameters, as depicted in Table 3.3.

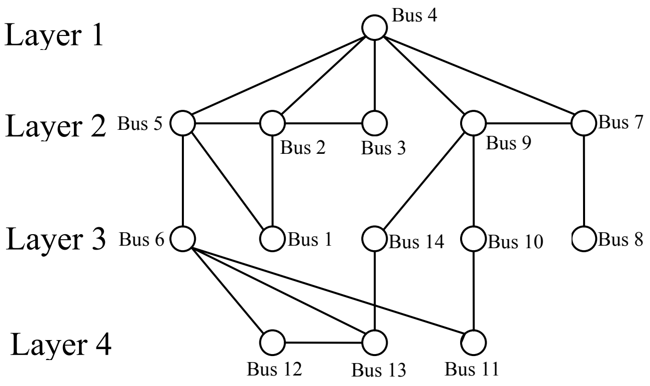


Figure 3.4 Hierarchical structure of the IEEE 14-bus system

Table 3.3 the database for the IEEE standard 14-bus system

	Bus	Degrees	Parent[a]	Brother[b]	Child[c]
Layer 1	4	5	NULL[d]	NULL	5,2,3,9,7
Layer 2	5	4	4	2	1,6
	2	4	4	3,5	1
	3	2	4	2	NULL
	7	3	4	9	8
	9	4	4	7	10,14
Layer 3	1	2	2,5	NULL	NULL
	6	4	5	NULL	11,12,13
	8	1	7	NULL	NULL
	14	2	9	NULL	13
	10	2	9	NULL	11
Layer 4	12	2	6	13	NULL
	13	3	6,14	12	NULL
	11	2	6,10	NULL	NULL

a: the parameter of parent indicates the bus is connected to upper layer.
b: the parameter of brother indicates the bus is connected to the same layer.
c: the parameter of child indicates the bus is connected to lower layer.
d: the parameter of NULL indicates no connection between two buses.

In this section, the IEEE standard 14-bus system is used as an example to illustrate the effectiveness of the proposed OPP method using a hybrid approach based on Breadth-First Search Method and Greedy Algorithm. The exact procedure is described in the following section.

Initially, the buses with only one connection (E.g. Degree of 1 as shown in Table 3.3) are to be located and PMUs need to be placed at their adjacent buses. In the IEEE 14-bus test system, a PMU is placed at bus 7 to cover the single bus 8. Next, the buses with the highest degree at each layer will be classified as initial buses and a PMU will be placed at these buses. Based on this criterion, a PMU should be placed on bus 4 in layer 1, bus 5 or bus 2 in layer 2, bus 6 in layer 3 and bus 13 in layer 4. Subsequently, the PMUs are placed in accordance to Greedy Algorithm so that the network is fully observed. All possible combinations for the IEEE 14-bus system are presented in Table 3.4.

Table 3.4 Example for the greedy algorithm in the 14-bus system under the condition of normal operation without zero-injection effect

<table>
<tr><td>1st placement</td><td colspan="7">Bus 7 [4, 9, 8 ,7]
(PMU is placed in bus 7 to cover the single connection at bus 8</td></tr>
<tr><td>Initial buses at Layer 1 to 4</td><td>Bus 4
[2,3,5]</td><td colspan="3">Bus 5
[1,2,5,6]</td><td>Bus 2
[1,2,3,5]</td><td>Bus 6
[5,6,12, 13,11]</td><td>Bus 13
[6,12,13, 14]</td></tr>
<tr><td>2nd placement</td><td>Bus 6
[6,12,13,11]</td><td>Bus 6
[11,12,13]</td><td colspan="2">Bus 13
[12,13,14]</td><td>Bus 6
[6,11,12,13]</td><td>Bus 2
[1,2,3]</td><td>Bus 2
[1,2,3,5]</td></tr>
<tr><td>3rd placement</td><td>Bus 9
[14,10]</td><td>Bus 9
[10,14]</td><td>Bus10
[10,11]</td><td>Bus11
[10,11]</td><td>Bus 9
[10,14]</td><td>Bus 9
[14, 10]</td><td>Bus 10
[10 11]</td></tr>
<tr><td>4th placement</td><td>Bus 1
[1]</td><td>Bus 3
[3]</td><td>Bus 3
[3]</td><td>Bus 3
[3]</td><td></td><td></td><td></td></tr>
<tr><td>Total Number of PMU Required</td><td>5</td><td>5</td><td>5</td><td>5</td><td>4</td><td>4</td><td>4</td></tr>
</table>

Note: Number in the bracket represents buses that are covered by the PMU

Looking closely at the results of Table 3.4, there are two possible solutions which give the total number of PMUs as 4. These combinations will have PMUs located at bus 2, 6, 7 and 9 or bus 2, 7, 10 and 13. In order to select the optimum solution from these two solutions, the redundancy index is calculated and the solution with the higher redundancy index will be selected as the optimal solution. For the IEEE 14-bus system, the placement at bus 2, 6, 7 and 9 gives a higher SORI value, as shown in Table 3.5.

Table 3.5 Comparison of SORI in 14-bus system

Optimal Location of PMUs	System Observability Redundancy Index (SORI)
2, 6 , 7, 9	19
2, 7, 10, 13	16

3.5.2 Normal Operation Condition with Zero-injection Effect

A zero-injection bus can be generalized as a bus without a generator and load connection. The zero-injection effect has been categorized as follows [39] :

1) For buses which are incident to an observable zero-injection bus are all observable except one, the unobservable bus will also be identified as observable by applying Kirchhoff's current law (KCL) at zero-injection bus.

2) For buses incident to an unobservable zero-injection bus are all observable, the zero-injection bus will also be identified as observable by applying the KCL at zero-injection bus.

In this proposed hybrid approach, once a zero-injection bus is detected inside the database, all of its neighboring buses are considered as virtually connected while the system hierarchy remains unchanged. For instance, in the IEEE 14-bus system as shown in Fig 3.3, bus 7 has no direct generator or load connection. Therefore, it is classified as a zero-injection bus and bus 8 will be considered as directly connected to bus 4 and bus 9. Hence, there is no single bus in this system, and based on the criterion of initial bus, a PMU should be placed on each of these buses (E.g. bus 4 in layer 1, bus 9 in layer 2, bus 6 in layer 3 and bus 13 in layer 4).

Table 3.6 Example of the greedy algorithm in the 14-bus test system in the condition of zero-injection effect

Initial bus:	bus 4 [2,3,4,5,7,8,9]	bus 9 [4,7,8,9,10,14]	bus 6 [5,6,11,12,13]	bus 13 [6,12,13,14]
1st placement:	bus 6 [6,11,12,13]	bus 6 [5,6,11,12,13]	Bus 9 [4,7,8,9,10,14]	bus 4 [2,3,4,5,7,8,9]
2nd placement:	bus 9 [10,14]	bus 2 [1,2,3]	bus 2 [1,2,3]	bus 10 [10]
3rd placement:	Bus 1 [1]			Bus 1 [1]

After regrouping the data, the optimization process will apply the same procedure as in the IEEE 14-bus system without zero-injection effect. Table 3.6 displays the optimal result through all the initial buses. As a result, the optimal solution is bus 2, 6 and 9. In summary, the total number of PMUs required to achieve full observability of the network with/without zero injection effect can be summarised in Table 3.7.

Table 3.7 Number of PMUs needed and locations in normal operation with/without zero-injection effect based on the 14-bus system

	Location of PMUs	Total Number of PMUs
Normal operation without zero-injection effect	2,6,7,9	4
Normal operation with zero-injection effect	2,6,9	3

3.6 Illustrative examples on IEEE 30-bus, 39-bus and 57-bus systems

In this section, the proposed hybrid algorithm combining Breadth-First Search Method and Greedy Algorithm is applied to other IEEE standard buses (the IEEE 30-bus, 39-bus and 57-bus system). The proposed algorithm is coded using MatLab/C++ program and the code is run on a Windows XP computer, which has 3.5 GB RAM and 3.16 GHz processor. The code is attached to Appendix A of this thesis.

3.6.1 IEEE 30-bus system

As the name suggests, the IEEE 30-bus system, as shown in Fig. 3.5, contains 30 buses in total and there are 6 zero-injection buses located at bus 6, 9, 22, 25, 27 and 28. Compared to the IEEE 14-bus system, the complexity of this system increases dramatically, the total number of branches rising from 20 to 41. The information related to the structure and connectivity of the 30-bus system is tabulated in Table 3.8.

As stated in Section 3.2, the connectivity matrix will be generated and the spanning tree topology will be constructed automatically by Breadth-First Search Algorithm in the software program, which is included in Appendix A. The final optimum results carried out under normal operation with and without zero-injection effect, is tabulated in Table 3.9.

Table 3.8 Summary of the IEEE 30-bus system

System	Number of branches	Number of zero-injection bus	Location of zero-injection bus
30 bus	41	6	6,9,22,25,27,28

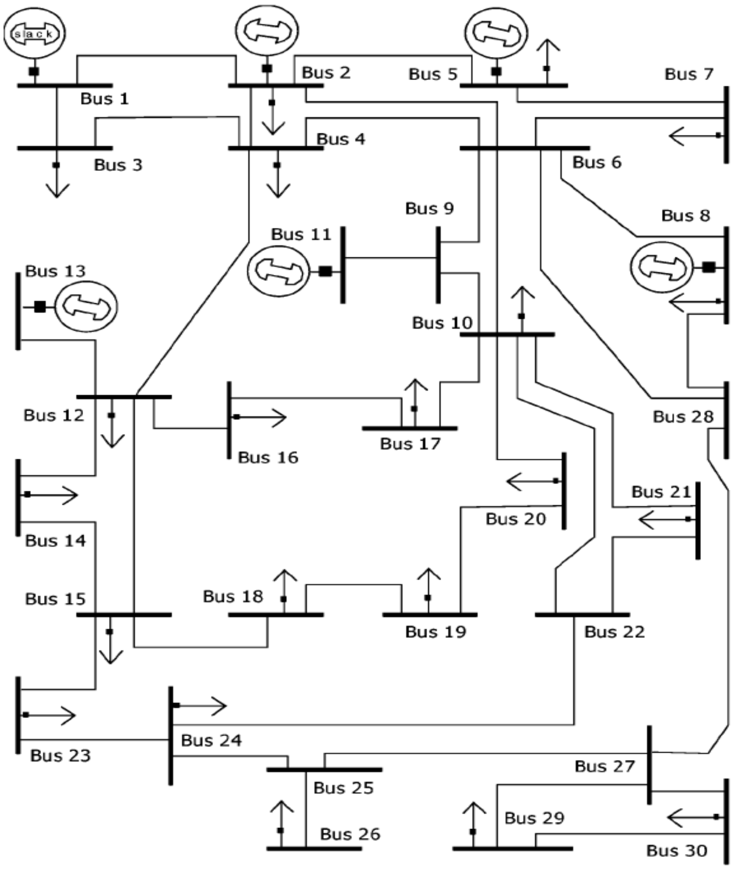


Figure 3.5 IEEE standard 30-bus test system

Table 3.9 Optimum PMU placement in normal operation with/without zero-injection effect

	Location of PMUs	Total Number of PMUs
Normal operation without zero-injection effect	2,4,6,9, 10,12,15, 19,25,27	10
Normal operation with zero-injection effect	2,4,10,12, 15,19,27	7

3.6.2 IEEE 39-bus system

The IEEE 39-bus system, as shown in Fig. 3.6, contains 39 buses in total and there are 12 zero-injection buses located at bus 1, 2, 5, 6, 9, 10, 11, 13, 14, 17, 19 and 22 as shown in Table 3.10. An identical procedure, stated in Section 3.3.1, is applied to the IEEE 39-bus system and the final results are tabulated in Table 3.11.

Table 3.10 Summary of IEEE 39-bus system

System	Number of branches	Number of zero-injection bus	Location of zero-injection bus
39 bus	46	12	1,2,5,6,9,10,11,13,14,17,19,22

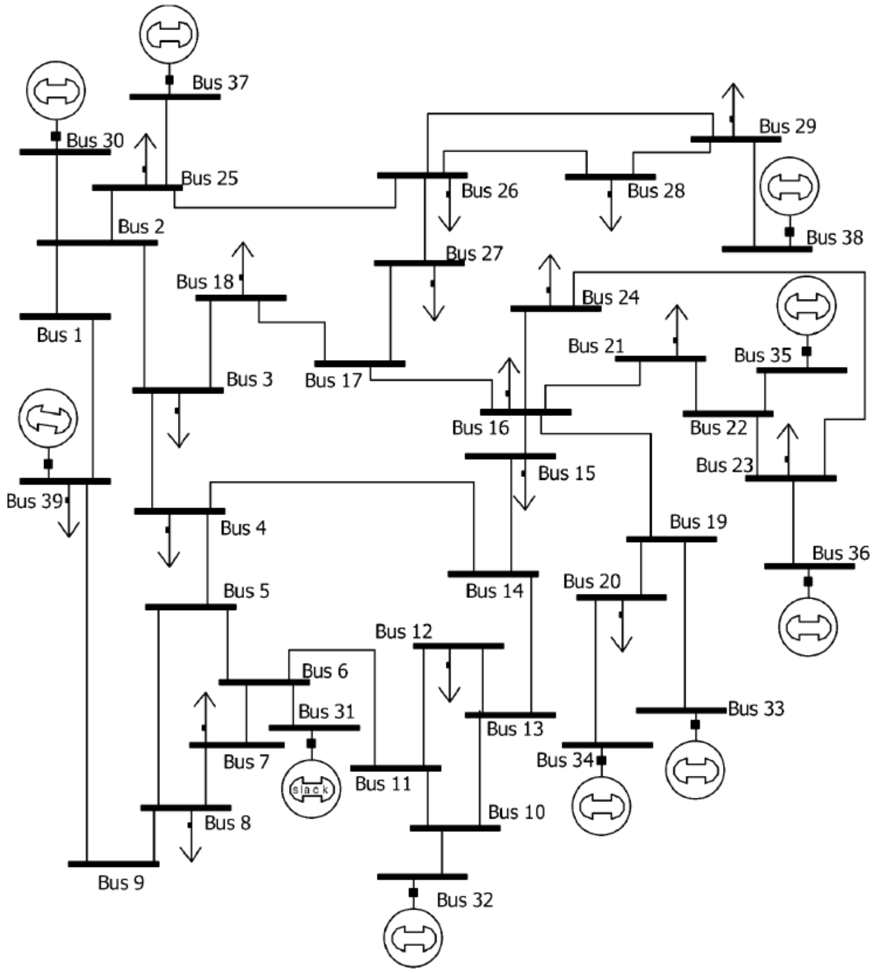


Figure 3.6 IEEE standard 39-bus test system

Table 3.11 Optimum number of PMUs needed in normal operation with/without zero-injection effect

	Location of PMUs	Total Number of PMUs
Normal operation without zero-injection effect	2,6,9,10,13, 14,17,19,20, 22, 23,25, 29	13
Normal operation with zero-injection effect	6,8,10,16, 20,23, 25,29	8

3.6.4 IEEE 57-bus system

The IEEE 39-bus system, as shown in Fig. 3.7, contains 57 buses in total and there are 15 zero-injection buses located at bus 4, 7, 11, 21, 22, 24, 26, 34, 36, 37, 39, 40, 45, 46 and 48 as shown in Table 3.12. The information of the system is tabulated in Table 3.9. An identical procedure, stated in Section 3.3.1, is applied to the IEEE 39-bus system and the final results are tabulated in Table 3.13.

An extensive comparison of the minimum number of PMUs obtained from this proposed hybrid approach, to other existing methods will be presented in Chapter 5 of this thesis.

Table 3.12 Summary of the IEEE 57-bus system

System	Number of branches	Number of zero-injection bus	Location of zero-injection bus
57 bus	80	15	4,7,11,21,22,24,26,34,36,37,39,40,45,46,48

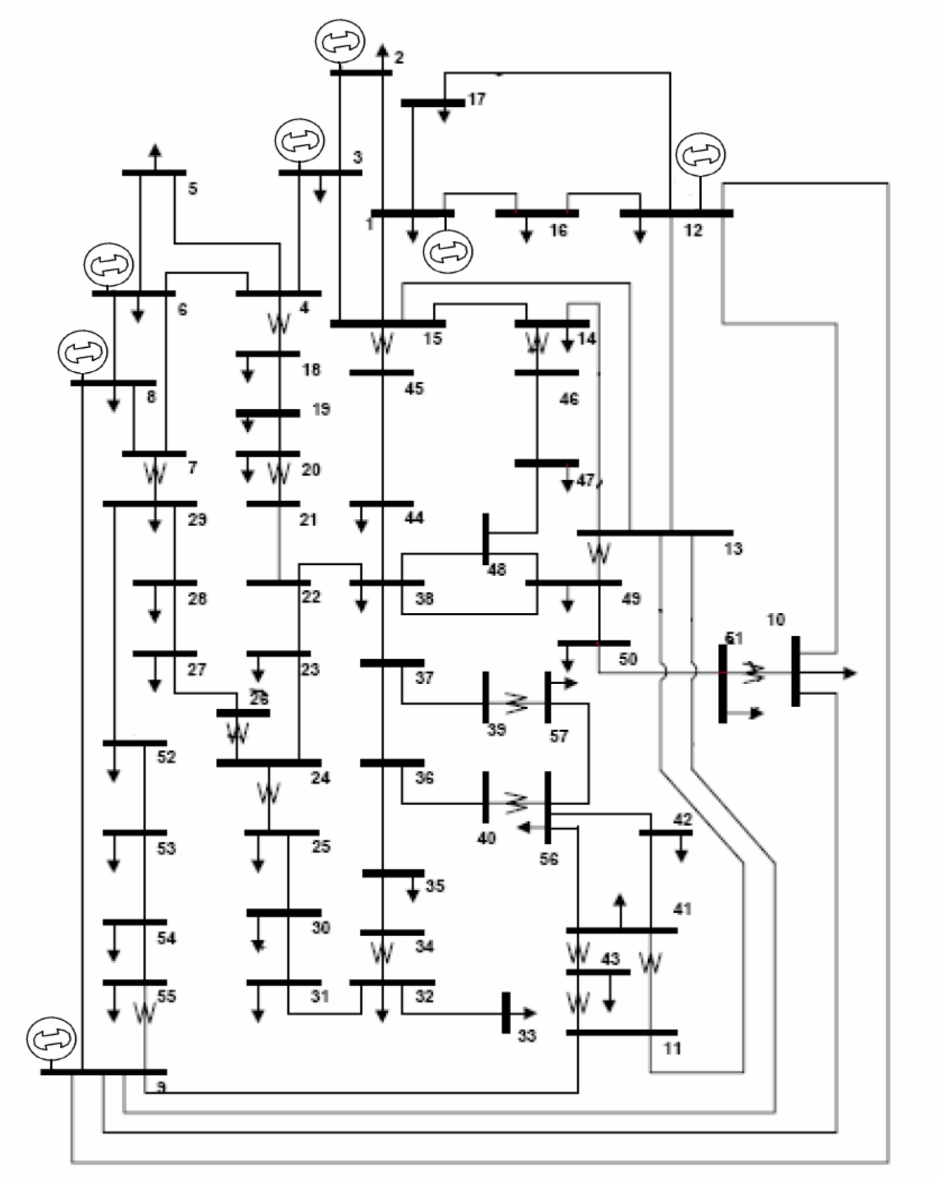


Figure 3.7 IEEE standard 57-bus test system

Table 3.13 Optimum number of PMUs needed in normal operation with/without zero-injection effect

	Location of PMUs	Total Number of PMUs
Normal operation without zero-injection effect	1,6,9,12,15,19,22,25,26,29,32, 36,38,41,47,50, 53,57	17
Normal operation with zero-injection effect	1,6,13,19,25,29, 32,38,51,54,56	11

3.7 Comparisons of the System Observability Redundancy Index

As discussed in Chapter 2, the System Observability Redundancy Index (SORI) will be applied to determine the best favorable outcome from different methods. As previously stated, a higher SORI value indicates that the PMU-based system is more reliable. Table 3.14 shows the results of the calculation of the system observability redundancy index, based on normal operation without zero-injection effect and normal operation with zero-injection effect.

When comparing the index value between other methods [39,42] and the proposed method, the redundancy in the 30-bus and 39-bus test system has been improved. The system with a higher redundancy index will generally cope better under circumstances in situations where bad data during transmission or loss of PMU due to faulty equipment occur. In the case of the IEEE 30-bus system, the SORI improves by a degree of 2 for normal operation without zero injection effect and a degree of 7 for normal operation with zero injection effect.

Table 3.14 System observability redundancy index in comparison

		14 BUS	30 BUS	39 BUS	57 BUS
Normal operation without zero-injection effect	Binary Search Method [42]	19	50	52	N/A
	Integer Programming [39]	N/A	N/A	N/A	N/A
	Proposed Method	19	52	52	72
Normal operation with zero-injection effect	Binary Search Method [39]	16	50	53	N/A
	Integer Programming [42]	16	54	53	64
	Proposed Method	16	57	54	64

N/A: not available

In this chapter, the problem of optimal PMU placement in a power network was investigated. A new PMU placement optimization method was introduced based on Breadth-First Search Algorithm and Greedy Algorithm. MatLab and C++ were written and used to automate the algorithm. The simulation results obtained from the IEEE standard 14-, 30-, 57- and the New England 39-bus test systems were presented and the results show that a hybrid approach involving the combination of two algorithms is effective in achieving optimum PMU placement.

Chapter 4 Hybrid approach based on the Global Search Algorithm

4.1 Introduction

Global Search Optimization is a branch of applied mathematics and numerical analysis that deals with the optimization of a function or a set of functions according to some criteria. Typically, a set of bounds and more general constraints is also present, and the decision variables are optimized considering also the constraints. In relation to the optimal PMU placement, a number of authors have confirmed that the minimal PMU location can be identified by the methodology used in Global Search Optimization [30].

In general, the Global Search Algorithm is able to determine the location of the minimal number of PMUs after searching all the possible combinations. In the searching process, a large number of combinations are generated, which leads to the consumption of extensive computational time and memory space in the user's computer, especially in a complex power network. As a solution, in this chapter, a group of innovative steps are introduced to locate, in an efficient way, the minimal number of PMUs needed, including: reducing single connection buses (Chapter 4.3); and the calculation of the upper and lower bounds using Domination Set (Chapter 4.4) and the Binary Search Algorithm (Chapter 4.5). This process will avoid the unnecessary combinations generated by the Global Search Algorithm. Furthermore, in the consideration of a normal operation with zero-injection effect, the author additionally applies a restriction (Chapter 4.6.2.1 and Chapter 4.6.2.3) which requires installing the devices of phasor measurement only at non zero-injection buses. In other words, the total size of the power network will scale down by eliminating the number of zero-injection buses, which enhances the resolving speed.

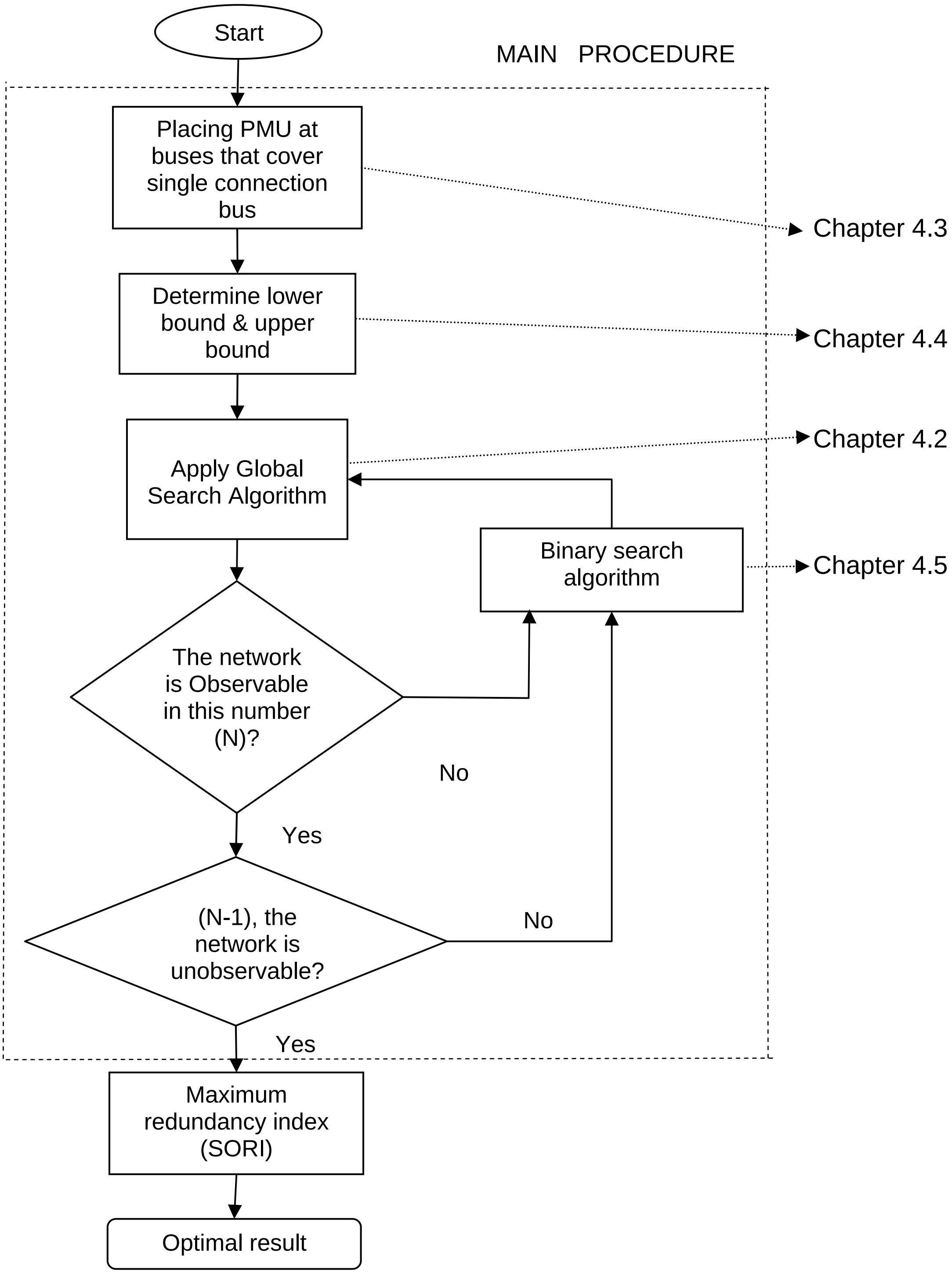


Figure 4.1 Flowchart for normal operation condition without zero-injection effect

4.2 The application of Global Search Algorithm

An Exhaustive Search [30], which is known as a ‘brute-force attack’, is defined as a branch of the Global Search Algorithm that can be employed to explore all the possible combinations. By checking any single combination, the optimal PMU placement will be determined if the combination is able to observe the entire power network with the least number of PMUs being required. The final solution is guaranteed to achieve a minimum of PMUs in the placement process. An example will be given to describe the procedure of Exhaustive Search, as follows:

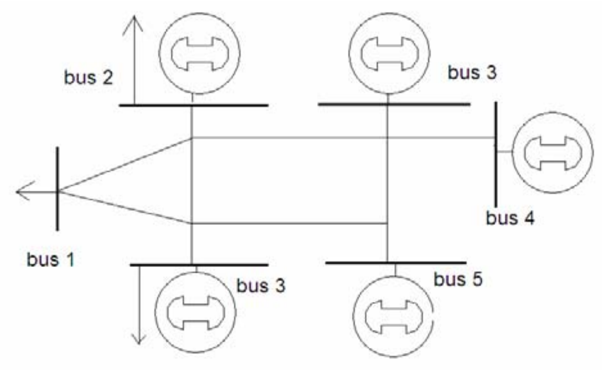


Figure 4.2 Single line diagram of a 6-bus

In the 6-bus system above, assume that the number of PMUs needed is 2. To generate all the possibilities, if the programme is executed from [1 2] as an initial value, the count of the last digit is going to increase until it reaches the number of buses in total. As a trigger, the previous digit is able to carry by 1. Repeat this process until the first digit is less than the next digit by one.

1 2
1 3
1 4
1 5
1 6
2 3
2 4
2 5
2 6
3 4
3 5
3 6
4 5
4 6
5 6

Based on the connectivity of each bus, any adequate combination will be selected if the buses are able to cover the entire network. (e.g. with the combination [2 4], bus 2 covers buses1 ,2, 3 and 4, while bus 4 covers buses 4, 5 and 6)
The following formula shows the total number of combinations in k_{th},

$$\sum_{k=1} C_k^N = \frac{N!}{k!(N-k)!} \tag{4.1}$$

where the parameter N represents the size of the network and the parameter k represents the number of PMUs needed.

Exhaustive Search can be implemented using a computer program, and will be executed for the full range of possibilities from k=1 to k=N, until the network is completely observable. However, the computer will spend a larger amount of computational time and use up a lot of computer memory space if it carries out all the possibilities for locating the value of k. To reduce the number of possible combinations, the principles of covering single connection buses, establishing the upper and lower bounds and the Binary Search Method should be implemented; these principles are discussed below.

4.3 PMU Placement to cover single connection buses

To begin with, a PMU will be placed at the adjacency of the single connected bus (degree=1 in the adjacency matrix) to avoid leaving out the single bus. Furthermore, installing the PMU at the adjacency of the single bus will achieve a better performance than it having been already installed, due to the increased occurrences of connectivity. After that, all the candidates without the single buses and their adjacent buses where PMUs have been installed can be identified as the candidate locations for placing PMUs.

4.4 Calculation of upper and lower bounds using Domination Set

Domination Set is known as the dominating set problem in Graph Theory and is a decision-making problem in computational complexity theory which is classical, non-deterministic, polynomial (NP) and complete in nature [44]. This concept can be brought into the placement of minimal measurement devices in an electric power network, due to their similar numerical structures. In previous published works, Haynes et al verified the bounds of the Domination Set in the topology as a tree graph and, after that, Yuan [27] successfully implemented the formula of upper and lower bounds in the Integer Programming. As listed below, equations (4.2) and (4.3) represent the lower bound and upper bound of the power dominating set respectively. Note that the result of such bounds will be an integer

$$\gamma p(T) \geq \frac{k+2}{3} \tag{4.2}$$

$$\gamma p(T) \leq \frac{n}{3} \tag{4.3}$$

where k counts the number of the bus which has at least 3 degrees and n is the total number of the bus.

According to the upper and lower bounds, the number of PMUs needed will be restricted in a set of bounds. This is an efficient way to downsize the total number of possibilities, which leads to a reduction in the entire execution time as well as in the use of computer memory space.

4.5 The application of Binary Search Algorithm

In computer science, a Binary Search finds a particular element in an orderly array [30]. In the program, the search process starts from the element in the middle of the array. The condition can be decided by the following requirements.

- Given a sorted sequence, compare the value in the array at position $n/2$ with a key k.
- If $A[n/2] > k$, compare k with the midpoint of the lower half.
- If $A[n/2] < k$, compare k with the midpoint of the upper half.
- If $A[n/2] = k$, return $n/2$ (the index of the position containing k).

In the proposed method, the interval of a sorted sequence has been defined by the approach of the upper and lower bounds in the previous section. Binary Search Algorithm simplifies the process of finding the optimal number of PMUs needed to render the network fully observable, instead of generating all possibilities. Basically, the checking procedure begins with the value of the lower bound; meanwhile, Global Search Algorithm generates all the possible combinations in this value. The optimal solution will be confirmed if the buses in any combination can observe the entire network. Otherwise, Binary Search Algorithm will be applied to find key k. However, the fact is that any sequential number above the minimal number of PMUs needed also leads to the generation of solutions which cover the entire network. In other words, it is not certain that the optimal position of PMUs with the least placement will be identified by using the conventional Binary Search Algorithm. To solve this issue, the author will give a definition to finalize the minimal number of PMU's in:

Under a sequence number (*N*), one combination leads to the network being fully covered; in contrast, there is no combination that leads to the network being fully covered under the sequence number N-1 (the previous number in order).

An example will be presented to describe the process of Binary Search Algorithm, as follows:

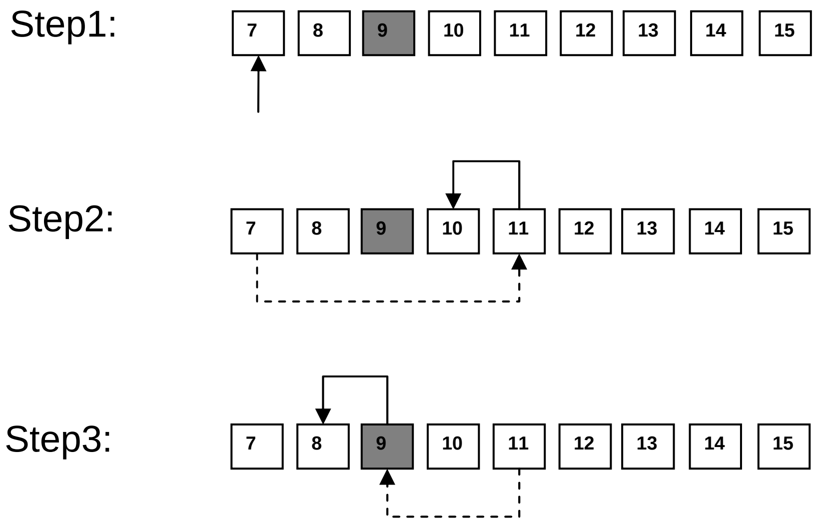


Figure 4.3 Example of binary search

In such a programming procedure (figure 4.3), assume that the number of optimal PMUs needed is nine in a sorted sequence from seven to fifteen. Note that, the command of 'detected' can be explained as one combination that makes the network fully observed and the command of 'undetected' can be explained as no combination that results in the network being fully observed

First of all, as the value of the lower bound, number seven, will be the starting point from which to search for any combination that makes the network fully observed. However, none of them can reach the requirement of observability. Secondly, the Binary Search Algorithm is applied and the search process starts from the element in the middle of the array (number eleven). By searching under the value of eleven, the solution is verified as "detected". Simultaneously, the previous sequence, number ten, will be examined with a label of 'detected'. After that, the searching position moves to the midpoint of the lower half, which is number nine. By the same procedure above, number nine and eight will be assessed as “detected” and “undetected” respectively. Accordingly, number nine is the minimal number for placing PMUs.

As demonstrated above, the optimal solution is only performed in the selected digit, which will ignore the generation of unnecessary combinations, and lead to a further reduction of the total execution time.

Last but not least, the System Observability Redundancy Index (definition in chapter 2) will also be applied in the implementation of this hybrid algorithm. It is a significant methodology for selecting the most efficient solution.

4.6 Implementation of Global Search Algorithm on IEEE 14-bus system

In this section, a solution to the problem of optimal PMU placement will be presented; this solution combines Global Search Algorithm, the approach of lower and upper bounds and Binary Search Algorithm. The IEEE standard 14-bus system, as shown in Fig. 4.4, will first be considered and studied under normal operation with and without zero-injection effect.

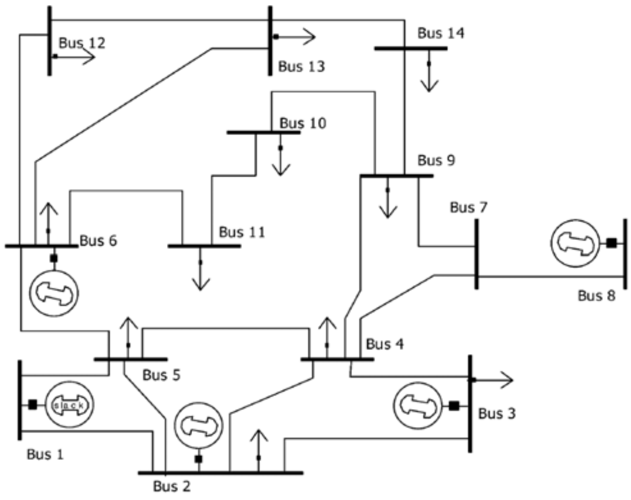


Figure 4.4 IEEE 14-bus test system

4.6.1 Normal operation condition without zero-injection effect

Initially, the equation 4.1 shows the way of generating the connectivity matrix A, as follows:

$$A_{k,m} = \begin{cases} 1 & if \quad k = m \ or \ k \ and \ m \ are \ connected \\ 0 & if \quad otherwise \end{cases} \qquad (4.4)$$

Meanwhile, for the purpose of programming, the connectivity matrix is converted into binary format. Matrix A presents the interconnectivity of the IEEE standard 14-bus system.

$$A=\begin{bmatrix}
1&1&0&0&1&0&0&0&0&0&0&0&0&0\\
1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&1&1&1&0&0&0&0&0&0&0&0&0&0\\
0&1&1&1&1&0&1&0&1&0&0&0&0&0\\
1&1&0&1&1&1&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&0&0&0&0&1&1&1&0\\
0&0&0&1&0&0&1&1&1&0&0&0&0&0\\
0&0&0&0&0&0&1&1&0&0&0&0&0&0\\
0&0&0&1&0&0&1&0&1&1&0&0&0&1\\
0&0&0&0&0&0&0&0&1&1&1&0&0&0\\
0&0&0&0&0&1&0&0&0&1&1&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&1&1&0\\
0&0&0&0&0&1&0&0&0&0&0&1&1&1\\
0&0&0&0&0&0&0&0&1&0&0&0&1&1
\end{bmatrix} \tag{4.5}$$

Secondly, a PMU will be placed at the adjacency of the single connected bus in a 14-bus system. Bus 8 is the single bus and bus 7 is the one which will be selected for the installation of a PMU. Moreover, both buses 7 and 8 will be eliminated from the array of the candidate locations for placing PMUs, as shown in table 4.1.

Table 4.1 Example of locations for placing PMUs in a 14-bus system in normal operation without zero-injection effect

Test system	The bus covers single connection Bus	Locations for placing PMUs
IEEE 14-bus	7	1,2,3,4,5,6,9,10,11,12,13,14

By utilizing the equations of lower & upper bound, the value of lower and upper bound can be calculated as follows,

$$\text{Lower bound: } \frac{k+2}{3} \qquad\qquad \text{Upper bound: } \frac{n}{3}$$

where k represents the vertices of degree of at least 3. In a 14-bus system, the index of degree in bus 2, 4, 5, 6, 7, 9 and 13 are equal or more than three.
n shows the total number of vertices which is calculated as fourteen in all.

$$\text{Lower bound: } \frac{7+2}{3} = 3 \qquad\qquad \text{Upper bound: } \frac{14}{3} = 4$$

As a result, the set of bound is 3 to 4 in integer

Beginning with the value of lower bound, the number of PMUs needed is determined as 3. The system will execute all the combinations by an Exhaustive Search and check the network observability by determining if any combination is able to cover the entire power network.

1	2	7
1	3	7
1	4	7
1	5	7
1	6	7
1	9	7
1	10	7
1	11	7
1	12	7
1	13	7
..	..	..

After checking the network observability, none of the combinations is competent to observe the entire system. Additionally, Binary Search Algorithm is applied to operate number four as the optimal number of PMUs needed. An example of a possible combination in a set of four is shown below:

1	2	3	7
1	2	4	7

1	2	5	7
1	2	6	7
1	2	9	7
1	2	10	7
1	2	11	7
1	2	12	7
1	2	13	7
1	2	14	7
..	..	..	..

The optimal consequences are processed as follows:

2	6	9	7
2	10	13	7
2	11	13	7

There are three possible solutions which give the total number of PMUs as 4. These combinations will have PMUs located at bus 2, 6, 7 and 9, bus 2, 7, 10 and 13 or 2, 11,13,7. In order to select the optimum solution from these three solutions, the redundancy index needs to be calculated and the solution with the higher redundancy index will be selected as the optimal solution. For the IEEE 14-bus system, the placement at bus 2, 6, 7 and 9 gives a higher SORI value, as shown in Table 4.3.

Finally, by applying the definition of SORI, bus [2 6 9 7] offers the achievement of a higher total degree value, indicates that the monitoring system is more reliable.

Table 4.2 Comparison of SORI in a 14-bus system

Optimal Location of PMUs	System Observability Redundancy Index (SORI)
2, 6 , 7, 9	19
2, 7, 10, 13	16
2, 7, 11, 13	16

4.6.2 Normal operation condition without zero-injection effect

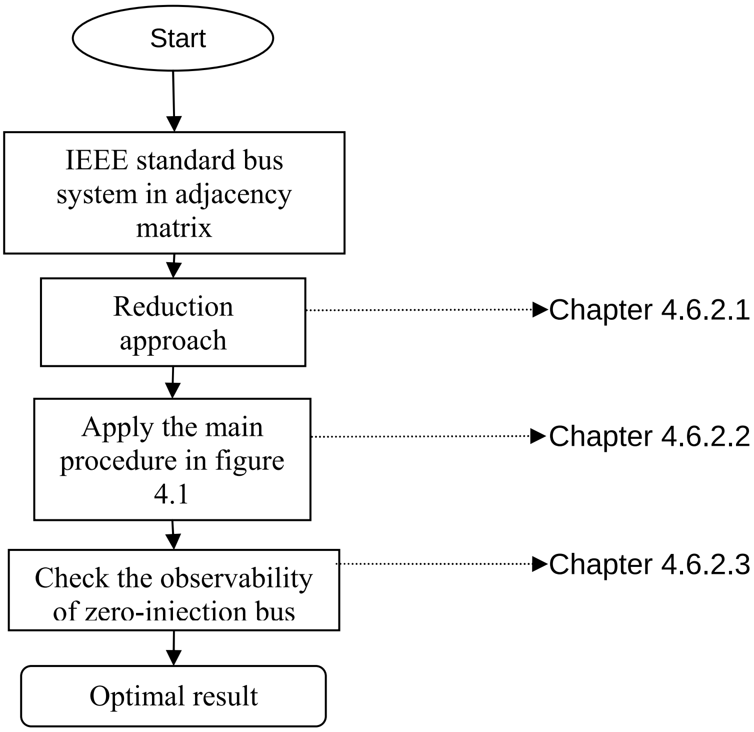


Figure 4.5 Flowchart for normal operation condition with zero-injection effect

A zero-injection bus can be generalized as a bus without a generator and load connection. The zero-injection effect has been categorized as follows [39]:

1) When buses which are incident to an observable zero-injection bus are all observable except one, the unobservable bus will also be identified as observable by applying Kirchhoff's current law (KCL) at the zero-injection bus.

2) When buses which are incident to an unobservable zero-injection bus are all observable, the zero-injection bus will also be identified as observable by applying the KCL at the zero-injection bus.

In this proposed method, once a zero-injection bus has been detected, all of its neighboring buses are considered as virtually connected. For instance, in an IEEE 14-bus system, bus 7 has no direct generator or load connection; therefore it is classified as a zero-injection bus and bus 8 will be treated as directly connected to bus 4 and bus 9 when applying the algorithm.

4.6.2.1 Reduction approach

A reduction approach is a new approach proposed in this thesis employed to downsize the full network in the condition of normal operation with zero-injection effect. As mentioned previously, all of the zero-injection neighboring buses are considered as virtually connected. This allows the avoidance of all the zero-injection buses from the group of placement locations. Consequently, the size of the power system will be efficiently reduced, especially for the network with more zero-injection bus.

Table 4.3 Example of locations for placing PMUs in a 14-bus system in normal operation with zero-injection effect

Test system	Zero-injection bus	Initial bus	Location for placing PMUs
IEEE 14-bus	7	none	1,2,3,4,5,6,8,9, 10,11,12,13,14

4.6.2.2 Main procedure

According to the main procedure described in Chapter 4.2 - 4.6, the interval between the lower and upper bounds can be confirmed as 3 to 4. The fully observed power network will be verified as follows:

```
1     2     3
1     2     4
1     2     5
1     2     6
1     2     8
1     2     9
1     2     10
..    ..    ..
```

In the group, the optimal consequence is processed as follows:

```
 2    6    9
```

In the end, the programme will terminate here due to the network being fully covered by this unique solution [2, 6, 9].

4.6.2.3 Check the observability of zero-injection bus

Due to all the zero-injection buses having been eliminated from the total, some buses may escape being monitored by measurement devices. A checking procedure is introduced to ensure that any neighbouring bus of zero-injection has been covered by the existing optimal solution, which is based on the rules of network observability described in chapter 2.4. In a 14-bus system, for instance, the existing optimal solution is bus 2, 6 and 9, which is measured by the main procedure, and bus 7 is the zero-injection bus, which connects to bus 4, 8, and 9. Accordingly, bus 4 is supervised directly by the PMUs in buses 2 and 9. The information of current and voltage in bus 8 can be calculated by bus 4 and bus 9 due to pseudo measurement (case 2 and 3 in chapter 2.4). Lastly, bus 9 is covered by the PMU which has already been installed.

After that, by employing the SORI, the solution of bus [2 6 9] is the only solution that offers a higher SORI value, which indicates that the monitoring system is more reliable.

As a result, the optimal solution is bus 2, 6 and 9. In summary, the total number of PMUs required to achieve full observability of the network with/without zero injection effect can be summarised in Table 4.4:

Table 4.4 Number of PMUs needed and locations in normal operation with/without zero-injection effect based on the 14-bus system

	Location of PMUs	Total Number of PMUs
Normal operation without zero-injection effect	2,6,7,9	4
Normal operation with zero-injection effect	2,6,9	3

4.7 Illustrative examples on IEEE 24-bus, 30-bus and 39-bus

In this section, the proposed hybrid algorithm combining Global Search Algorithm, the approach of lower and upper bound and Binary Search Algorithm is applied to other IEEE standard buses (the IEEE 30-bus, 39-bus and 57-bus system). The proposed algorithm is coded using the MatLab program and the code is run on a Windows XP computer, which has 3.5 GB RAM and 3.16 GHz processor. The code is attached in Appendix B of this thesis.

4.7.1 IEEE 24-bus System

The IEEE 24-bus system is displayed in Fig.4.6 and the information of the system is tabulated in Table 4.5. As seen in Table 4.6 and 4.7, the optimal PMU placement for the 24-bus system under consideration with or without zero-injection effect is presented. The total number of PMUs needed is counted as well.

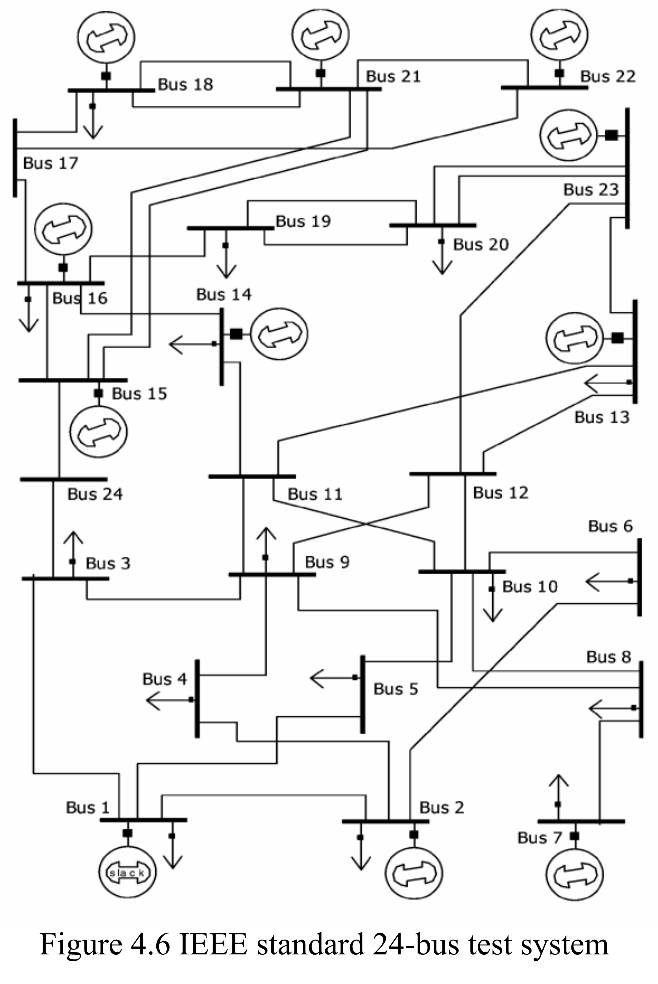


Figure 4.6 IEEE standard 24-bus test system

Table 4.5 The information of a 24-bus system

System	Number of branches	Number of zero-injection bus	Location of zero-injection bus
24 bus	38	4	11,12, 17, 24

Table 4.6 The initial bus and location for placing PMUs in normal operation with/without zero-injection effect

	Test system	Zero-injection bus	PMU to cover single connection bus	Location for placing PMUs
Normal operation without zero-injection effect	IEEE 24-bus		8	1,2,3,4,5,6,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24
Normal operation with zero-injection effect		11,12,17,24	8	1,2,3,4,5,6,9,10, 13,14,15,16,18,19,20,21,22,23

Table 4.7 Number of PMUs needed and locations in normal operation with/without zero-injection effect based on 24-bus system

Normal operation without zero-injection effect		Normal operation with zero-injection effect	
Number of PMUs	Location of PMUs	Number of PMUs	Location of PMUs
7	2,3,8,10,16,21,23	6	2,8,10,15,20,21

4.7.2 IEEE 30-bus System

The IEEE 30-bus system is displayed in Fig.4.7 and the information of the system is tabulated in Table 4.8. As seen in Table 4.9 and 4.10, the optimal PMU placement for the 30-bus system under consideration with or without zero-injection effect is presented. The total number of PMUs needed is counted as well.

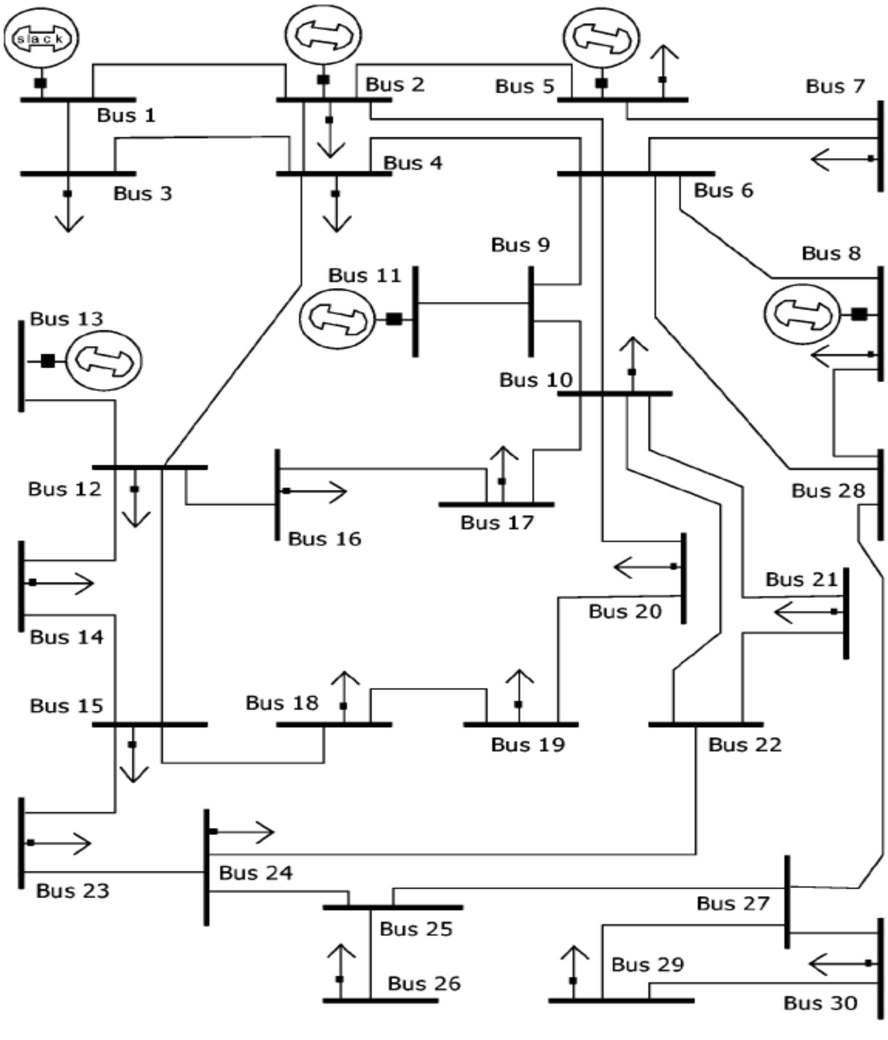


Figure 4.7 IEEE standard 30-bus test system

Table 4.8 the information of 30-bus system

System	Number of branches	Number of zero-injection bus	Location of zero-injection bus
30 bus	41	6	6,9,22,25,27,28

Table 4.9 the initial bus and location for placing PMUs in normal operation with/without zero-injection effect

	Test system	Zero-injection bus	PMU to cover single connection bus	Location for placing PMUs
Normal operation without zero-injection effect	IEEE 30-bus		9,12,25	1,2,3,4,5,6,7,8,10,14,15,16,17,18, 19,20,21, 22,23,24,27,28,29,30
Normal operation with zero-injection effect		6,9,22, 25,27,28	12	1,2,3,4,5,7,8,10,11,14,15,16,17, 18,19,20,21,23,24,26,29,30

Table 4.10 Number of PMUs needed and locations in normal operation with/without zero-injection effect based on 30-bus system

Normal operation without zero-injection effect		Normal operation with zero-injection effect	
Number of PMUs	Location of PMUs	Number of PMUs	Location of PMUs
10	2,4,6,9,10,12,15, 18,25,27	7	2,4,10,12,18,24,29
			2,4,10,12,18,24,30
	2,4,6,9,10,12,15, 19,25,27		2,4,10,12,19,24,29
			2,4,10,12,19,24,30
	2,4,6,9,10,12,15, 20,25,27		2,4,10,12,20,24,29
			2,4,10,12,20,24,30

4.7.3 IEEE 39-bus System

The IEEE 39-bus system is displayed in Fig.4.8 above. The information of the system is tabulated in Table 4.11. In addition, Table 4.12 and 4.13 represents the optimal PMU placement under consideration with or without zero-injection effect. The total number of PMUs needed is counted as well.

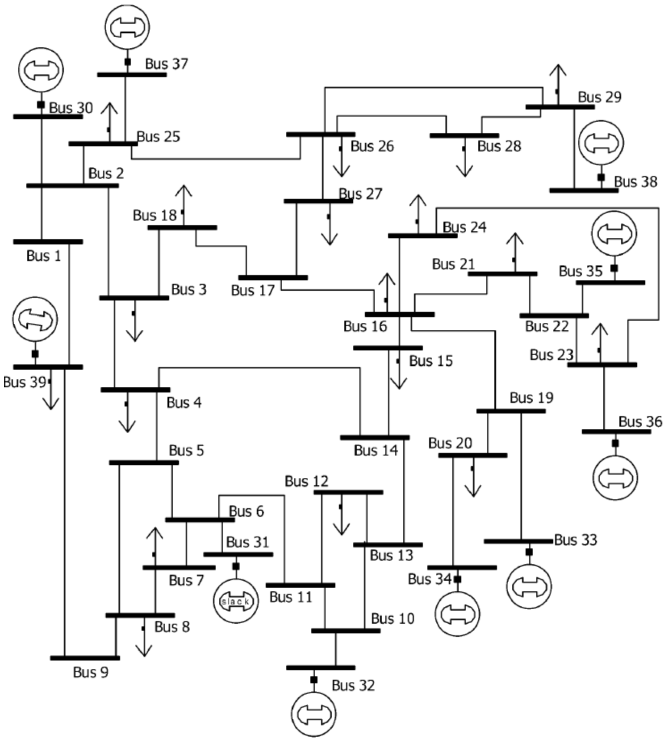


Figure 4.8 IEEE standard 39-bus test system

Table 4.11 the information of a 39-bus system

System	Number of branches	Number of zero-injection bus	Location of zero-injection bus
39 bus	46	12	1,2,5,6,9,10,11,13,14,17,19,22

Table 4.12 the initial bus and location for placing PMUs in normal operation with/without zero-injection effect

	Test system	Zero-injection bus	Initial bus	Location for placing PMUs
Normal operation without zero-injection effect	IEEE 39-bus		2,6,10,19,20,22,23,25,29	1,3,4,5,7,8,9,11,12,13,14,15,16,17,18,21,24,26,27,28,39
Normal operation with zero-injection effect		1,2,5,6,9,10,11,13,14,17,19,22	20,23,25,29	3,4,7,8,12,15,16,18,21,24,26,27,28,30,31,32,33,35,39

Table 4.13 Number of PMUs needed and locations in normal operation with/without zero-injection effect based on a 39-bus system

Normal operation without zero-injection effect		Normal operation with zero-injection effect	
Number of PMUs	Location of PMUs	Number of PMUs	Location of PMUs
13	2,6,9,10,11,14,17,19,20,22,23,25,29	8	3,8,12,16,20,23,25,29
	2,6,9,10,13,14,17,19,20,22,23,25,29		

In this chapter, a novel PMU placement optimization method was introduced based on the Global optimization algorithm with additional approaches. By employing the MatLab, the simulation results on IEEE standard 14-, 24-, 30- and the New England 39-bus test systems were tabulated and this assisted in proving that the hybrid algorithm works effectively and adequately. The main comparison will be presented in Chapter 5.

Chapter 5 Discussion and Conclusion

In this chapter, to investigate the optimal PMU placement in the power system, a comprehensive comparison between the proposed methods and the existing methods reported in the literature review has been tabulated in Table 5.1. Due to the similarity of results in the consideration of normal operation without zero-injection effect, as well as there being only a few published papers on this subject, the information regarding the number of PMUs needed and the system observability redundancy index (SORI) will be applied, herein, under the condition of normal operation with zero-injection effect.

Table 5.1 Comparison between proposed methods and the methods in published papers in minimal PMUs needed

		IEEE 14-bus system	IEEE 24-bus system	IEEE 30-bus system	IEEE 39-bus system	IEEE 57-bus system
Proposed methods	Hybrid Approach based on Approximation Algorithm[43]	3	N/A	7	8	11
	Hybrid Approach based on Global Search Algorithm [chapter 4]	3	6	7	8	N/A
Published paper in Literature Review	Depth-First Algorithm (DFS)[25]	6	N/A	N/A	N/A	21

	Domination Set Approach[27]	3	N/A	7	8	N/A
	Greedy Algorithm[28]	3	N/A	8	12	N/A
	Genetic Algorithm[31]	3	N/A	7	N/A	12
	Particle Swarm Optimization (PSO)[37]	3	N/A	7	N/A	13
	Integer Programming[39]	3	N/A	7	8	11
	Binary Search Algorithm[42]	3	6	7	8	N/A

N/A: not available

As the table demonstrated, the proposed methods are able to achieve the optimal PMU placement in IEEE 14-Bus, 24-Bus, 30-Bus, 39-Bus and 57-bus system. For a hybrid approach based on an Approximation Algorithm which includes Breath-First Algorithm and Greedy Algorithm, the minimal number of PMUs needed achieves a better performance compared to the procedure which is carried out by Greedy Algorithm alone [28]. The optimum number of PMUs is significantly reduced by 1 and 4 in IEEE 30-bus system and 39-bus system respectively. Besides, a hybrid approach based on a Global Search Algorithm, which is represented in chapter 4, is capable of achieving the requirement of the least number of PMUs from IEEE 14-bus system to IEEE 39-Bus system in PMU count 3, 6, 7, 8 relatively.

In addition, in Table 5.2, Integer Programming and Binary Search will be implemented to evaluate the system observability redundancy index (SORI), because of the optimal number of PMUs in the former simulated result as shown in Table 5.1. Note that a higher SORI value indicates that the PMU-based monitoring system is more reliable.

Table 5.2 Comparison between proposed methods and the methods in published papers in system observability redundancy index (SORI)

		IEEE 14-bus system	IEEE 24-bus system	IEEE 30-bus system	IEEE 39-bus system	IEEE 57-bus system
Proposed methods	Hybrid Approach based on Approximation Algorithm	16	N/A	57	54	64
	Hybrid Approach based on Global Search Algorithm	16	30	54	53	N/A
Related Methods	Integer programming[39]	16	N/A	54	53	64
	Binary Search Algorithm [42]	16	30	50	53	N/A

In comparison with the latest relevant papers, the redundancy indexes in both proposed hybrid approaches are adequate and capable of analysing the investigation of optimal PMU placement. In table 5.2, the proposed method of a hybrid approach based on an approximation algorithm achieves the best performance, which exceeds the Integer Programming by 3 degrees and the Binary Search Algorithm by 7 degrees in an IEEE 30-bus system, as well as by 1 degree in IEEE 39-bus system.

In comparison with the two proposed methods, the solution of a hybrid approach based on a Global Search Algorithm, which is simulated without PMUs installed in any zero-injection bus, has a lower redundancy index in IEEE 30-Bus and 39-Bus system. Thus, an assumption of installing a PMU in any location will obtain a reliable and stable solution to lead to the network being fully observable.

Finally, to contrast the approximation algorithm and global search algorithm generally, searching and locating the optimal position in polynomial time is a benefit of implementing the approximation algorithm. Consequently, the final solution is tractable, and efficient in terms of computation time, even in the case of a large-scale power network. This algorithm is targeting the local minimal placement to lead to the final optimization which is no guarantee that the PMU placement is minimal. On the other hand, there is no doubt that the global optimization algorithm will search all the combinations until it locates the most suitable optimal solution. Nevertheless, the more possibilities that have been explored, the more execution time will be required. Accordingly, for a large-scale network, a super fast computer should be applied to lead to a reduction in the total computation time.

In this day and age, computer technology is growing fast. Hence, the computational speed will not be a barrier to executing large amounts of information. More and more adaptive Global Search Algorithms are implemented to solve the problem of optimal PMU placement. The author believes that Global Search Optimization will become the chief resource to handle the optimization problem.

Conclusion

Due to the increasing development of power networks, their control systems and protection requirements are becoming complex. In recent years, the theory of synchronized phase angle measurement has verified that it brings a deep-seated advantage for the network real-time protection. Furthermore, along with the device of Phasor Measurement unit, these applications will enhance, in reality, the power system monitoring, control and protection. In this thesis, the author has aimed to ensure that the number of PMUs required for supervising the power network is minimal. If the premise is network observability, installation with less PMUs can be transferred in combinational optimization problems. Two varied combination algorithms have been introduced, separately, in Chapter 3 and Chapter 4. To summarize, by researching the Approximation Algorithm and the Global Optimization Algorithm, it has been found that both algorithms are able to deal successfully with the problem of optimal PMU placement.

In the future, the Optimal Placement Algorithm should be implemented into real networks, owing to the absence of information in the models of IEEE standard bus. Accordingly, this is the way to ascertain that the selected locations of PMUs qualify to guard the entire system against any information being lost in the transmission line.

Reference

[1] A. G. Phadke, J. S. Thorp, and K. J. Karimi, “State Estimation with Phasor Measurements”, in *IEEE Transactions on Power Systems*, Vol. 1, No. 1, pp. 233-241, February 1986.

[2] A. G. Phadke, “Synchronized phasor measurements in power systems,” in *IEEE Computer Applications in Power*, Vol. 6, Issue 2, pp. 10-15, April 1993.

[3] “Calculation of phase angle”, http://www.sengpielaudio.com/calculator-timedelayphase.htm Retrieved 2012-04-07

[4] A. G. Phadke and J. S. Thorp*, Synchronized Phasor Measurements and Their Applications*. New York: Springer, 2008.

[5] "USNO NAVSTAR Global Positioning System". U.S. Naval Observatory. http://tycho.usno.navy.mil/gpsinfo.html. Retrieved 2012-03-27.

[6] D. Allan, N. Ashby, C. Hodge, “The Science of Timekeeping”, Hewlett Packard Application Note 1289, 1997.

[7] Y. Liu, L. Mili, J. Ree, R. Nuqui, "State Estimation and Voltage Security Monitoring Using Synchronized Phasor Measurement", Virginia Polytechnic Institute and State University.

[8] “Phasor Measurement Unit model”, http://www.macrodyneusa.com/model_1690.htm. Retrieved 2011-6-13.

[9] “Consortium for Electric Reliability Technology Solution”, http://www.phasor-rtdms.com/phaserconcepts/phasor_adv_faq.html. Retrieved 2012-01-13.

[10] Abbasy, N.H.; Ismail, H.M.; , "A Unified Approach for the Optimal PMU Location for Power System State Estimation," *Power Systems, IEEE Transactions on* , vol.24, no.2, pp.806-813, May 2009

[11] J.J. Allemong, et. al., “A fast and reliable state estimation algorithm for AEP’s new control center”, IEEE Transactions on PAS, Vol. 101, No. 4, April 1982, pp 933–944.

[12] E.Handshin, et.al., “Bad data analysis for power system static state estimation”,IEEE Transactions on PAS, Vol. 94, No. 2, March/April 1975, pp 329–337.

[13] J. Chen; A. Abur , "Placement of PMUs to Enable Bad Data Detection in State Estimation," *Power Systems, IEEE Transactions on* , vol.21, no.4, pp.1608-1615, Nov. 2006

[14] J.F. Dopazo, et. al., “Implementation of the AEP real-time monitoring system”,IEEE Transactions on PAS, Vol. 95, No. 5, September/October 1975,pp 1618–1529.

[15] Y. Cheng; X. Hu; B. Gou; , "A new state estimation using synchronized phasor measurements," *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on* , vol., no., pp.2817-2820, 18-21 May 2008

[16] H. Yin; L. Fan; , "PMU data-based fault location techniques," *North American Power Symposium (NAPS), 2010* , vol., no., pp.1-7, 26-28 Sept. 2010

[17] Y. Lin; C. Liu; C. Chen; , "A new PMU-based fault detection/location technique for transmission lines with consideration of arcing fault discrimination-part I: theory and algorithms," *Power Delivery, IEEE Transactions on* , vol.19, no.4, pp. 1587- 1593, Oct. 2004

[18] Y. Lin; C. Liu; C. Chen; , "A new PMU-based fault detection/location technique for transmission lines with consideration of arcing fault discrimination-part II: performance evaluation," *Power Delivery, IEEE Transactions on* , vol.19, no.4, pp. 1594- 1601, Oct. 2004

[19] A.O.Ibe and BJ.Cory, "A traveling wave-based fault locator for two and three-terminal networks," IEEE Trans. Power Syst., vol. PWRD-I,no. 2, pp. 283-288, April 1986.

[20] T.Takagi, Y.Yamakoshi, M.Yamaura, R.Kondow, and T.M.atsushima,"Development of a new type fault locator using the one-terminal voltage and current data," IEEE Trans. Power App. Syst., vol. PAS-101, no. 8,pp. 2892-2898, August 1982.

[21] De La Ree, J.; Centeno, V.; Thorp, J.S.; Phadke, A.G.; , "Synchronized Phasor Measurement Applications in Power Systems," *Smart Grid, IEEE Transactions on* , vol.1, no.1, pp.20-27, June 2010

[22] “Keithley's Ultra-Fast Current-Voltage System Combines Three Essential Characterization Capabilities in One Chassis”, http://www.keithley.com/news/prod021810. Retrieved 2012-06-01.

[23] N.M. Manousakis, G.N. Korres, P.S. Georgilakis, "Optimal placement of phasor measurement units: A literature review," *Intelligent System Application to Power Systems (ISAP), 2011 16th International Conference on* , vol., no., pp.1-6, 25-28 Sept. 2011

[24] M. Farsadi, H.Golahmadi, H. Shojaei, , "Phasor Measurement Unit (PMU) allocation in power system with different algorithms," *Electrical and Electronics Engineering, 2009. ELECO 2009. International Conference on* , vol., no., pp.I-396-I-400, 5-8 Nov. 2009

[25] T.T.Cai and Q.Ai, “Research of PMU optimal placement in power systems,” in 2005 World Scientific and Engineering Academy and Society Int. Conf., pp.38-43.

[26] T.W. Haynes et. al., 2002. “Domination ingraphs applied to electric power networks”. *SIAM J. Disc. Math. 15*, 4, 519–529.

[27] X.A. Yuan, "A linear algorithm for minimum Phasor measurement units placement," *Innovative Smart Grid Technologies (ISGT), 2010* , vol., no., pp.1-3, 19-21 Jan. 2010

[28] M. Zhou, V. A. Centeno, A. G. Phadke, H. Yi, D. Novosel, and H. A. R. Volskis, “A preprocessing method for effective PMU placement studies,” in *Proc. 3rd Int. Conf. Electric Utility Deregulation and Restructuring and Power Technologies (DRPT 2008)*, pp. 2862–2867, Apr. 6–9, 2008.

[29] W. Yuill, A. Edwards, S. Chowdhury, S.P. Chowdhury, "Optimal PMU placement: A comprehensive literature review," *Power and Energy Society General Meeting, 2011 IEEE* , vol., no., pp.1-8, 24-29 July 2011

[30] T. H. Cormen, C. E. Leisserson, and R. L. Rivest, *Introduction to Algorithms*, MIT Press, Third Edition, 2009.

[31] F.J Marin, F.Garcia-Lagos, G. Joya, F. Sandoval, "Genetic algorithms for optimal placement of phasor measurement units in electrical networks," *Electronics Letters* , vol.39, no.19,pp.1403-1405,18Sept.2003

[32] B. Milosevic, M. Begovic, "Nondominated sorting genetic algorithm for optimal phasor measurement placement," *Power Systems, IEEE Transactions on* , vol.18, no.1, pp. 69- 75, Feb 2003

[33] J.Kennedy and R.C.Eberhart, "Particle Swam Optimization," Proceedings of the 1995 IEEE International Conference on Neural Networks, Perth, Australia,1995,pp.1942-1948.

[34] A, Sadu; R ,Kumar and R.G.,Kavasseri; , "Optimal placement of Phasor Measurement Units using Particle swarm Optimization," *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on* , pp.1708-1713, 9-11 Dec. 2009.

[35] L.N. de Castro, and F.J Von Zuben, "Learning and optimization using the clonal selection principle," in *IEEE Transaction on Evolutionary Computation*, Vol.6 (3), pp.239-251, June 2002.

[36] L.N. de Castro, and F.J Von Zuben, "Learning and optimization using the clonal selection principle," in *IEEE Transaction on Evolutionary Computation*, Vol.6 (3), pp.239-251, June 2002.

[37] A. Ahmadi, Y. Alinejad-Beromi, and M. Moradi, "Optimal PMU placement for power system observability using binary particle swarm optimization and considering measurement redundancy," Expert Syst. Appl., vol. 38, pp. 7263–7269, 2011.

[38] "The TOMLAB Optimization Environment", http://tomlab.biz/. Retrieved 2011-10-10.

[39] F. Aminifar, A. Khodaei, M. Fotuhi-Firuzabad and M. Shahidehpour, "Contingency-constrained PMU placement in power networks," in *IEEE Trans Power Syst,* pp. 516–523, 2010.

[40] S. Chakrabarti, E. Kyriakides, D.G. Eliades, "Placement of Synchronized Measurements for Power System Observability," *Power Delivery, IEEE Transactions on* , vol.24, no.1, pp.12-19, Jan. 2009

[41] D. Dua, S. Dambhare, R. K. Gajbhiye, and S. A. Soman, "Optimal multistage scheduling of PMU placement: An ILP approach," in *IEEE Trans. Power Del.*, vol. 23, no. 4, pp. 1812–1820, Oct. 2008.

[42] S. Chakrabarti and E. Kyriakides, "Optimal placement of phasor measurement units for power system observability," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 1433–1440, Aug. 2008.

[43] Jiangxia Zhong, K.L.Wong, “A Hierarchical Analysis of Phasor Measurement Unit Placement Optimization in Transmission Network”, 2011 Australasian Universities Power Engineering Conference (AUPEC), September 25-28, 2011, Brisbane, Australia.

[44] A. Jochen, R. Michael; R. Niedermeier, "Polynomial-time data reduction for dominating set", Journal of the ACM 51 (3): 363–384, 2004.

Appendix A: Program of Hybrid Approach based on Approximation Algorithm

```
#include <stdio.h>
#include <stdlib.h>


int main(int argc, char **argv) {

FILE *fileptr;

int X = 0;
int Y = 0;

char *filename = argv[1];

        if ((fileptr = fopen(filename, "r")) == NULL) {
                printf("File %s cannot be open\n", filename);
                system("pause");
                exit(1);
        } //end if
        else {
                printf("File %s has been open\n", filename);
                int i = 0;
                while (i < 2) {
                        i++;
                        if (i == 1) {
                                fscanf(fileptr, "%d\n", &X);
                                printf("X is %d.\n", X);
                        }//end if
                        if (i == 2) {
                                fscanf(fileptr, "%d\n", &Y);
                                printf("Y is %d.\n", Y);
                        }//end if
                }//while
        }//end else

        int matrix[X][Y];

        for (int p = 0; p < X; p++) {
                for (int q = 0; q < Y; q++) {
                        matrix[p][q] = 0;
                } //end for
        }//end for


        if ((fileptr = fopen(filename, "r")) == NULL) {
                printf("File %s cannot be open\n", filename);
                system("pause");
                exit(1);
        } //end if
        else {
                printf("File %s has been open\n", filename);
```

```
                int i = 0;
                int e, f;
                while (!feof(fileptr)) {
                        i++;
                        if (i > 2) {
                                fscanf(fileptr, "%d %d\n", &e, &f);
                                //printf("%d %d\n",e,f);
                                if(e!=f)
                                matrix[e-1][f-1] = 1;
                        }//end if
                }//while
        }//end else

        fclose(fileptr);

        for (int p = 0; p < X; p++) {
                for (int q = 0; q < Y; q++) {
                        printf("%d ", matrix[p][q]);
                } //end for
                printf("\n");
        }//end for

        for (int p = 0; p < X; p++) {
                for (int q = X-p; q < Y; q++) {
                        if(matrix[p][q]!=matrix[q][p])
        printf("errors in %d %d", p+1, q+1);
                } //end for
                printf("\n");
        }//end for

int neighbours[X];        //m1 --> neighbours
int status[X];            //m1 --> status

int bus_id[X];            //bus_number

int m2[X];                //in queue


        for (int p = 0; p < X; p++) {
   int n = 0;
                for (int q = 0; q < Y; q++) {
        if(matrix[p][q]==1){
          n++;
        }
                } //end for
                neighbours[p] = n;
                bus_id[p] = p+1;
                status[p]= 0;  //status = layer
                m2[p] = 0;
                //printf("%d %d %d\n", bus_id[p], neighbours[p], m2[p]);
        }//end for
```

```
         printf("---------------------------------\n");

   int temp1, temp2;

   for( int pass = 1; pass <= X - 1; pass++){
     for(int k=0; k < X - 1 ; k++){
      if (neighbours[k]<neighbours[k+1]){
         temp1 = bus_id[k];
         bus_id[k]= bus_id[k+1];
         bus_id[k+1]= temp1;

         temp2 = neighbours[k];
         neighbours[k]= neighbours[k+1];
         neighbours[k+1]= temp2;
        }
      }//inner for loop
      }//for


         for (int p = 0; p < X; p++) {
                  printf("%d %d\n", bus_id[p], neighbours[p]);
         }

   int num = 0;

/*
   //layer1
            printf("layer %d\nbus %d\n", 1, bus_id[0]);
            m2[0] = bus_id[0];
            status[bus_id[0]-1]= 1;
            num = 1;
*/

for (int t =0; t<X; t++)
{
   printf("------Traversing bus %d-----\n", t+1);
   //layer1
   printf("layer %d\nbus %d\n", 1, t+1);
            m2[0] = t+1;
            status[t]= 1;
            num = 1;

   //int layer = 1;  l ---layer
         for (int l = 2; l < X; l++) {
       //display layer
       printf("layer %d\n", l);
       //find father nodes
       for(int f = 0; f < X; f++)
       {
          if( status[f] == (l-1) )
```

```
               {

                 for ( int p=0;p<X;p++ )
                 {

                  if(matrix[f][p]==1)
                  {
                     if ( status[p] ==0 )
                     {
                        printf("bus %d ", p+1);
                        m2[num] = p+1;
                        num ++;
                        status[p]= l;
                     } //end if -- print bus id
                  } //end if -- find neighbours
                 } //end for -- find neighbours
              } //end if -- find father node
         } //end for -- find father node
         printf("\n");
         if (num == X) break;
    }//end for -- layer count

    for (int s=0; s<X; s++){
        status[s] = 0;
        m2[s] = 0;
    }

} //end for


/*

            for (int p = 0; p < X; p++) {
                        printf("%d ", m2[p]);
            }

*/
    printf("\n");

            system("pause");
            return 0;
}
```

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
using namespace std;

#define MAX 100

int link[MAX][MAX];
int links[MAX];
int premier[MAX];
int isinstall[MAX];
int cover[MAX];
int stat[MAX];
int n;
int rlt[MAX];
int out[5000][MAX];
FILE *fileptr;
int k;
time_t start;
time_t end;

int isOK()
{

        for (int i = 0; i < n; i++)
        if (cover[i]<1)
                return false;

        out[k][0] = rlt[0];
        for (int i = 1; i <= rlt[0]; i++)
                out[k][i] = rlt[i];
        k++;

//      for (int i = 0; i < n; i++)
//              printf("%d ",cover[i]);
//      printf("\n");
//      printf("----------------------------------\n");
        return true;
}

int sol()
{
        if (!isOK())
        {
                int tmp2 = 0;
                int label[MAX];
                int tmp3[MAX];
```

```
int tmp4 = 0;
memset(label,0,sizeof(label));
for (int i = 0; i < n; i++)
if (! isinstall[i])
{
        tmp3[i] = 0;
        if (cover[i]<1)
                tmp3[i]++;
        for (int j = 0; j < n; j++)
        if (link[i][j])
        if (cover[j]<1)
                tmp3[i]++;

        if (tmp3[i] == tmp2)
                tmp4++;

        if (tmp3[i] > tmp2)
        {
                tmp2 = tmp3[i];
                tmp4 = 0;

        }


}

for (int i = 0; i < n; i++)
if (! isinstall[i])
if (tmp3[i] == tmp2)
{
        //printf("label=%d\n",i+1);
        rlt[0]++;
        rlt[rlt[0]] = i;
        isinstall[i] = 1;
        cover[i]++;
        for (int j = 0; j < n; j++)
        if (link[i][j])
                cover[j]++;

        sol();


        {
                rlt[0]--;
                isinstall[i] = 0;
                cover[i]--;
                for (int j = 0; j < n; j++)
                if (link[i][j])
                        cover[j]--;
        }
}
```

```
	}
	else
	{
		//printf("----------------------------\n");
		return 0;
	}
}

int main(int argc, char **argv)
{
	time(&start);
	char *filename = argv[1];
	filename = "14bus_7.txt";
	memset(links,0,sizeof(links));
	if ((fileptr = fopen(filename, "r")) == NULL) {
		printf("File %s cannot be open\n", filename);
		system("pause");

	} //end if
	else
		printf("File %s has been open\n", filename);


	memset(cover,0,sizeof(cover));
	memset(isinstall,0,sizeof(isinstall));
	memset(rlt,0,sizeof(rlt));
	memset(out,0,sizeof(out));
	k = 0;

	fscanf(fileptr,"%d",&n);
	for (int i = 0; i < n; i++)
	{
		for (int j = 0; j < n; j++)
			fscanf(fileptr,"%d",&link[i][j]);
		fscanf(fileptr,"\n");
	}
	for (int i = 0 ; i<n; i++)
	{
		for (int j = 0; j < n; j++)
		if (link[i][j])
			links[i]++;
	}
	int m;
	int nn;
	int min = MAX;
	int maxlinks=0;
	fscanf(fileptr,"%d %d",&m,&nn);
	while (nn--)
	{
		memset(cover,0,sizeof(cover));
```

```
            memset(isinstall,0,sizeof(isinstall));
            memset(rlt,0,sizeof(rlt));
            memset(out,0,sizeof(out));
            k = 0;
            for (int i = 0; i < m; i++)
            {
                    int tmp1;
                    fscanf(fileptr,"%d",&tmp1);
                    premier[i] = tmp1;
                    isinstall[tmp1] = 1;
                    cover[tmp1]++;
                    for (int j = 0; j < n; j++)
                    if (link[tmp1][j])
                            cover[j]++;
            }

            sol();

            for (int i = 0; i < k; i++)
            if (out[i][0] < min)
                    min = out[i][0];
            for (int i = 0; i < k; i++)
            if (out [i][0] == min)
            {
                    int tmp4 = 0;
                    for (int j = 1; j <= out[i][0]; j++)
                    {
                            printf("%d ",out[i][j]+1);
                            tmp4 += links[out[i][j]];
                    }
                    if (tmp4>maxlinks) maxlinks = tmp4;
                    printf("   %d\n",tmp4);
            }
    }
    time(&end);
    //printf("%d solutions in total\n",k);
    printf("max links is %d\n",maxlinks);
    printf("time is %ds\n",end - start);
    system("pause");
    return 0;

}
```

Appendix B: Program of Hybrid Approach based on Global Search Algorithm

```
clear all, clc, close all;
tic
fid=fopen('14bus_7.txt','r');
s=fscanf(fid,'%c');
s=str2num(s);

%load structmax.mat
%load zeroinjection.mat

%s=structmax;
%zi=zeroinjection;

zi=[7 8];
zid=[];

% for i=1:length(zi)
%     t=find(zid==zi(i));
%     if isempty(t)==1
%         a=zi(i);    %find the ZI bus
%         zib=find(s(a,:)~=0);    %find all the connections on the ZI bus
%         for k=1:length(zib)
%             s(zib(k),:)=s(zib(k),:)+s(a,:); %reflash connection map
%             zid=[zid,zib];
%
%         end
%     else
%     end
% end
%
% nonzero=find(s);
% s(nonzero)=1;   %push all non-zero back to one
% diag(ones(1,14),0)
% s
c=0;
len=length(s);
ib=[];
is=[];
obs=zeros(1,len);
k=0;



for i=1:len
    findone=find(s(i,:));
    if length(findone)==1
        ib=[ib,findone];
        is=[is,i];
    end
    if length(findone)>=3
        k=k+1;
    end
end

LB=ceil((k+2)/3);

UB=floor(len/3);

maxb=1:len;
maxb=setdiff(maxb,is);
```

```
maxb=setdiff(maxb,ib);

n=LB;
found=0;
flagif=0;

while 1
    for t=1:1000    %random
%         tstart=tic;

        while 1
        ind=randperm(length(maxb));
        ind=ind(1:(n-length(ib)));
        if isempty(find(ind==7,1))
            break
        end
        end

        indx=[maxb(ind),ib];
        for ii=1:(n);
            obs=obs+s(indx(ii),:);
            obs(indx(ii))=1;
        end
        if isempty(find(obs==0,1))
            found=1;
            obs=zeros(1,len);
            break
%         elseif toc(tstart)>=5
%             telap=toc(tstart);
%             break
        else
            found=0;
            obs=zeros(1,len);
        end
    end

    if found
        if flagif~=1
            n=n-1;
            flagif=1;
            sol=indx

        else
            n=floor((n+LB)/2);
            sol=indx
        end

    else
        if flagif~=1
            n=ceil((UB+n)/2);
        else
            break
        end
    end
    %c=c+1 %debugging counter

end


n=length(sol);
```

```
cad=[];
csum=[];

posb=nchoosek(maxb,n-1);

for lol=1:length(posb)
    indxx=[posb(lol,:),ib];
    for iii=1:n
    obs=obs+s(indxx(iii),:);
    obs(indxx(iii))=obs(indxx(iii))+1;
    end
    if isempty(find(obs==0,1))
        cad=[cad;indxx];
        csum=[csum,sum(obs)];
        %obs %uncomment to see which buses can be observed
    end
    obs=zeros(1,len);
end

cad

indbes=find(csum==max(csum));


sort(cad(indbes,:))

toc
```

Appendix C: Publications

1. Jiangxia Zhong, K.L.Wong, “A Hierarchical Analysis of Phasor Measurement Unit Placement Optimization in Transmission Network”, 2011 Australasian Universities Power Engineering Conference (AUPEC), September 25-28, 2011, Brisbane, Australia.

A Hierarchical Analysis of Phasor Measurement Unit Placement Optimization in Transmission Network

Jiangxia Zhong
Department of Electrical and Computer Engineering
RMIT University
Melbourne, Australia
Jiangxia.zhong@gmail.com

K.L. Wong
Department of Electrical and Computer Engineering
RMIT University
Melbourne, Australia
alan.wong@rmit.edu.au

Abstract-- This paper presents a new phasor measurement unit (PMU) placement algorithm based on Hierarchical Analysis, a hybrid algorithm combining the breadth-first algorithm (BFA) and greedy algorithm. With the optimized number of PMU, the system is to maintain full network view during normal operation with and without conventional zero-injection effect. The IEEE standard 14-, 30-, 57- and the New England 39-bus test systems are investigated. Based on the comparison of the proposed method in this paper with other published methods, the proposed method has achieve good results with minimal PMU required and maximum system observability redundancy index (SORI).

***Index Terms*--Phasor Measurment Unit, Breadth-First Algorithm, Greedy Algorithm, Network Observability, Maximum Redundancy.**

I. INTRODUCTION

As the demand of electricity usage is increasing, the state estimation for transmission network is required in order to achieve higher reliability and stability of a power system [1]. Phasor Measurement Unit (PMU) is a device employed to detect the power failure by monitoring phase changes in the current and voltage waveform continuously. Integrated with the global positioning system (GPS) [2], the base station is able to receive the synchronous data from each PMU in real time. If a phase difference is detected, the location of malfunction circuits or transmission line can be immediately identified. In order to reduce the costs of system implementation and maintenance, the optimal PMU placement is applied by minimizing the number of PMUs while the network is fully functional and observable [3].

In recent years, there has been overwhelming interest in the problem of PMU optimization. A binary search algorithm, which was proposed by S. Chakrabarti and E. Kyriakides [4] was used to minimize the total number of PMUs required and to maximize the measurement redundancy at the power system buses. Complete observability of the system is ensured under normal operating conditions as well as under the outage of a single transmission line or a single PMU. F. Aminifar et al [5] formulated the base-case state, loss of measurement and line outage by applying integer linear programming. In this paper, the required number of PMUs in each scenario has been comprehensively studied and compared. M. Zhou [6] introduced the matrix reduction algorithm to effectively reduce the computational efforts and greedy algorithm had been used to solve the minimum number in PMU necessary for full observability. In [7], the authors introduced the Integer Linear Programming (ILP) framework to explain the zero injection constraints. In addition, the System Observability Redundancy Index (SORI) was also introduced to expand the bus coverage in the minimal solution. Several heuristic algorithms also have been utilized to minimize the PMU set for network fully observability. As discussed by A.Sadu, R.Kumar and R.G.Kavasseri [8], particle swarm optimization is able to solve the optimal placement of PMUs. The optimal result achieved the high performance by less of number of iterations which compared to the other heuristic algorithms such as original clonal algorithm (CLONALG) [9] and adaptive clonal algorithm (CLONALG) [10].

This paper presents a new method for optimal PMU placement. The novel algorithm is named Hierarchical Analysis and is developed based on the combination of breadth-first algorithm (BFA) [11] and greedy algorithm [6]. BFA is used to explore and list the connectivity between a root node and all the neighboring nodes graphically and intuitively. Greedy algorithm is applied to locate the buses with the most coverage until the network is fully observable.

II. PROPOSED PMU PLACEMENT METHOD

The new approach in optimal PMU placement, hierarchical analysis is developed using both breadth-first algorithm and greedy algorithm. Refer to the bus admittance matrix in [3], the IEEE standard 14-bus system [4] as shown in Fig.1. It can be organized by BFA into a tree topology, which is represented as a top to bottom hierarchical structure. Any of the highest incident buses can be used as the top node in the hierarchical analysis as illustrated in Fig. 2.

A database is established to represent the multilevel structure where all the interconnections and the degrees of connectivity are recorded as parameters. The sample of the database is presented in Table I.

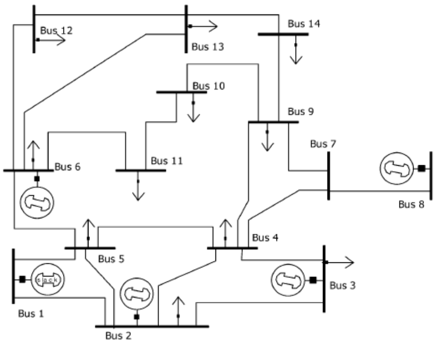


Fig.1. IEEE 14-bus test system [4].

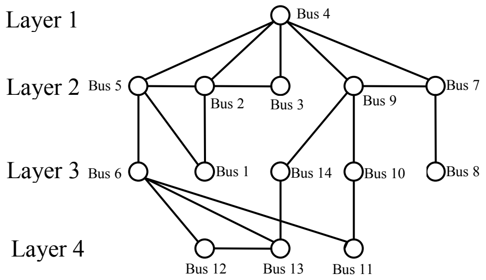


Fig.2. Layer structure for IEEE 14-bus system.

TABLE I
THE DATABASE FOR IEEE FOURTEEN-BUS STANDARD SYSTEM

	Bus	Degrees	Parent[a]	Brother[b]	Children[c]
Layer 1	4	6	NULL[d]	NULL	5,2,3,9,7
Layer 2	5	5	4	2	1,6
	2	5	4	3,5	1
	3	3	4	2	NULL
	7	4	4	9	8
	9	5	4	7	10,14
Layer 3	1	3	2,5	NULL	NULL
	6	5	5	NULL	11,12,13
	8	2	7	NULL	NULL
	14	3	9	NULL	13
	10	3	9	NULL	11
Layer 4	12	3	6	13	NULL
	13	4	6,14	12	NULL
	11	3	6,10	NULL	NULL

a: the parameter of parent indicates the bus is connected to upper layer.
b: the parameter of brother indicates the bus is connected to the same layer.
c: the parameter of children indicates the bus is connected to lower layer.
d: the parameter of NULL indicates no connection between two buses.

Greedy algorithm produces the best choice at each stage to achieve the overall optimum. In the case of a multilevel structural system, greedy algorithm can be interpreted as an algorithm searching for the highest coverage bus till the system is fully inspected. Once a bus has been selected, the interconnections will be eliminated. However, one execution loop by starting from highest incident node is not sufficient to acquire the best solution. On the other hand, if the greedy algorithm is applied for multiple times using every single node as starting points, the total execution time will be much extended. A solution has been proposed by the author utilizing the buses with highest degree in each layer as the initial bus when applying greedy algorithm. As a result, the hierarchical analysis is able to accomplish the minimum PMU placement with less execution time.

As discussed in paper [7], system observability redundancy index (SORI) standards for the total sum of bus coverage for all the implemented buses in an active system. A higher SORI value achieved indicates monitoring system is more reliable. Therefore, the final optimal result will be selected according to the SORI in maximum.

The procedure of the Hierarchical Analysis for optimizing PMU placement can be clearly defined in the flow chart shown in Fig. 3.

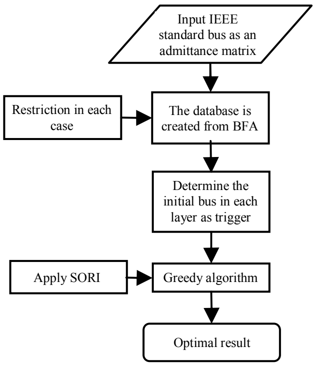


Fig.3. Flowchart of PMU optimal placement by combined algorithm.

III. IMPLEMENTATION AND CASE STUDY

A. Normal Operation without Zero-Injection Effect

Firstly, the buses with only one connection are to be located and PMUs are required to be placed at their adjacent bus. In the 14-bus test system, PMU is placed at bus 7 to cover the single connection bus 8. Next, the initial buses in highest degree will be selected as the highest incident buses which refer as bus 4, 5 or 2, 6 and 13 from top to bottom layer respectively. The complete steps with the bus coverage involved greedy algorithm are shown in Table II. The optional solutions of optimal PMU placement are represented as bus 2, 6, 7 and 9 or bus 2, 7, 10 and 13. According to the definition of SORI, the result with higher redundancy index will be the optimal solution, which is the placement at bus 2, 6, 7 and 9.

TABLE II
EXAMPLE FOR GREEDY ALGORITHM IN FOURTEEN-BUS SYSTEM
IN LAYER TWO AND LAYER FOUR

Initial bus:	**layer 2: bus 2**	**OR bus 5**		**layer 4: bus 13**
Bus Coverage:	1,2,3,5	1,2,5,6		6,11,12,13,14
2nd placement:	**bus 6**	**bus 6 OR bus 13**		**bus 2**
Bus Coverage:	6,11,12,13	11,12,13	12,13,14	1,2,3,5
3rd placement:	**bus 9**	**bus 9**	**Bus10 OR bus11**	**bus 10**
Bus Coverage:	10,14	10,14	10,11	10
4th placement:		**bus 3**	**bus 3**	
Bus Coverage:		3	3	

B. Normal Operation with Zero-Injection Effect

Zero-injection bus can be generalized as a bus without generator and load connection. In reference [5], the zero-injection effect has been categorized as following:

> 1) When buses, which are incident to an observable zero-injection bus, are all observable except one, the unobservable bus will also be identified as observable by applying the Kirchhoff's current law (KCL) at zero-injection bus.
>
> 2) When buses incident to an unobservable zero-injection bus are all observable, the zero-injection bus will also be identified as observable by applying the KCL at zero-injection bus.

In the Hierarchical Algorithm, once a zero-injection bus is detected inside the database, all of its neighboring buses are considered as virtually connected while the system hierarchy remains unchanged. For example, in IEEE 14-bus system, bus 7 has no direct generator or load connection therefore it is classified as a zero-injection bus and bus 8 will be treated as directly connected to bus 4 and bus 9 when applying the algorithm.

TABLE III
EXAMPLE FOR GREEDY ALGORITHM IN FOURTEEN-BUS TEST SYSTEM
IN LAYER TWO

Initial bus:	**layer 2: bus 9**
Bus Coverage:	4,7,8,9,10,14
2nd placement:	**bus 6**
Bus Coverage:	5,6,11,12,13
3rd placement:	**bus 2**
Bus Coverage:	1,2,3

After regrouping the data, the optimization process is completed following the similar procedure from case A. Table-III displays the optimal result based on the initial bus in layer 2.

IV. SIMULATION RESULTS AND COMPARISON

The entire system functional test is simulated by running MatLab/C++ program on a Windows XP operating system based computer, which has 3.5 GB RAM and core 2 duo 3.16 GHz processor. Table IV represents the optimal PMU placement under consideration with or without zero-injection effect. The optimal solution is bus 2, 6 and 9.

TABLE IV
OPTIMAL PMU PLACEMENT IN CASE A AND CASE B

	14 BUS	**30 BUS**	**39 BUS**	**57 BUS**
CASE A	2,6,7,9	2,4,6,9, 10,12,15, 19,25,27	2,6,9,10,13, 14,17,19,20, 22, 23,25, 29	1,6,9,12,15,19, 22,25,26,29,32, 36,38,41,47, 50, 53,57
CASE B	2,6,9	2,4,10,12, 15,19,27	6,8,10,16, 20,23, 25,29	1,6,13,19,25,29, 32,38,51,54,56

Table V summarizes the comparison results of the proposed method and existing method in literature [4] based on IEEE standard 14-, 30-, 57- and the New England 39-bus test systems. Evidently, the results in the proposed method achieved the same value as published method. Therefore, hierarchical algorithm is adequate to solve the problem of optimal PMU placement.

TABLE V
MINIMAL NUMBER OF PMU IN COMPARISON

		14 BUS	**30 BUS**	**39 BUS**	**57 BUS**
CASE A:	Presented in[4]	4	10	13	N/A
	Proposed Method	4	10	13	17
CASE B:	Presented in[4]	3	7	8	N/A
	Proposed Method	3	7	8	11

N/A: not available

Table VI shows the outcome of system observability redundancy index based on case A and B. When comparing the index value between reference [4] and proposed method, the performance of SORI in 30-bus and 39-bus test system has been successfully improved.

TABLE VI
SYSTEM OBSERVABILITY REDUNDANCY INDEX IN COMPARISON

		14 BUS	**30 BUS**	**39 BUS**	**57 BUS**
CASE A:	Presented in[4]	19	50	52	N/A
	Proposed Method	19	52	52	72
CASE B:	Presented in[4]	16	50	53	N/A
	Proposed Method	16	57	54	64

N/A: not available

V. CONCLUSION

In this paper, the problem of optimal PMU placement in power network observability was investigated. A new PMU placement optimization method was introduced base on breadth-first algorithm and greedy algorithm. By employing the MatLab and C++, the simulation results on IEEE standard 14-, 30-, 57- and the New England 39-bus test systems were presented to demonstrate the effectiveness of hierarchical algorithm. Large-scale network will be implemented in the future work.

ACKNOWLEDGMENT

The author is grateful to Mr. Chongzheng Fan and Mr. Zhe Zhang for the technical support.

REFERENCES

[1] A. G. Phadke, J. S. Thorp, and K. J. Karimi, "State Estimation with Phasor Measurements", in *IEEE Transactions on Power Systems*, Vol. 1, No. 1, pp. 233-241, February 1986.

[2] A. G. Phadke, "Synchronized phasor measurements in power systems," in *IEEE Computer Applications in Power*, Vol. 6, Issue 2, pp. 10-15, April 1993.

[3] B. Xu and A. Abur, "Optimal placement of phasor measurement units for state estimation," *Final Project Report, PSERC*, Oct. 2005.

[4] S. Chakrabarti and E. Kyriakides, "Optimal placement of phasor measurement units for power system observability," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 1433–1440, Aug. 2008.

[5] F. Aminifar, A. Khodaei, M. Fotuhi-Firuzabad and M. Shahidehpour, "Contingency-constrained PMU placement in power networks," in *IEEE Trans Power Syst,* pp. 516–523, 2010.

[6] M. Zhou, V. A. Centeno, A. G. Phadke, H. Yi, D. Novosel, and H. A. R. Volskis, "A preprocessing method for effective PMU placement studies," in *Proc. 3rd Int. Conf. Electric Utility Deregulation and Restructuring and Power Technologies (DRPT 2008)*, pp. 2862–2867, Apr. 6–9, 2008.

[7] D. Dua, S. Dambhare, R. K. Gajbhiye, and S. A. Soman, "Optimal multistage scheduling of PMU placement: An ILP approach," in *IEEE Trans. Power Del.*, vol. 23, no. 4, pp. 1812–1820, Oct. 2008.

[8] A, Sadu; R ,Kumar and R.G.,Kavasseri; , "Optimal placement of Phasor Measurement Units using Particle swarm Optimization," *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on* , pp.1708-1713, 9-11 Dec. 2009.

[9] L.N. de Castro, and F.J Von Zuben, "Learning and optimization using the clonal selection principle," in *IEEE Transaction on Evolutionary Computation*, Vol.6 (3), pp.239-251, June 2002.

[10] X. Bian and J. Qiu, "Adaptive clonal algorithm and its application for optimal PMU placement," in Proc. IEEE Int. Conf. Communication&Circuits,Syst., 2006, vol. 3, pp. 2102–2106.

[11] T. H. Cormen, C. E. Leisserson, and R. L. Rivest, *Introduction to Algorithms*, MIT Press, Third Edition, 2009.