PREVENTION OF CASCADING OUTAGES USING SPARSE WIDE AREA SYNCHROPHASOR MEASUREMENTS

A DISSERTATION

Submitted in Partial Fulfillment of The

Requirement for the Award of the Degree

of

MASTER OF TECHNOLOGY

in

ELECTRICAL ENGINEERING

(with specialization in Power Systems)

By

Rajendra Prasad Sharma
(17529011)

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE

ROORKEE – 247667 (INDIA)

May, 2019

CANDIDATE DECLARATION

I hereby certify that the work being presented in thesis report, entitled **"Prevention of Cascading Outages using Sparse Wide Area Synchrophasor Measurements"** in partial fulfillment of the requirement for the award of the degree of **Masters Of Technology** with specialization in **"Power System Engineering",** submitted in **Department of Electrical Engineering, Indian Institute of Technology, Roorkee** is an authentic record of my own work carried out during the period from May 2018 to May 2019 under the supervision of **Dr. Bhavesh Bhalja, Dr. Premalata Jena, Professor, Department of Electrical Engineering, Indian Institute of Technology, Roorkee, India**.

The matter embodied in the dissertation report has not been submitted by me for the award of any other degree or diploma.

Dated: 21/05/2019

Place: ROORKEE

(RAJENDRA PRASAD SHARMA)

CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Dr. Bhavesh Bhalja
Associate Professor
Dept. of Electrical Engineering
Indian Institute of Technology
Roorkee- 247667

Dr. Premalata Jena
Assistant Professor
Dept. of Electrical Engineering
Indian Institute of Technology
Roorkee- 247667

ACKNOWLEDGEMENT

I would like to express my sincere and deep sense of gratitude and indebtedness to my supervisors **Dr. Bhavesh Bhalja, Dr. Premalata Jena, Professor, Department of Electrical Engineering, Indian Institute of Technology Roorkee** for guiding me to undertake this dissertation report work as well as providing me all the necessary guidance and support throughout this work. They have displayed unique tolerance and understanding at every step of progress, without which this work would not have been in the present shape.

I would also be thankful to all staff of **Department of Electrical Engineering** for their constant support at and all my friends, for their help and encouragement at the need of the hour.

Dated: 21/05/2019

RAJENDRA PRASAD SHARMA

17529011

M. Tech 2nd year

EED (PSE), IITR

Abstract

Power System cascading outages can result in major system loss and the other indirect losses. These outages are unpredictable in nature and hence their modeling is a fairly complicated task. The modeling of these outages is based on statistical inferences. These outages can start from very simple outages and can grow into a big outage if some chaotic parameter is touched. They can also start from inter tie oscillations and can grow into big outages. The health of the power system is represented by frequency and whenever there are some oscillations which are small in frequency then that means some dynamic is there. If these don’t get damped out, then there is a chance that they will grow and that will definitely cause system wide cascade if some critical equipment failure occurs. In this report a small part of a big problem of preventing cascading outage is discussed. Under this, the fault location needs to be found for real time topology update of the system network. This ensures that if some critical state is present, it can be sensed with the help of generator coherency information. The method uses the electromechanical wave velocity concept and the concept of multilateration to determine the fault location. The system on which this approach is tested is IEEE- 9 bus system. The method to detect the fault location does require the use of PMU and IED which can be used for measurement of signals. With very less amount of data and less number of PMUs, but sufficient enough to make the overall power system observable, fault location can be calculated. The report only focuses on the fault locations at the buses, but in actual system fault could also occur between the buses and the method can be modified to include line faults. Also after determining the location of fault, system data can be modified with new calculations. This modified data can be used further to estimate the state of the system. The problem of state estimation has not been dealt with in this work. It is assumed already that the system is in vulnerable state or in emergency state and different areas can lose synchronism with each other which will result in different generators falling apart. In order to save the system, different buses and generators which are coherent need to be grouped. This is done by studying the coherency information of the system and finally the system is studied for optimum isolation under such condition (optimum in terms of minimum power flow disruptions).

Table of Contents

List of Figures-

List of Tables-

1. CHAPTER

1.1 INTRODUCTION

Cascading-is a term used in literature to show the occurrence of objects or some phenomena or events one after the other. A cascading outage is a sequence of events in which initial disturbance, or set of disturbances, triggers a sequence of one or more dependent component outages.

Although outages generally occur in the system due to one or other reason such as equipment failure and malfunctions of protection equipment but most of them do not develop into a cascade. It is only when some important or vulnerable component in the power system gets into trouble, cascading chances become high and even then one can’t say with surety that it will become cascade. But the reverse seems and found to be true by observing actual events in the sense that, whenever the cascade event occurred, it did due to some important component getting out of service for one reason or the other. Usually the first initial event is found to be random, but the subsequent events can be connected by a causal link and can be explained. Researches first find some vulnerable elements that will develop into some cascade of events [25], [26]. Therefore, cascading outages are influenced by the details of the system state, such as components out for maintenance and the patterns of power transfers, and the automatic and manual system procedures. Subsequent events can include transmission line outages, frequency oscillations leading to instability, overloads and malfunctions of protective relays.

The initiating events for a cascading outage can include a wide variety of disturbances such as high winds, overloads, lightning, natural disasters, contact between conductors and vegetation or human error [26]. The subsequent events are dependent events and they form a chain which seems to be numerous before actually they occur. The Risk is considered to be a combination of probability and the cost it causes if it occurs. Hence characterization of all the uncertainties and their cost is studied under risk assessment of power system.

To prevent cascading outages controlled islanding is done which determines best possible cut in the network. The real time information is obtained to determine if any fault is occurring in the network. The fault location is determined and topology information is updated to include the fault. Then coherency detection helps in finding the optimal cut to determine the lines that needs to be cut to separate the differently coherent areas and hence prevent the system from losing towards instability. In this report, the problem of determining fault location is dealt with. Hence, this problem can be divided into three parts-

1) Determination of location of fault,
2) Determine possible islands combinations, these can be more than one,
3) Find the optimal cut that can reduce power disruption between different areas.

The report is organized to explain each part one by one. A brief introduction of continuum modelling of electromechanical waves is presented in next section. The explanation of all the methods is given in chapter 3 and finally the test results are given in chapter 4.

1.2 Electromechanical Waves-

Since any cause and effect are separated by some time gap which is necessary (considering the causal behavior of the system) and important according to modern theories of physics. The affects which are produced by some cause in material world travel with some speed which depends on the properties of the material. The travelling of this disturbance is called wave propagation. Some waves are transient in nature. However, if some energy storage elements are presents together in the system then it causes harmonic travel of energy between different point of energy storages. These waves are sustaining and move in the system. In power system as the grids got interconnected sufficient amount of generation and load gets distributed throughout the system. It now seems as though the inertia of generators and the sinks of electricity are distributed throughout the system. Hence, in terms of parameters, rotor angle δ and loads P_L are distributed in space. When these energy interchanging quantities are distributed in space, then surely the disturbance travels in the form of waves. This was recognized by electrical engineers earlier and they even thought that the waves of disturbance travel with same speed but it has been possible to measure them now and verify the prevalence of speed all over the network with the help of wide area measurement systems.

Model for Electromechanical waves-

Although this model assumes the continuity to be embedded in alternators and load positions which is far from the reality [4], it does give good estimate of the speed of the disturbances that has been measured at different buses in the power system. It makes inertia, damping coefficient, mechanical power and impedance as dependent on the spatial coordinates as shown below-

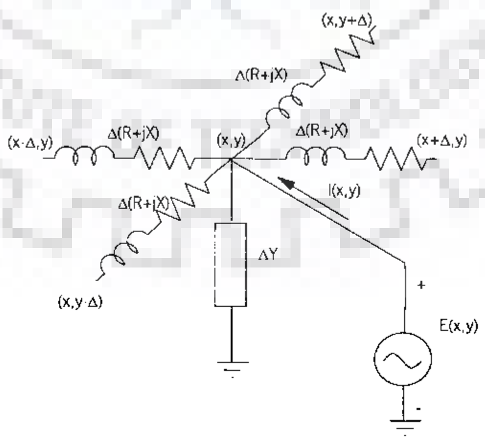


Fig. 1. 1 Model for explaining electromechanical waves

This modeling gives

$$I(x,y) = -\frac{\Delta^2}{Z}\left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2}\right) + (\Delta)YV \quad (1.1)$$

, which gives Pe = Re(VI*) and substituting this equation in dynamic equation of alternator gives,

$$\frac{2h}{\omega}\frac{\partial^2 \delta}{\partial t^2} + \omega d\frac{\partial \delta}{\partial t} = P - Pe \quad (1.2)$$

$$\frac{2h}{\omega}\frac{\partial^2 \delta}{\partial t^2} + \omega d\frac{\partial \delta}{\partial t} - \frac{V^2}{Z}\left(\sin(\theta)\left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2}\right) - \cos(\theta)\left(\left(\frac{\partial V}{\partial x}\right)^2 + \left(\frac{\partial V}{\partial y}\right)^2\right)\right) = p - GV^2 \quad (1.3)$$

This equation, although does not model the entire disturbance phenomena for discrete system, like our power system generally is, but it gives the value of the wave velocity as the coefficient of the double derivative of V with respect to spatial coordinates x and y. This velocity is given in equation (1.5)

$$\frac{\partial^2 \delta}{\partial t^2} + \mu\frac{\partial \delta}{\partial t} - (v^2\nabla^2\delta - u^2(\nabla\delta)^2) = P \quad (1.4)$$

The value of v, described in (1.4) is given in (1.5).

$$v^2 = \frac{\omega V^2 \sin(\theta)}{2h|z|} \quad (1.5)$$

2. LITERATURE REVIEW-

Urban Rudez, Rafael Mihalic,[1] uses electromechanical wave propagation to determine the fault location. The author proposes the use of critical points on the elctromechanical angle wave and gives a formula determined by simulation to calculate the time of arrival of disturbance wave at a particular bus. Thus this method makes the time of arrival determination independent of amplitude and frequency. Also the effect of inertia and line reactance on electromechanical wave propagation is examined. To produce the simulation results homogenous system is assumed. The author gives reason of why approach of constant threshold is wrong and it can give wrong results due to noise measurements. The proposed critical points on the elctromechanical wave are maximum point, maximum speed point and bifurcation point on the wave. Time of arrival of an electromechanical wave shouldn't depend on its own waveshape but on the waveshapes of other localized generator form where the wave is coming. But in constant threshold approach, one measures the time of arrival from the intersection of constant threshold with its own wave shape which make arrival time dependent on the waveshape[2].

J.S.Thorp, C.E. Seyler, A.G.Phadke, [5] developed the model of electromechanical waves using the concept of continuum systems. Although a bit unrealistic, this model idealizes the discrete nature of the power system and makes it continuous in order to get explicit insight into the phenomena. The power system loads are assumed to be continuous with zero spacing between them. The line impedance is assumed to be distributed all throughout the network and both unidimensional and two- dimensional distribution cases are considered for developing the differential equations representing the phenomena. The equation closely resembles the equation of wave when some reductions are performed and it confirms the fact that disturbance travels with some velocity like a wave along the electrical network spanned by the power system. The quantitative result of speed is obtained through the modelling. Author further investigates the propagation characteristics of wave, equilibrium solutions of the wave are calculated using analytical methods and stability of resulting equations. It also provides asserts to the observed fact that the waves could grow in certain directions, depending upon the rotor position in steady state with respect to their stator(stiff stator rotor coupling causes resistance to the propagation of the wave) on the network and other parameters

Manu Parashar, J.S. Thorp,[3] developed the continuum model for real power systems. Since the power system is having discreteness related to the presence of generation and load as well as inertia and impedance of transmission lines, it is highly irregular and anisotropic (system whose parameters are not regular with respect to geometry). Previous models are based on the assumption that power system is a continuous sheet of impedance distributed all over the world with electrical and mechanical inertia along with loads, but it doesn't really model the actual system. In this paper the author used a continuum model which on the large seems to be continuous but has finite discreteness embedded in it. The differential equations thus obtained contains spatial real information in its coefficients. A pair of non- linear partial differential equation is obtained which can be numerically analyzed. The first PDE is the continuum equivalent of the load flow equations of the power system and is a boundary value problem. The second equation is the continuum equivalent of the swing equations of the power system. The parameters of these equations are functions of spatial coordinates and the network topology is embedded in them. The model also shows that there can be phenomena just like an electromagnetic wave can have. Dispersion can be quantitatively modelled in these continuum model equations and it reveals the fact that the disturbances propagate in the phase angles and with constant finite velocity which is much less than the speed of light.

H. Song, M. Kezunovic, [18] used the vulnerability index and margin index to evaluate the vulnerability of the system parts as well as the whole system during the operation of the system. It identifies the vulnerable parts of the system using the topology processing and operation indexing method. In the paper power flow is used to evaluate the vulnerability and security of the system, identifies the vulnerable part, find the transmission line and bus voltage problems and predicts the possible successive events.

L. Mili, Q. Qiu, A. Phadke, [19] used another vulnerability based analysis which determines the condition of power system based on short circuit calculations and reserves of power being low due to fault or successive outages caused in the system. The cause of failure is taken to be the removal or lessening of reserves of power due to successive outages leading to voltage collapse. The algorithm aims at finding weak links in the power network. In this paper, the author proposes a methodology for a steady-state-risk assessment of multiple contingencies in large- scale power systems. The author also presents the statistical method for estimating the probabilities of hidden failures from historical data. The vulnerable link or line is described as the one whose disconnection can cause a large loss in load with highest conditional probability.

K. Sun, D. Zheng, and Q. Lu, [23] proposed a method to determine splitting points in the system to avoid any load generation imbalance in the subsystems resulting after islanding. The time based layer structure of problem solving is introduced in this work. Author focussed on the use of ordered binary decision diagrams(OBDD) for the splitting of 3-phase power systems. The author first reduced the graph of power system by removing irrelevant nodes and edges, merging of two nodes and removal of redundant edges in a cut-set. In phase-2 the balanced partition(BP) problem is solved on the reduced graph to find the partition, satisfying in addition, the PBC and SSC constraints. If no solution is found in phase 2, phase 1 is executed again by relaxing some constraints and changing parameters to change the reduced network. In the third phase the author checks the power flow results and find the proper Splitting strategies. Then a threshold limit is chosen for power flow in transmission lines and only those SS are considered feasible for which the power flow in transmission line in every island is within permissible or chosen threshold limits.

X. Wang, B. Qian, I. Davidson, [14] explained how traditional spectral clustering can be used to satisfy some constraints in order to cluster graph according to some properties different from geometrical patterns. The use of constrained spectral clustering is provided on various elementary example as well as on big applications such as image segmentation. The method is presented in two forms one weighted and other direct method. Weighted method weights the lagrangian according to the value of the found eigenvectors and then applies clustering on it. Author also presented the geometrical interpretation example of this method. The mathematical proof is included along with applications.

A Peiravi, R Ildarabadi [24] proposed multilevel kernel k- means method to perform intentional islanding and compared it with the spectral method developed earlier. The proposed method is fast and the time saved increases as the system size increases. The author proposes three phases of the method in which first is graph aggregation, then spectral partition is performed then merging or retrieval algorithm is applied on the graphs.

G.N. Ramaswamy, G.C. Verghese, L. Rouco, C. Vialas, C.L. DeMarco, [27] In this paper the author made the aggregate model of system base on inter area modes. The concept of synchrony is explored and approximate synchrony is found using clustering algorithm. Using for illustration a 23-generator power system model with 325 state variables, the paper demonstrates the effectiveness of a synchrony-based approach to decomposing the eigenvector analysis of the electromechanical modes, separating the computation of inter-area and intra-area modes in the style of multi-area Selective Modal Analysis.

N.Senroy, [28] In this paper the author applied Huang's empirical mode decomposition in order to extract dominant oscillatory modes from inter area oscillations from swing cuves or instantaneous phase angle curves. Hilbert transform on the modes extracted provides their instantaneous phases. The phase angles are used here for the analysis of inter and intra area modes. The analysis presented in the paper points to the fact that generator coherency can be extracted from insstantaneous phase difference between dominant modes of oscillations. The analysis of wide-area measurements demonstrate that it is possible to extract coherency between different areas, using distributed frequency measurements.

H. A. Alsafih, R. Dunn, [29] In this paper the author presented a method which uses a hierarchical clustering technique to classify the synchronous generators in power system into a number of coherent groups irrespective of the number of synchronous generators. The author also discusses the effect of type of disturbance on the clustering. Response of generators motion due to some disturbance is recorded and used to generate coherency information. The indexes which are used to evaluate the degree of coherency between any pair of generators are utilized in this method. Thereafter, the clustering algorithm is used to cluster these coherent generators into coherent groups.

K. L. Lo, Z. Z. Qi and D. Xiao, [30] The author presented a method of identifying the coherency of a group of generators using spectrum analysis technique. First, the rotor angles of generators in the early part of the transients are predicted by the Taylor series expansion of the power system model. The values obtained are taken as sample data for the spectrum analysis with an FFT algorithm. The author uses the coherency criteria from the spectrum found and the coherent groups of generators is found through the spectra.

K. K. Anaparthi, B. Chaudhuri, N. F. Thornhill and B. C. Pal, [32] The author used a technique of spectral analysis to identify coherent generators in large interconnected power system using measurements of generator speed and bus angle data. Based on the application of principal component analysis (PCA) to measurements obtained from simulation studies that represent examples of inter area events this method is used separately on bus angle and speed data. The approach of PCA was able to highlight clusters of generators showing common features when compared with the conventional modal analysis.

M. Jonsson, M. Begovic, J. Daalder, [31] Author proposed a method of using generator speed measurements combined with Fourier analysis technique. The author test the spectrum analysis method on three test cases where it is compared to conventional methods based on the generator speed, modal analysis, and phasor angle measurement.

H. You, V. Vittal, and X. Wang,[20] In this paper author used the analytical basis for an application of slow coherency theory to the design of an islanding scheme, which is employed as an important part of a corrective control strategy to deal with large disturbances. Various networking conditions and different types of loading are used to test this method. The results indicate that the slow coherency based grouping is almost insensitive to locations and severity of the initial faults. However, because of the loosely coherent generators and physical constraints the islands formed change slightly based on location and severity of the disturbance, and loading conditions. The description of the procedure to determine the groups using the slow coherency is also given in the paper. The verification of the islanding scheme is proven with simulations on large bus systems.

C Juarez, A. R. Messina, R Castellanos and G Espinosa-Pérez, [22] In this paper, an online hierarchical clustering method based on pattern recognition techniques is proposed for the automated clustering of system motion trajectories. Using the concept of minimum average distance between machine oscillations exhibiting a common behavior, a hierarchical clustered structure of the system that can be used for online determination of multi machine dynamic equivalents is suggested. This method accounts for complex inter machine oscillations and is suitable for a wide range of problems such as wide-area stability analysis and online dynamic security assessment and control. The method is used on Mexican power system. The clustering procedure is applied to identify the coherent motion of system machines following critical contingencies. The temporal modal behavior is separated using clustering technique. The method presented correctly identifies system dynamic behavior and hence can be used for grouping power system into different buses. The method is currently being extended to perform model reduction, directly in time-domain using a multimachine representation.

Ahad Esmaeilean, and M. Kezunovic,[4] used the synchrophasor measurements and concept of electromechanical wave to determine the fault locations. By calculating the time of arrival of electro mechanical wave to reach at a particular bus locations and taking the topology of the network into account, the method is proposed to detect the faulty line. The system is provided with optimal PMU locations in the power system. This paper proposes to use neural network for time of arrival detection and simulation is done in MATLAB on 118 bus system. The author also used the statistical techniques to remove bad data measurements. In another paper author proposed the optimization problem in which constrained spectral clustering is employed to optimize the amount of power disruption (minimize) and determining such cuts which isolates different areas of power systems while having minimum disruption in power.

Ahad Esmaeilean, and M. Kezunovic, [15] used the spectral clustering algorithm to determine the preliminary clusters using dynamic stability considerations. The method uses stiffness coefficients as the weights of the graph connections. The method divides the system first into two islands and then uses recursive bisection to separate each island into further two groups until the required number of groups are found. The use of dynamic stability constants helps in deciding the value of preliminary clusters which can be used to form constraint matrix for its further application in constrained spectral k- embedded clustering to find optimal solutions.

Ahad Esmaeilean, and M. Kezunovic[7] used synchrophasor measurements to utilize the coherency of the generators in the power system to find the cuts which will lead to minimum disruption in power.   A method to predict cascade event outage at early stage and mitigate it with proper control strategy is developed. In the first step, methodology employs sparsely located phasor measurement units to detect disturbances using electromechanical oscillation propagation phenomena. The obtained information is used to update system topology and power flow. Next, a constrained spectral k-embedded clustering method is defined to determine possible cascade event scenarios.

3. Methodology

3.1 Part 1- Determination of fault location-

3.1.1 Fault location-

The topology of electrical network at which the fault occurs is as shown in the fig. 1. The shown network is standard IEEE 9 bus system. All the buses are assumed to be either generator buses or load buses. No ZIB is considered in order to make PMU placement algorithm simple. Also, the line or links connecting the buses are assumed to be purely without bypass losses i.e., their shunt conductance is assumed to be zero. The buses can undergo any type of fault out of three phase, single line to ground or faults involving double lines.

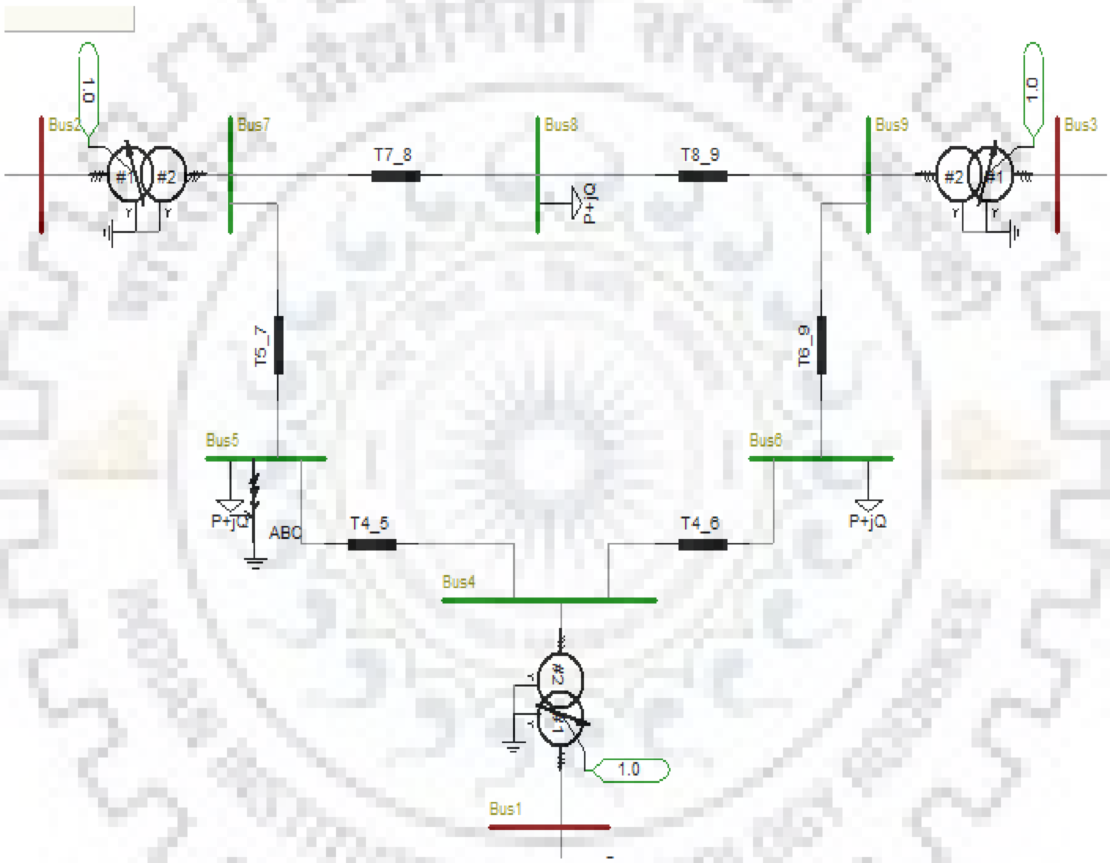


Fig. 3. 1 IEEE 09 bus test system with fault at bus 5

Now, if a fault occurs at any bus k, then there may be possibility that lines connecting bus k(k = 5) to the buses originally connected may get overloaded. The lines if overloaded will be tripped by the connected relay signals which are seeing the fault area as their nearest zones. Due to this some other lines which are originally carrying the large amount of current may get overloaded, which will cause ultimately another tripping if the issue of fault is not taken care of by now. This can continue going on involving not only lines but other electrical equipment such as transformer, generator (in the way that it might lose synchronism and hence need to be isolated first) etc. So our methodology is applied in the interval that begins after several outages and before the next outage can occur in the system. Now we can see that in order to separate the bus from the rest of the system, one can't just disconnect the bus from the system by tripping all the lines connected to the bus. To remove the bus and prevent outages from cascading into the system, proper islanding of different coherent areas of the power system can be a solution.

Now, at first the location of PMUs can be selected such that the power system is completely observable and we have to apply minimum number of PMUs, by solving a simple optimization problem which is related to the Integer Linear Programming.

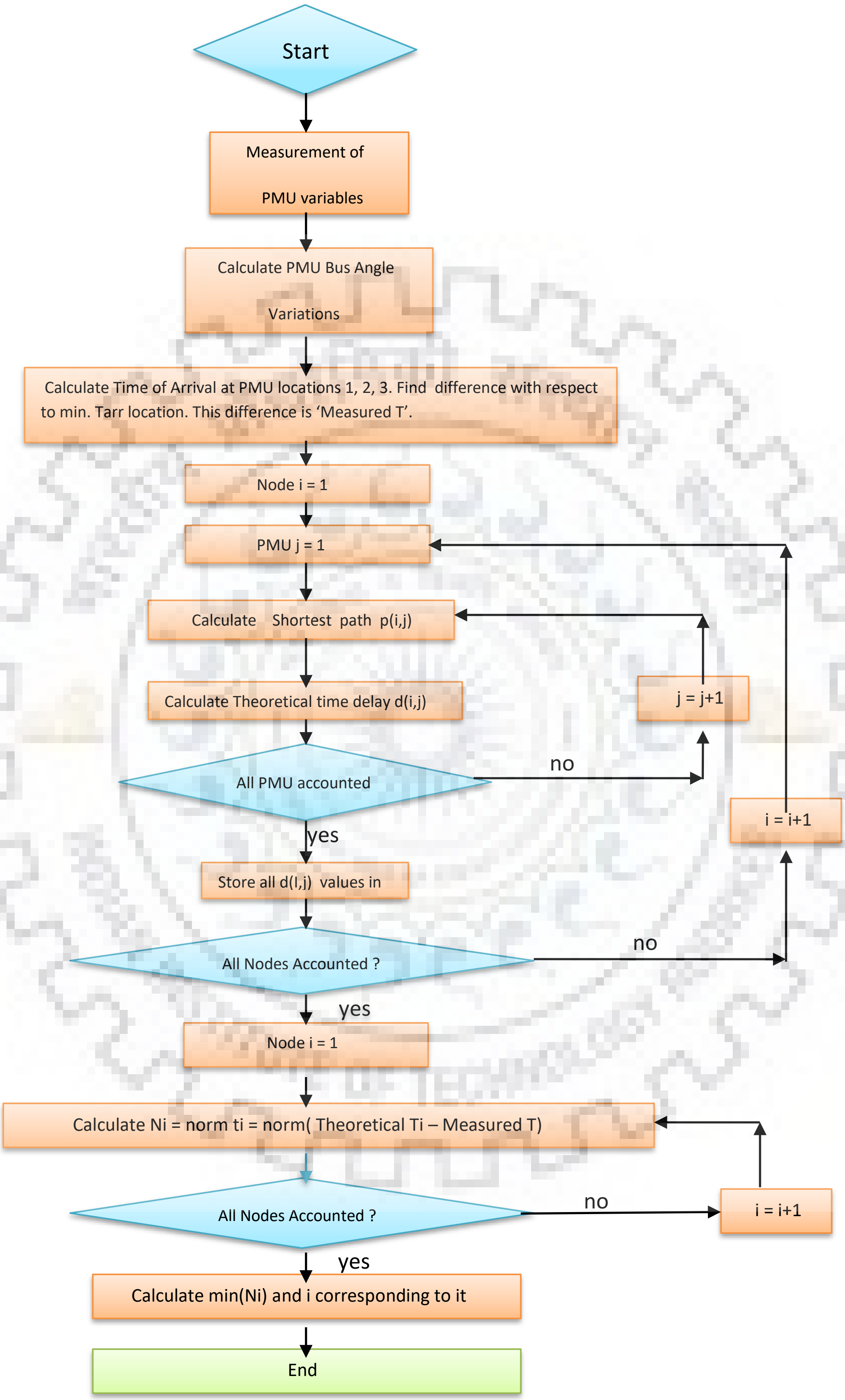
After the PMU locations are known one can determine the steady state condition of a power system. Steady state algebraic equations can be written in order to find the different variables in the power system. Now if a fault occurs at a bus no. k, it causes electromechanical disturbance in the power system which propagates in the system with some speed. Hence if the fault bus voltage, current or load angle changes according to a function $v_k(t)$, $i_k(t)$, $\delta_k(t)$ then one can find the similar changes at some other bus m superimposed over their steady state values at some later time. Hence all signals of changes look similar but seems shifted in time with respect to each other.

The time delay between different buses can be calculated using electromechanical wave velocity but it requires the determination of distance between bus k and bus m. But there can be many paths in the power system to reach bus m by starting at bus k. Hence, the simplest assumption that can be made here is that electromechanical disturbance wave travels in the power system along the shortest path that is available and the time the first disturbance in frequency or phase angle analogous to some other bus is found, it will be due to electromechanical wave travelling along this shortest path. This can be used to calculate the time delay between different buses in power system. Also, measurement of the bus signal information of the observable buses in the system is available through PMUs placed at bus locations. It is assumed that there is no limitations to channel availability in the PMUs and one can find as many channels as are required according to the application. Usually, we want at least the bus voltage and the line currents of all the lines connected to bus k, because in such way more measurements are providing more observability into the system.

This time delay calculated form the velocity is the theoretical time delay for a disturbance to travel in the circuit from bus k to bus m in the network. Also the measurements can provide the information about the time at which the fault arrives at the bus m. Note, here is one important point that one can't calculate absolute time of the arrival of fault at the bus and only time delay for the disturbance to arrive at the concerned bus from the fault bus, because the bus at which the fault occured is not known. But we can calculate the time difference of arrival time of the PMUs[4]. Now if the procedure of calculating the time of arrival of the PMUs with respect to the time of arrival of the observable bus where disturbance reaches first (bus with minimum time of arrival) is done, then we will be having the data points of the difference of time of arrival. This difference data set is calculated by assuming the faults at different buses. So, we have the data set of difference in arrival times for every bus in the system. This will be obtained by using the shortest path database made for the network. Now if the assumed theoretical fault is the actual fault i.e., if the fault assumed to calculate the delay at bus k is actually the fault happening in real- time in the system, then the difference of measured data set and the calculated data set must be zero, or in statistical sense, it must be minimum.

This is the reason we check for all the buses by assuming fault at every bus and calculating the difference for each bus, the minimum of which will give indication about the bus nearest to the fault.

Flowchart 3.1- Fault location Algorithm-



3.1.2 Measurement of PMU variables-

A PMU can be placed at a bus and it can measure voltage, current and even power flow in the lines connected to the corresponding bus. Theses PMUs have output channels which can be employed according to the need. The data is collected in the Phasor Data Concentrators (PDCs) Units and then computation is done on the data for different purposes.

To have complete observability of the system network in limited resources, optimal PMU placement algorithm is run on the IEEE test bus system. The PMU locations turn out to be bus 2, bus 4 and bus 9 for standard IEEE 9 bus system. These bus locations are found without any consideration of Zero Injection Buses (ZIBs) The ZIBs if included in algorithm will further reduce the PMUs found from the original optimal PMU problem. ZIBs are those buses at which neither the load nor the generation is connected.

Rules for Optimal PMU placement[6]-

a) A bus installed with a PMU measures the voltage of the bus and all of the line currents injected to the Bus.
b) If the voltage at one bus terminal of a line is known and the current through the line is also known, then the voltage at the other end of the line can be determined form KVL on steady-state phasors.
c) If the voltage phasors at the two ends of the line are also known, then one can determine the current phasor of the line.

Based on these it can be concluded that if any node of the power system is connected to other different nodes then one can determine the voltages of all the other nodes if a PMU is placed at the location of the former node.

Now the connectivity vector can be obtained from the adjacency matrix, e.g., for node 5 , the adjacency matrix row is,

$$A_5 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} \quad (3.1)$$

Now if at any of the bus 5 one PMU is located then all other nodes connected to bus 5 are observable.

To formulate the problem one can find the adjacency matrix of the system, denoted by A.

Also, let us introduce n binary variable matrix, where n is the number of nodes in the system. These variables' Boolean value denotes the presence of PMU at the particular node. The below matrix is for IEEE 9 bus standard test system.

$$A = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}, \quad X = \begin{pmatrix} x1 \\ x2 \\ x3 \\ x4 \\ x5 \\ x6 \\ x7 \\ x8 \\ x9 \end{pmatrix} \tag{3.2}$$

Then optimal PMU problem becomes,

$$\min \left(\sum x_k \right) \tag{3.3}$$

, subjected to the constraints

$$AX \geq b, \quad where \;\; x_k \in \{0\,,1\} \tag{3.4}$$

, and

$$b = (1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1)^T \tag{3.5}$$

After solving in MATLAB with linear programming problem, one gets optimal bus locations as bus 2, bus 4 and bus 9.

Hence, with these we know E2, E4, E9 and line currents I54, I14, I64, I72, I89, I39. After obtaining these variables one can find the other node voltages and line currents as

$$E5 = E4 + I(54) * (-Ybus(5,4)^{-1})$$

$$E1 = E4 + I(14) * (-Ybus(1,4)^{-1})$$

$$E7 = E2 + I(72) * (-Ybus(7,2)^{-1})$$

$$E8 = E9 + I(89) * (-Ybus(8,9)^{-1}) \qquad \ldots(3.6)$$

$$E7 = E2 + I(72) * (-Ybus(7,2)^{-1})$$

$$E8 = E9 + I(89) * (-Ybus(8,9)^{-1})$$

$$E3 = E9 + I(39) * (-Ybus(3,9)^{-1})$$

$$E6 = E4 + I(64) * (-Ybus(6,4)^{-1})$$

$$I(57) = (E5 - E7) * (-Ybus(1,4))$$

$$I(69) = (E6 - E9) * (-Ybus(6,9)) \quad \ldots(3.7)$$

$$I(87) = (E8 - E7) * (-Ybus(8,7))$$

If ZIBs are included then the rules which help in optimization are as follows[6]-

a) If a ZIB is connected to a PMU and it is connected to other buses which are observed by PMUs except one, then the bus which is not measured by any PMU can be made observable by applying KCL at the Zero Injection bus and finding the current in the line connecting the ZIB to the unobserved bus. Then KVL will provide the voltage of the previously unobserved bus.
b) If a ZIB is surrounded by the buses which are all observed by some PMU then the ZIB can also be made observable by applying KCL at the ZIB or by using Millman's Theorem at the ZIB.

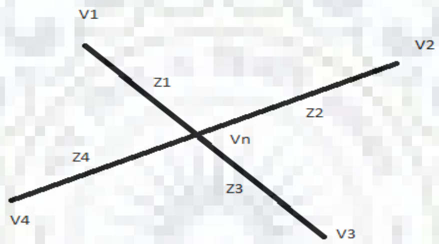


Fig. 3. 2 ZIB rule (b)

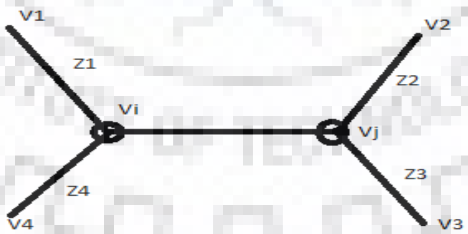


Fig. 3. 3 ZIB rule (c)

$$V_n = \frac{\frac{V_1}{Z_1} + \frac{V_2}{Z_2} + \frac{V_3}{Z_3} + \frac{V_4}{Z_4}}{\frac{1}{Z_1} + \frac{1}{Z_2} + \frac{1}{Z_3} + \frac{1}{Z_4}} \quad (3.8)$$

c) If there are two or more ZIBs connected to each other and are not measured by any PMUs and they all are surrounded by neighbor buses that are observable, then they can be made visible by applying KCL at all the ZIBs connected together.

$$V_i = \frac{\frac{V_1}{Z_1} + \frac{V_j}{Z_{ij}} + \frac{V_4}{Z_4}}{\frac{1}{Z_1} + \frac{1}{Z_{ij}} + \frac{1}{Z_4}} \quad (3.9)$$

$$V_j = \frac{\frac{V_2}{Z_2} + \frac{V_3}{Z_3} + \frac{V_i}{Z_{ij}}}{\frac{1}{Z_2} + \frac{1}{Z_3} + \frac{1}{Z_{ij}}} \quad (3.10)$$

Since eqn. (8) and (9) are two linear equations in two unknown variables Vi and Vj, the solutions can be obtained easily. What ZIB is doing is that it tries to remove the overlapping area of individual PMU observations. That means one PMU gives measurement of all unknown variables whose values can't be obtained from the measurement of any of the other PMUs.

3.1.3 Calculation of Bus Angle Variations-

After the data of the PMU variables is imported to the MATLAB, bus voltage angles need to be calculated in order to observe the swing caused by the fault or disturbance. There can be two thought processes to calculate the bus voltage angles variation with time.

3.1.3.1 Using Discrete Fourier Transform-

The Discrete Fourier Transform is the signal with period 2π, with respect to ω, whose Fourier Series representation is the original discrete time signal. If our discrete signal would have been a continuous time periodic signal, then its coefficients in the Trigonometric Fourier Series representation of itself is the discrete Fourier transform. In mathematical terms,

$$f(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos\left(\frac{2\pi nt}{T}\right) + b_n \sin\left(\frac{2\pi nt}{T}\right), \quad (3.11)$$
$$where\ f(t+T) = f(t)$$

$$a_0 = \frac{1}{T}\int_{(T)} f(t)\,dt \tag{3.12}$$

$$a_n = \frac{1}{T}\int_{(T)} f(t)\cos\left(\frac{2\pi nt}{T}\right)dt \tag{3.13}$$

$$b_n = \frac{1}{T}\int_{(T)} f(t)\sin\left(\frac{2\pi nt}{T}\right)dt \tag{3.14}$$

If this signal f(t) is sampled at M samples per cycle then,

$$Ts = h = \frac{T}{M} \tag{3.15}$$

, where Ts is the sampling time.

Now, at k_{th} sample time t = kh and denoting f(kh) by f_k,

$$a_n = \frac{1}{Mh}\int_{(T)} \rightarrow \sum f(kh)\cos(\frac{2\pi nkh}{Mh})\ h$$
$$= \frac{1}{M}\sum_{k=0}^{M} f(kh)\cos(\frac{2\pi nk}{M}) \tag{3.16}$$

$$b_n = \frac{1}{Mh}\int_{(T)} \rightarrow \sum f(kh)\sin(\frac{2\pi nkh}{Mh})\ h$$
$$= \frac{1}{M}\sum_{k=0}^{M} f(kh)\sin(\frac{2\pi nk}{M}) \tag{3.17}$$

If only fundamental component needs to be seen,

$$a_1 = \frac{1}{M}\sum_{k=0}^{M} f(k)\cos(\frac{2\pi k}{M}) \tag{3.18}$$

$$b_1 = \frac{1}{M}\sum_{k=0}^{M} f(k)\sin\left(\frac{2\pi k}{M}\right) \tag{3.19}$$

Then ,

$$f(t) = a_1 \cos(\omega_0 t) + b_1 \sin(\omega_0 t) = A \sin(\omega_0 t + \varphi) \tag{3.20}$$

where,

$$A = \sqrt{a_1{}^2 + b_1{}^2} \text{and } \varphi = tan^{-1}\left(\frac{a_1}{b_1}\right) \tag{3.21}$$

Hence, we can see that if φ is constant we can have actual amplitude and phase angle of fundamental component of any signal using DFT.

This can be seen in reverse too. Let $v(t) = A \sin(\omega_0 t + \varphi)$, then

$$= \sum_{k=0}^{M} (\sin(2\omega_0 kh + \varphi) + \sin(\varphi))$$

$$= M \sin(\varphi) + \frac{1}{2\sin(\omega_0 h)} \sum_{k=0}^{M} 2\sin(2\omega_0 kh + \varphi)\sin(\omega_0 h) \tag{3.22}$$

$$= M \sin(\varphi) + \frac{1}{2\sin(\omega_0 h)} \{2\sin(\varphi)\sin(\omega_0 h) + \cos(\omega_0 kh + \varphi) - \cos(3\omega_0 kh + \varphi) + \cos(3\omega_0 kh + \varphi) - \cos(5\omega_0 kh + \varphi) + \ldots. + \cos((2M - 1)\,\omega_0 h + \varphi)) - \cos((2M + 1)\,\omega_0 h + \varphi)\} \tag{3.23}$$

$$= 2M \sin(\varphi) + \frac{1}{2\sin(\omega_0 h)} (cos(\omega_0 kh + \varphi) - \cos((2M + 1)\,\omega_0 h + \varphi))) \tag{3.24}$$

$$= 2M \sin(\varphi) + \frac{1}{2\sin(\omega_0 h)} \{2\sin(M\omega_0 h)\sin((M + 1)\omega_0 h + \varphi))\} \tag{3.25}$$

Also, $\sin(M\omega_0 h) = \sin(2\pi) = 0$

Hence ,

$$\frac{2M}{A} a_1 = 2M \sin(\varphi) \Rightarrow a_1 = A\, sin(\varphi) \tag{3.26}$$

Similarly, $b_1 = A\, cos(\varphi)$

Hence

$$A = \sqrt{a_1{}^2 + b_1{}^2}$$

$$\varphi = tan^{-1}\left(\frac{a_1}{b_1}\right) \tag{3.27}$$

Hence, if signal have constant phase then it is easier to determine the voltage magnitude and phase angle from discrete Fourier Transform.

But when signal phase angle is changing it cannot produce good results as integral could not be solved elementarily and the period of integration is large during which φ cannot remain constant. Fig.4 shows a plot of signal $v(t) = \sin(100\pi t + \frac{\pi}{6})$ that shows the angle is constant and accurately calculated by DFT.

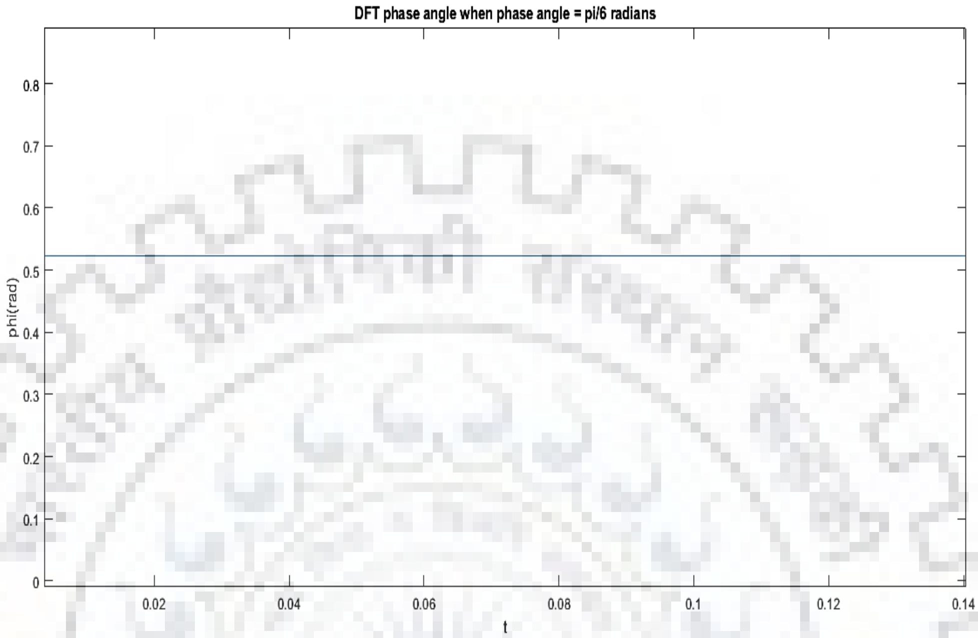


Fig. 3. 4 DFT calculated phase angle when phase angle = pi/6 radians

Fig.3.5 shows how wrong the results are when phase angle is varying with half the frequency,

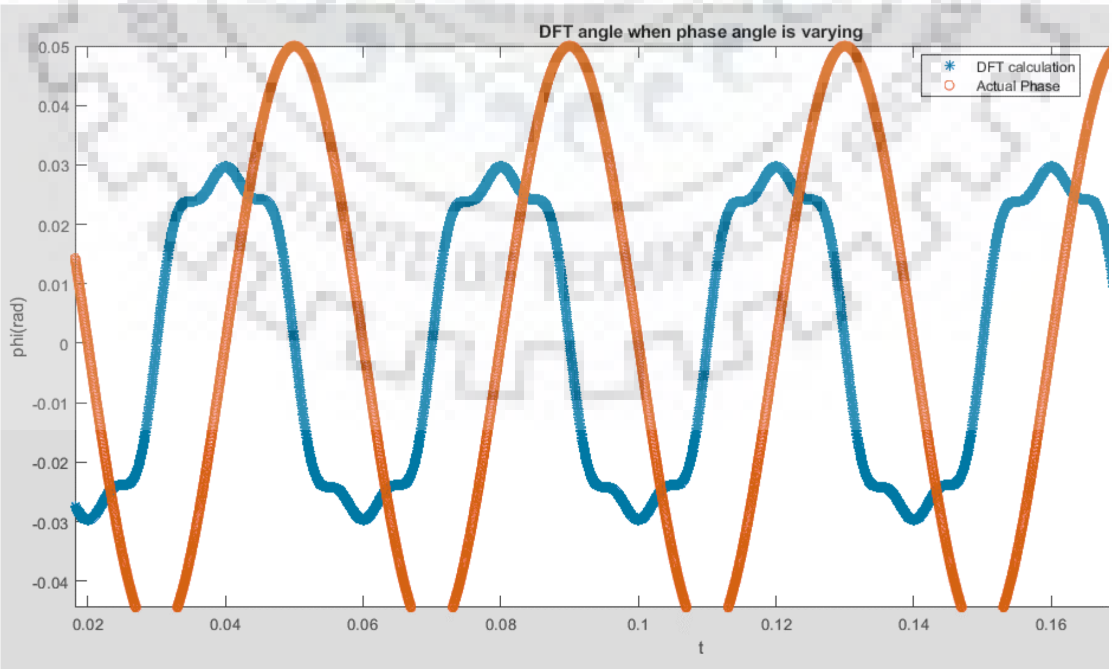


Fig. 3. 5Varying DFT angle when actual angle is varying at half the power frequency

Similarly, Fig. 3.6 shows the difference between DFT calculation and the actual time varying phase angle when the phase angle was varying with one fifth of power frequency.

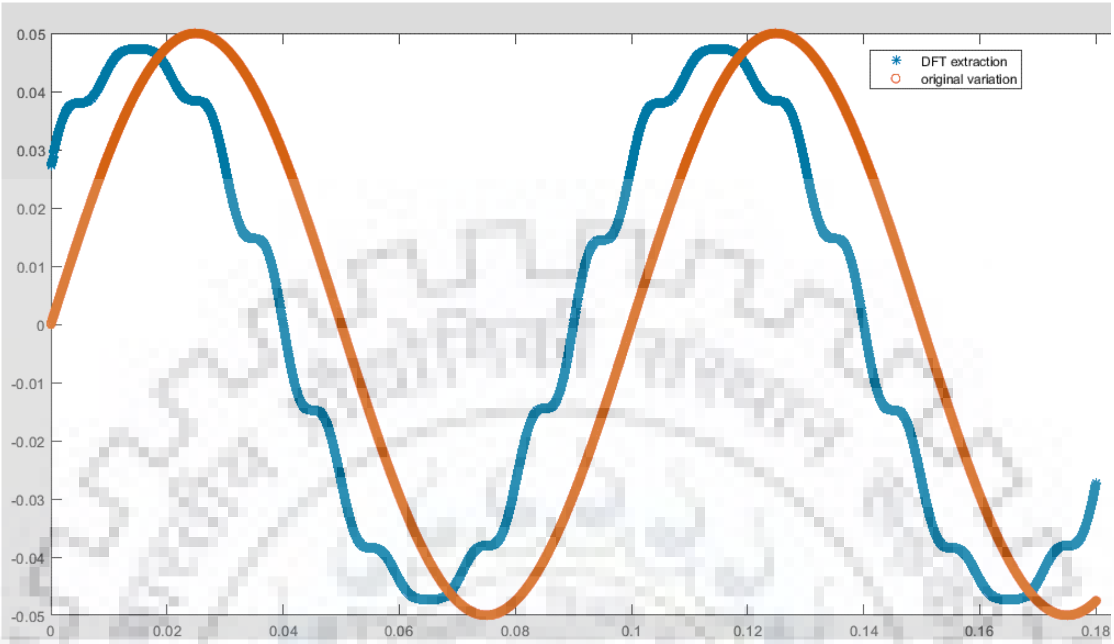


Fig. 3. 6DFT angle when actual phase angle is varying at one fifth of power frequency

3.1.3.2 Calculation Through Hilbert Transform-

Hilbert transform can give more accurate results when the phase angle of sinusoidal signal is varying in time. It is used in communication signal phase demodulation. Hilbert transform is based on conjugate analytic function. The fact that a function and its Hilbert transform form an analytic function conjugate pair is used to extract phase angle variations from the signal.

Hilbert transform of a signal $f(t)$ is given by

$$H(x,y) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{f(t)(x-t)}{(x-t)^2 + y^2} dt \tag{3.28}$$

It can be seen that the $H(x,y)$ is the convolution of $f(t)$ with $\frac{i}{\pi z}$. Hence,

$$H(x,y) = f(x) * \left(Im\left(\frac{i}{\pi z}\right)\right) \tag{3.29}$$

Hilbert transform for signals can be derived form the above definition,

$$H(x) = \lim_{y \to 0} H(x,y) \tag{3.30}$$

Now the point is, $F(t) = f(t) + jH(t)$ is analytic and its angle gives us the phase variation of a sinusoidal signal.

i.e., if $f(t) = A\sin(g(t))$, then $g1(t) = \tan^{-1} H(t)/f(t)$, g1(t) represents the variation of g(t). For a simple sinusoidal signal, one can mathematically prove it.

Now if we consider the signal $v(t) = \sin(\omega_0 t)$ then mathematically,

$$H(x,y) = \frac{1}{\pi}\int_{-\infty}^{\infty} \sin\omega_0 t \frac{(x-t)}{(x-t)^2 + y^2} \mathrm{dt}$$
$$= \frac{1}{\pi}\int_{-\infty}^{\infty} \sin\omega_0 (x-t) \frac{t}{t^2 + y^2} \mathrm{dt} \tag{3.31}$$

$$H(x,y) = \frac{1}{\pi}\int_{-\infty}^{\infty} \sin\omega_0 (x-t) \frac{t}{t^2 + y^2} \mathrm{dt}$$
$$= \frac{1}{\pi}\int_{-\infty}^{\infty} \sin\omega_0 (x+t) \frac{-t}{t^2 + y^2} \mathrm{dt} \tag{3.32}$$

$$H(x,y) = \frac{1}{\pi}\int_{-\infty}^{\infty} \cos\omega_0 x \ \sin\omega_0 t \frac{t}{t^2 + y^2} \mathrm{dt}$$
$$= \frac{1}{\pi}\cos\omega_0 x \int_{-\infty}^{\infty} \sin\omega_0 t \frac{t}{t^2 + y^2} \mathrm{dt} \tag{3.33}$$

Now consider the integral

$$I1 = \int_{-\infty}^{\infty} \sin\omega_0 t \frac{t}{t^2 + y^2} \mathrm{dt}$$
$$\Rightarrow H(x,y) = \frac{1}{\pi}\cos(\omega_0 x)\ I1(y,\omega_0) \tag{3.34}$$

$$I1(y,\omega_0) = \int_{-\infty}^{\infty} \sin\omega_0 t \frac{t}{t^2 + y^2}\, dt$$
$$\Rightarrow I1(y,\omega_0) = 2\int_{0}^{\infty} \sin\omega_0 t \frac{t}{t^2 + y^2}\, dt \tag{3.35}$$

Taking Laplace Transform on both the sides with respect to ω_0, we get

$$L(I1)(s) = L1(s) = 2\int_{0}^{\infty} \frac{t}{t^2 + y^2} \frac{L(\sin(t\omega_0))}{1}\, dt$$
$$= 2\int_{0}^{\infty} \frac{t}{t^2 + y^2} \frac{t}{s^2 + t^2}\, dt \tag{3.36}$$

$$L1 = 2\int_0^{\infty} \frac{t^2}{(t^2 + y^2)(t^2 + s^2)} dt$$

(3.37)

$$= 2\frac{-y^2}{s^2 - y^2}\int_0^{\infty} \frac{1}{(t^2 + y^2)} dt + 2\frac{s^2}{s^2 - y^2}\int_0^{\infty} \frac{1}{(t^2 + s^2)} dt$$

$$L1(s) = 2\frac{-y^2}{s^2 - y^2}\left(\frac{1}{y}\tan^{-1}\frac{t}{y}\Big|_0^{\infty}\right) + 2\frac{s^2}{s^2 - y^2}\left(\frac{1}{s}\tan^{-1}\frac{t}{s}\Big|_0^{\infty}\right)$$

(3.38)

$$= 2\frac{-y}{s^2 - y^2}\left(\frac{\pi}{2}\right) + 2\frac{s}{s^2 - y^2}\left(\frac{\pi}{2}\right) = \pi\frac{s - y}{s^2 - y^2}$$

$$= \frac{\pi}{s + y}$$

$$\Rightarrow L1(s) = \frac{\pi}{s + y} \qquad (3.39)$$

Since L1 is the Laplace transform of I1 with respect to ω_0, hence

$$I1(y, \omega_0) = L^{-1}(L1(s)) = L^{-1}\left(\frac{\pi}{s + y}\right) = \pi e^{-y\omega_0} \qquad (3.40)$$

Here, we assumed $\omega_0 > 0$, while for $\omega_0 < 0$,

$$L(\sin(\omega_0 t) = \int_{-\infty}^{0} e^{s\omega_0}\sin(\omega_0 t)\, d\omega_0 = -\frac{t}{s^2 + t^2} \qquad (3.41)$$

Hence, the result will be negative,

$$I1(y, \omega_0) = -\pi e^{-y\omega_0}$$

(3.42)

Now, our original integral H(x, y) is

$$H(x, y) = \frac{1}{\pi}\cos(\omega_0 x)\,(I1(y, \omega_0)) \qquad (3.43)$$

$$= \cos(\omega_0 x)\, e^{-y\omega_0} \quad , where\ \omega_0 > 0$$

Hence the Hilbert transform = $H(x) = \lim_{y \to 0} H(x, y) = \cos(\omega_0 x)$

Hence, to find phase we see F(t) = f(t) + j H(t)

$$\Rightarrow \mathrm{F(t)} = \sin\omega_0 t + j\cos(\omega_0 t) = e^{j(-\omega_0 t + \frac{\pi}{2})}$$

$$\arg(F(t)) = \left(-\omega_0 t + \frac{\pi}{2}\right) \qquad (3.44)$$

, which determines the variation in the angle except for some finite shift.

The transform can be used on varying phase as can be shown by fig.7, fig.8 and fig.9 for the same signals on which DFT was applied,

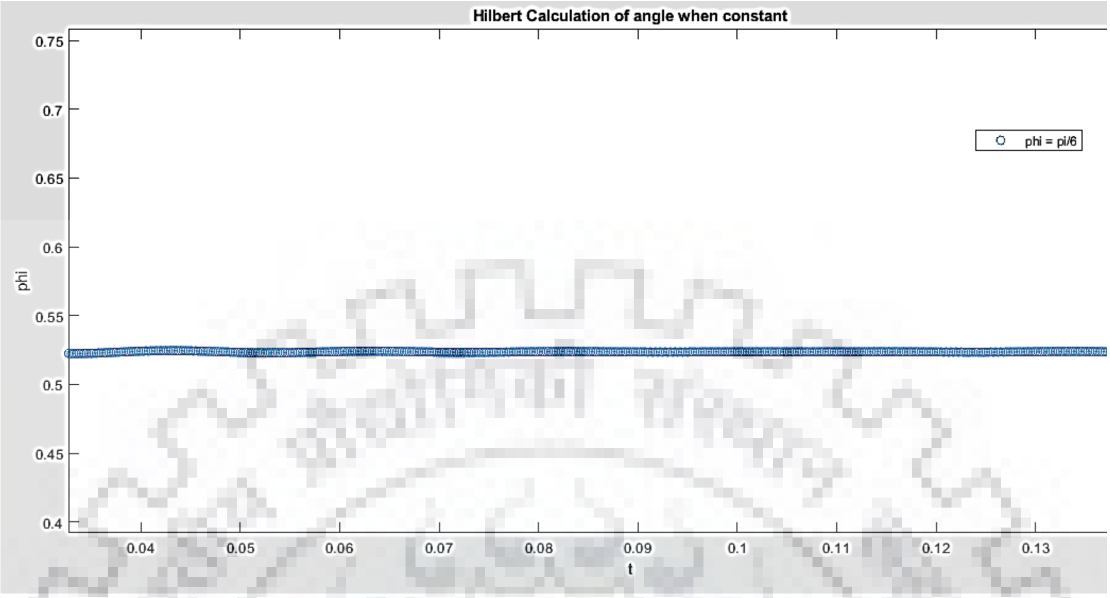


Fig. 3. 7 Angle calculation through Hilbert when angle phi is constant = pi/6

Fig. 3.7 shows phase extraction through Hilbert transform of constant phase angle.

When $\varphi(t) = 0.05\sin(50\pi t)$, fig. 3.8 shows the variation of angle calculated through Hilbert transform,

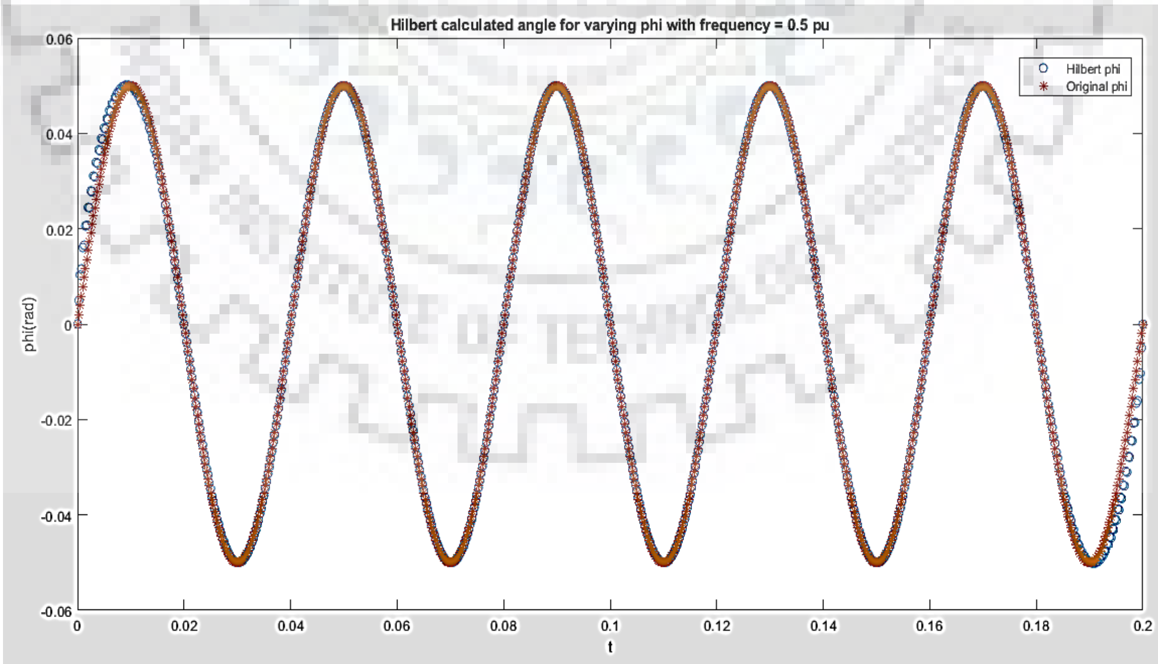


Fig. 3. 8 Phase angle variation calculated through Hilbert Transform for half frequency

Similarly, Fig. 3.9 shows Hilbert calculated values of Phase angle varying with one-fifth frequency.

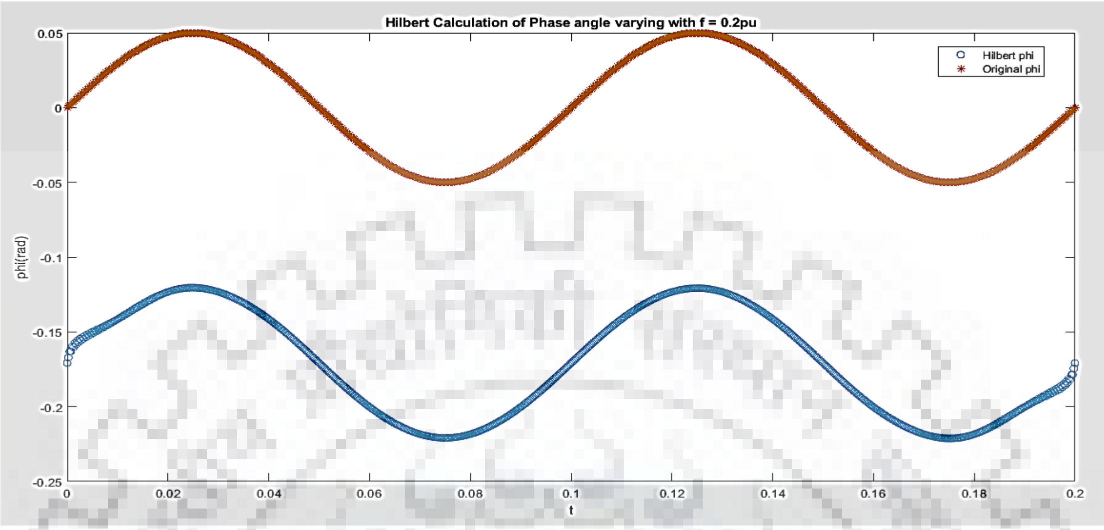


Fig. 3. 9 Phase Angle varying with one fifth frequency extracted through Hilbert Transform

3.1.4 Formation of Shortest Path Database-

It is assumed that a wave when originated travels along the shortest path from source to any destination point under observation. For this Shortest path algorithm is employed to find the path in the network from bus i to bus j. One such algorithm is the Dijkstra Algorithm[2]. Dijkstra algorithm is a greedy search algorithm based on the fact that it searches the optimal solution by taking optimal smaller solutions into account. The principle behind is, the steps taken towards optimal solutions must also be optimal.

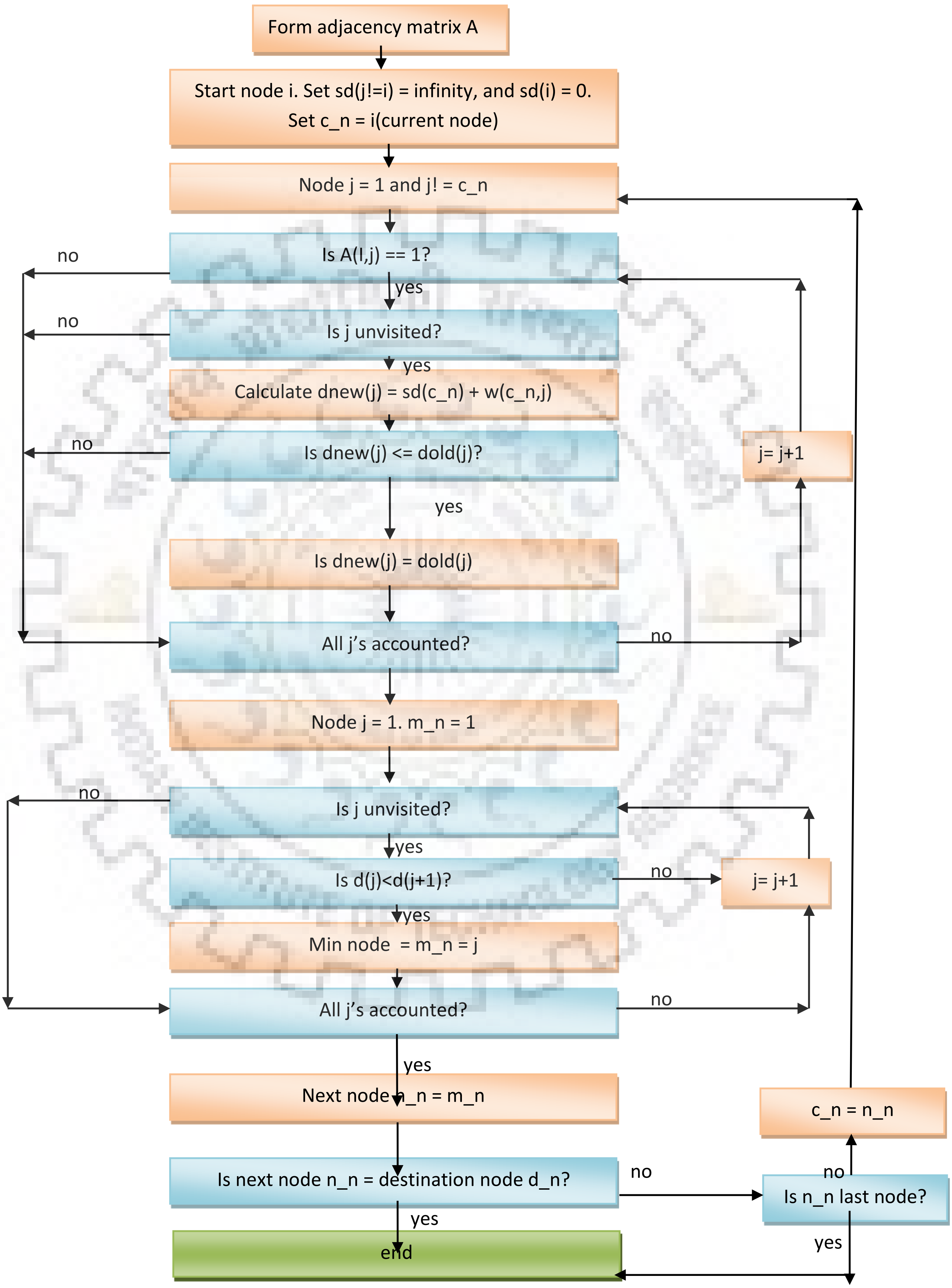
3.1.4.1 Dijkstra Shortest Path Algorithm-

This Dijkstra Path is applied on a graph of weighted branches having only positive weights. Each node of the graph is given a initial value of infinity except the starting node which is initialized to zero. To find the shortest distance path you move from one node to the next. The node at which one is present is called the current node. The node which is selected for next position is called next node. Hence, every time one moves form present node to next node. The present node left behind is called visited node and will form a part of the shortest path obtained at the end of the iteration [33]. The procedure to find the next node is as follows-

One proceeds by counting the next possible shortest distance to every neighbor node connected to the current node. The distance to a neighbor node is calculated by adding the value of current node and the weight of the link or branch connecting the current node to the observed neighbor node. Observed neighbor node is the node for which you are calculating the distance value. If the value of the distance to the neighbor node is less than the already allotted value of the node, then the node value is replaced by the calculated distance value, otherwise, the node value is left unchanged. After every neighbor node of the present node is updated for its distance (changed or unchanged), the graph is analyzed. Now the search is carried out for the next node in the graph by finding the node whose present value is minimum. When the next node is searched, already visited nodes are removed from consideration, i.e., finding the minimum value of only those nodes is carried out which are not visited yet. After leaving the current node, the current node also becomes visited. This process is carried out until one reaches the destination node.

To find neighbor nodes of the present node, the adjacency matrix is formed from the topology data matrix top_data. The topology matrix gives the data of nodes for every branch in the network graph. The diagonal elements of the adjacency matrix are set to zero.

Flowchart 3.2- The Algorithm in the flowchart is as shown-



The adjacency matrix definition is given here,

$$A(i,j) = \begin{cases} 1 \quad , if\ i\ and\ j\ nodes\ are\ connected \\ 0, if\ nodes\ i\ and\ j\ are\ not\ connected \\ 0, \qquad\qquad\qquad\qquad if\ i = j \end{cases} \tag{3.45}$$

Now related to every graph there is a tree called shortest path tree for every starting node, i.e., if a starting node is selected then a tree which gives the shortest path from starting node to any other node can be deduced.

In this tree every node has a characteristic shortest distance assigned to it after running the algorithm. In case, start and destination is not given and algorithm is run until all nodes are spanned by the algorithm. After running the algorithm, each node is assigned a previous node to it, which tells about the node which will fall just before the node under observation in the shortest path. Suppose if the shortest path tree for starting node j is obtained after running the algorithm, and suppose node p is assigned the node q as the previous node in the shortest path tree. Then that means p is arrived at via q when one moves on the shortest path from node j to node p. Now suppose q is assigned a node r, then that means q is arrived at via r when shortest path from node j to node q is followed. Hence, in this way one can back trace the shortest path for any node pair if shortest path tree for every starting node of the graph is known.

This can be understood from the example graph shown in fig 3.10-

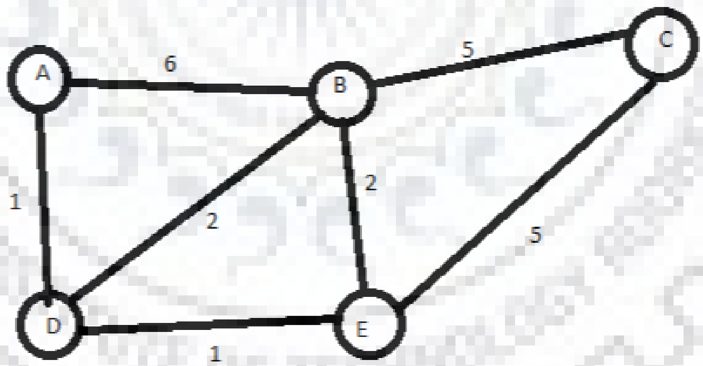


Fig. 3. 10 A weighted graph for Dijkstra Algorithm

Table 3. 1- Shortest Path Tree for node A

Nodes	Pre_nodes	S. dist.
A		0
B	D	3
C	E	7
D	A	1
E	D	2

Now, the above table Table. 3.1 shows the shortest path tree span for bus A(starting node). By viewing this table one can determine the shortest path from A to any node and also the shortest distance from A to any other node can be obtained. If the shortest path from Node A to Node C needs to be found. Then using the table, C can be arrived at via E, E can be arrived at via D, D can be arrived at via A. Hence the shortest path is ADEC and the shortest distance is 7.

3.1.5 Calculation of Time of Arrival-

The time at which a disturbance starts rising at a particular bus is called the time of arrival of the disturbance at a particular bus. Since the disturbance is assumed to travel with some velocity, it will reach different buses at a different time. This time of arrival can be determined from the first swing of the bus angle variation. There are 3 critical points on the bus angle curve that determines the time of arrival of the wave at a particular bus[1]. These are the maximum point, the max velocity point or the max rate of change point, or the bifurcation point.

These points are denoted as-

a) point T1 – peak point (the maximum value of bus angles during the first swing of the curve),
b) point T2- maximum rate of change point or the point at which the rate of change of the phase angle is maximum,
c) point T3- maximum bifurcation point or maximum acceleration point or max double derivative point,

Earlier the same time of arrival is determined using maximum point, bifurcation point and the point of crossing a constant threshold. This approach is described as erroneous as the time of arrival becomes dependent of the value of threshold. So the less was the threshold, the more accurate Time of arrival is obtained[3].

Fig.3.11 shows the three points on the curve of delta variation-

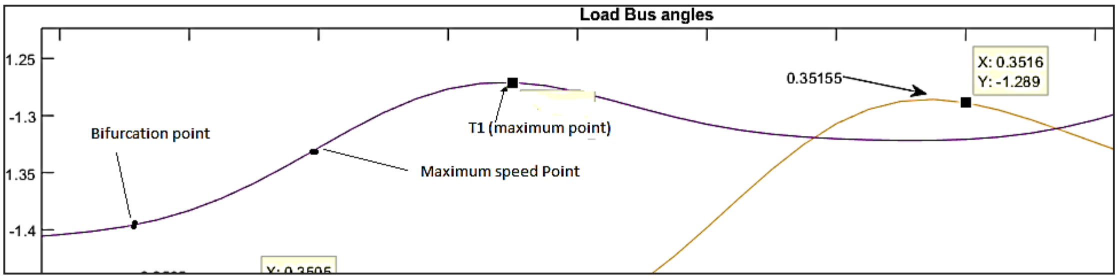


Fig. 3. 11 Variation of load angle and critical points on the curve

After obtaining these three points, we get the time of arrival using an experimental fomula[1].

$$Measured\ Time\ of\ arrival = t_{T3} - \Delta T_{23} - \frac{\Delta T_{12}}{2} \qquad (3.46)$$

The author arrived at this formula after the results of careful simulations. Now after obtaining measured time of arrival, the bus at which the time of arrival is minimum is found and the time of arrival of all the other PMU buses are represented with respect to this minimum arrival time bus. This is discussed in the next section in Multilateration approach to find the bus at which fault occurs.

3.1.6 Multilateration based approach to determine the fault location-

This technique is based on the assumption of a constant velocity of disturbance wave propagating in the power system[4][7]. Let the fault occurred at a bus k = 5 in the power system. Then PMU locations will provide the data and one can calculate the measured time of arrival of the disturbance at all the buses from eq. 3.46. Suppose the fault initiation time is t_k. Time delay for bus 2, 4 and 9 and 3 are-

$T2k = t2 - tk\ ;\ T4k = t4 - tk\ ; T9k = t9 - tk\ ; T3k = t3 - tk$

But since the time of inception of fault is not known, hence tk is unknown. Let the value of t2 is minimum. Then the time of arrival of a disturbance at a particular bus with respect to the the time of arrival of the disturbance at bus 2 is calculated. This is also equal to the time delay at the rest of the buses with respect to the time delay of bus 2.

$T32 = T3k - T2k;\ T42 = T4k - T2k;\ \ T92 = T9k - T2k$

Hence,

$$\begin{aligned} T32 &= t3 - t2 \\ T42 &= t4 - t2 \\ T92 &= t9 - t2 \end{aligned} \qquad (3.47)$$

Hence, a vector for these relative time of arrivals can be calculated,

$$T_M = \begin{bmatrix} T32 \\ T42 \\ T92 \end{bmatrix} \tag{3.48}$$

Now, the same matrix will be evaluated through theoretical consideration and then these matrices will be compared.

The electromechanical wave velocity is given by[3],

$$v = \sqrt{\frac{\omega V^2 \sin\theta}{2h|z|}}$$

ω is the angular power frequency or nominal system frequency,

h is the inertia constant of the generator per unit length,

z is the impedance of the line per unit length,

θ is the line impedance angle,

V is the rated system voltage in pu(= 1pu).

Hence, it can be seen that the electromechanical wave velocity varies with line impedance and if a path contains more than one line then the velocity must be calculated accordingly by taking different line impedances in consideration.

In order to find out the faulty bus, theoretical time delays are calculated. These time delays are calculated from every bus of the power system network to all the PMU locations. Another way of saying it is that the fault is assumed to happen at a particular bus location 'p' and the time delays for this assumed fault is calculated for all the PMU locations. For this, the shortest path database is used to determine the shortest path from bus assumed as the faulty bus to all the PMU locations.

Hence if bus 'p' is assumed to be faulty, then one needs to find the shortest path from bus p to all the PMU buses. For our IEEE 9 bus system, let the 5^{th} bus is assumed to be faulty. Then shortest path P1, P2, and P3 needs to be calculated using the shortest path database, where,

P1 = shortest path from bus 5 to bus 2,

P2 = shortest path from bus 5 to bus 4,

P3 = shortest path from bus 5 to bus 9,

From the Dijkstra algorithm, a previous node matrix is generated for finding the shortest path tree from all the nodes, which gives,

P1 = 5, 7, 2 (to 2)

P2 = 5, 4 (to 4)

P3 = 5 ,7, 8, 9 (to 9)

P4 = 5, 7, 8, 9, 3 (to 3)

This can be found easily from the previous node matrix generated after the shortest path algorithm is applied to the network. In this matrix, each row represents the previous nodes table for different starting nodes.

$$prev\ node = \begin{bmatrix} 1 & 7 & 9 & 1 & 4 & 4 & 5 & 7 & 6 \\ 4 & 2 & 9 & 5 & 7 & 4 & 2 & 7 & 8 \\ 4 & 7 & 3 & 6 & 7 & 9 & 8 & 9 & 3 \\ 4 & 7 & 9 & 4 & 4 & 4 & 5 & 7 & 6 \\ 4 & 7 & 9 & 5 & 5 & 4 & 5 & 7 & 8 \\ 4 & 7 & 9 & 6 & 4 & 6 & 5 & 9 & 6 \\ 4 & 7 & 9 & 5 & 7 & 4 & 7 & 7 & 8 \\ 4 & 7 & 9 & 5 & 7 & 9 & 8 & 8 & 8 \\ 4 & 7 & 9 & 6 & 7 & 9 & 9 & 9 & 9 \end{bmatrix}$$

In order to find out the shortest path from node 5 to node 9, the following procedure is followed-

1) The destination node is 9 and source node is 5, hence start at the 9th column and 5th row to find the node by looking at element (5,9), which is equal to 8, hence the previous node to 9 in the shortest path is 8.
2) Now 8th column and 5th row are analyzed and the element to be focused on is (5,8) which is 7, hence the second previous node along the shortest path is 7.
3) Move to the 7th column and find the element (5,7), which is 5, hence the starting node is arrived at and the process is stopped once the source node is obtained.
4) Hence, the path became 5,7, 8, 9.

Once the shortest path is obtained, the theoretical time delay from the source node to the destination node is obtained as follows-

a) First, the matrix of the path from all the nodes to the PMU nodes must be calculated in which rows represent the nodes and column contains the information of path to be followed. The number of such matrices is equal to the number of PMU locations. For every location, one matrix is generated. Hence path matrix is generated for node 2, 4, 3 and 9 in the above discussed 9 bus system. Let these be path2, path3, path4, and path9.

b) Then the velocity along the links that are in the shortest path is calculated. Hence, for path P1 v_{57}, v_{72} needs to be calculated. Then the time delay along P1 $= \frac{l_{57}}{v_{57}} + \frac{l_{72}}{v_{72}}$. This is calculated for every row in path4, path9, and path2 matrix. After this operation three vector matrices is obtained that gives the time delay from all the nodes to the respective node, i.e. tp4 provides time delay from every node to node 4. Similarly, for other nodes 2, 3 and 9 tp2, tp3 and tp9 are obtained.

Now, assuming the fault occurred on bus k, then time delays to the PMU locations can be obtained through the k^{th} row of tp4, tp2, tp3 and tp9, which are t_{4k}^{th}, t_{2k}^{th}, t_{3k}^{th} t_{9k}^{th}.

The time delay of PMU nodes with respect to the min. delay node is found. Let it be node 2. Then

$$T_{th}32 = t_{3k}^{th} - t_{2k}^{th}$$

$$T_{th}42 = t_{4k}^{th} - t_{2k}^{th}$$

$$T_{th}92 = t_{9k}^{th} - t_{2k}^{th}$$

$$T_{thk} = \begin{bmatrix} T_{th}32 \\ T_{th}42 \\ T_{th}92 \end{bmatrix} \tag{3.49}$$

The difference in theoretical time and measured time is,

$$\Delta T_k = T_{thk} - T_M$$

$$\Delta_k = \|\Delta T_k\| = \|T_{thk} - T_M\| \tag{3.50}$$

This Δ_k is calculated for every bus k and if bus k is the actual fault bus, Δ_k must be minimum. Hence, by finding the minimum value of Δ_k fault bus k can be determined.

3.2 Part 2- Finding Coherent Groups-

3.2.1 Method to Determine the Coherent areas-

The information about the coherency of the areas can be obtained by obtaining information about major inter-area modes. These are slow oscillations that can cause an area to fall out of synchronism from the other area easily during a fault or an emergency state. The presence of oscillation between two areas signifies that they are weakly electromechanically coupled. There can be three forms of intra-mode oscillation that can arise in the power system-

a) **Decaying Oscillation** – In cases where the oscillation is decaying with time. Every Inter oscillation needs to be decayed as fast as possible in the system. Generally, it happens always with an electromechanical wave that it loses some of its energy on encountering a large capacity or inertia generator. In such cases, the oscillation is decaying.
b) **Sustained Oscillation**- Such oscillation remains for a large period of time in the system and signify that there may be a fault or disturbance prevailing in the system which needs to be checked. They can start well before a major fault or outage is to occur.
c) **Growing oscillation**- This oscillation grows in amplitude and causes the system to become unstable.

The effect of inter oscillations present in the system can be seen on different signals of the power system that are obtained from the PMU measurement. But the change in rotor angle position contains only these oscillations. Because a system fault can also cause electromagnetic transients which are having a very small time constant, the rotor can't follow these oscillations and hence only electromechanical oscillation are followed by the rotor. Hence rotor angle can be used to study the inter modes or can be used to extract the coherency information of the power system. Also as the rotor is subjected to these oscillations, its velocity increases and decreases about the steady velocity. Hence velocity signals of the alternators can be used to know about inter harmonic frequency range. This band can be obtained by performing the spectrum analysis of the velocity signals of all the generators. Also, the total Kinetic energy for the overall system also provides information about the frequency band of inter-area harmonics.

The methodology used here is to obtain preliminary information about the coherency of the system and the number of coherent groups through the use of alternator velocity and the use of load angles of the buses. The information about the coherency of the generator buses can be obtained through the velocity signals and kinetic energy signals which can be used to build the total kinetic energy of the system. The generators are grouped with above information signals and then the load buses are assigned to one of these groups based on their coherency with the areas. The coherency with the areas is extracted by studying the spectral coefficients of load angles of these buses.

3.2.1.1 Coherency Determination of Generator Buses-

The velocity of signals is denoted by w1, w2, ..., wn for n generator system.

Then the information of frequency band of low-frequency modes can be obtained through the use of discrete Fourier transform of the velocity signals[7].

$$w_k(f) = \sum_{i=0}^{N-1} H * w_k(i) * e^{-j\frac{2\pi f}{N}i}, \qquad 0 \leq f \leq N-1 \tag{3.51}$$

, where k represents for the index of generator bus,

H is the window function,
N is the total number of samples in the window,
f is the Sampling frequency of the signal.

Plotting ω_k(f) as a function of frequency gives information about the oscillation that is in low frequency range. Individually, there can be many low frequency modes in the frequency spectrum but only that mode is significant which is present in all the speeds. The total kinetic energy can be obtained for every mode from the ω_k(f),
Kinetic energy for mode f in the speed of generator k is,

$$KE_k(f) = \frac{1}{2} * J_k * w_k^2 \tag{3.52}$$

The total Kinetic Energy of the overall system is,

$$KE(f) = \sum_{k=1}^{n} KE_k \tag{3.53}$$

The individual kinetic energy is plotted for viewing the overlapping low frequency mode. Also in the same mode, the total kinetic energy must be maximum as all the magnitude are present at the same frequency location in individual kinetic energy, so they all add up to produce a higher magnitude. However, there can be dominant modes present in a speed spectrum that may be less dominant in another speed spectrum. These modes can't add up and hence they produce a magnitude which is less. Hence, total kinetic energy can be used to verify for the major dominant low frequencies inter-area modes that are affecting all of the generators. This mode can be said to be major electromechanical mode. The number of samples that a window contains must be greater so as incorporate even the lowest frequency inter-area mode[9]. Hence a higher number of samples in the window means finer will be the frequency spectrum with respect to frequency, or, in other words, the frequency resolution is higher. In the application, we can take the entire time sequence of the speed-time waveform and then find its Discrete Fourier Transform.

For e.g., if the total number of samples of the speed- time signal is Ns = 403000 and the window function takes over the entire speed-time sample signal,
and let the sampling period be Ts = 5us, then the frequency resolution is

$$f_{res} = \frac{f_s}{N_s - 1} = \frac{1}{N_s - 1} \times \frac{1}{T_s} \tag{3.54}$$

Here, it will be equal to f_{res} = 0.4963Hz, which can easily incorporate the frequency band of low frequency inter-area modes (0.1-5Hz).

After the band of inter harmonic frequency is found the velocity signals can be checked for correlation with each other [7]. One of the ways to determine the correlation between two functions or sequences is to measure it with the correlation coefficient. The more the signals are related, the more is the value of correlation coefficient, the more the signals are the same with each other in terms of shape, spectra, phase, etc. Unity value of this coefficient means that the two sequences or signals fed to the operation are identical.

Mathematically, the correlation coefficient of two sequences x[n] and y[n] is equal to,

$$r_{xy}[n] = \frac{Cov(x,y)}{\sqrt{var(x) * var(y)}} \tag{3.55}$$

where,

$$Cov(x,y) = \frac{\sum_{k=1}^{N}(x[k] - E_X)(y[k] - E_Y)}{N} \tag{3.56}$$

$$var(x) = \frac{\sum_{k=1}^{N}(x[k] - E_X)^2}{N} \tag{3.57}$$

From Cauchy Schwarz Inequality,

$$(x,y) \leq (x,x).(y,y) \tag{3.58}$$

$$x \rightarrow x - E_X, \ \ y \rightarrow y - E_Y$$

$$\Rightarrow (x - E_X, y - E_Y) \leq (x - E_X, x - E_X).(y - E_Y, y - E_Y) \tag{3.59}$$

$$\Rightarrow \frac{(x - E_X, y - E_Y)}{(x - E_X, x - E_X).(y - E_Y, y - E_Y)} \leq 1 \tag{3.60}$$

$$Cov(x,y) \leq 1 \tag{3.61}$$

Hence, the covariance of the two signals is always less than 1.

Now, the correlation coefficient is calculated for every pair of generators from their speed signals[9],

$$r_{ij}[n] = \frac{\sum_{k=1}^{N}(\omega_i[k] - W_i)(\omega_j[k] - W_j)}{\sqrt{\sum_{k=1}^{N}(\omega_i[k] - W_i)^2 * \sum_{k=1}^{N}\left(\omega_j[k] - W_j\right)^2}} \tag{3.62}$$

$$; 1 \leq \ i,j \leq n$$

These coefficients provide the information of how well they are correlated by providing an index for every pair of generator, which can decide how many areas are present on. The finer boundary still is not clear as it doesn't provide the information about which coefficients are considered to be as a group and that means the range of these coefficients to be called in one group is not very clear and hence only provide fuzzy information about coherent generators.

To obtain further information so that a solid grouping can be decided coherence functions needs to be found. The Coherence function can be calculated through the use of power spectral density functions[9].

3.2.1.2 Cross-Correlation and Auto-Correlation Functions-

The Cross-Correlation function between two signals provides the information about the periodicity in the signals and it also gives a visual approach to compare for the signals about how much the signals are identical. Also, the value of cross-correlation at zero provides a way of measure of similarity between the signals. If the signals are same then the cross-correlation is known as auto-correlation. Auto-correlation function is an even function and its value for only positive values suffice to give the information for the overall sequence.

Mathematically,

$$R_{xy}[n] = \sum_{k=1}^{N} x[k].\,y[k-n] = \sum_{k=1}^{N} x[k+n].\,y[k] \qquad ;-N \leq n \leq N \tag{3.63}$$

The Cross-Correlation function is near to zero or very small if the signals are not related in any way. Also if the signal is containing harmonics or periodic components then the cross-correlation or auto-correlation sequences will have maxima and minima at regular intervals which can be used to estimate the frequency of dominant harmonics. The maxima and minima are usually of decreasing amplitude because of the finite length of sequences.

Also if the two signals are shifted from each other which otherwise are identical, then the global maxima of the whole Cross-Correlation will not be at zero instant. The global maxima get shifted by the same phase as is the original signals phase difference. Also, if a dominant frequency is present in the signal, the autocorrelation sequence of the signal has periodic maxima and minima although not equal in magnitude, but their occurrence repeats after same time as the period of dominant harmonic in the signal (Fig.3.12).

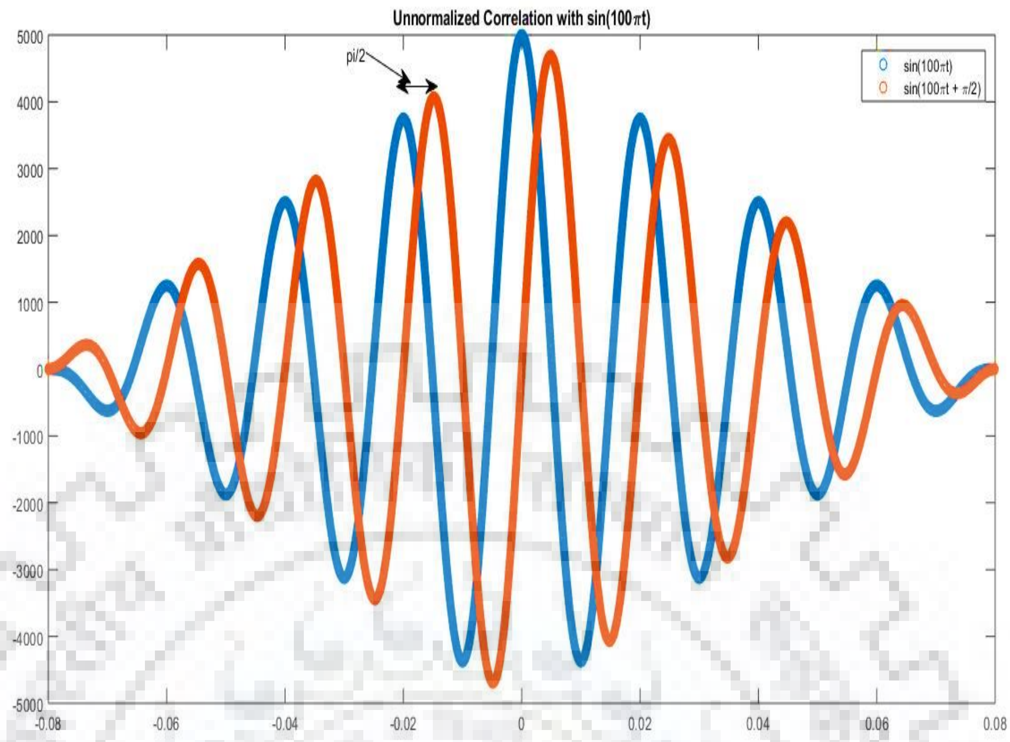


Fig. 3. 12 Correlation between sin(100*pi*t) and sin(100*pi*t + pi/2)

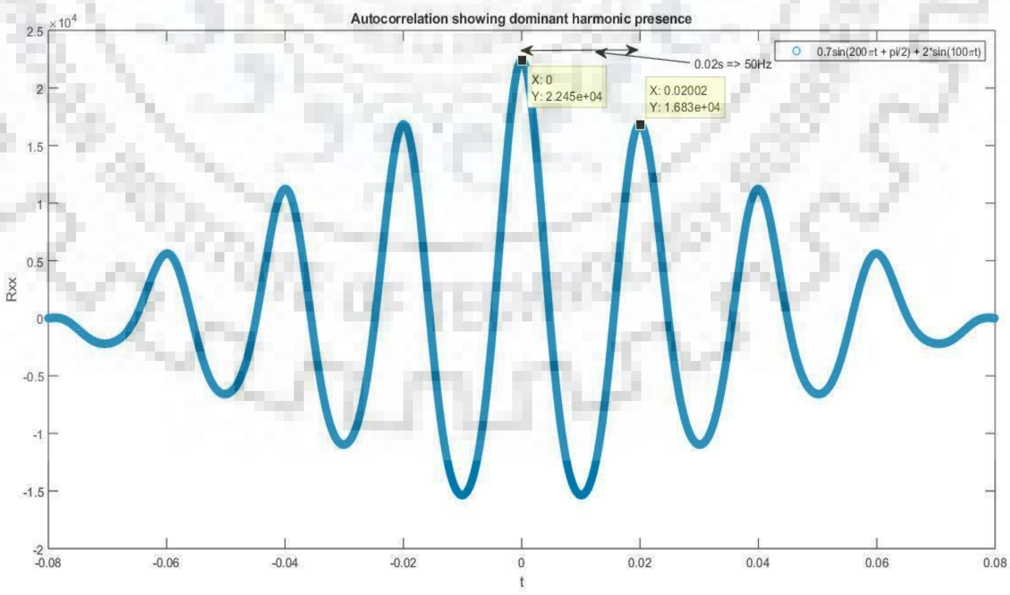


Fig. 3. 13 Autocorrelation function determines the most weighted frequency or period in signal

In the system, the auto-correlation and cross-correlation of the speed signal of each generator speed is calculated after low pass filtering of the speed signals. Thus if i and j denote the indexes of the generators and ω_i and ω_j denote the speed signals of the generators i and j respectively, then

$$\omega_{fi}[n] = \mathcal{F}^{-1}(H_{buw}(f, f_0, r) * \omega_i(f)) \tag{3.64}$$

, where ω_{fi} is the filtered velocity signal of the generator i,

$H_{buw}(f, f_0, r)$ is the transfer function of the filter (Butterworth) with order r and cutoff frequency f_0,

$\omega_i(f)$ is the discrete Fourier transform of the signal velocity $\omega_i[n]$.

Then correlation sequences of these filtered signal are found using,

$$R_{ij}[n] = \sum_{k=0}^{N-n-1} \omega_{fi}[k].\omega_{fj}[k-n] \tag{3.65}$$

$$;1 \le i,j \le Ng; \quad -(N-1) \le n \le 0$$

$$R_{ij}[n] = \sum_{k=n}^{N-1} \omega_{fi}[k].\omega_{fj}[k-n] \tag{3.66}$$

$$;1 \le i,j \le Ng; \quad 0 < n \le N-1$$

, where, N_g = number of generators,

N = total number of samples in the sequence of speed signals w_i or w_j,

After this, the Discrete Fourier Transform of the Correlation sequences is calculated which will provide the information about the frequencies in the correlation sequences. This is because if two signals have that common frequency component then its periodic trend will definitely come in Cross-Correlation Sequences of the two signals and Discrete Fourier Transform explores the presence of frequency component in the signals. So the Discrete Fourier Transform of the correlation sequence will also contain the component frequencies present in the correlation sequence signals and because the cross-correlation sequence signals contain the modes which are common to both the signals of which the cross-correlation is found, it will lead to the revelation of largely those component frequencies common to both the signals (fig. 3.14). Now if the signals are near to each other they will lead to the same Discrete Fourier Transform sequences.

Again the Discrete Fourier Transform sequence is calculated with all the samples present in the cross-correlation sequence or auto-correlation sequence. Also, only positive part of the cross and auto-correlation sequence is analyzed.

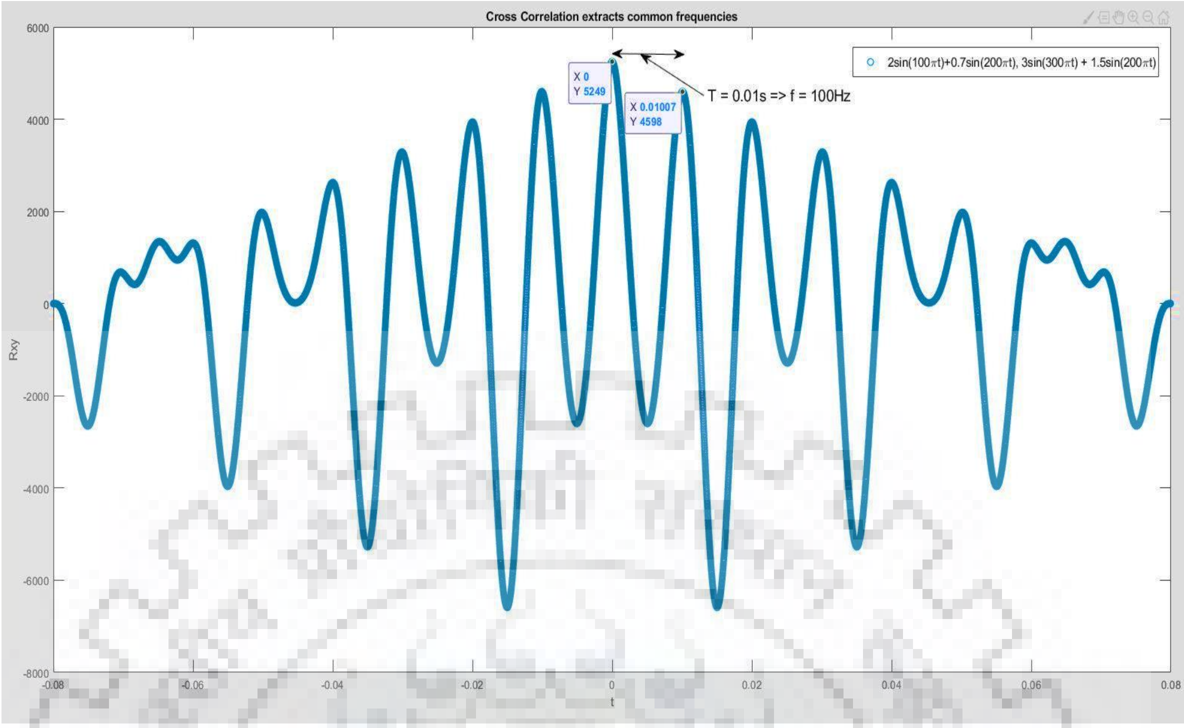


Fig. 3. 14Cross correlation of 2sin(100pi.t)+0.7sin(200*pi*t) and 3sin(300*pi*t)+1.5 sin(200pi.t)

The Discrete Fourier Transform of cross-correlation sequence and auto- correlation sequences of the two signals are called Cross- Spectral Density and Power Spectral Density of the signals respectively.

Hence,

$$P_{xy}(f) = \sum_{n=0}^{N-1} H\, R_{xy}\,[n]\; e^{-j\frac{2\pi f}{N}n}, \qquad 0 \le f \le (N-1);\, f \in \mathbb{Z} \qquad (3.67)$$

$$P_{xx}(f) = \sum_{n=0}^{N-1} H\, R_{xx}\,[n]\; e^{-j\frac{2\pi f}{N}n}, \qquad 0 \le f \le (N-1);\; f \in \mathbb{Z} \qquad (3.68)$$

It is to be noted that the cross-correlation function has values on the negative arguments also but only causal sequences are considered here and hence its value is taken to be zero before n<0.

Now, these spectral density functions tell about the density of modes in the spectrum of signals. For every pair of signals, the corresponding spectral density functions, whether cross or power, specifies the strength of a particular frequency in the signal spectrum. More values of it at a particular frequency indicates the prevalence of the frequency components to both the signals in case of cross-spectral density or in the same signal in case of power spectral density function.

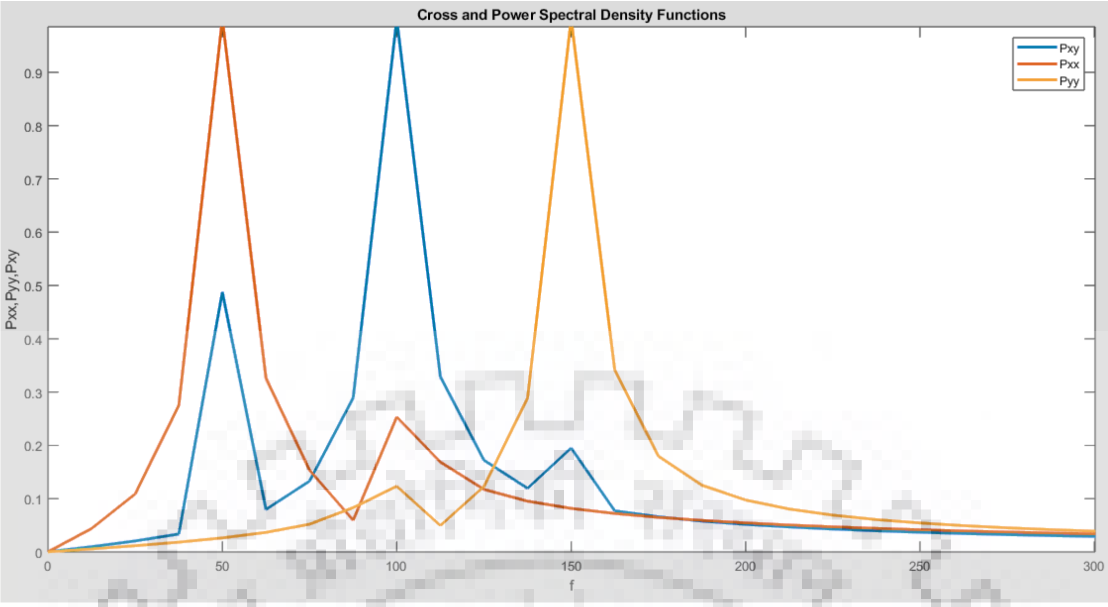


Fig. 3. 15 Cross and Power Spectral Density functions

Fig. 3.15 shows the Cross and Power- Spectral Density functions of the signals

$$x(t) = 3\sin(100\pi t) + 1.5\sin(200\pi t)$$
$$y(t) = 2\sin(300\pi t) + 0.7\sin(200\pi t) \quad (3.69)$$

It can be seen from Power and Cross-Spectral Density functions that the cross-spectral density contains the largest magnitude for 100Hz which is common to both the x(t) and y(t) and have small magnitudes for other two frequencies 150Hz and 50Hz as they are not common to both the signals and hence had not appeared dominantly in the cross-correlation function which ultimately led to diminished magnitude of these frequencies in the Cross-Spectral Density functions.

Also, the power spectral density functions contain the component frequencies of respective signals having magnitude according to their dominance in the corresponding signal. Hence, Power Spectral Density of x contains 50Hz as the dominant frequency and 100 Hz as a low magnitude frequency. Similarly, the Power Spectral Density function of y shows 150Hz as the dominant frequency and 100Hz as the less contributing frequency. This contribution is said to be with respect to the power of the signal. It is to be noted here that the Power and Cross-Spectral density functions are shown above in Fig. are normalized by their magnitude. Even with these individually dominant frequencies, cross-spectral density function has only one dominant frequency which is common to both the signals. Also individually the frequency 100Hz is not contributing largely to the signals when compared to other frequencies in the respective signals which makes clear, a fact that it is not necessary that the common frequencies should have a large share in the original signals in order to appear in the cross-correlation function. Hence, cross-correlation detects all the common frequencies.

3.2.1.3 Coherence Function-

For every frequency in the domain of cross-spectral density and power spectral density functions, the coherence function is a normalized function and gives a magnitude to measure the correlation between the two signals for a particular frequency.

Mathematically,

$$C_{xy}(f) = \frac{|P_{xy}(f)|^2}{|P_{xx}(f)P_{yy}(f)|}, \quad \forall \; f = 0, 1, 2, \dots, (N-1) \tag{3.70}$$

Cauchy- Schwarz inequality guarantees that C_{xy} will always be less than 1 even if the noise signal is present in x and y[10]. When the same signal is used to find coherence function, it will turn out to be 1. Also if the signals x and y are linearly related, i.e., the relation between them can be expressed by a linear constant coefficient differential equation, then the coherence function turns out to be 1.

For the signal x and y in Eq. (3.69) above, the graph is shown below,

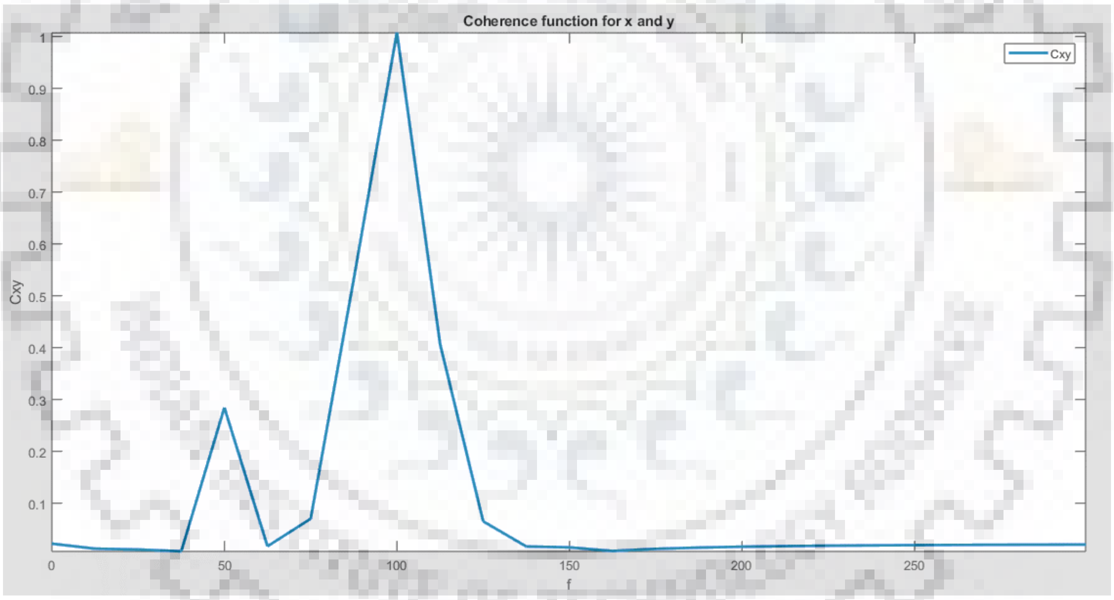


Fig. 3. 16 Coherence Function for x and y

In summary, all of these function will be used as follows-
For every pair of speed signals ω_i and ω_j,

a) First, calculate the Discrete Fourier transform of the signals and individual and total energy as a function of frequency, there would be N_g of it, where N_g is the number of generators in the grid. Find the low frequency inter harmonic band in the total energy. This also gives the value of dominant low inter harmonic frequency present in all the generators.

b) Now filter the frequency signals ω_{fi} and ω_{fj}, and evaluate the correlation coefficients r_{ij}. If the groups are clearly visible from the 'r' matrix, then form the groups. If the groups are ambiguous then go for step (c).

$$r_{ij}[n] = \frac{\sum_{k=1}^{N}(\omega_i[k] - W_i)(\omega_j[k] - W_j)}{\sqrt{\sum_{k=1}^{N}(\omega_i[k] - W_i)^2 * \sum_{k=1}^{N}\left(\omega_j[k] - W_j\right)^2}} \quad (3.71)$$

$$; 1 \leq \ i, j \leq n$$

c) Evaluate the Correlation Functions for every pair of ω_i and ω_j according to equation (58) and (59).

d) Evaluate the Power Spectral Density and Cross-Spectral Density functions for every speed signal and pair of speed signals respectively according to the equation given below,

$$P_{ij}(f) = \sum_{n=0}^{N-1} H\, R_{ij}\,[n]\, e^{-j\frac{2\pi f}{N}n}, \qquad 0 \leq f \leq (N-1); f \in \mathbb{Z} \quad (3.72)$$

$$P_{ii}(f) = \sum_{n=0}^{N-1} H\, R_{ii}\,[n]\, e^{-j\frac{2\pi f}{N}n}, \qquad 0 \leq f \leq (N-1);\ f \in \mathbb{Z} \quad (3.73)$$

where, $0 \leq i \leq Ng;\ \ i \in \mathbb{Z}$

e) Evaluate the Coherence function for every pair of speed signals according to the following equation,

$$C_{ij}(f) = \frac{\left|P_{ij}(f)\right|^2}{\left|P_{ii}(f)P_{jj}(f)\right|}, \quad \forall\ \ f = 0, 1, 2, \ldots, (N-1) \quad (3.74)$$

and as was pointed earlier that

$$C_{ii}(f) = 1 \quad (3.75)$$

f) Now visual inspection of coherence function can give information about the group of the generators, but in order for a computer to do it a method which can group them together should be applied on the coherence functions. For this, kmeans algorithm is used to determine the groups by feeding the data of the coherence function of one node with all the nodes of the grid. That is to say, the coherence function of node i with all other nodes of the power system is calculated and then it is stored in the form of a matrix.

$$data_i = \left[C_{ij}\right]_{(N_g \times N)} \quad ,1 \leq \ j \leq Ng; \quad (3.76)$$

where,
N_g is the number of generators and,
N = Total number of samples in Coherence function.
Now, this matrix is fed to k- means algorithm having Ng number of vectors to be clustered into optimum no. of groups k and all vectors have dimension N.

The clustering takes place along the row of the $data_i$ matrix which implies that each row is considered one vector and there are Ng rows (because there are Ng generators) and these rows are clustered.

Hence, related to every coherent node there will be a $data_i$ matrix and each matrix may provide a different set of clustered nodes. Those generators will be considered in the same group which gives the same set of clustered nodes.

3.2.2 **Coherency for Non- Generator/Load Buses-**

The coherency for generator buses can be found using the method in the previous section. If this determination of coherent groups leads to k groups(k<Ng) of generators in the system, then the areas are determined. But each of the load bus can also be having oscillations as are having on the nearest coherent area in case of fault or outage. So, each load bus needs to be assigned to one of the coherent groups from the groups found in the previous section. In this scenario, the whole power system buses are allotted their particular areas and these areas are considered to be in synchronism within them, i.e., any two buses of an area are not violently oscillating against another bus within the same area. Such a formulation of areas will lead to the definition of clear cut boundaries which are connected via some links. These links even if disconnected will not cause much harm if the generation load balance is present in the subsystems created by such disconnection. It can help take a decision during intentional islanding.

So in order to assign the load buses to one of the coherent groups of generators, the load angle is calculated for each load bus using PMU measurements. It can also be calculated from generator bus voltage measurements if all loads connected to load buses are assumed to be constant impedances [9]. In case, the generator bus voltages are known, the admittance matrix equations help find the voltage phasors for non–generator buses.

$$I = YV \Rightarrow \begin{bmatrix} Ig \\ Il \end{bmatrix} = \begin{bmatrix} Ygg & Ygl \\ Ylg & Yll \end{bmatrix} \begin{bmatrix} Vg \\ Vl \end{bmatrix} \quad (3.77)$$

If all load injections are assumed to be admittances, then

$$\begin{bmatrix} I_{l1} \\ \vdots \\ I_{lN_l} \end{bmatrix} = \begin{bmatrix} Y_{l1}{}' & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & Y_{lN_l}{}' \end{bmatrix} \begin{bmatrix} V_{l1} \\ \vdots \\ V_{lN_l} \end{bmatrix} \Rightarrow [Il] = [Yll'][Vl] \quad (3.78)$$

From this equation, when substituted in eq.(3.77),

$$[Il] = [Yll][Vl] + [Ylg][Vg] \Rightarrow [Yll'][Vl] = [Yll][Vl] + [Ylg][Vg] \quad (3,79)$$

$$\Rightarrow \{[Yll] - [Yll']\}[Vl] + [Ylg][Vg] = 0$$

$$\Rightarrow \{[Yll1]\}[Vl] + [Ylg][Vg] = 0 \Rightarrow [Vl] = [Yll1]^{-1}[Ylg][Vg] \quad (3,80)$$

Hence, finding the angle of these phasors gives the value of phasor angles of load buses or non –generation buses,

$$[\delta_L] = angle([Vl]) \quad (3,81)$$

After the load angles of non- generator buses are found, the same procedure as that have been applied on speed signals is used to for the allotment of these buses to one of the groups of generators already found.

Firstly, the filtered angles along with generator angles are used to calculate the correlation coefficients which gives the general information about the coherency of non-generator buses. Then, find the Cross-Correlation and Auto-Correlation functions[9]. Then coherence function is calculated which is used as a data for k-means-algorithm. Then those buses can be considered to be in the same group which has the same value of cluster number. Also, k-means is run with the intention to divide the system into k-areas and the value of k was determined from the generator grouping done earlier with speed signals.

3.2.3 K- Means Algorithm (Lloyd's Algorithm)-

This algorithm is a graph clustering algorithm which aims at grouping the nodes of the graph into a pre-specified number of clusters or groups. So for example, if a graph contains 100 nodes, then k means algorithm will allot every node to a group out of k groups and the number of groups k is pre-specified [35]. The procedure minimizes the squared Euclidean L^2 distance of all points from their respective cluster centers. The algorithm selects k random centers or centroids(means) from the given group of points. Each centroid or mean represents one group and every point of the data will be given to these centroids or groups. Then every points instance(vector) is checked for the minimum distance centroid among the selected centroids. The point is allotted the group whose centroid has a minimum distance from the data point. Then new centroid is found and this virtual process is repeated again and again. It can be proved that the k- means algorithm will always converge and have a solution always. The algorithm procedure is given in the following points-

STEP 1: INITIALIZE

1) First randomly assign k centers from the data points. These will be centroids for the first iteration. Let k_{th} center be C_k and let r_{th} data point be given by D_r. r varies from 1 to n, where n is the number of data points in the data.
2) Find L^2 norm between the C_k and D_r , i.e., find the Cartesian distance between every centroid C_k and point D_r for every k from 1 to k.
3) Find the centroid C_r which has a minimum value of L^2 norm from the point D_r. and allot the point D_r to the group C_r. This can be done by storing D_r and C_r in one matrix.
4) Do step 2 and step 3 with all the data points D_r, where r varies from 1 to n. Allot them one of the k matrices according to point number 3.
5) After finishing step four, k groups are obtained or k matrices each containing one group is obtained.

STEP 2: LOOP

1) Find D_{Cr} by calculating the distances from the centroid of one group to the data points in the same group, i.e., in the same matrix. This step should be done for every r from 1 to k, as there are k groups. Then find the sum D of all these distances.
2) Check if the change in D is smaller than a predefined error threshold.
3) If yes, exit the loop. In this case, the k matrices obtained are the groups required.
4) If no, then proceed to next point.
5) Find the new centroid of every group by averaging all the data points in one matrix. So there are new k centroids obtained by averaging the vectors present in one individual matrix. It is to be noted that only vectors or data points in individual matrices needs to be averaged and should not be done for all matrices' data points in which case only one average will be there which is not intended.
6) Find L^2 norm between the C_k and D_r , i.e., find the Cartesian distance between every centroid C_k and point D_r for every k from 1 to k.
7) Find the centroid C_r which has a minimum value of L^2 norm from the point D_r. and allot the point D_r to the group C_r. This can be done by storing D_r and C_r in one matrix. Do step 6 and step 7 with all the data points D_r, where r varies from 1 to n After finishing this step, k groups are obtained or k matrices each containing one group is obtained. Go to step 1.

The above algorithm using random initialized data points is called Lloyd's Algorithm. Apart from its simplicity of application and scalability, there are a few issues with the application of this algorithm.

The Lloyd's algorithm is sensitive for local extremum and can stop at such a local minimum. Also, it is too sensitive to the choice of the randomly chosen initial cluster. This doesn't mean that it will not converge sometimes. In fact, Lloyd's algorithm always converges to a minimum of this problem. The sensitivity issue leads to unreliable results. Because Lloyd's algorithm is based on local search, it changes its results on different initializations. So, k clusters obtained from one initialization may have a different set of points when finding through some different initialization of centroids. Also, k means on its own takes a lot of iteration (fig.3.17). Below fig. 3.18 shows an example of k-means run on a data set. Ideally, the star should have been grouped in one cluster and circle should have been grouped in another cluster. But different initialization leads to different clusters. Hence, k- means on its own is unreliable. Also, k-means group the data points based on the concept of nearness, i.e., points which are nearer to each other than all other points are grouped in a cluster. But sometimes nearness or crowdedness is not the only feature in the data. Sometimes, there can be patterns in the data which the human brain can easily find but if that pattern is to be made the rule to find the group then the concept of nearness gets violated. For example, in figure 3.18 the points on the top of the star are nearer to the points of the circle on the top, so k- means will group them together. But it can be clearly seen that these groups are not the groups required.

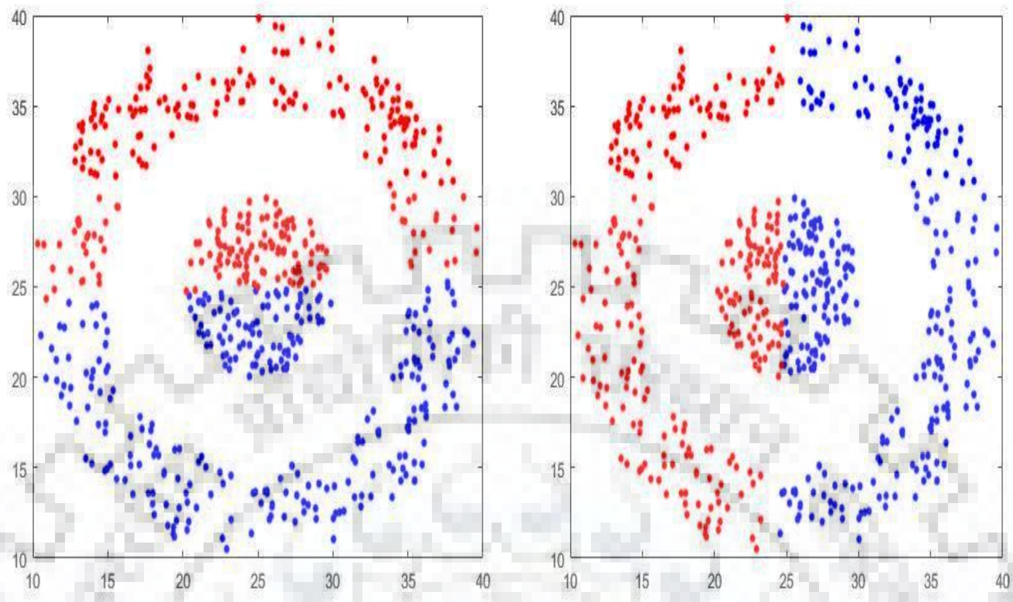


Fig. 3. 17 k means gives wrong results for different patterns and result is not unique.

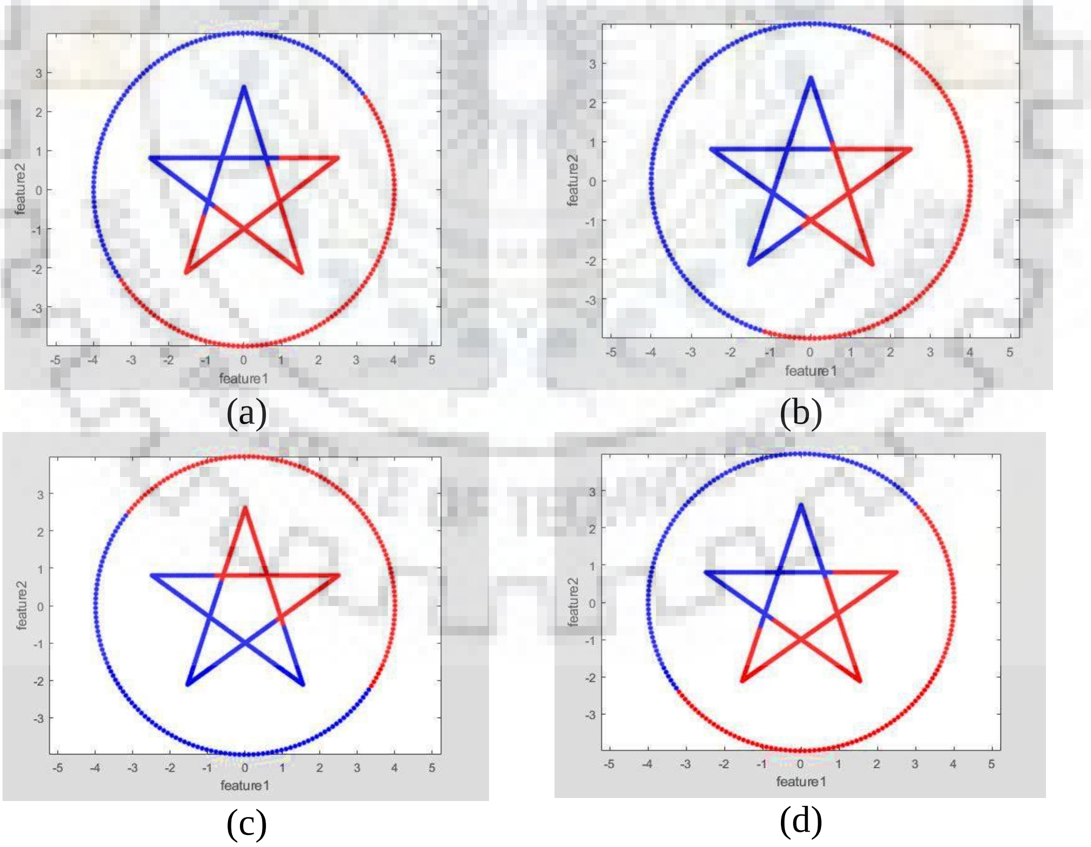
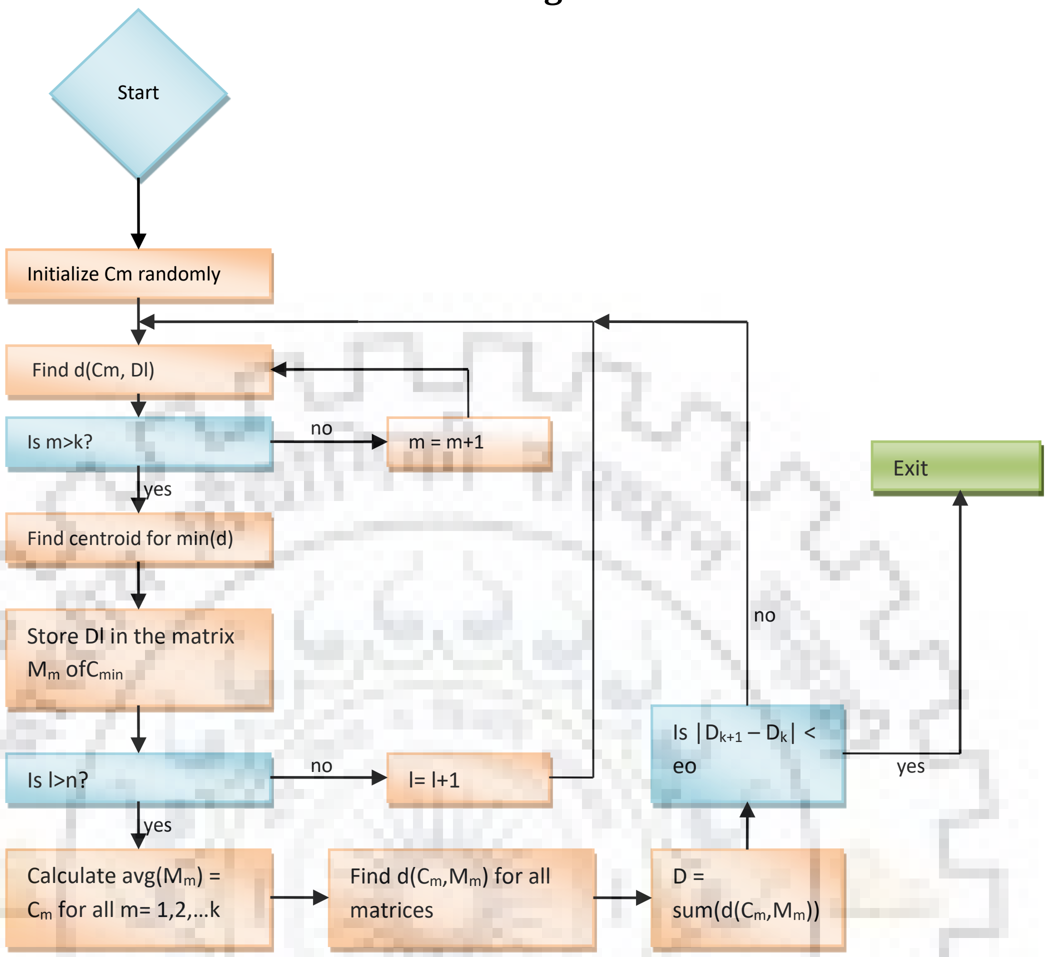


Fig. 3. 18 k- means algorithm leading to different clusters and the results are also not satisfactory

Flowchart 3.3- Flowchart for k- means algorithm-



So k- means is incapable to find such groups where the pattern is the rule to follow. It is to be noted that what the previous statement implies is not some drawback of k-means but it is not meant for use in these pattern finding and hence it is incapable for such application. The fact that grouping is totally wrong can be seen from Fig. 3.17. Furthermore, different initialization led to different groups in the data which echo the fact that traditional k- means or Lloyd's algorithm without any intelligent initialization is totally unreliable.

It is to be noted that the good centroid initialization can be found by k-means++ algorithm. The problem is, different random initialization leading to a different set of groups formed in the result. K-means++ is an algorithm which is used to initialize the centroid for Lloyd's algorithm [14][11]. It initializes the centroids based on the probabilistic approach and tries to spread the initial centroids as much as possible in different areas of the data space. In this way, all the clusters which are widely separated can be found and hence always provides more or less the same result. The algorithm is as follows:

1. Pick any random points as first center C_k.
2. Find the distance $d(C_1, D_k)$ for all points k from 1 to n, and choose the point which has the highest probability,

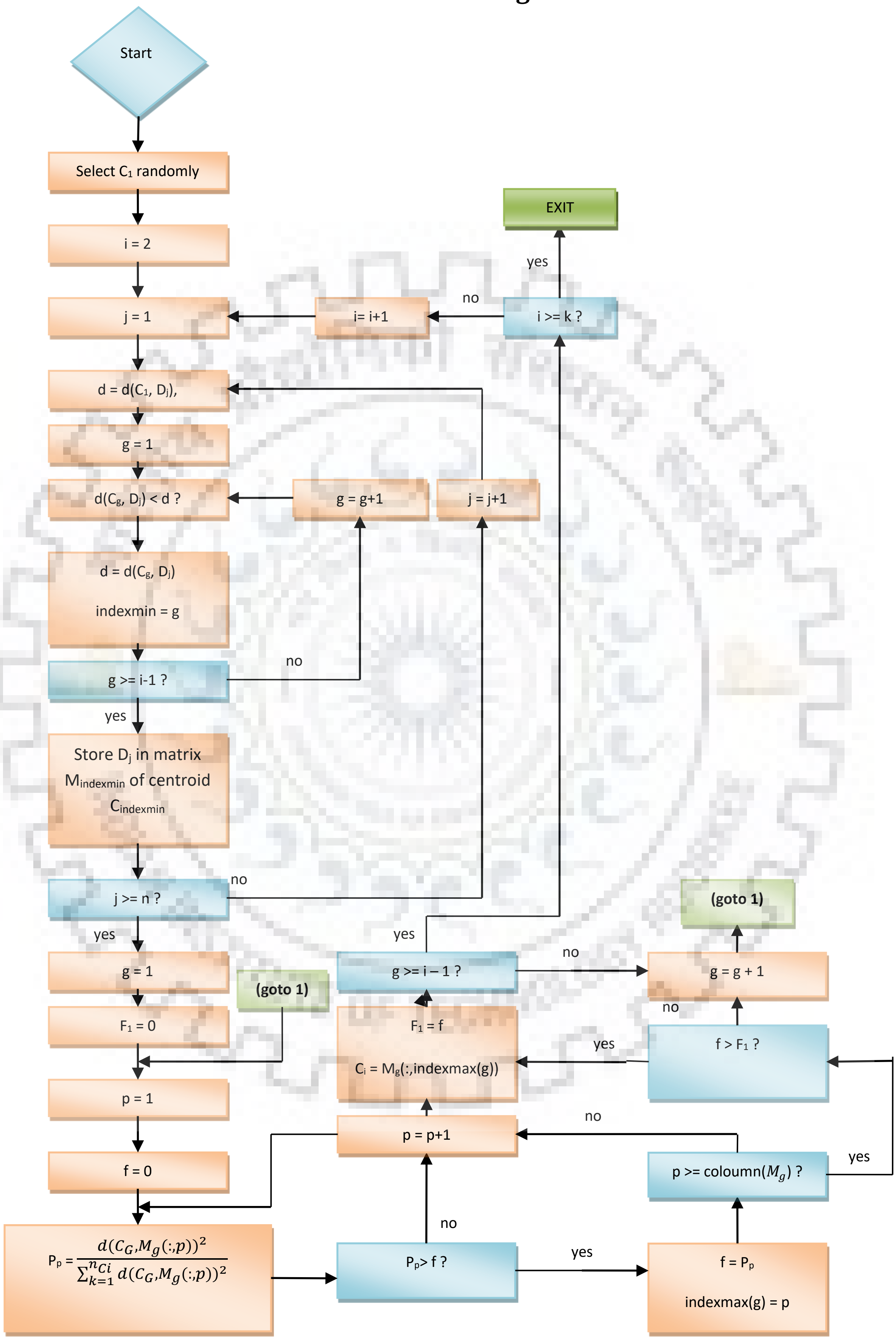
$$P(C_1, D_k) = \frac{d(C_1, D_k)^2}{\sum_{k=1}^{n} d(C_1, D_k)^2} \tag{3.82}$$

3. Loop for i = 3,4,5,…,k. For finding j^{th} centroid, all previous j-1 centroids are used.
4. First assign every data points to one of the j-1 centroids which is nearest to it, i.e., calculate distance $d(C_i, D_k)$ for i = 1 to j-1, and then select that i for centroid for which distance is minimum. Assign D_k to that centroid.
5. Since every data point is now in one of the centroids, now a probability for every point is to be found according to the group in which that point is placed. Let $S(C_i)$ be the group of points or set of data points assigned to the centroid C_i, i.e., $S(C_i)$ is the group of points which are nearest to C_i than rest other data point. Let those data points be denoted by D_{kCi} and n_{Ci} is the number of data points assigned to the group of centroid C_i then the probability of D_{kCi} can be calculated as,

$$P(D_{kC_i}) = \frac{d(C_i, D_{kC_i})^2}{\sum_{k=1}^{n_{Ci}} d(C_i, D_{kC_i})^2}, \quad where\ D_{kC_i} \in S(C_i) \tag{3.83}$$

6) Find this probability for all data points according to their nearest centroid.
7) Choose j^{th} centroid as the point having the maximum likelihood of getting selected, i.e., it has a maximum probability.
8) If i<k, then go to step 3, else go to next step.
9) Exit. The k centroid obtained are the initialized centroids to be used in Lloyd’s algorithm.

Flowchart 3.4- Flowchart for kmeans++ algorithm-



3.2.4 **Spectral clustering-**

A graph can be assigned weights to cluster it into two groups, and concept of spectral clustering can be understood from the point of view of separating the nodes of the graph into two groups according to some characteristics which will be embedded in the weight matrix[12]. So, suppose that there are N nodes and they need to be grouped into two groups which are decided by the geometrical pattern they are making. Fig. 3.19 shows a bunch of data points which can easily be recognized to belong to one of the two groups. Let the points in the data be given indexes v_i whose value will decide whether they are in group 1 or group 2.
Then

$$v_i = \begin{cases} +1 \quad , \quad if\ i \in group\ 1 \\ \ \ 0 \quad , \quad if\ i \in group\ 2 \end{cases} \tag{3.84}$$

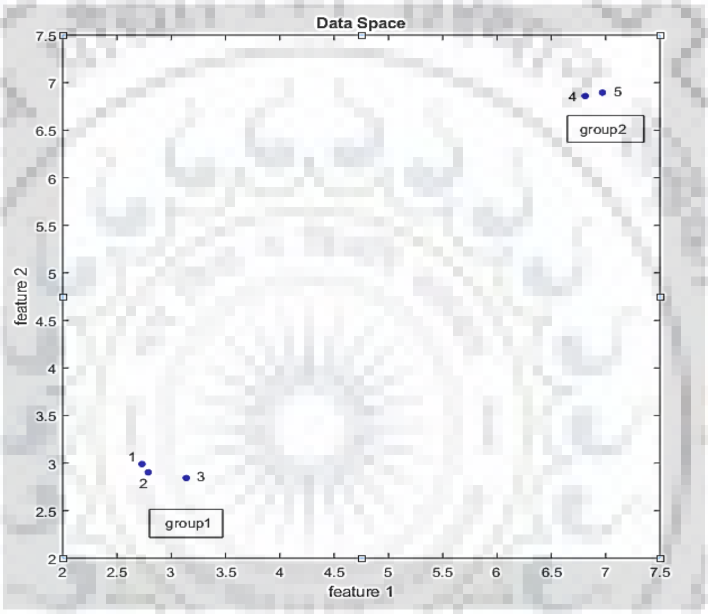


Fig. 3. 19 Groups of data points for spectral clustering

It can be visually identified in fig. 3.19 that f_i =1 for i = 1,2, 3 and f_i = 0 for i = 4,5.
With every graph, there is an associated weight matrix W, which gives information about their connectedness or closeness according to some property solely decided by the formula used to find the weights for the graph. The formula used to find the weights for the graph is called the kernel. It associates two different nodes of the graph by some real number.
Weights for this graph can be found from many available kernels such as,
$\varepsilon -$ neighborhood Kernel-

$$w_{ij} = \begin{cases} +1 \quad , \quad if\ \|\mathbb{x}_i - \mathbb{x}_j\| < \varepsilon \\ \ \ 0 \quad , \quad if\ \|\mathbb{x}_i - \mathbb{x}_j\| > \varepsilon \end{cases} \tag{3.85}$$

Gaussian Distance Kernel-

$$w_{ij} = e^{-\frac{\|\mathbb{x}_i - \mathbb{x}_j\|^2}{2\sigma^2}} \tag{3.86}$$

where σ is a parameter which can be varied to obtain the different degree of results[13]. Once the weight matrix is formed, the objective function should be defined such that it will be extremum at the correct solution which in this case is easily visible. It is to be

noted that the value of weights is maximum for the points in the same group while it is very low for two points which belong to different groups.
Hence according to Gaussian kernel and sigma 2, the weight matrix of the graph is

$$W = \begin{bmatrix} 1.0000 & 0.9930 & 0.9129 & 0.0015 & 0.0011 \\ 0.9930 & 1.0000 & 0.9311 & 0.0012 & 0.0009 \\ 0.9129 & 0.9311 & 1.0000 & 0.0003 & 0.0002 \\ 0.0015 & 0.0012 & 0.0002 & 1.0000 & 0.9318 \\ 0.0011 & 0.0009 & 0.0002 & 0.9318 & 1.0000 \end{bmatrix} \quad (3.87)$$

The objective function is,

$$F(\mathbb{v}) = \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}\,(v_i - v_j)^2 \quad (3.88)$$

It can be verified easily that this function will be minimum at the true solution, because at true solution $v = [1\ 1\ 1\ 0\ 0]$. So if i and j belong to the same cluster their difference will be zero, so the only terms which will be remaining in equation 3.88 will be the terms corresponding to the weights of the edges joining group1 to group2. But weights of such cross edges is very low as can be seen from equation 3.87 and hence F will be minimum at the real true solution. So for obtaining true solution F must be minimized.

$$F = w_{14} + w_{15} + w_{24} + w_{25} + w_{34} + w_{35}$$

$$F = 0.0015 + 0.0011 + 0.0012 + 0.0009 + 0.0003 + 0.0002 = 0.0052$$

In order to see this is minimum, suppose another solution different from the true solution, e.g., 1 and 2 in group1 and 3,4,5 in group2, then $v = [1\ 1\ 0\ 0\ 0]$.

$$F = w_{13} + w_{14} + w_{15} + w_{23} + w_{24} + w_{25}$$

$$F = 0.9129 + 0.0015 + 0.0011 + 0.9311 + 0.0012 + 0.0009 = 1.8447$$

Suppose another solution 1,2,3,4 in group1 and 5 in group2, then $v = [1\ 1\ 1\ 1\ 0]$ and

$$F = w_{15} + w_{25} + w_{35} + w_{45}$$

$$F = 0.0011 + 0.0009 + 0.0002 + 0.9318 = 0.9340$$

Clearly, the last two wrong solutions give a higher value F than the true solution. Hence for finding a true solution, minimization of F must be done. The function $F(\mathbb{v}) = \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}\,(v_i - v_j)^2$ can be represented in matrix form as follows,

$$F(\mathbb{v}) = \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}\,({v_i}^2 + {v_j}^2 - 2v_i v_j) \quad (3.89)$$

$$F(\mathbb{v}) = \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}{v_i}^2 + \sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}{v_j}^2 - 2\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}v_i v_j \quad (3.90)$$

Since i and j are independent of each other, the order of summation can be changed in the second summation because sum has to be taken over all values of i and j and the index with which sum is taken first is not important, the result remains the same.

$$F(\mathbb{v}) = \sum_{i=1}^{N} {v_i}^2\left(\sum_{j=1}^{N} w_{ij}\right) + \sum_{j=1}^{N}\sum_{i=1}^{N} w_{ij}{v_j}^2 - 2\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}v_i v_j \quad (3.91)$$

$$F(\mathbb{v}) = \sum_{i=1}^{N} {v_i}^2\left(\sum_{j=1}^{N} w_{ij}\right) + \sum_{j=1}^{N} {v_j}^2\left(\sum_{i=1}^{N} w_{ij}\right) - 2\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}v_i v_j \quad (3.92)$$

Since W matrix is symmetric, hence $w_{ij} = w_{ji}$ and substituting it in the second term will give,

$$F(\mathbb{v}) = \sum_{i=1}^{N} {v_i}^2 \left(\sum_{j=1}^{N} w_{ij} \right) + \sum_{j=1}^{N} {v_j}^2 \left(\sum_{i=1}^{N} w_{ji} \right) - 2 \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} v_i v_j \tag{3.93}$$

Now it is very easy to verify that $\sum_{i=1}^{N} {v_i}^2 \left(\sum_{j=1}^{N} w_{ij}\right) = \sum_{j=1}^{N} {v_j}^2 \left(\sum_{i=1}^{N} w_{ji}\right)$
Hence,

$$F(\mathbb{v}) = 2 \sum_{i=1}^{N} {v_i}^2 \left(\sum_{j=1}^{N} w_{ij} \right) - 2 \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} v_i v_j \tag{3.94}$$

$$\Rightarrow F(\mathbb{v}) = 2 \left[\sum_{i=1}^{N} {v_i}^2 \left(\sum_{j=1}^{N} w_{ij} \right) - \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} v_i v_j \right] \tag{3.95}$$

Discarding the factor of 2 as it won't affect optimization will lead to and defining $d_{ii} = \sum_{j=1}^{N} w_{ij}$ will reduce equation 3.95 to the following form,

$$\Rightarrow F(\mathbb{v}) = \sum_{i=1}^{N} d_{ii} {v_i}^2 - \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} v_i v_j \tag{3.96}$$

The second term is a quadratic form of the weight matrix W, i.e.,

$$v^T W v = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} v_i v_j \tag{3.97}$$

And the first is the quadratic from of the matrix $D = [d_{ij}]$, where

$$d_{ij} = \begin{cases} +d_{ii} \quad , & if \;\; i = j \\ \;\; 0 \quad , & if \;\; i \neq j \end{cases} \tag{3.98}$$

i.e., D is a diagonal matrix and

$$v^T D v = \sum_{i=1}^{N} \sum_{j=1}^{N} d_{ij} v_i v_j \qquad , given \; that \; i = j \tag{3.99}$$

$$\Rightarrow v^T D v = \sum_{i=1}^{N} d_{ii} v_i v_i = \sum_{i=1}^{N} d_{ii} {v_i}^2 \tag{3.100}$$

Hence Eq. 3.96 becomes

$$F(\mathbb{v}) = v^T D v - v^T W v \tag{3.101}$$
$$\Rightarrow F(\mathbb{v}) = v^T (D - W) v$$

Defining L = D – W as the laplacian matrix will give,

$$F(\mathbb{v}) = v^T L v \;\; , where \quad L = D - W \tag{3.102}$$

This is the objective function and it needs to be minimized.
So to apply spectral clustering to a graph, the objective function is defined through the laplacian matrix and it is minimized. To minimize it one needs to differentiate equation 3.102 with respect to all the components of v,

$$v^T L v = \sum_{i=1}^{N} \sum_{j=1}^{N} l_{ij} v_i v_j = \sum_{i=1}^{N} v_i \sum_{j=1}^{N} l_{ij} v_j \tag{3.103}$$

Differentiating 3.103 with respect to v_k,

$$\frac{d}{dv_k}(v^T L v) = \frac{d}{dv_k}\left(\sum_{i=1}^{N} v_i \sum_{j=1}^{N} l_{ij} v_j \right) = \frac{d}{dv_k}\left(\sum_{\substack{i=1 \\ i \neq k}}^{N} v_i \sum_{j=1}^{N} l_{ij} v_j + v_k \sum_{j=1}^{N} l_{kj} v_j \right) \tag{3.104(a)}$$

$$\frac{d}{dv_k}(v^T Lv) = \sum_{\substack{i=1\\i\neq k}}^{N} \frac{d}{dv_k}\left(v_i \sum_{j=1}^{N} l_{ij} v_j\right) + \frac{d}{dv_k}\left(v_k \sum_{j=1}^{N} l_{kj} v_j\right) \tag{3.105(b)}$$

From product rule of differentiation applied on the second term,

$$\begin{aligned}\frac{d}{dv_k}(v^T Lv) &= \sum_{\substack{i=1\\i\neq k}}^{N} v_i \frac{d}{dv_k}\left(\sum_{j=1}^{N} l_{ij} v_j\right) + \frac{d}{dv_k}(v_k)\sum_{j=1}^{N} l_{kj} v_j + v_k \frac{d}{dv_k}\sum_{j=1}^{N} l_{kj} v_j \\ &= \sum_{\substack{i=1\\i\neq k}}^{N} v_i \frac{d}{dv_k}\left(\sum_{j=1}^{N} l_{ij} v_j\right) + \sum_{j=1}^{N} l_{kj} v_j + v_k \frac{d}{dv_k}\sum_{\substack{j=1\\j\neq k}}^{N} l_{kj} v_j + v_k \frac{d}{dv_k}(l_{kk} v_k) \\ &= \sum_{\substack{i=1\\i\neq k}}^{N} v_i \frac{d}{dv_k}\left(\sum_{j=1}^{N} l_{ij} v_j\right) + \sum_{j=1}^{N} l_{kj} v_j + 0 + v_k l_{kk}\end{aligned} \tag{3.106}$$

$$\frac{d}{dv_k}(v^T Lv) = \left(\sum_{\substack{i=1\\i\neq k}}^{N} v_i\, l_{ik} \frac{d}{dv_k}(v_k) + v_k l_{kk}\right) + \frac{d}{dv_k}(v_k)\sum_{j=1}^{N} l_{kj} v_j \tag{3.107}$$

$$\frac{d}{dv_k}(v^T Lv) = \sum_{i=1}^{N} l_{ik} v_i + \sum_{j=1}^{N} l_{kj} v_j \tag{3.108}$$

Since L is a symmetric matrix, therefore $\sum_{i=1}^{N} l_{ik} v_i = \sum_{j=1}^{N} l_{kj} v_j$

$$\frac{d}{dv_k}(v^T Lv) = 2\sum_{i=1}^{N} l_{ki} v_i = 2\, L_{(k,:)}\, v \tag{3.109}$$

So the differentiation is equal to the linear combination of elements of k_{th} row of L matrix. This is found for every k = 1,2,3, …., N. When all these differentiations are arranged in a column matrix the resulting equation will lead to,

$$\frac{d}{dv}(v^T Lv) = 2\, L\, v \tag{3.110}$$

The solution has to satisfy the constraint $v^T v = 1$, which means that the vector is normalized. This is done so that the maximization and minimization will depend on eigenvalues only. This can be seen from the following,

The optimization problem is,

$$\underset{v}{argmin}(v^T Lv) \quad , subject\ to\ constraints\ v^T v = 1 \tag{3.111}$$

So the Lagrangian is,

$$\mathcal{L}(v) = v^T\, Lv - \lambda(v^T\, v - 1) \tag{3.112}$$

Differentiating it w.r.t v and λ will give

$$\frac{d}{dv}\mathcal{L}(v) = \frac{d}{dv} v^T\, Lv - \lambda \frac{d}{dv}(v^T I\, v) = 0 \tag{3.113}$$

$$\begin{aligned} 2\, Lv - 2\lambda v = 0 \quad &\Rightarrow\ Lv = \lambda v \\ v^T\, v - 1 = 0 \quad &\Rightarrow\ v^T\, v = 1 \end{aligned} \tag{3.114}$$

The solution of this equation are the eigenvectors of matrix L corresponding to eigenvalues λ of matrix L. There will be N eigenvectors.

Now to minimize $v^T\, Lv$, multiply both sides of equation 3.113 by v^T,

$$v^T\, Lv = \lambda(v^T\, v) = \lambda \tag{3.115}$$

Hence, to minimize $v^T Lv$, eigenvector corresponding to minimum eigenvalue should be chosen as the solution of equation 3.113. By visual inspection, the two groups can be identified.

For our example in fig. 3.17, degree matrix is,

$$D = \begin{bmatrix} 2.9086 & 0 & 0 & 0 & 0 \\ 0 & 2.9262 & 0 & 0 & 0 \\ 0 & 0 & 2.8445 & 0 & 0 \\ 0 & 0 & 0 & 1.9348 & 0 \\ 0 & 0 & 0 & 0 & 1.9339 \end{bmatrix} \quad (3.116)$$

and the Laplacian matrix is L = D – W,

$$L = \begin{bmatrix} 1.9086 & -0.9930 & -0.9129 & -0.0015 & -0.0011 \\ -0.9930 & 1.9262 & -0.9311 & -0.0012 & -0.0009 \\ -0.9129 & -0.9311 & 1.8445 & -0.0003 & -0.0002 \\ -0.0015 & -0.0012 & -0.0002 & 0.9348 & -0.9318 \\ -0.0011 & -0.0009 & -0.0002 & -0.9318 & 0.9339 \end{bmatrix} \quad (3.117)$$

The eigenvector matrix is (eige vectors are in columns)

$$v = \begin{bmatrix} 0.4472 & -0.3649 & -0.0000 & 0.4816 & -0.6595 \\ 0.4472 & -0.3650 & -0.0001 & 0.3306 & 0.7466 \\ 0.4472 & -0.3656 & 0.0003 & -0.8116 & -0.0872 \\ 0.4472 & 0.5475 & -0.7073 & -0.0004 & 0.0001 \\ 0.4472 & 0.5479 & 0.7069 & -0.0001 & 0.0000 \end{bmatrix} \quad (3.118)$$

The eigenvalues are

$$\lambda = [0.0000 \quad 0.0043 \quad 1.8661 \quad 2.7655 \quad 2.9121] \quad (3.119)$$

The least eigenvalue except zero is the second eigenvalue $\lambda = 0.0043$ and the corresponding eigenvector is the second column of v. This vector is

$$v1 = \begin{bmatrix} -0.3649 \\ -0.3650 \\ -0.3656 \\ 0.5475 \\ 0.5479 \end{bmatrix} \quad (3.120)$$

It can be seen from this vector that the values of nodes 1, 2 and 3 are nearly the same and negative. And the other two are positive. Hence 1,2 and 3 belongs to group1 and 4,5 belongs to group2, which is the true solution. If we let negative values to represent 1 and positive values to represent zeros, then the solution from v1 is

$$v2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (3.121)$$

Hence spectral clustering can solve the grouping problem and not only this, spectral clustering can also solve the problem of patterns, and fig. 3.21 and 3.23 provides two applications of it which k-means algorithm was not able to solve.

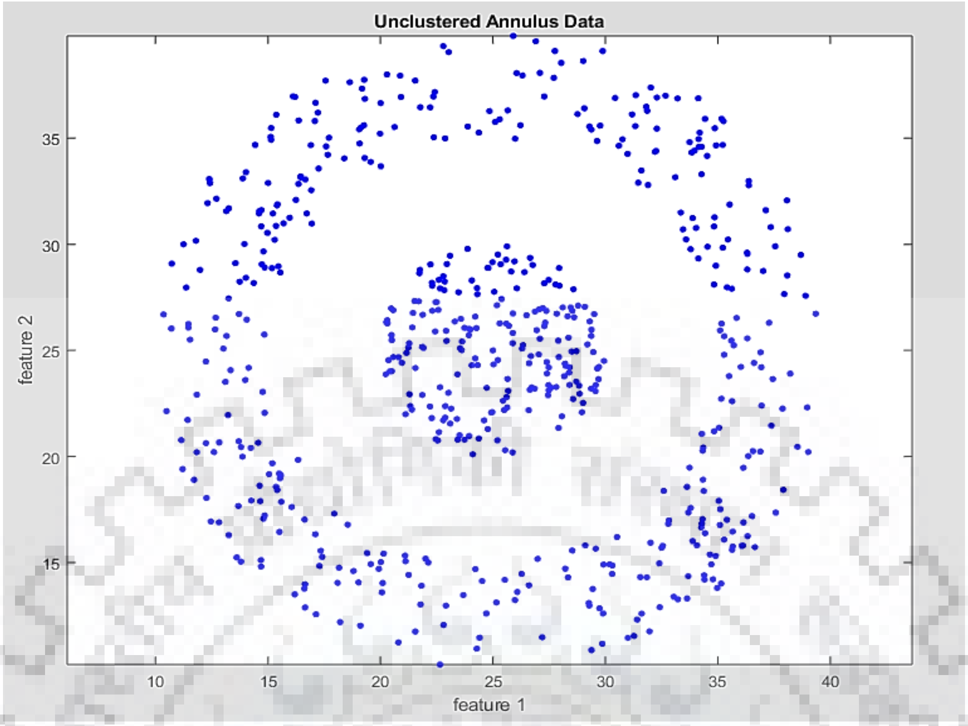


Fig. 3. 20 Original data of annular ring

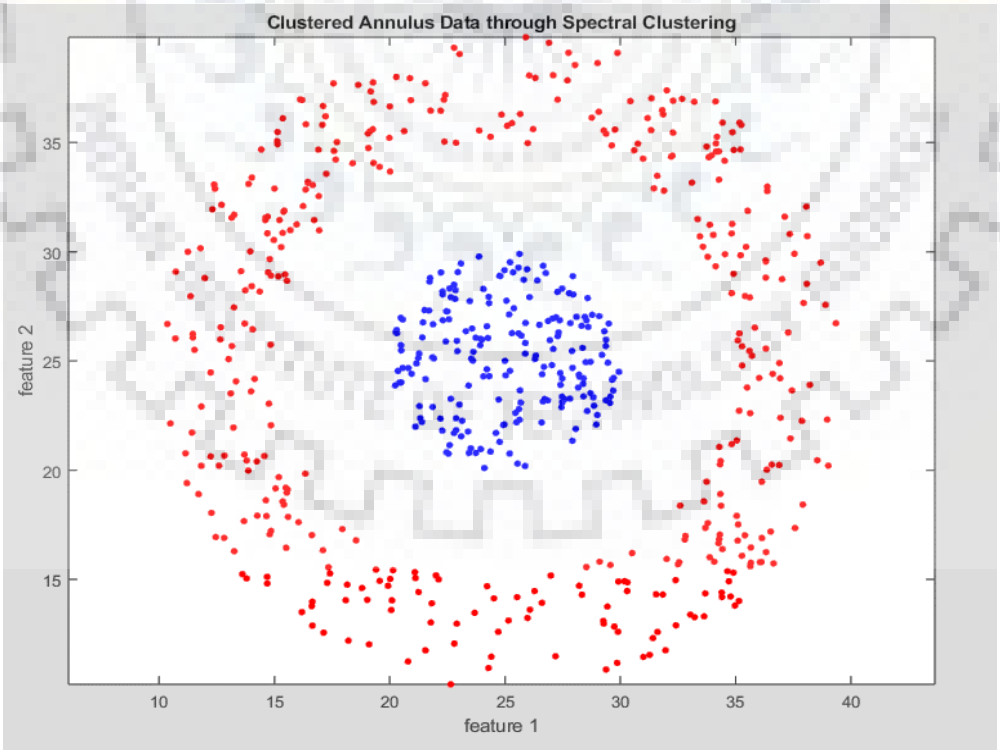


Fig. 3. 21Annulus data clustered through Spectral Clustering

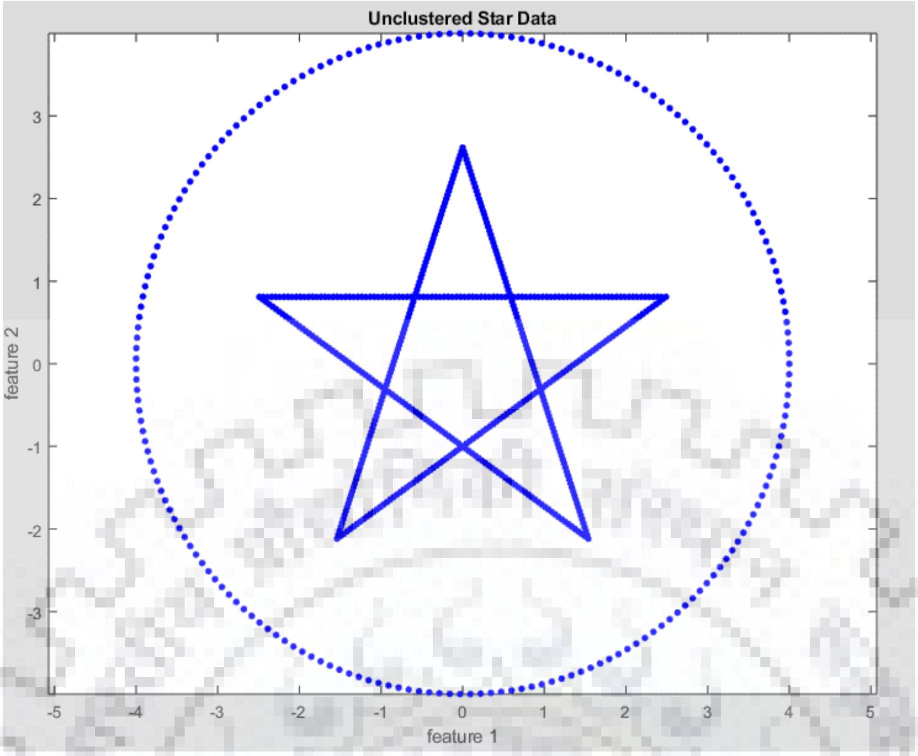


Fig. 3. 22 Unclustered Star data

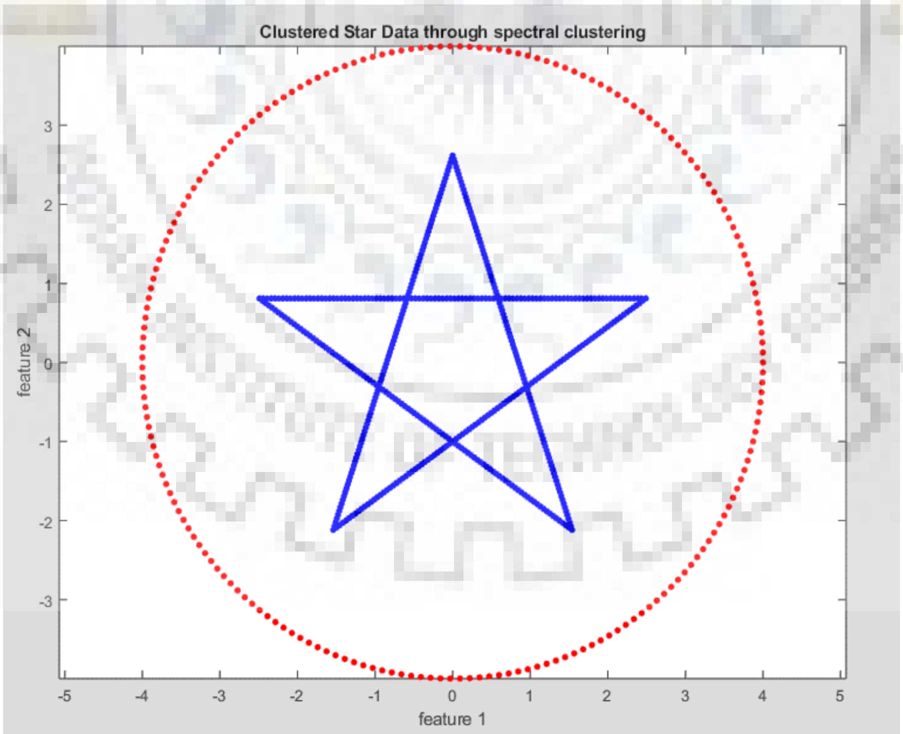


Fig. 3. 23 Clustered Star data through Spectral clustering

3.3 Part 3 – Finding Optimum cut for minimum power disruption

3.3.1 Constrained Spectral Clustering-

Since k- means algorithm can't be applied for different pattern groups another method of clustering which clusters based on characteristics different from nearness is employed. One such method is Spectral Clustering. In spectral clustering, a characteristic is selected to decide for the weights of the graph. In this method, certain constraints can also be satisfied besides grouping of nodes.

The Power system is represented in the form of the graph, and each bus corresponds to the node of the network, and each line represents an edge of the graph. The weights of the edge are decided on the basis of the characteristic which will help minimize some related characteristics. In this study, the line powers are taken to be weights[7]. This is because once the power system is clustered in the form of groups, some lines connecting those groups need to be cut. It is then required to minimize this disruption in the power flow in these lines. So such lines will be cut such that the sum of removed line powers is minimum.

In spectral clustering, constraints can be used to find the optimum solution. Here, Constraints are the information of coherency. In constrained spectral clustering, two matrices are required one is weight or adjacency matrix and other is constraint matrix[7], [14].

Let Constraint matrix be denoted by B, then B is obtained on the basis of coherency information which is known by applying previous methods of correlations to the power system speed and phase angle signals. This matrix is a square matrix having a dimension equal to the number of buses of the system. Two buses if present in the same area will be given a value of +1 and if present in different areas will be given a value of -1. So the i^{th} row and j^{th} column entry will be +1 if bus i and bus j are coherent and -1 otherwise[7]. If the buses are totally unrelated then they can be given a mutual value of zero.

Mathematically,

$$B = \left[b_{ij}\right]_{N\times N} \tag{3.122}$$

$$b_{ij} = \begin{cases} +1 \quad , & if\ i,j \in ML \\ -1 \quad , & if\ i,j \in CL \end{cases} \tag{3.123}$$

Here ML means a must link constraint and CL means a cannot link constraint. If the buses are coherent then they are said to belong to the set ML and to the set CL otherwise.

In this study, the weight matrix is defined on the basis of line power flows[7]. So the weight matrix is[15], [7]

$$A = \left[a_{ij}\right]_{N\times N} \tag{3.124}$$

$$a_{ij} = \begin{cases} \dfrac{|P_{ij}| + |P_{ji}|}{2} & ,if\ i \neq j \\ 0 & ,if\ i = j \end{cases} \tag{3.125}$$

There is a matrix associated with every graph called its Laplacian matrix whose spectra gives much information about the graph. This is denoted by L,

The degree matrix is obtained by

$$L = D - A \qquad , D\ is\ the\ degree\ matrix \tag{3.126}$$

$$D = [d_{ij}]_{N\times N} \tag{3.127}$$

$$d_{ij} = \begin{cases} 0 & , if\ i \neq j \\ d_{ii} & , if\ i = j \end{cases} \quad ; d_{ii} = \sum_{k=1}^{N} w_{ik} \tag{3.128}$$

Then Laplacian Matrix L is,

$$L = D - A = \begin{cases} -\dfrac{|P_{ij}| + |P_{ji}|}{2} & , if\ i \neq j \\ d_{ii} & , if\ i = j \end{cases} \tag{3.129}$$

Also, the volume, vol, of the graph is defined as

$$vol = \sum_{i=1}^{N} d_{ii} = \mathbf{1}^T D\, \mathbf{1} \tag{3.130}$$

where **1** is the vector containing all 1's. In simpler terms, vol is the sum of all the weights taken from both sides of the edges.

It is to be noted that whatever follows from here is the general spectral clustering method and the following points are not restricted for power system grid application only.

The normalized Laplacian and constraint matrices are defined by[7][14],

$$L' = D^{-1/2}\, L\, D^{-1/2}\, , B' = D^{-1/2}\, B\, D^{-1/2} \tag{3.131}$$

Spectral Clustering requires the minimization of an objective function formed by normalized Laplacian as,

$$\mathcal{L}(\mathbb{v}) = \mathbb{v}^T L' \mathbb{v} \tag{3.132}$$

and the constraints[14] are

$$\mathbb{v}^T B' \mathbb{v} > \alpha, \quad and\ \mathbb{v}^T \mathbb{v} = vol \tag{3.133}$$

The first constraint requires that constraint matrix should be satisfied to some minimum extent and α must be chosen accordingly so as to satisfy as much constraints as possible. The second constraint just normalizes the $\mathbb{v}$ so that they remain within some magnitude and not exceed indefinitely. Also, it will lead to feasible eigenvalue solutions.

So the complete optimization problem can be written as[16][14],

$$\underset{\mathbb{v}\in\ \mathbb{R}^N}{argmin}\ \mathbb{v}^T L' \mathbb{v} \quad subject\ to \quad \mathbb{v}^T B' \mathbb{v} > \alpha, \mathbb{v}^T \mathbb{v} = vol, \mathbb{v} \neq D^{1/2}\mathbf{1} \tag{3.134}$$

So the Lagrangian is[7][10],

$$\mathcal{L}(\mathbb{v}) = \mathbb{v}^T L'\mathbb{v} - \lambda(\mathbb{v}^T B'\mathbb{v} - \alpha) - \mu(\mathbb{v}^T\mathbb{v} - vol) \tag{3.135}$$

Differentiating this Lagrangian with respect to $\mathbb{v}$ gives

$$L'\mathbb{v} - \lambda\left(B' + \frac{\mu}{\lambda}I\right)\mathbb{v} = 0 \tag{3.136}$$

$$L'\mathbb{v} = \lambda\left(B' + \frac{\mu}{\lambda}I\right)\mathbb{v} \tag{3.137}$$

Also, the Kraush Kuhn Tucker conditions require [34],

$$\lambda(\mathbb{v}^T \boldsymbol{B}'\mathbb{v} - \alpha) \geq 0 \tag{3.138}$$

and

$$\mathbb{v}^T\mathbb{v} = vol \tag{3.139}$$

According to these equations the possibility of interest is $\lambda \neq 0$ and let $\lambda > 0$. It will be clear later why $\lambda > 0$ is required.

Also since $\lambda \neq 0$ let's select the boundary condition for a solution, i.e, converting inequality constraint into equality constraint for the boundary,

$$\mathbb{v}^T B'\mathbb{v} - \alpha = 0 \Rightarrow \mathbb{v}^T B'\mathbb{v} = \alpha$$

But the main point is to bound $\mathbb{v}^T B'\mathbb{v}$ by some constant or since now it is equal to α lower bounding $\mathbb{v}^T B'\mathbb{v}$ by some constant would be same as lower bounding α by some constant. Let β be a constant such that[10],

$$\beta = -\frac{\mu}{\lambda} vol \tag{3.140}$$

Then from equation (3.98) we get

$$L'\mathbb{v} = \lambda\left(B' - \frac{\beta}{vol}I\right)\mathbb{v} \tag{3.141}$$

Now it can be seen mathematically that the introduction of constant β causes the lower bounding of α as follows[14],

Multiplying both sides of equation 3.102 by $\mathbb{v}^T$

$$\mathbb{v}^T L'\mathbb{v} = \lambda\,\mathbb{v}^T\left(B' - \frac{\beta}{vol}I\right)\mathbb{v} = \lambda\left(\mathbb{v}^T B'\mathbb{v} - \frac{\beta}{vol}\mathbb{v}^T\mathbb{v}\right) \tag{3.142}$$

since $\mathbb{v}^T\mathbb{v} = vol$ eq. 3.103 reduces to

$$\mathbb{v}^T L'\mathbb{v} = \lambda\,(\mathbb{v}^T B'\mathbb{v} - \beta) \tag{3.143}$$

Since L is symmetric and positive definite, so is L', which can be shown as

- because $D^{-1/2}$ is the square root of a positive diagonal matrix, its entries will always be positive,
- the elements of $u = D^{-1/2}\mathbb{v}$ is the linear combination of elements of vector v,
- since L is positive definite $\mathbb{v}^T L\mathbb{v} > 0$ for any vector $\mathbb{v}$,
- now $L' = D^{-1/2}\, L\, D^{-1/2} = (D^{-1/2})^T\, L\, D^{-1/2}$ which means $\mathbb{v}^T L'\mathbb{v} = \mathbb{v}^T (D^{-1/2})^T\, L\, D^{-1/2}\mathbb{v}$
- This means $\mathbb{v}^T L'\mathbb{v} = (D^{-1/2}\mathbb{v})^T\, L\, D^{-1/2}\mathbb{v} = (u)^T\, L\, u$ >0 for any u.
- Hence L' is positive definite.

Using the above fact in eq. 3.104 requires

$$\lambda\,(\mathbb{v}^T B'\mathbb{v} - \beta) = \mathbb{v}^T L'\mathbb{v} > 0 \tag{3.144}$$

Now since λ (this was the reason to take it positive) is assumed greater than zero initially, the eq. 3.105 reduces to[14]

$$\mathbb{v}^T B'\mathbb{v} - \beta > 0 \tag{3.145}$$
$$\mathbb{v}^T B'\mathbb{v} > \beta$$

Hence the introduction of β leads to the lower bounding of the constraint, which was required. The constraint satisfaction is valid provided $\lambda > 0$.

Hence,

$$\lambda > 0 \qquad (important\ condition\ for\ the\ solution) \tag{3.146}$$

So now the original optimization problem gets reduced to,

Find $\mathbb{v}$ such that

$$L'\mathbb{v} = \lambda\left(B' - \frac{\beta}{vol}I\right)\mathbb{v} \quad for\ some\ specified\ \beta \tag{3.147}$$

This is a generalized eigenvalue problem which can be solved. But it is to be noted that only those solutions should be considered feasible whose eigenvalue is positive only. This is due to the fact that $\lambda > 0$ and hence it is a condition for the solution or for the feasible eigenvalues. So only those eigenvectors are chosen for which the generalized eigenvalues λ are positive.

After choosing feasible eigenvectors, the vector which will lead to the minimum value of objective function are the ones which have lowest eigenvalues out of all the feasible eigenvalues except the zero eigenvalue[14][7], [15], which was the trivial solution and hence to be neglected. If the objective function is to be maximized, then the eigenvectors corresponding to maximum positive eigenvalues should have been selected. Also, there may be many positive eigenvectors but the number of eigenvectors selected should be one less than the number of clusters required. So to minimize F and form k clusters at the same time, k-1 lowest positive non- zero eigenvalues must be selected.

There is one remaining constraint condition that needs to be satisfied and that is, $\mathbb{v}^T\mathbb{v} = vol$. For finding each of selected k-1 eigenvectors $\mathbb{v}$ are normalized according to,

$$\mathbb{v} \leftarrow \frac{\mathbb{v}}{\mathbb{v}^T \mathbb{v}} \; vol \tag{3.148}$$

Finally, to form k clusters, these k-1 eigenvectors are placed in the column of a data matrix. Suppose V denote the matrix of k-1 eigenvectors arranged in columns of V, then there are N rows in the V matrix because each of k-1 vectors has N components obeying to the fact that L has dimension N.

$$V = [\mathbb{v}_i]_{N \times k-1} \tag{3.149}$$

Now, the rows of this matrix need to be grouped and grouping can be done using k- means clustering as no pattern finding is required. So the N rows will be grouped into k clusters and this equivalently means that the N nodes of the graphs are grouped into k clusters according to the properties embedded in Laplacian matrix.

How to select β and k?

Now, there are two independent selection that needs to be made at the starting of using this method for clustering and these are- the number of clusters k, and the value of β. Here, the emphasis on the value of β should be given. β should be chosen such that the generalized eigenvalues remain positive[14]. From the structure of the right side of the equation it can be seen that it contains the term $B' - \frac{\beta}{vol} I$ which will directly determine the values of generalized eigenvalues. So in order for them to be positive the eigenvalues of $B' - \frac{\beta}{vol} I$ must be positive and hence,

$$eig\left(B' - \frac{\beta}{vol} I\right) = \lambda_{B'} - \frac{\beta}{vol} > 0 \tag{3.150}$$

$$\beta < \lambda_{B'} * \; vol \tag{3.151}$$

For this to be true,

$$\beta < (\lambda_{B'})_{max} * vol$$

But the constraint $\mathbb{v}^T \, B' \mathbb{v} > \; \beta$ needs to be satisfied so β should not be less than zero or very small value. So a good choice for selecting β is,

$$(\lambda_{B'})_{min} * vol \; < \beta < \; (\lambda_{B'})_{max} * vol \tag{3.152}$$

The more is the value of β in this range, the more constraints are satisfied. So beta should be chosen higher if constraint satisfaction is strongly desired. But increasing the value of β also reduces the feasibility of the eigenvectors obtained. Because the higher value of beta leads to more negative generalized eigenvalues. Hence less number of eigenvectors to select from[14].

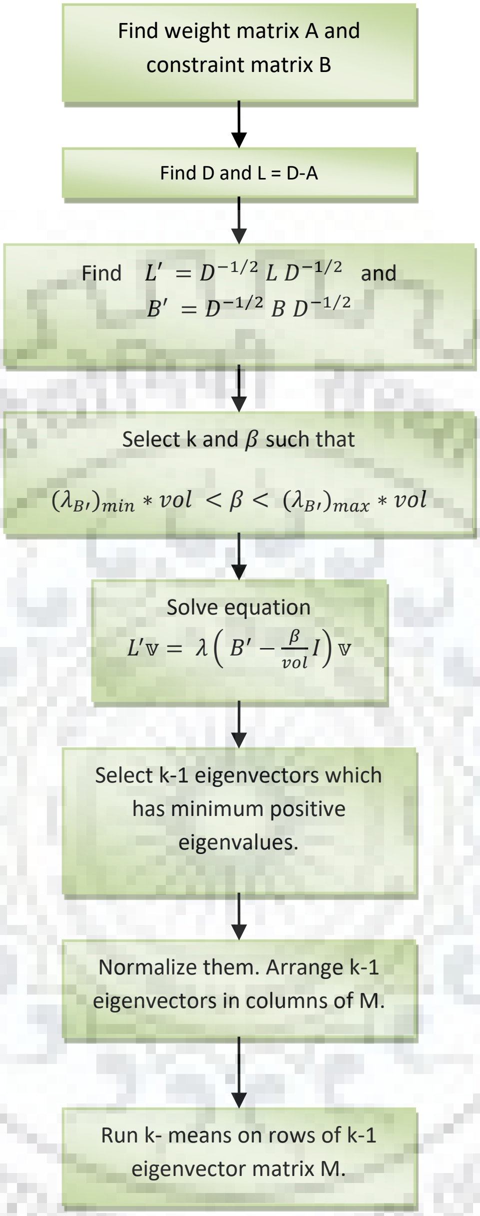
Selection of k can be done on the basis of k- means run for clustering the graph prior to the application of spectral clustering on the graph.

In summary, constrained spectral clustering methods involve the following steps[7], [14]-

a) Prepare Adjacency matrix or weight matrix,
b) Prepare Constraint matrix B,

c) Find laplacian matrix using L = D – W,
d) Find normalized Laplacian and constraint matrices,
e) Find eigenvalues of normalized constraint matrix, find the maximum and minimum eigenvalues.
f) Chose β such that $(\lambda_{B'})_{min} * vol < \beta < (\lambda_{B'})_{max} * vol$
g) Solve the generalized eigenvalue problem, and find the generalized eigenvectors of, $L'\mathbb{v} = \lambda\left(B' - \frac{\beta}{vol}I\right)\mathbb{v}$
h) Discard the non-feasible, i.e., negative eigenvalues and chose first minimum k-1 eigenvalues and arrange them in columns of matrix V,
i) Then apply k-means clustering on the rows of V,
j) The required indexes for every node is obtained, i.e., whether they belong to cluster 1, 2, …..or, k.

Flowchart 3.5- Flow Diagram for Constrained Spectral Clustering-



3.3.2 Application of Constrained Spectral clustering in finding N-cut to minimize power disruption-

Since coherency information about the power system is obtained through correlation coefficients earlier. Constrained spectral clustering can be employed for finding the normalized cut satisfying coherency constraint and minimizing power disruption. Also, the topology information needs to checked again and again for any fault and modification in the system during cascading outages. This helps in finding new coherency information correctly and also removing unnecessary data in the system.

Suppose if the line is removed from the system. Due to fault in the line, relays removed the line from the system, then there is unnecessary power flow data in the power flow weight matrix. Including such data can give a cut passing through the removed line, which is already removed and hence should not be considered for application of this method. Also, coherency can change due to the removal of a line. For example, a bus is connected to two areas A1 and A2 only through link L1 and L2. Earlier, it was coherent with A1, but due to the occurrence of a fault in link L1, it is removed by the switchgear and relays. Then now that same bus which was earlier coherent with the area A1 will now be coherent with area A2 as it is joined to only A2 now. Hence, the checking of fault location is a part working here independently for the overall application.

The second part is the Coherency determination which will provide the value of coherent areas k to the Constrained Spectral Clustering algorithm[7].

Then what is the need for Constrained Spectral Clustering algorithm? It is needed because the coherency determination doesn't take into account the minimization of power if all areas obtained through coherency information needs to be cut down for islanding to prevent a blackout. Constrained Spectral Clustering algorithm minimizes this power flow disruption between the areas, in addition to satisfying coherency of buses which is provided to this algorithm in the form of constraints[7]. Some coherency constraint might get relaxed in order to minimize power flow disruption.

In the case of cascading outage scenario, the operator has to manually run the algorithm which will first check the topology and information found will be sent to the coherency determination module. This module will provide the value of k to the spectral clustering algorithm which will decide which lines need to be cut. The constant topology check through fault location determination also secures the possibility in case the outages stops in between. Then the system becomes less vulnerable and the algorithm can be stopped.

3.3.3 Another way to determine Preliminary Coherency-

The coherency or coherent areas are calculated to find k to apply constrained spectral clustering to the system. Earlier in section, this was calculated using correlation coefficient, cross-correlation, Cross-spectral density and Coherence function. Another way to find is to spectrally cluster the system according to the dynamic coupling between the different buses[15]. The dynamic coupling between different buses is given by stiffness coefficients.

$$K_{ij} = \frac{\partial P_{ij}}{\partial \delta_{ij}} = \frac{V_i\, V_j}{Z_{ij}} \sin(\delta_i - \delta_i + \phi_{ij}) \qquad (3.153)$$

More dynamically coupled buses are stiffer together with each other and hence follow oscillations closely. This coefficient can be used for grouping power system buses into coherent groups [11]. To use stiffness coefficient as the property of classification, which is definitely is not a distance measure, k- means clustering can't be used as it strongly classifies on the basis of geometrical distance. But spectral clustering can be used by embedding the property of stiffness coefficient into the Laplacian matrix. Because the stiffness coefficient is a mutual characteristic, Laplacian can be formed with them.

The Laplacian matrix L_D employing stiffness coefficient is given by [11],

$$L_D = [l_D^{ij}]_{N\times N} \tag{3.154}$$

$$l_D^{ij} = \begin{cases} \dfrac{|K_{ij}| + |K_{ji}|}{2} & , if\ i \neq j \\ -\sum\limits_{\substack{m=1 \\ m\neq i}}^{N} l_D^{im} & , if\ i = j \end{cases} \tag{3.155}$$

After forming laplacian, spectral clustering is run to divide this system into two clusters. To do this[15][13],

a) Solve for eigenvalues and eigenvectors of L_D,
b) Choose two lowest magnitude eigenvalues and corresponding eigenvectors,
c) Arrange the eigenvectors in the columns of data matrix V,
d) Perform k-means on the rows of V with the value of k = 2,
e) If the clusters are satisfactory, then exit.
f) If more clusters are required, then apply the whole method to the two clusters obtained above in step (e) to further divide one of them or both of them into two more clusters. Repeat this process until a satisfactory number of clusters are obtained.

4. TEST RESULTS-

4.1 System Studied and Data-

The System studied is IEEE 9 bus system whose parameters are as follows-

Table 4. 1- Terminal conditions of IEEE 9- bus system

Bus	V[kV]	δ(deg)	P[pu]	Q[pu]
1	17.16	0.0000	0.7163	0.2791
2	18.45	9.3507	1.6300	0.0490
3	14.145	5.142	0.85	-0.1145

Table 4. 2- Transmission Line Characteristics of IEEE- 9 Bus System

Line		R[pu/m]	X[pu/m]	B[pu/m]
From Bus	To Bus			
4	5	0.01	0.168	0.176
4	6	0.0170	0.0920	0.1580
5	7	0.0320	0.1610	0.3060
6	9	0.0390	0.1738	0.3580
7	8	0.0085	0.0576	0.1490
8	9	0.0119	0.1008	0.2090

Table 4. 3- Load Characteristics of IEEE 9-bus System

Bus	P[pu]	Q[pu]
5	1.25	0.50
6	0.90	0.30
8	1.00	0.35

The fault is created at bus 8 of the system. The fault type is 3 phase to ground fault. It is applied at 0.35s and cleared at 0.4s. The line data is on base 100MVA, 230kV [21].

The shortest path database consists of previous node matrix and visited node matrix formed during the shortest path tree generation. The weight matrix for the shortest path algorithm is obtained by weighing each link with its line impedance in the positive sequence network. The transformer is replaced by its leakage reactance. Also, the weights are taken after multiplying the line reactances in per unit by 100, i.e., converting them to percentage. The generated weight matrix is shown on the next page.

The common velocity is found by the averaging of time of disturbance movement between different nodes. The fault causes the load angles to change as shown below and the time of arrival is calculated using these results.

The weight matrix for links of the graph of IEEE 9 bus system is,

$$weights = \begin{bmatrix} 0 & 0 & 0 & 48 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5.2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4.88 \\ 48 & 0 & 0 & 0 & 5.75 & 7.85 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5.75 & 0 & 0 & 13.79 & 0 & 0 \\ 0 & 0 & 0 & 7.85 & 0 & 0 & 0 & 0 & 14.99 \\ 0 & 5.2 & 0 & 0 & 13.79 & 0 & 0 & 4.87 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4.87 & 0 & 8.48 \\ 0 & 0 & 4.88 & 0 & 0 & 14.99 & 0 & 8.48 & 0 \end{bmatrix}$$

The previous node matrix is =

$$prev\ node = \begin{bmatrix} 1 & 7 & 9 & 1 & 4 & 4 & 5 & 7 & 6 \\ 4 & 2 & 9 & 5 & 7 & 4 & 2 & 7 & 8 \\ 4 & 7 & 3 & 6 & 7 & 9 & 8 & 9 & 3 \\ 4 & 7 & 9 & 4 & 4 & 4 & 5 & 7 & 6 \\ 4 & 7 & 9 & 5 & 5 & 4 & 5 & 7 & 8 \\ 4 & 7 & 9 & 6 & 4 & 6 & 5 & 9 & 6 \\ 4 & 7 & 9 & 5 & 7 & 4 & 7 & 7 & 8 \\ 4 & 7 & 9 & 5 & 7 & 9 & 8 & 8 & 8 \\ 4 & 7 & 9 & 6 & 7 & 9 & 9 & 9 & 9 \end{bmatrix}$$

The visited node matrix =

$$prev\ node = \begin{bmatrix} 1 & 4 & 5 & 6 & 7 & 9 & 8 & 2 & 3 \\ 2 & 7 & 8 & 9 & 5 & 3 & 4 & 3 & 1 \\ 3 & 9 & 8 & 7 & 6 & 2 & 4 & 5 & 1 \\ 4 & 5 & 6 & 7 & 9 & 8 & 2 & 3 & 1 \\ 5 & 4 & 6 & 7 & 8 & 2 & 9 & 3 & 1 \\ 6 & 4 & 5 & 9 & 3 & 8 & 7 & 2 & 1 \\ 7 & 8 & 2 & 9 & 5 & 3 & 4 & 6 & 1 \\ 8 & 7 & 9 & 2 & 3 & 5 & 6 & 4 & 1 \\ 9 & 3 & 8 & 7 & 6 & 2 & 4 & 5 & 1 \end{bmatrix}$$

For Optimal PMU algorithm top_data matrix is generated which tells the information about the end nodes of every link present in the system.

Optimal PMU suggests the PMU locations 2, 4, 9 are best for network observability. From these points, all the calculations for all the buses can be made.

The bus angles signals are first low pass filtered to remove any high-frequency changes so that their smooth variation can be seen. The filter cutoff frequency is selected to be 2Hz and a sampling rate of 50 samples is selected for Matlab built-in command.

Angle variations at PMU locations due to 3- phase fault at Bus 8-

Actual PMU bus Angles measured through PMUs during a fault on bus 8-

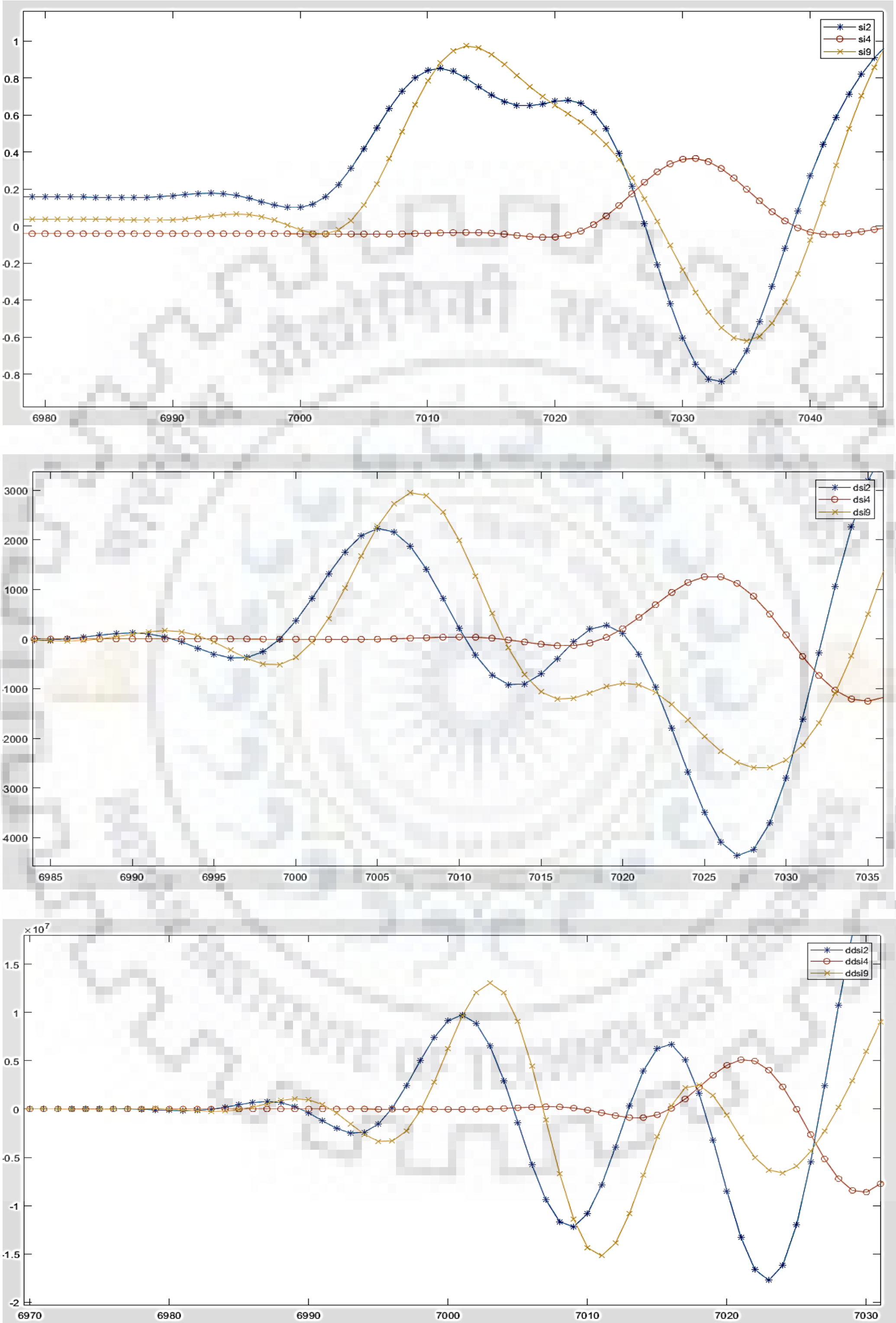


Fig. 4. 1 Bus angle(top), its derivative and double derivative measured from PMU

Bus 8 fault's PMU angles calculated through Hilbert transform-

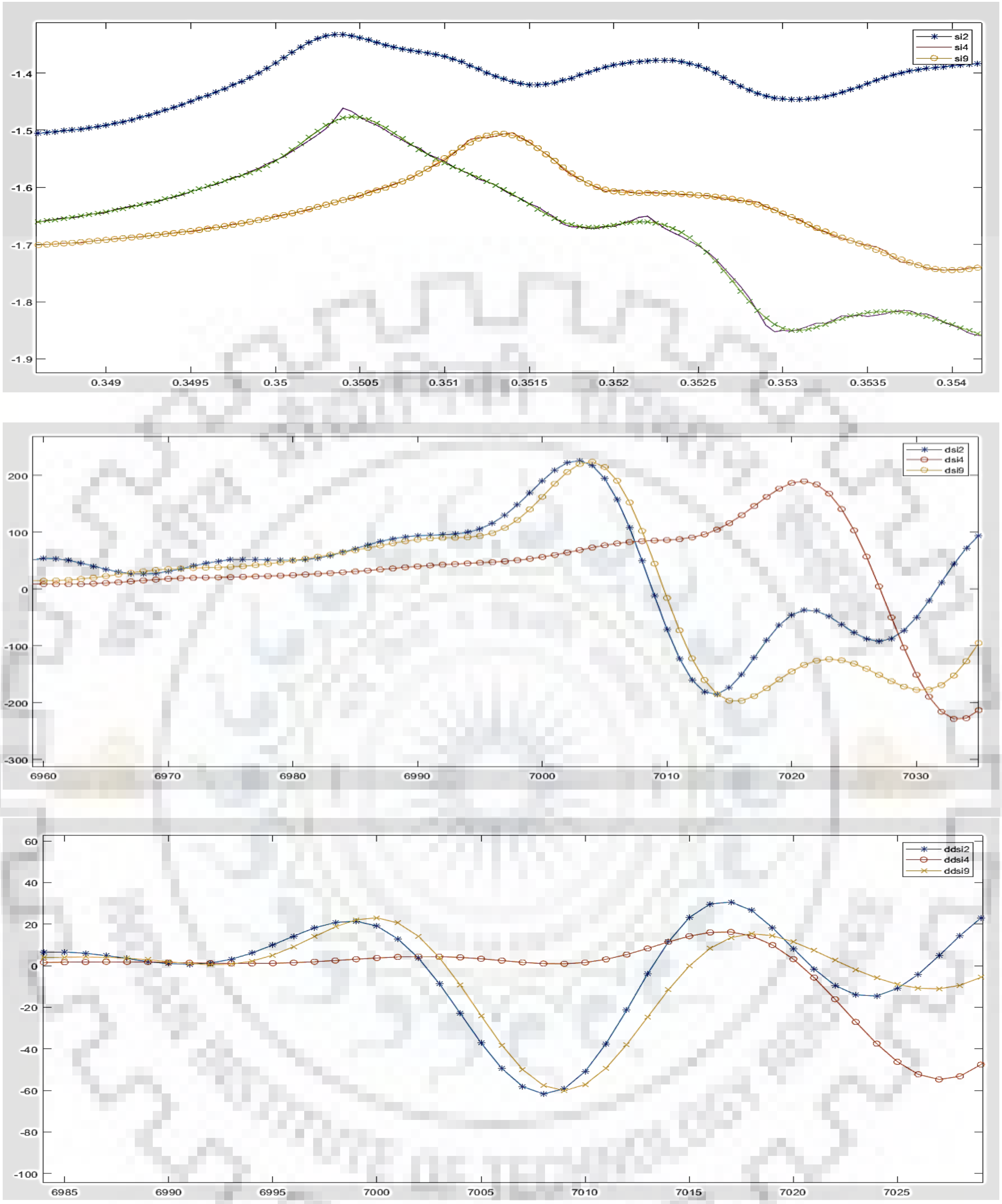


Fig. 4. 2 Bus angle(top), its derivative and double derivative calculated from voltages through Hilbert Transform

It can be observed that although the magnitude of the variation is a bit suppressed by Hilbert transform but the information about critical points is more or less preserved in this extraction. This can be seen by comparing previous curves which are actual data with these curves.

After running Matlab code on the data for finding T1, T2 and T3 points, the following data is obtained for an 8 bus system fault-

Table 4. 4– Measured Time data of critical points at node 4, node 2 and node 9

	T1(max. point)	T2(max.speed point)	T3(bifurcation point)
Node 2	0.3606 s	0.3503 s	0.3500 s
Node 4	0.3515 s	0.3513 s	0.3510 s
Node 9	0.3505 s	0.3503 s	0.3502 s

These results are used to calculate the time of arrival for each PMU node from eq.(31). These time of arrivals are –

Table 4. 5- Time of arrival of a disturbance at bus PMU locations

Node	Time of arrival
Node 2	0.3501
Node 4	0.3511
Node 9	0.3501

This toa is used to calculate T_M, the relative time of arrival matrix, the minimum time node is taken as node 9,

Table 4. 6- Relative time of arrival matrix at the PMU location w.r.t bus 9

Node	Relative time of arrival
T29	0.000
T49	0.00095

The generated path matrix for node 2, node 4 and node 9 is-

$$path2 = \begin{bmatrix} 14572 \\ 22000 \\ 39872 \\ 45720 \\ 57200 \\ 64572 \\ 72000 \\ 87200 \\ 98720 \end{bmatrix}; \quad path4 = \begin{bmatrix} 1400 \\ 2754 \\ 3964 \\ 4400 \\ 5400 \\ 6400 \\ 7540 \\ 8754 \\ 9640 \end{bmatrix}; \quad path9 = \begin{bmatrix} 1469 \\ 2789 \\ 3900 \\ 4690 \\ 5789 \\ 6900 \\ 7890 \\ 8900 \\ 9900 \end{bmatrix}$$

These matrices are used to calculate the theoretical time of arrival,

$$tp2 = \begin{bmatrix} 0.20 \\ 0.00 \\ 0.20 \\ 0.15 \\ 0.10 \\ 0.20 \\ 0.05 \\ 0.10 \\ 0.15 \end{bmatrix} ms; \quad tp4 = \begin{bmatrix} 0.05 \\ 0.15 \\ 0.15 \\ 0.00 \\ 0.05 \\ 0.05 \\ 0.10 \\ 0.15 \\ 0.10 \end{bmatrix} ms; \quad tp9 = \begin{bmatrix} 0.15 \\ 0.15 \\ 0.05 \\ 0.10 \\ 0.15 \\ 0.05 \\ 0.10 \\ 0.05 \\ 0.00 \end{bmatrix} ms$$

Then, the theoretical relative time of arrival is calculated by,

$$Tth = [tp4 - tp9, tp2 - tp9]$$

Then,

$$\Delta_k = \|Tth_k - T_M\|^2$$

The value of k for which Δ_k is minimum is the solution.

4.2 Determination of coherency and coherent areas-

The data of the system studied is given in Appendix A. The IEEE standard 9- bus system is fed with two alternators at bus 2 and bus 3 and an infinite bus at the bus no. 1. The disturbance is applied to the system on bus 6. This disturbance is produced by changing the phase angle of the 6th bus voltage instantly from 0 to π. The turbine model used is Cv – Iv model and the governor used is Mechanical Hydraulic type of governor. The simulation is done in PSCAD and for processing and programming, MATLAB is used. The run time is 60s and the sampling period is 100us.

The data of the system run is given in table 4.1 to 4.3, and Fig. 4.3 shows the variation of speed signals of the two synchronous generators. Also, the phase angle of every bus is shown in fig.4.4.

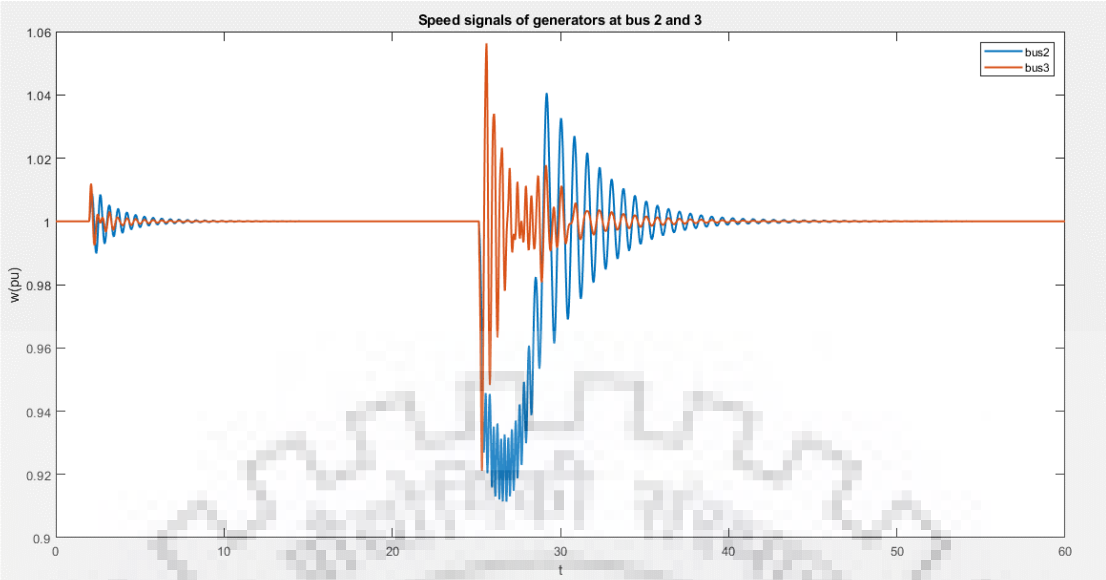


Fig. 4. 3 Speed Signals of alternators at bus 2 and bus 3

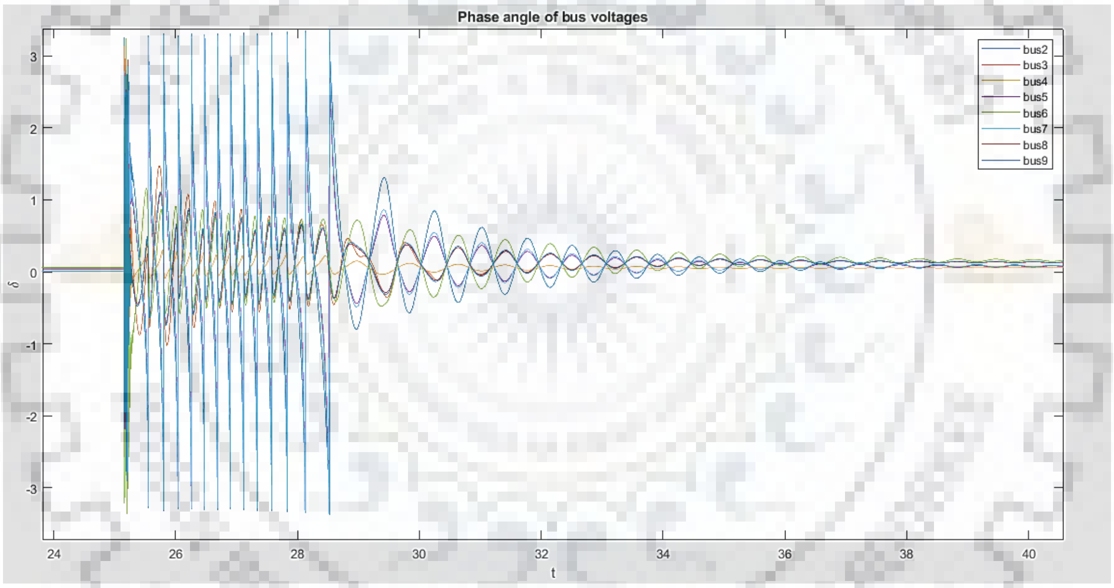


Fig. 4. 4 Load angles of buses

The correlation coefficient between load angles of all buses in the system is as follows,

$$R_{ij} = \begin{bmatrix} 1.0 & -0.47 & -0.52 & 0.92 & -0.67 & 0.94 & -0.56 & -0.58 \\ -0.47 & 1.0 & 0.50 & -0.38 & 0.51 & -0.39 & 0.91 & 0.90 \\ -0.52 & 0.50 & 1.0 & -0.45 & 0.82 & -0.46 & 0.69 & 0.69 \\ 0.92 & -0.38 & -0.45 & 1.0 & -0.56 & 0.99 & -0.47 & -0.48 \\ -0.67 & 0.51 & 0.82 & -0.56 & 1.0 & -0.58 & 0.69 & 0.71 \\ 0.94 & -0.39 & -0.46 & 0.99 & -0.58 & 1.0 & -0.48 & -0.49 \\ -0.56 & 0.91 & 0.69 & -0.47 & 0.69 & -0.48 & 1.0 & 0.99 \\ -0.58 & 0.90 & 0.69 & -0.48 & 0.71 & -0.49 & 0.99 & 1.0 \end{bmatrix}$$

From this matrix, a coherency table can be prepared. The correlation coefficient matrix consists of rows and columns representing bus 2 to bus 9.

Table 4. 7- Coherency check from the correlation coefficient

S.no.	Correlated buses
Bus 2	(5,7) have high values
Bus 3	(8,9) have high values
Bus 4	6 has high value but 3,8,9 seems to be more related to it than 2,5,7
Bus 5	(2,7) have high values
Bus 6	(4) has high value, but 3,8,9 seems to be more related to it than (2,5,7)
Bus 7	(2,5) have high values
Bus 8	(3,9) have high values
Bus 9	(3,8) have high values

From this table, it seems there can be two groups (2,5,7) and (3,8,9,1,4,6) if two clusters are to be chosen. While there can be three groups (2,5,7), (3,8,9) and (1,4,6) if three clusters need to be chosen.

Coherency through correlation is evaluated according to equation 3.65 and equation 3.66 with speed replaced by load angle of buses.

The correlation functions are evaluated and their trend is shown in fig. 4.5 and finally the cross-spectral density and coherence functions are shown in fig. 4.6 and 4.7 respectively. The cross-correlation functions show that buses 2,5,7 have functions whose shape is the same. While 3,8,9,6 follow each other closely in shape. A similar trend can be seen in Cross-spectral density functions which tells that the composition of frequencies (low-frequency range) of buses (2,5,7) are nearly same and so is the composition of frequencies of buses (3,8,9,6).

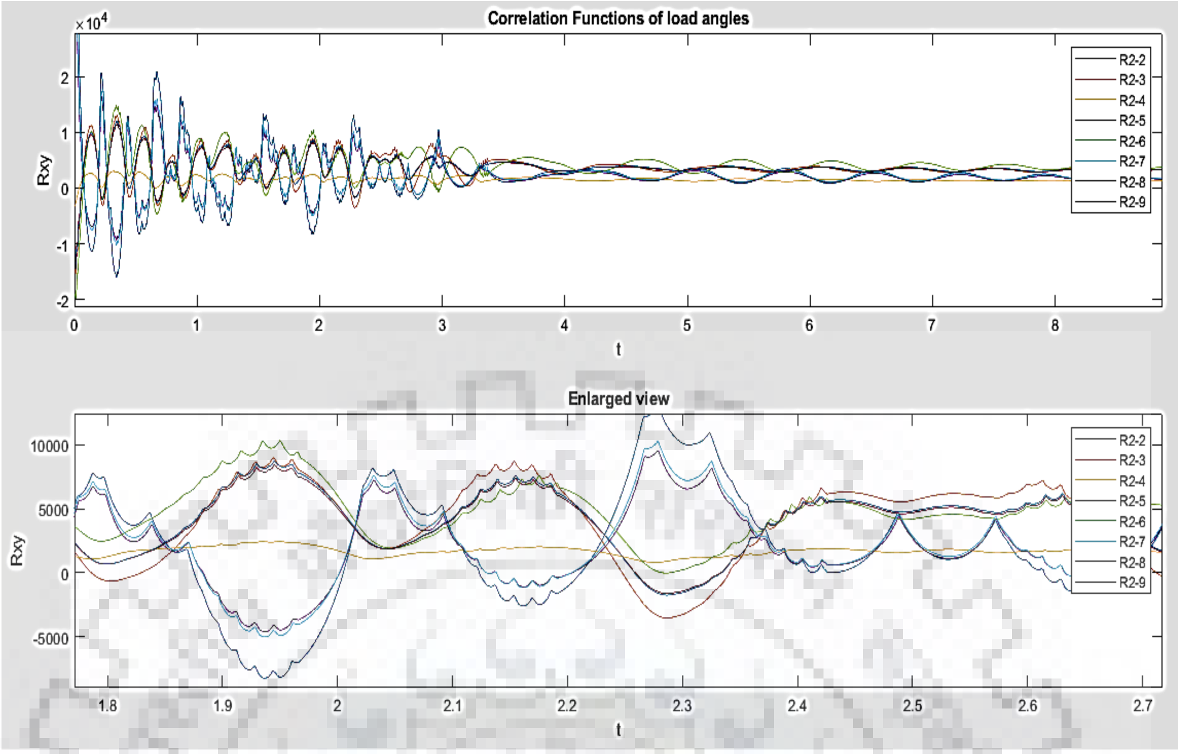


Fig. 4. 5 Cross- Correlation functions of load angles of buses.

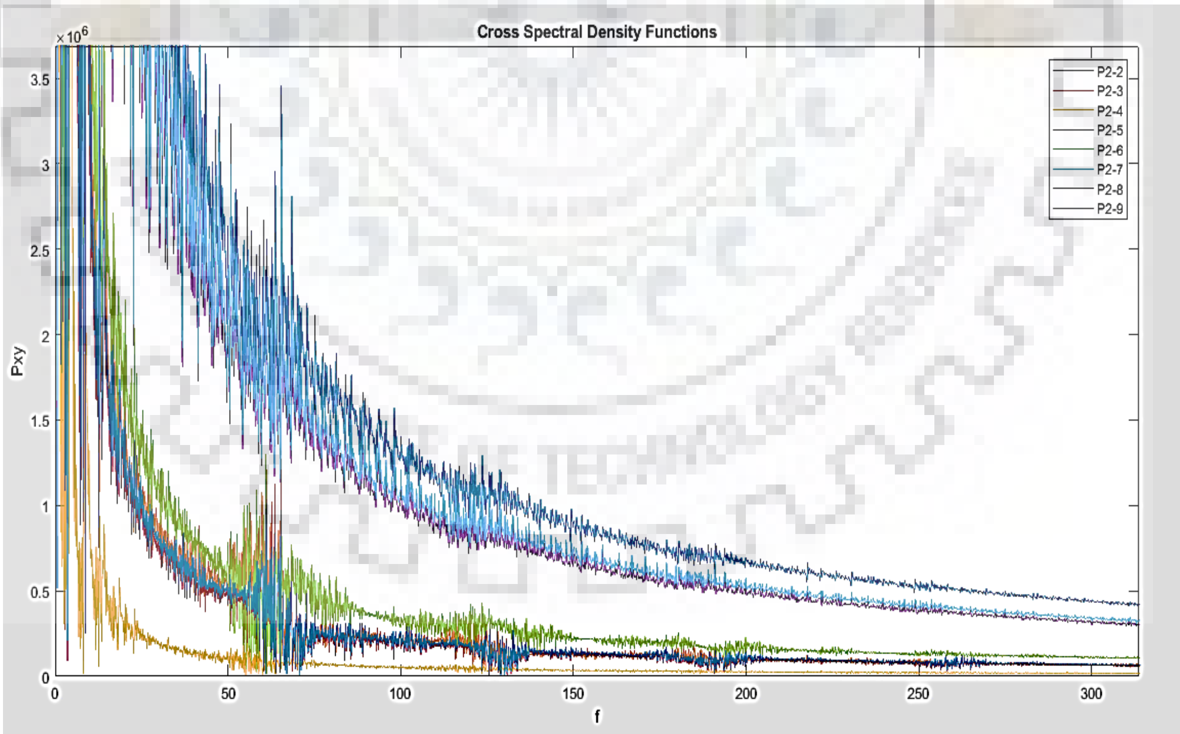


Fig. 4. 6 Power Spectral Density of load angle of buses

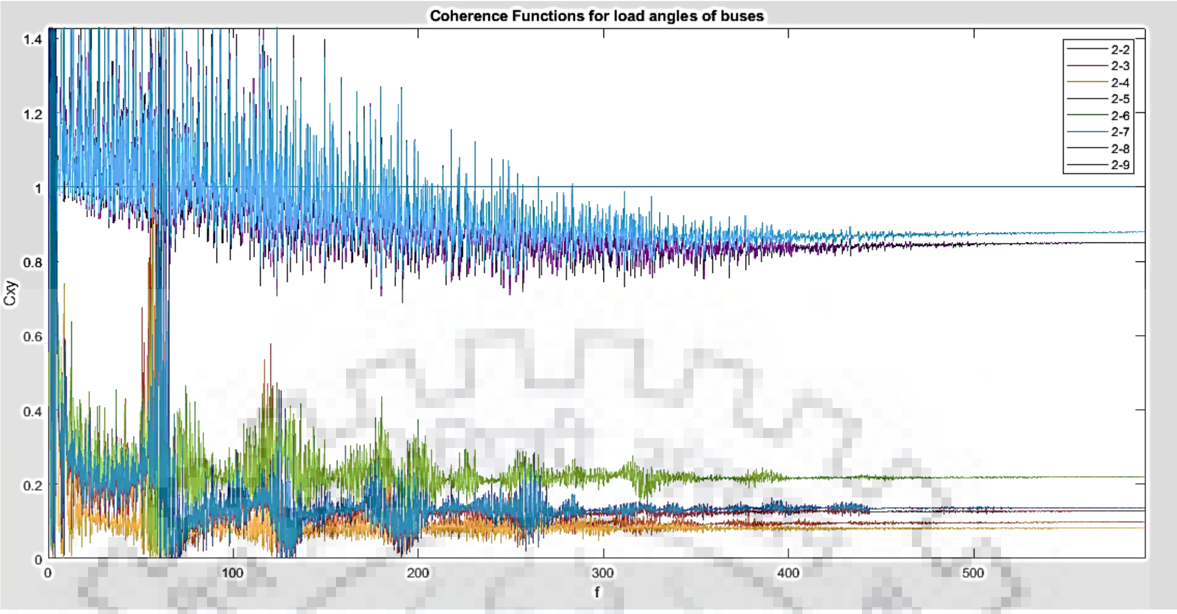


Fig. 4. 7 Coherence functions calculated with respect to bus 2

It can be seen from coherence functions that buses 2, 5 and 7 are more related to each other than buses 1, 4, 7, 8, 9.

But these coherence functions are functions of bus 2,3,4,5,6,7,8,9 with respect to bus 2. There can be coherence functions with respect to bus 3, 4, 5, 6, 7, 8, 9. For determining which buses are near to each other these coherence functions can be k-means clustered.

The clustering result is

Table 4. 8- K- means cluster obtained through coherence functions

S.no.	2 Cluster	3 Cluster
Bus 2	(2,5,7); (1,3,4,6,8,9)	(2,5,7); (3,4,8,9); (6)
Bus 3	(3,8,9);(1,2,4,5,6,7)	(2,5,7); (3,8,9); (1,4,6)
Bus 4	(2,5,7); (1,3,4,6,8,9)	(2,5,7); (1,3,4,6,8,9)
Bus 5	(2,5,7); (1,3,4,6,8,9)	(5,7); (1,3,4,6,8,9);(2)
Bus 6	(2,3,5,7); (1,4,6,8,9)	(2,5,7); (3,8,9); (1,4,6)
Bus 7	(2,5,7); (1,3,4,6,8,9)	(2,5,7); (1,3,4,8,9);(6)
Bus 8	(2,5,7); (1,3,4,6,8,9)	(2,5,7); (3,8,9); (1,4,6)
Bus 9	(2,5,7); (1,3,4,6,8,9)	(2,5,7); (3,8,9); (1,4,6)

From this table, it can be strongly stated that the buses which are coherent are 2,5,7 and 3,8,9. Buses 1,4,6 change groups in three cluster groups coherence functions calculated w.r.t to different buses. Also, groups can be identified by looking not only in every cell but also in different rows. The rows involving bus 2, bus 5 and bus 7 gives nearly the same result and hence can be said to strongly belong to one of the groups. Similarly, rows involving bus 3, bus 8 and bus 9 also have nearly the same group cells in third column. In 2 cluster, i.e., the second column, the grouping seems to be much stiffer. So k = 2 should be the used for spectral clustering.

4.3 Coherency Determined through Dynamic Constraints-

The Laplacian matrix is formed according to equation 3.154 of section 3. These use stiffness coefficients as their elements. The Laplacian matrix is

$$L_D = \begin{bmatrix} -31.15 & 0 & 0 & 31.15 & 0 & 0 & 0 & 0 & 0 \\ 0 & -49.72 & 0 & 0 & 0 & 0 & 49.72 & 0 & 0 \\ 0 & 0 & -57.39 & 0 & 0 & 0 & 0 & 0 & 57.39 \\ 31.15 & 0 & 0 & -49.63 & 8.91 & 9.57 & 0 & 0 & 0 \\ 0 & 0 & 0 & 8.91 & -119.19 & 0 & 110.28 & 0 & 0 \\ 0 & 0 & 0 & 9.57 & 0 & -40.56 & 0 & 0 & 30.99 \\ 0 & 49.72 & 0 & 0 & 110.28 & 0 & -171.35 & 11.34 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 11.34 & -307.84 & 296.49 \\ 0 & 0 & 57.39 & 0 & 0 & 30.99 & 0 & 296.49 & -384.88 \end{bmatrix}$$

Its eigenvalues are,

$$eig(L_D) = [-649.52 \quad -266.03 \quad -92.43 \quad -76.28 \quad -66.15 \quad -43.62 \quad -9.32 \quad -8.36 \quad 0.00]$$

The two lowest magnitude non zero eigenvalues are -8.36 and -9.32. Picking the eigenvectors associated with these two eigenvalues and arranging them in matrix V gives,

$$V = \begin{bmatrix} -0.5800 & -0.4109 \\ 0.4600 & -0.2741 \\ 0.4600 & 0.4436 \\ -0.4000 & 0.4436 \\ 0.3400 & -0.2511 \\ -0.1400 & 0.2755 \\ 0.3700 & -0.2280 \\ -0.0064 & 0.3666 \\ -0.0209 & 0.3790 \end{bmatrix}$$

This matrix can be considered to be 9 points in two-dimensional space, where each vector or data point has a dimension of 2. Then this matrix can be plotted in the x- y Cartesian plane for visualization of grouping. The graph showing the points is shown in fig. 4.6. These points can clearly be seen to agree with the analytical results. The points are designated by the respective row number in which they are present in V matrix or, equivalently by the node number to which these rows points. In other words, the points present in the graph indirectly represent the buses. The encircled groups are cluster 1(on the lower right corner) and cluster 2 of Table 4.9.

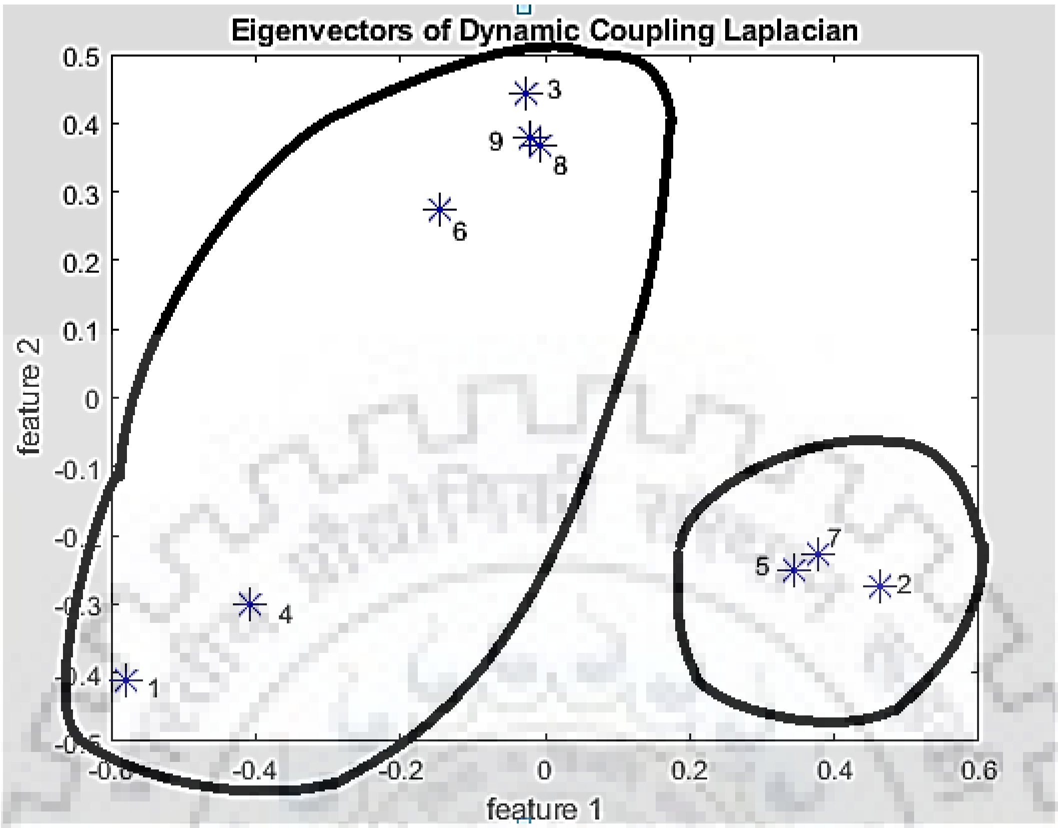


Fig. 4. 8 Points plotted through components of two eigenvectors

Clustering its rows, i.e., k-means is run on the rows of V considering one row as one data point gives the following two clusters. K-means can be run on MATLAB via built-in command

```
kmeans(V, k, 'replilcates', 5);
```

It will return a vector whose row represents a node in the power system, and value in each row is the cluster number to which it belongs.

$$id = [2 \quad 1 \quad 2 \quad 2 \quad 1 \quad 2 \quad 1 \quad 2 \quad 2]^T$$

Table 4. 9- Clustered group using dynamic coupling

Cluster 1	(2, 5, 7)
Cluster 2	(1, 3, 4, 6, 8, 9)

Now k is known for this system and that is k = 2.

For spectral clustering, first power flow data is used to obtain power flow in each of the lines/transformers connecting two buses after the outage in steady state is taken and stored in Pij matrix.

Then weight matrix is formed using the equation 3.124 and then laplacian is calculated and it comes out to be,

$$L = \begin{bmatrix} 1.674 & 0 & 0 & -1.674 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.0865 & 0 & 0 & 0 & 0 & -0.0865 & 0 & 0 \\ 0 & 0 & 0.0923 & 0 & 0 & 0 & 0 & 0 & -0.0923 \\ -1.674 & 0 & 0 & 3.5012 & -0.408 & -1.419 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.408 & 1.0238 & 0 & -0.6150 & 0 & 0 \\ 0 & 0 & 0 & -1.419 & 0 & 2.3710 & 0 & 0 & -0.9518 \\ 0 & -0.0865 & 0 & 0 & -0.615 & 0 & 1.4316 & -0.7294 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.7294 & 1.5881 & -0.8587 \\ 0 & 0 & -0.0923 & 0 & 0 & -0.951 & 0 & -0.8587 & 1.9028 \end{bmatrix}$$

From the cluster indicator matrix id constraint matrix B should be formed according to equation 3.122. Hence,

$$b_{ij} = \begin{cases} +1 & , \quad if\ (i,j) \in [2,5,7] \times [2,5,7] \\ +1\ , & if\ \ (i,j) \in [3,8,9,1,4,6] \times [3,8,9,1,4,6] \\ -1 & , \quad if\ (i,j) \in [2,5,7] \times [3,8,9,1,4,6] \end{cases}$$

After normalization of L and B matrices, the maximum eigenvalue of B is calculated, which is $(\lambda_{B'})_{max} = 9$.

Taking $\beta = 100(< (\lambda_{B'})_{max} * vol\ )$ and finding the volume by Laplacian matrix as

$$vol = sum\ of\ diagonal\ elements\ of\ L = 13.6712$$

The generalized eigenvalue problem in equation 3.146 can be solved and its eigenvalues are

$$gen\ eig = [\,-0.273 \quad -0.238 \quad -0.198 \quad 0.035 \quad 0 \quad -0.037 \quad -0.0873 \quad -0.136 \quad -0.136]$$

There are only two clusters are required, so one eigenvector is needed to decide for the clusters. Now the eigenvalue corresponding to which the eigenvector will be selected must be minimum among the non-zero positive eigenvalues. There is only one non-zero positive eigenvalue which is the fourth element of the $gen\ eig$ matrix. The eigenvector corresponding to the fourth eigenvalue is

$$V = \begin{bmatrix} 0.5778 \\ -0.9742 \\ 1.0000 \\ 0.6817 \\ -0.3047 \\ 0.7061 \\ -0.3981 \\ 0.3524 \\ 0.7550 \end{bmatrix}$$

It can directly be seen from this matrix or k-means can be applied on the rows of V to know that final clusters are (2,5,7) and (1,3,4,6,8,9). The direct visualization is the recognition of the fact that only (2,5,7) rows have negative values and all other are positive values. Nevertheless, k-means can be applied on V and it returns the id matrix and that is,

$$id = [2 \quad 1 \quad 2 \quad 2 \quad 1 \quad 2 \quad 1 \quad 2 \quad 2]^T$$

So finally, the power disruption turns out to be minimum if line 7-8 and line 4-5 is cut in case of emergency satisfying coherency constraints too.

Test result on 14 bus system-

The same analysis is performed on a 14 bus system with the use of dynamic stiffness coefficients. The disturbance is the removal of line TL6-12 with the help of breaker. The whole 14 bus system is operating on 18kV. The simulation is done in RSCAD and the analysis is carried out. There is one difference in the model available in RSCAD simulation software from the standard IEEE test system that it doesn't contain the 7th bus. Instead, the 4th, 7th, and 9th buses are connected via a three-winding transformer. So bus 7 voltage needs to be calculated using the impedance used for three- winding transformer. It is as follows-

Z_{12}, Z_{13}, and Z_{23} are given, so individual winding resistances can be calculated as,

$$Z_1 = \frac{Z_{12} + Z_{13} - Z_{23}}{2}; \tag{4.1}$$

$$Z_2 = \frac{Z_{12} + Z_{23} - Z_{13}}{2} \tag{4.2}$$

$$Z_3 = \frac{Z_{13} + Z_{23} - Z_{12}}{2} \tag{4.3}$$

After calculating these impedances, V_7 can be easily calculated as,

$$V_7 = V_9 - I_{97} * Z_1 = V_4 - I_{47} * Z_2 = V_8 - I_{87} * Z_3 \tag{4.4}$$

After the fault, the voltage profile, phase angles, and speeds are as shown in figure below,

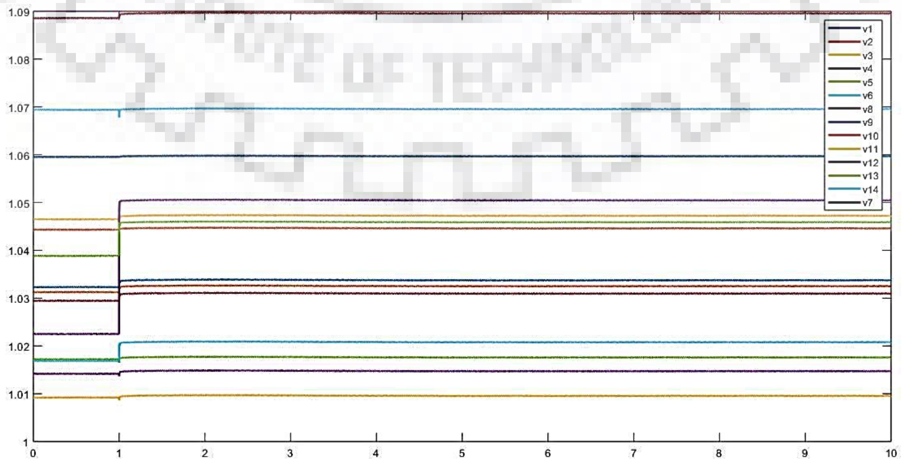


Fig. 4. 9- Voltages of buses after line outage

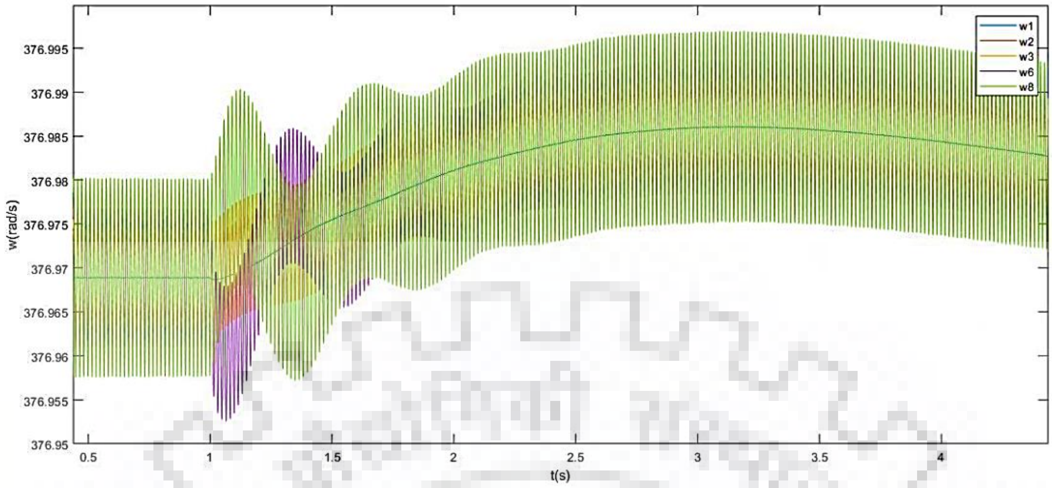


Fig. 4. 10- Speed variation of generators after the outage

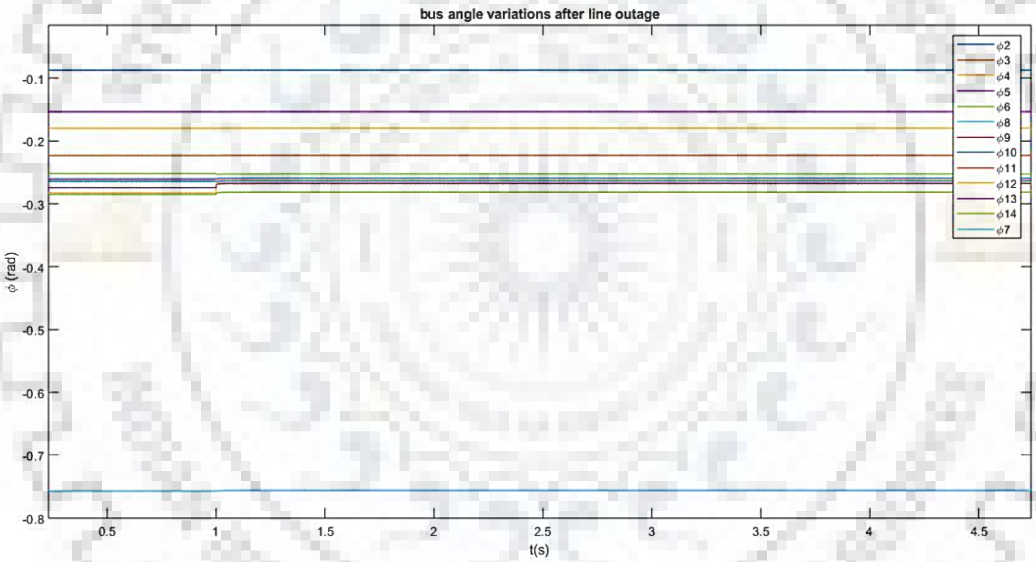


Fig. 4. 11- Bus angle variations after line outage

This data along with line data in Appendix 6.2 is used with eqn. 3.152 and 3.154 in forming the L_d matrix which is shown below as-

$$L_d = \begin{bmatrix} -21.3 & 16.8 & 0 & 0 & 4.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 16.8 & -32.9 & 5.1 & 5.3 & 5.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5.1 & -10.5 & 5.4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5.3 & 5.4 & -39.8 & 22.2 & 0 & 5 & 0 & 1.9 & 0 & 0 & 0 & 0 & 0 \\ 4.5 & 5.5 & 0 & 22.2 & -36.3 & 4.1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4.1 & -18.3 & 0 & 0 & 0 & 0 & 4.4 & 6.4 & 3.5 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 & -20.6 & 5.9 & 9.7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5.9 & -5.9 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.9 & 0 & 0 & 9.7 & 0 & -25.8 & 11.1 & 0 & 0 & 0 & 3.2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 11.1 & -15.8 & 4.8 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4.4 & 0 & 0 & 0 & 4.8 & -9.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6.4 & 0 & 0 & 0 & 0 & 0 & -5.8 & 2.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3.5 & 0 & 0 & 0 & 0 & 0 & 2.5 & -11.4 & 2.5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3.2 & 0 & 0 & 0 & 2.5 & -5.6 \end{bmatrix}$$

The lowest magnitude eigenvectors of it are,

$$eig(Ld) = \begin{bmatrix} -0.25 & 0.32 \\ -0.23 & 0.32 \\ -0.24 & 0.33 \\ -0.11 & 0.22 \\ -0.17 & 0.21 \\ -0.16 & -0.25 \\ 0.31 & 0.10 \\ 0.59 & 0.16 \\ 0.26 & -0.01 \\ 0.25 & -0.08 \\ 0.08 & -0.21 \\ -0.31 & -0.47 \\ -0.16 & -0.37 \\ 0.15 & -0.27 \end{bmatrix}$$

When plotted in two-dimensional plane and considering each row as coordinates of one point, the 14 buses can be visualized as,

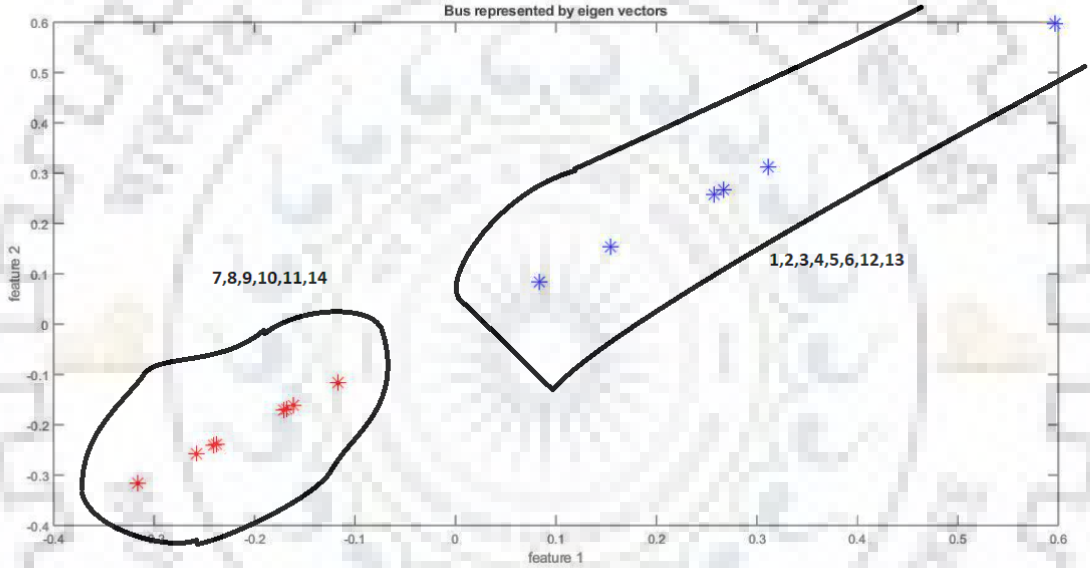


Fig. 4. 12- Bus groups found from stiffness coefficients after line outage

It suggests that lines 11-6, 13-14, 5-6, 4-7, and 4-9 should be disconnected for maintaining coherent operation in respective areas.

After this, the constraint matrix will be formed and then laplacian is formed which are used to solve the generalized eigenvalue problem in Eq. 3.146 and the lowest magnitude eigenvalue and corresponding eigenvector is selected which is

$$v = [0.02 \quad 0.02 \quad 0.03 \quad 0.02 \quad 0.02 \quad 0.03 \quad -0.07 \quad -1 \quad -0.06 \quad -0.10 \quad -0.08 \quad 0.2 \quad 0.06 \quad -0.07]^T$$

β is taken as 140 which satisfies Eq. 3.131 and the plot of the eigenvectors is as shown

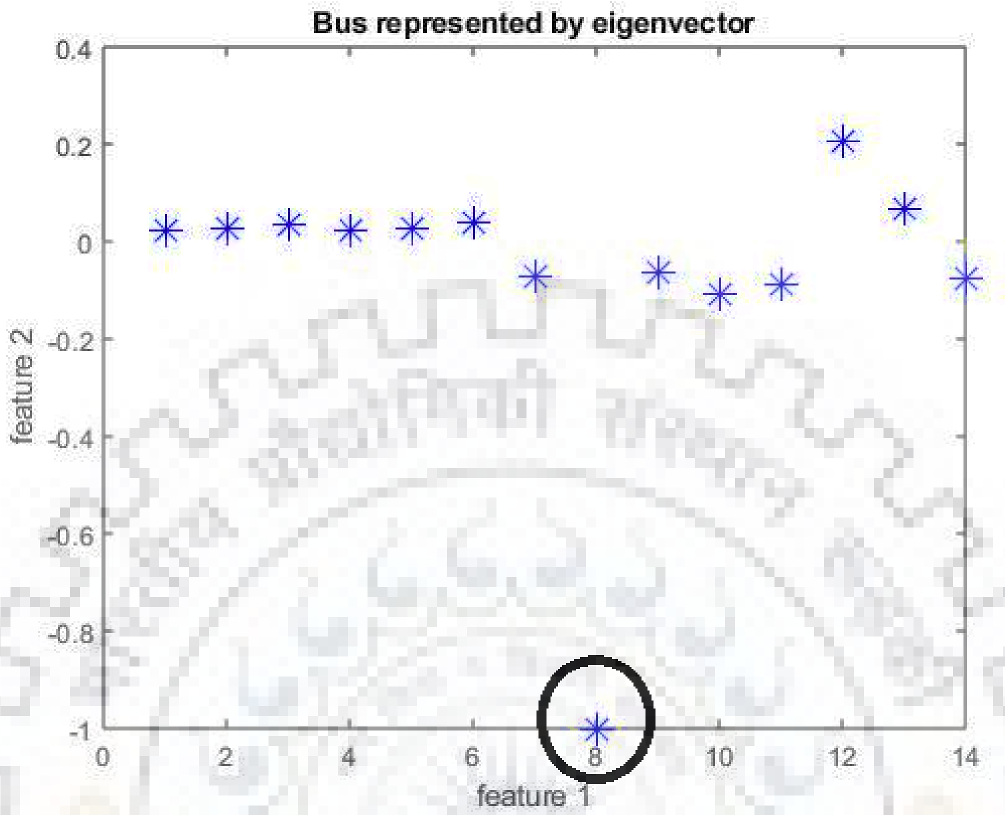


Fig. 4. 13 Buses represented by eigenvector after constraint spectral clustering

This suggests that only bus 8 is to be removed for minimizing power disruption and still maintaining coherency which is acceptable as all generators are nearly coherent and only generator 8 was falling in group 2 which is not supplying much power and working as synchronous condenser. If one was to cut according to constraint matrix only i.e., according to previously suggested clusters as shown in fig. 4.12 then a large amount of load connected to buses 11, 14 10 and 9 buses would have been shed because the synchronous condenser won't be able to supply this much load. So in effect, only generator 8 should be removed if an emergency arises.

5. CONCLUSION

The method of multilateration depends on the velocity of propagation of disturbance which is assumed to be constant in the system studied. Also, the inertia of the system is not constant and its value must be found for the system by extensive study first and then that value can be used for calculation for such methods and determination of electromechanical wave velocity. Optimal PMU locations provide node 4, node 2 and node 9 as the best locations, but it can be seen from equation (6) that some variable measurements overlap, i.e., I(87) can be determined either from voltages as done in equation 3.7 or it can be found from the currents by applying KCL at node bus 7 since bus 7 is a Zero Injection Bus. Hence I(78) is the sum of currents I(57) and I(27). Further reduction of the number of PMUs can be obtained and it can be further analyzed. The results are satisfactory for the velocity of disturbance wave. A neural network can be trained to measure the time of arrival at a particular bus since it can optimize its weight to nearly know the relation between time of arrival and the three critical points of the disturbance wave coming to it. Since the power system is not changing suddenly, the nature of variation of the curve of bus angles is more nearly constant from faults originated in different locations in the power system, or it can be said that the faults originating in different directions have an effect on the disturbance reaching the bus, but the nature of variation might remain the same for a particular direction for a significant period of time. Also, the nature of variation depends on the fault type and a neural network can optimize itself to distinguish the faults in the system from the information of disturbance in the phase angle [4]. Hence, a law exists between the nature of variation of the bus angle curve and the time of arrival of the disturbance and also the type of fault in the system.

The calculation of phase angles from the Discrete Fourier Transform gives wrong results. It is only helpful when the variation in the phase angle is very slow over time and hence can be assumed to be constant at least for a cycle during which the DFT takes its signal window. Moreover, Hilbert transform is more satisfactory when variation in phase angle is not very large and also the frequency should be small in comparison to the power frequency. The Hilbert Transform introduces oscillations in the extracted signal phase angle, but that is large enough to be noticeable only at the ends of the data and not in the middle or steady state. It reproduced much of the similarity of variation with the actual data.

The determination of coherent areas from correlation functions and coherence functions are needed only when correlation coefficients doesn’t provide a crisp decision for some of the buses. Although, they can also give a blurry decision boundary. But coherence functions can be subjected to k-means to obtain more firm decision about the buses. There may be some groups which are infeasible.

The method of spectral clustering using dynamic coupling coefficients to provide the value for k asserts good solution. The clustered groups can be subjected to go through constrained spectral k- embedded clustering to find lines which can be cut maintaining coherency constraints as well as minimum disruption of power. The results are quite satisfactory in that the system remains stable even after removal of the lines predicted by the spectral clustering algorithm. On large systems, there may be more than two clusters. And the solution of preliminary clustering can provide such cluster groups which are infeasible. In those cases, out of all solutions, only those solutions are taken which are feasible and minimize power disruption.

6. APPENDIX

6.1 MATLAB Codes used-

Adjacency Matrix and Optimal PMU Algoritm-

```
% generation of adjacency matrix
top_data =  [   1   1   4;
                2   2   7;
                3   3   9;
                4   4   5;
                5   4   6;
                6   5   7;
                7   6   9;
                8   7   8;
                9   8   9];
adjmat = zeros(9);
nb = max(max(top_data(:,2:3)));% no. of buses
nl = length(top_data(:,1));% no. of links
for i = 1:max(top_data(:,1))
    adjmat(top_data(i,2),top_data(i,3)) = 1;
end

adjmat = adjmat+adjmat'
for i = 1:length(top_data(:,1))
    adjmat(i,i) = 1;
end
adjmat



%------------------------------- Optimal PMU algorithm----------------
--
pmuprob = 'optimprob';
clc;
clear(pmuprob);
pmuprob = optimproblem;
pmus = optimvar('pmus',9,'Type','integer',...
'LowerBound',0,'UpperBound',1);

tot_pmu = ones(1,9)*pmus;
pmuprob.Objective = tot_pmu;
showexpr(tot_pmu);
cons1 = adjmat*pmus >= ones(9,1);
showconstr(cons1);
pmuprob.Constraints.cons1 = cons1;
showproblem(pmuprob);
[sol,cost] = solve(pmuprob);
npmus = sol.pmus;
cost
```

Shortest Path Database Algorithm (Dijkstra Algorithm)-

```
%edges and their weights
%clear all;
close all;
clc;
%s.no.     f_N     t_N     weight
%%data = [   1        1        2        6;
%            2        1        4        1;
%            3        2        3        5;
%            4        2        4        2;
%            5        2        5        2;
%            6        3        5        5;
%            7        4        5        1   ];

data = [    1        1        4        48;
            2        2        7        5.20;
            3        3        9        4.88;
            4        4        5        5.75;
            5        4        6        7.85;
            6        5        7        13.79;
            7        6        9        14.99;
            8        7        8        4.87;
            9        8        9        8.48];
% making the weight matrix
n = max(max([data(:,2), data(:,3)]));%no. of nodes
w_n = zeros(n);
for i = 1:length(data(:,1))
    w_n(data(i,2),data(i,3)) = data(i,4);
end
w_n = w_n + w_n';
display(w_n);
%################----SHORTEST PATH ALGORITHM----
#####################
v1_n = zeros(n);
prev1_n = zeros(n);
sd1 = zeros(n);
%n_n = 1;
for i = 1:n;% we will apply for loop here
    unv_n = 1:n;
    v_n = 0;
    sd = inf*ones(1,n);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%i = 2;% we will apply for loop here
    c_n = unv_n(i);% start by the start node unv_n(i)
    prev_n(c_n) = c_n;
    sd(i) = 0;
while(length(unv_n) > 1)
for j = 1:n
if(w_n(c_n,j)>0&&sd(j)>(sd(c_n)+w_n(c_n,j))...
&&chk_unv_n(unv_n,j) == 1)%only sd of nodes which are connected to
c_n will be modified
                sd(j) = sd(c_n) + w_n(c_n,j);
                prev_n(j) = c_n;
end
end
        v_n  = [v_n,c_n];%this matrix stores the order in which those
are accessed during shortest tree formation
%removing c_n from unvisited node
        unv_n = remove_c_n(unv_n,c_n);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% finding the next current node 'c_n' by findig shortest distance
        n_n = unv_n(1); %next node
for k = 1:length(unv_n)
if (sd(unv_n(k))<sd(n_n)) n_n = unv_n(k);
end
end
        c_n = n_n; % setting next shortest distance node as the
current node 'c_n'
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
end
    v_n = [v_n(2:length(v_n)),unv_n];
%display(v_n);
%display(prev_n);
%display(sd);
    v1_n(i,:) = v_n;
    prev1_n(i,:) = prev_n;
    sd1(i,:) = sd;
end


function g = chk_unv_n(unv_n,j)
for i = 1:length(unv_n)
if(unv_n(i) == j)
            g = 1;% found j node as unvisited
return;
end
end
    g = 0;
end


function y =  remove_c_n(unv_n,c_n)
for i= 1:length(unv_n)
if (unv_n(i) == c_n)
            y = [unv_n(1:i-1),unv_n(i+1:length(unv_n))];
end
end
end
```

Backtracing Shortest Path Database Function- Backtrace(x,y,z)

```
%s_n- source node, d_n- destination node
% This functin takes input as previous node matrix, source node and
% and destination node and provide output as the path vector
% giving successive nodes for shortest path from source to
destination node
function [y] = backtrace(prev1_n,s_n,d_n)
    path = d_n;
    main_row = prev1_n(s_n,:);
    d = d_n; %dummy d to change in the loop
while(main_row(d) ~= s_n)
        path = [main_row(d), path];
        d = main_row(d);
end
    path = [main_row(d),path];
    y = path;
end
```

Implementation of K- means in MATLAB-

```
clc;
clear all;
close all;
dim = 8;
num_data = 100;%number of data points in 9 dimensional space
x = rand(dim,num_data);
n = 3; %number of clusters
g = floor(10*rand(1,n));
display(g);
error = 100;
k = 2; % no. of iterations

   for i = find(g == 0)
        p = floor(10*rand);
        while(find(g == p)~=0)
            p = floor(10*rand);

        end
        g(i) = p;
    end

    %this block removes any similar element from the matrix
    for i = 1:length(g)
        for j = g(i+1: length(g))
            if(j>g(i) || j<g(i))
                continue;
            else
                p = floor(10*rand);
                while(p == 0 || length(find(g == p))~=0)
                    p = floor(10*rand);

                end
                g(i) = p;
                continue;
            end
        end

    end
    %####################################################################

    cent_i = x(:,g);
```

```
    display(cent_i);
    cent = cent_i;

while(abs(error(k-1))>=1e-3)
    clus = cell(1,n);
    clus = {[],[],[]};

    %#######################################################################
    %this block removes any zero element from the matrix

    for i = x
        dist = sum((cent - i).*(cent - i));
        m = find(dist == min(dist));
        clus{1,m} = [clus{1,m},i];
    end
    %checking for empty centroids
    emp_num = 0;
    emp = [];

    for i = 1:length(clus)
        if(size(clus{1,i}) == size([]))
            emp_num = emp_num + 1;
            emp = [emp, i];
        end
    end
    display("Number of empty clusters = ");
    display(emp_num);
    clus(:,emp) = [];
    cent(:,emp) = [];

    n = length(clus);

    cent_i = cent;
    %calculation of new centroids
    for i = 1:length(clus)
        cent(:,i) =
((clus{1,i})*ones(length(clus{1,i}(1,:)),1))/length(clus{1,i}(1,:));%here
was
    end

    %calculation of evaluation index
    eval_index(k) = 0;
    for i = 1:length(cent(1,:))
        eval_index(k) = eval_index(k) + ones(1,length(cent(:,i)))*...
            ((cent(:,i) - clus{1,i}).*(cent(:,i) - clus{1,i}))*...
            ones(length(clus{1,i}(1,:)),1);
    end

    error(k) = eval_index(k) - eval_index(k-1);

    k = k+1;


end

display("Number of iterations = ");
display("centroid = ");
display(cent);
 display(eval_index);
display(error);
```

Matlab code to calculate eigenvector in Spectral clustering-

```
%Weighr matrix- t= 46s is choosen as fault settlling time
for v = 1:n_nod
    g1 = [Pij{v,:}]; %Pij is a cell matrix already stored in workspace
    W(v,:) = g1(460000,:); %change this state according to new steady value
end
W = (abs(W) + abs(W'))/2;%laplacian must be symmetrical, so is W

display(W);% values will be in per Unit

%%
                   %  FORM DEGREE MATRIX
D = sum(W,2);
D = diag(D);
N = inv(sqrt(D));
%%
             %VOLUME OF THE DATASET
vol = sum(diag(D));
%%
L = D - W;
%%
                    %UNNORMALIZED EMBEDDED CONSTRAINT MATRIX

%UNCONSTRAINED CLUSTER FROM INITIAL COHERENCY
% this section is written because 'i' vector contained only (n_nod-1) rows
and the bus 1 was connected to bus 4 so i(1) must be equal to i(4), so they
belong to the same cluster. Omit this section if all i has n_nod rows
Q = zeros(n_nod);

Q = zeros(n_nod);
if (length(i) == n_nod -1)
    i =  [i(3); i]; %this makes 1st element of resulting 'i' same as 4th
end

for k = 1:3 %k represent the kth cluster
    Q(find(i == k), find(i == k)) = 1;% notice that i is a matrix
end

Q(find(~Q)) = -1;

display(Q);

%%
% Now Calculate the normalized Laplacian and Constrain matrices
% formin normalized laplacian
LN = N*L*N;
QN = N*Q*N;

%%
%solve for maximum eigenvalue of QN
lamQ = (eig(QN));
betarange = [min(lamQ(find(lamQ>0.1)))*vol ,max(lamQ)*vol];
%%
%select beta and solve generalized eigen value problem
beta = 15; %beta must be chosen from betarange variable

%%
% solving the generalized eigen value problem A * vi = lami * B*vi
% The problem is LN*v = lam*(QN-(beta/vol)*I)*v
% this is same as solving considering A = LN, B = QN-(beta/vol)*I
A = LN; B = (QN-(beta/vol)*eye(size(QN)));
[V, lam] = eig(A,B);
```

For running this code, MATLAB workspace must contain beforehand, the Power flow cell(not matrix) Pij, the cluster indicator vector i, the number of nodes of the graph n_nod . It is better if the code is run section wise. The code was written for 9- bus system.

6.2 System Data Used-

6.2.1 IEEE- 14 Bus System-

Table 6.2. 1- Line Data

From Bus	To Bus	R(pu)	X(pu)	B(pu)
1	2	0.01938	0.05917	0.0264
1	5	0.05403	0.22304	0.0246
2	3	0.04699	0.19797	0.0219
2	4	0.05811	0.17632	0.0187
2	5	0.05695	0.17388	0.0170
3	4	0.06701	0.17103	0.0173
4	5	0.01335	0.04211	0.0064
6	11	0.09498	0.19890	-
6	12	0.12291	0.25581	-
6	13	0.06615	0.13027	-
9	10	0.03181	0.08450	-
9	14	0.12711	0.27038	-
10	11	0.08205	0.19207	-
12	13	0.22092	0.19988	-
13	14	0.17093	0.34802	-

Table 6.2. 2– Transformer Data

From Bus	To Bus	R(pu)	X(pu)	Tap Ratio
4	7	0	0.20912	0.978
4	9	0	0.55618	0.969
5	6	0	0.25202	0.932
7	8	0	0.17615	1
7	9	0	0.11001	1

Table 6.2. 3- Generator Dynamic Data

Gen	Xa(pu)	Xd(pu)	Xd'(pu)	X_d''(pu)	X_q''(pu)	X_q'''(pu)	X_q''(pu)	Base (MVA)
1	0.1450	1.7241	0.2586	0.2029	1.6587	0.4524	0.2029	615
2	0.1540	1.7241	0.2586	0.2029	1.6587	0.4524	0.2029	60
3	0.1540	1.7241	0.2586	0.2029	1.6587	0.4524	0.2029	60
6	0.1540	1.7241	0.2586	0.2029	1.6587	0.4524	0.2029	25
8	0.1540	1.7241	0.2586	0.2029	1.6587	0.4524	0.2029	25

Table 6.2. 4- Generator Data -2

Gen.	Ra(pu)	T_{do}'(s)	T_{do}'' (s)	T_{qo}'(s)	T_{qo}'' (s)	H(s)	D(pu/pu)	Base(MVA)
1	0.000125	3.826	0.0225	0.5084	0.0225	3.41	0.0	615
2	0.000125	3.826	0.0225	0.5084	0.0225	3.41	0.0	60
3	0.000125	3.826	0.0225	0.5084	0.0225	3.41	0.0	60
6	0.000125	3.826	0.0225	0.5084	0.0225	3.41	0.0	25
8	0.000125	3.826	0.0225	0.5084	0.0225	3.41	0.0	25

Table 6.2. 5- Exciter Data-

Gen	Tr	Ka	Ta	Vmax	Vmin	Ke	Te	Kf	Tf
1	0	6.2	0.05	5.2	-4.16	1	0.83	0.057	0.5
2	0	6.2	0.05	5.2	-4.16	1	0.83	0.057	0.5
3	0	6.2	0.05	5.2	-4.16	1	0.83	0.057	0.5
6	0	6.2	0.05	5.2	-4.16	1	0.83	0.057	0.5
8	0	6.2	0.05	5.2	-4.16	1	0.83	0.057	0.5

Table 6.2. 6- Governor Data-

Gen	R	T1	Vmax	Vmin	T2	T3	Dt
1	0.05	0.49	15	0	2.1	7.0	0.0
2	0.05	0.49	15	0	2.1	7.0	0.0

7. REFERENCES

[1] U. Rudez and R. Mihalic, "A method of detecting the time of arrival for an electromechanical wave in large power systems," *2013 IEEE Grenoble Conference*, Grenoble, 2013, pp. 1-6.

[2] U. Rudez and R. Mihalic, "Understanding the electromechanical wave propagation speed," *2013 IREP Symposium Bulk Power System Dynamics and Control - IX Optimization, Security and Control of the Emerging Power Grid*, Rethymno, 2013, pp. 1-8.

[3] M. Parashar, J. S. Thorp and C. E. Seyler, "Continuum modeling of electromechanical dynamics in large-scale power systems," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 9, pp. 1848-1858, Sept. 2004.

[4] A. Esmaeilian and M. Kezunovic, "Fault Location Using Sparse Synchrophasor Measurement of Electromechanical-Wave Oscillations," in *IEEE Transactions on Power Delivery*, vol. 31, no. 4, pp. 1787-1796, Aug. 2016.

[5] J. S. Thorp, C. E. Seyler and A. G. Phadke, "Electromechanical wave propagation in large electric power systems," in *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 45, no. 6, pp. 614-622, June 1998.

[6] H.A. Rahman, Nadia & Zobaa, Ahmed. (2016). Optimal PMU Placement using Topology Transformation Method in Power Systems. Journal of Advanced Research. 7. 10.1016.

[7] A. Esmaeilian and M. Kezunovic, "Prevention of cascading outages using sparse wide area synchrophasor measurements," *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, Ljubljana, 2016, pp. 1-5.

[8] J. S. Thorp, C. E. Seyler and A. G. Phadke, "Electromechanical wave propagation in large electric power systems," in *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 45, no. 6, pp. 614-622, June 1998.

[9] A. Vahidnia, G. Ledwich, E. Palmer and A. Ghosh, "Generator coherency and area detection in large power systems," in *IET Generation, Transmission & Distribution*, vol. 6, no. 9, pp. 874-883, September 2012.

[10] “Coherence (signal processing),” Wikipedia, 22-Jan-2019. [Online]. Available: https://en.wikipedia.org/wiki/Coherence_(signal_processing). [Accessed: 07-Mar-2019].

[11] G. A. Wilkin and X. Huang, "K-Means Clustering Algorithms: Implementation and Comparison," *Second International Multi-Symposiums on Computer and Computational Sciences (IMSCCS 2007)*, Iowa City, IA, 2007, pp. 133-136.

[12] Ng AY, Jordan MI, Weiss Y (2001) On spectral clustering: analysis and an algorithm. In: Advances in neural information processing systems 22 (NIPS 2001), pp 849–856

[13] C. Guo and S. Zheng, “A Survey on Spectral Clustering,” *World Autom. Congr. 2012*, pp. 20–23, 2010.

[14] X. Wang, B. Qian, and I. Davidson, “On constrained spectral clustering and its applications,” *Data Min. Knowl. Discov.*, vol. 28, no. 1, pp. 1–30, 2014.

[15] A. Esmaeilian and M. Kezunovic, “Prevention of power grid blackouts using intentional islanding scheme,” *IEEE Trans. Ind. Appl.*, vol. 53, no. 1, pp. 622–629, 2017.

[16] Jianbo Shi and J. Malik, "Normalized cuts and image segmentation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888-905, Aug. 2000.

[17] David Arthur , Sergei Vassilvitskii, k-means++: the advantages of careful seeding, Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, p.1027-1035, January 07-09, 2007, New Orleans, Louisiana.

[18] Hongbiao Song, Mladen Kezunovic, A new analysis method for early detection and prevention of cascading events, Electric Power Systems Research, Volume 77, Issue 8, 2007, Pages 1132-1142.

[19] Mili, Lamine & Qiu, Q & G. Phadke, A. (2004). Risk assessment of catastrophic failures in electric power systems. IJCIS. 1. 38-63. 10.1504/IJCIS.2004.003795.

[20] H. You, V. Vittal and Xiaoming Wang, "Slow coherency-based islanding," in *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 483-491, Feb. 2004.

[21] Jena, Premalata & Kumar Pradhan, Ashok. (2010). A Positive-Sequence Directional Relaying Algorithm for Series-Compensated Line. Power Delivery, IEEE Transactions on. 25. 2288 - 2298. 10.1109/TPWRD.2010.2047122.

[22] Juarez, Carlos & Messina, Arturo & Castellanos, Rafael & Espinosa-Perez, Gerardo. (2011). Characterization of Multimachine System Behavior Using a Hierarchical Trajectory Cluster Analysis. Power Systems, IEEE Transactions on. 26. 972 - 981. 10.1109/TPWRS.2010.2100051.

[23] Kai Sun, Da-Zhong Zheng and Qiang Lu, "Splitting strategies for islanding operation of large-scale power systems using OBDD-based methods," *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No.03CH37491)*, Toronto, Ont., 2003, pp. 1812-1812.

[24] Peiravi, Ali & R, Ildarabadi. (2009). A Fast Algorithm for Intentional Islanding of Power Systems Using the Multilevel Kernel k-Means Approach. Journal of Applied Sciences. 9. 10.3923/jas.2009.2247.2255.

[25] R. Baldick *et al.*, "Initial review of methods for cascading failure analysis in electric power transmission systems IEEE PES CAMS task force on understanding, prediction, mitigation and restoration of cascading failures," *2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, Pittsburgh, PA, 2008, pp. 1-8.

[26] Vaiman *et al*., "Risk Assessment of Cascading Outages: Methodologies and Challenges," in *IEEE Transactions on Power Systems*, vol. 27, no. 2, pp. 631-641, May 2012.

[27] Ramaswamy, G.N., Verghese, G.C., Rouco, L., Vialas, C., DeMarco, C.L.: 'Synchrony, aggregation, and multi-area eigenanalysis', IEEE Trans. Power Syst., 1995, 10, (4), pp. 1986–1993.

[28] N. Senroy, "Generator Coherency Using the Hilbert–Huang Transform," in *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1701-1708, Nov. 2008.

[29] H. A. Alsafih and R. Dunn, "Determination of coherent clusters in a multi-machine power system based on wide-area signal measurements," *IEEE PES General Meeting*, Providence, RI, 2010, pp. 1-8.

[30] K. L. Lo, Z. Z. Qi and D. Xiao, "Identification of coherent generators by spectrum analysis," in *IEE Proceedings - Generation, Transmission and Distribution*, vol. 142, no. 4, pp. 367-371, July 1995.

[31] M. Jonsson, M. Begovic and J. Daalder, "A new method suitable for real-time generator coherency determination," in *IEEE Transactions on Power Systems*, vol. 19, no. 3, pp. 1473-1482, Aug. 2004.

[32] K. K. Anaparthi, B. Chaudhuri, N. F. Thornhill and B. C. Pal, "Coherency identification in power systems through principal component analysis," in *IEEE Transactions on Power Systems*, vol. 20, no. 3, pp. 1658-1660, Aug. 2005.

[33] E. W. Dijkstra, "A note on two problems in connexion with graphs," Numer. Math., vol. 1, pp. 269–271, 1959.

[34] H. Kuhn and A. Tucker, "Nonlinear programming," ACMSIGMAP Bull., pp. 6–18, 1982.

[35] "Kmeans," Mathworks. [Online]. Available: http://www.mathworks.com/help/stats/kmeans.html. [Accessed: 16-Feb-2019].