8-2010

High Order Contingency Selection using Particle Swarm Optimization and Tabu Search

Ashwini Chegu
University of Tennessee - Knoxville, achegu@utk.edu

To the Graduate Council:

I am submitting herewith a thesis written by Ashwini Chegu entitled "High Order Contingency Selection using Particle Swarm Optimization and Tabu Search." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Electrical Engineering.

Fangxing (Fran) Li, Major Professor

We have read this thesis and recommend its acceptance:

Yilu Liu, Kevin Tomsovic

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a thesis written by Ashwini Chegu entitled "High Order Contingency Selection using Particle Swarm Optimization and Tabu Search". I have examined the final electronic copy of this thesis form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Electrical Engineering.

Fangxing "Fran" Li, Major Advisor

We have read this thesis and recommend its acceptance:

Kevin Tomsovic

____

Yilu Liu

____

Acceptance for the Council:

____

Carolyn R. Hodges, Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records)

High Order Contingency Selection using Particle Swarm Optimization and Tabu Search

A Thesis
Submitted for
Master of Science Degree
The University of Tennessee, Knoxville

Ashwini Chegu
August 2010

Dedication

This thesis is dedicated to my beloved parents, sister and my dearest friends who have blessed my life with their love, support and sacrifice.

Acknowledgements

Firstly, I am grateful to my graduate advisor, Dr. Fangxing Li for giving me this opportunity to work on this project and also supporting through my Masters. His patience, encouragement, and valuable guidance helped me learn a lot through my research.

I would like to thank my committee members Dr. Kevin Tomsovic and Dr. Yilu Liu for reviewing my thesis.

I express my thanks to all my professors for influencing me and helping me in achieving my goals. Thank you to those in Power Engineering Laboratory for their help and suggestions in my research work.

Finally, I owe my heartfelt regards for my family who had constantly been my force of inspiration, determination, and encouragement. I extend my special thanks to all my friends for their care and support which has helped me through this arduous task.

Abstract

There is a growing interest in investigating the high order contingency events that may result in large blackouts, which have been a great concern for power grid secure operation. The actual number of high order contingency is too huge for operators and planner to apply a brute-force enumerative analysis. This thesis presents a heuristic searching method based on particle swarm optimization (PSO) and tabu search to select severe high order contingencies. The original PSO algorithm gives an intelligent strategy to search the feasible solution space, but tends to find the best solution only. The proposed method combines the original PSO with tabu search such that a number of top candidates will be identified. This fits the need of high order contingency screening, which can be eventually the input to many other more complicate security analyses.

Reordering of branches of test system based on severity of N-1 contingencies is applied as a pre-processing to increase the convergence properties and efficiency of the algorithm. With this reordering approach, many critical high order contingencies are located in a small area in the whole searching space. Therefore, the proposed algorithm tends to concentrate in searching this area such that the number of critical branch combinations searched will increase. Therefore, the speedup ratio is found to increase significantly.

The proposed algorithm is tested for N-2 and N-3 contingencies using two test systems modified from the IEEE 118-bus and 30-bus systems. Variation of inertia weight, learning factors, and number of particles is tested and the range of values more suitable for this specific algorithm is suggested.

Although illustrated and tested with N-2 and N-3 contingency analysis, the proposed algorithm can be extended to even higher order contingencies but visualization will be difficult because of the increase in the problem dimensions corresponding to the order of contingencies.

Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1. Background

Power systems are among the most complex and largest technological systems ever developed. The present power systems in the United States and many other developed countries are running close to their operational limits. This raises many concerns within the power industry as well as general public, since power systems are important elements of the national and global infrastructures. Due to the continuously reducing operating margin, the US power system sometimes suffers from unplanned, large-scale disturbances, which have considerable affects on power grid and cause direct and indirect consequences on the economy and national security [1]. Occurrence of blackouts is very rare but they have a huge impact. The interconnected nature of the infrastructure makes the power system more integrated and complex to understand the entire system and the blackout.

Power system blackout occurs due to successive failure of a set of individual components in a very short duration of time, where the first failure occurs unexpectedly. High ordered contingency or N-k contingency may be defined as multiple component failure that coincidently occurs near simultaneously. But in reality the true randomly multiple component failure is extremely rare. So, it does not reflect the fact that failures are related, while the term “cascading failure” does reflect such consequence and dependence. However, since N-k contingency or high order contingency is an understandable term and also commonly used when analyzing cascading failures, both terms are considered exchangeable in this research [2][3]. Since a cascading failure may be due to hidden failures which are difficult to identify by their nature as well as the lack of data, a general searching algorithm of high order contingency can be employed to find the most severe high order contingency events.

In recent years, this has been a particular concern for power transmission operators as evidenced by many researches in high order contingency analysis as well as the utility practices. For instance, many power transmission operators have expanded contingency criterion from N-1 to some N-2 contingencies and even N-3 contingencies. High order contingency events are difficult to analyze and model. If we take possible combinations of N-k contingency, then the total number of possible combination is $N!/[k!\times(N-k)!]$, which is as huge as 499,500 for a relatively small system with N=1000 and k=2. And the number of cases are much worsened to 166,167,000 if k=3. Hence, brute-force enumeration is not an efficient approach especially for short term operations. Therefore, there is a need for efficient high order contingency screening approach, especially considering potential short-term operation.

1.2. Generic Scenario of Cascading Failures

"Cascading failures are sequence of dependent failures of individual components that successively weakens the power system." On analyzing the blackouts occurred by cascading trips of generators or transmission facilities in the year 2003, the August 14th blackout in USA and Canada, the August 28th blackout in London, and the September 28th blackout in Italy, a generic scenario of the causes and the effects of blackouts is suggested in [5].

The blackout generally occurs when the system experiences some form of instability. Because of an initial triggering event, load is shifted to its neighboring elements in the system and subsequent failures occur due to power flow surges, equipment overloads, and voltage problems. These elements with load exceeding its capacity in-turn transfer the load to all its neighboring elements and causes sudden spikes across all the nodes in the system. This may cause more overloads and may result in blackouts in the system in a very short duration of time.

Power system protection devices such as relays play an important role in the development of blackouts. When a fault occurs, protection systems are used to disconnect the equipment from the rest of the system due to the action of breakers. This may trigger multiple outages and may cause voltage instabilities and overloading. Some load loss may occur during this process that in turn causes more power flow surges and overloads. Load loss due to islanding could help in balance generation and load and relieve system problems in remaining part of interconnection as well as in some isolated islets within separated grid. Figure 1.1 shows the mechanism of cascading failures.

Some of the remedial actions to solve contingencies are use of shunt capacitor switching which solves low voltage problems occurred at the buses due to lack of reactive power supplied. Under-load tap changing transformers are used to change the supplied voltage to load or system.

When the current flowing through a line is over the specified line limit, generator re-dispatch is done to send the power to the load through changing generation. Load shedding is generally done when all other methods to solve violations for contingencies fail. Also, line overload problems may be solved using Distributed Generators (DG) or local generators since power can be generated nearer to the load.

Failures are most commonly seen in high voltage systems with a single point of failure (SPF) and occur in fully or slightly loaded system. A sudden spike may occur across all the nodes in the system leading to failure.

The Figure 1.1 shows the graphical representation to demonstrate the process of cascading failures. When the line trips due to a fault, there is a possibility of cascading failure to happen or else the system can recover. If the system recovers, there might be a few lines which are out and with another initiating event, this process might continue until the failure leads to blackout.

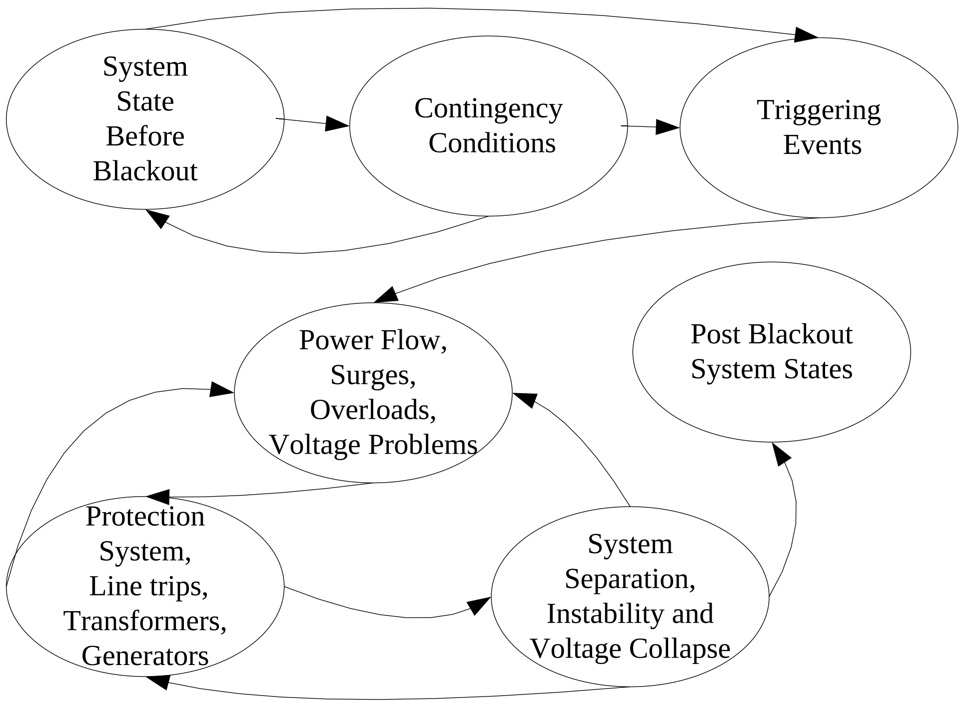

Figure 1.1. Generic scenario of cascading failures [5]

1.3. Causes of Cascading Failures

There are various causes for occurrence of cascading failures. A cascading failure is usually initiated by an outage of a single component. It is interesting to discuss why subsequent failures occur. From various previous works [2-6], we summarize the possible reasons into the following categories.

Hidden failures are the equipment failure that is not known or visible to operators, but will cause follow-up outages after the initial triggering contingency event. Once the initial contingency occurs, the protective device may not function correctly and timely to mitigate the impact such that the disturbance propagates to another transmission facility.

Backup protection includes Zone 2, Zone 3, and even Zone 4 relays that serve as backup options to clear a fault in case the primary relay cannot. Sometimes backup protection unnecessarily trips a line. It may trip even after the main protection correctly clears a fault [4]. Relay settings are usually re-adjusted periodically due to the change of system operation state. However, it is difficult to ensure that the relay settings are perfect under every scenario, especially considering the increasing stress of many power systems.

When there is a switching operation needed in a substation due to maintenance or attempt to mitigate undesirable conditions, it leads to a change of the topology such that it is different from the original design. Then, when the system needs to respond to a fault, it may incorrectly remove multiple components. In this case, the possibility of N-k contingency is close to an N-1 contingency, because they are no longer truly independent events [6]. If one relay fails, it has to send a tripping signal to adjacent circuit breakers to isolate the fault. Due to the lack of such communication between different components, there is a higher chance of occurrence of cascading failures.

Some factors like tree contact, line contact, and also excessive line sagging due to expansion in summer can also cause failures. The reasons leading to cascading failure are summarized in the Table 1.1.

The violations that cause cascading failures are at buses with low voltages or line overloads. If the voltage of bus is less than the specified value, the low voltage violations take place. Reactive power causes voltage problems. In case of low voltage problems reactive power is supplied to the bus to increase the voltage profile at the bus and in case of high voltage reactive power is absorbed at the buses to maintain normal voltage [7].

Line MVA limit violations occur when the load on the line increases beyond its limit. In general the lines are designed so that the line withstands 125% of the line MVA limit. This mainly happens due to the increase in amplitude of current flowing in the line. Remedial actions to identify such overloaded lines and rank them depending on how much the lines are loaded is discussed in this research.

Table 1.1 Causes of cascading failures [7]

	Primary causes	**Causes of cascading**	
		Deterministic factor	**Probabilistic factors**
Blackouts	Primary protective relay failure	Under-frequency	Failure of the tap-changing mechanism
	Line fault	Overload	Additional lighting
	High winds causing line failure	Over-current	Failure of Communication channel
	Line sagged into trees	Low voltage	Failure of Backup device
	Hidden failure		Operators unawareness of failures
	Lightning		Failure of EMS system
	Phase-to-ground fault		
	Tower causing multiple lines out		
	A sequence of line trappings		

1.4. Effect of Loading

The power system is said to be stable at a specific loading level called base case loading. Loading margin is calculated by measuring the amount of load increase that might cause voltage collapse. The voltage curve as a function of loading has a sharp change in direction called nose point at voltage collapse. The contingency condition is when there is transient and the system re-stabilizes after it. Under such conditions the loading margin decreases. The nominal voltage shown in the Figure 1.2 is loading on a specific bus as a function of total system loading. It is measured to be the distance of curve from base case operating point till the nose point. Loading margin can be assumed as a function of change in line admittance caused by removal of one line. The change in sensitivity is a useful measure to calculate the change in loading [8].

The probability of occurrence of blackout is roughly proportional to blackout size as illustrated in Figure 1.2 and hence the blackout distribution probability has an exponential tail. The point at which there is a sudden change in intensity of loading is called as a point of critical loading. Critical loading point is characterized by operation with lines close to their line limits. It is thus considered as reference for power system operating limit with respect to cascading failure. The plot showing how the mean blackout size changes with loading is as shown in Figure 1.3. It is observed that the blackout size increases sharply at critical loading point.

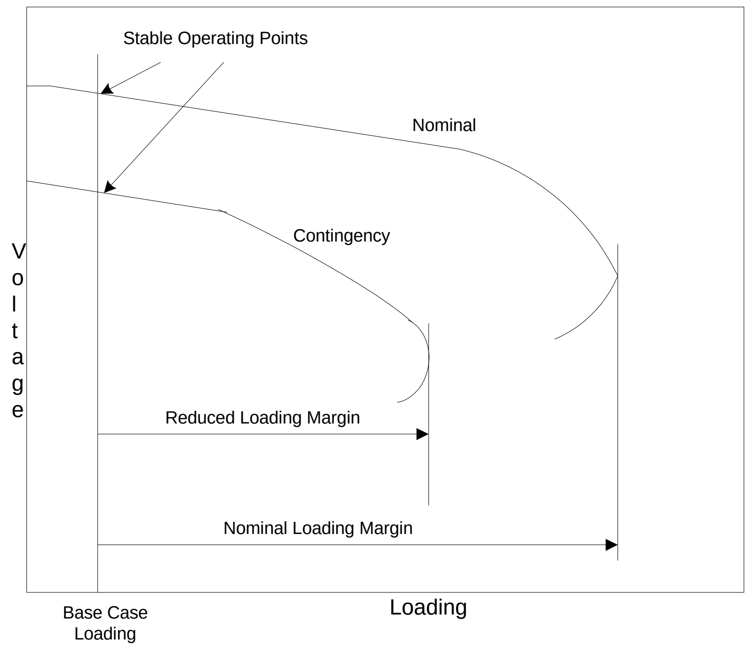

Figure 1.2 Nominal and contingency nose curves [8]

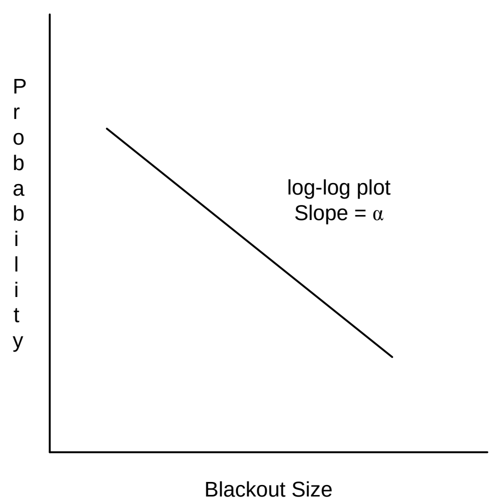

Figure 1.3 Kink in blackout size [9]

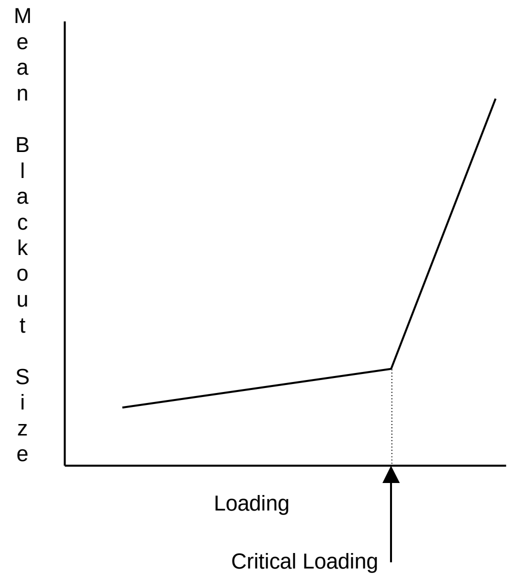

Figure 1.4 Probability of blackout size at critical loading [9]

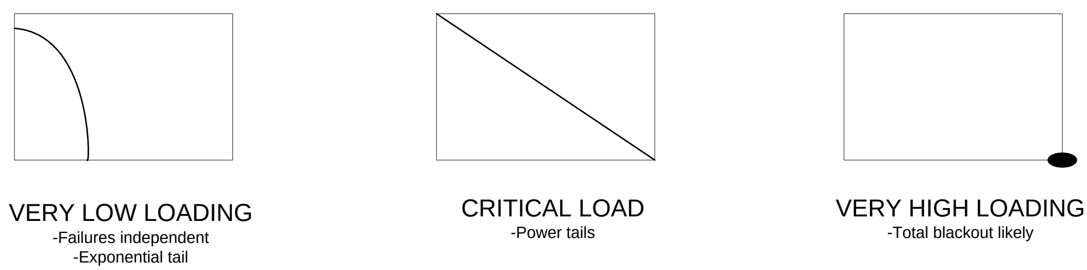

Figure 1.5 Log-log plots showing effect of loading [9][10]

The probability of occurrence of blackout is low at minimal loading and has a very less impact on components having large operating margins. As the loading increases, the probability of occurrence of blackout increases. Power tails are exhibited at critical loading [10].

1.5. Motivation

Operation of power system in normal state is very important for maintaining its security. Developing a decision support system to make right decisions for taking corrective actions or remedial methods is necessary to reduce the probability of blackouts. It is essential to identify the lines with high rate of failure and contingencies which could cause severe damage. Some necessary corrective actions are to be taken to restore them in case of failure. Ranking of the cascading scenarios based on severity is useful in making an offline study for evaluating system impact on utilities or analysis of the probable extreme situations in a power system which requires immediate attention.

Due to technological and economical restrictions, it is not possible to completely eliminate blackouts. However, strategies can be applied to identify the severe high order contingencies in real time for possible preventive actions. In this thesis an efficient algorithm is developed for screening the high order contingencies. It is used to rank the contingencies based on severity of overloading and this could be very useful to the operators to take the preventive actions to decrease the overloading of lines and eventually prevent cascading failures.

1.6. Thesis Contribution and Outline

The purpose of this thesis is to screen all high order contingencies caused due to overloading of power lines. The particle swarm optimization (PSO) algorithm is chosen in this thesis to identify some critical N-2 and N-3 combinations within a reasonable searching effort. DC power flow is implemented for calculating redistribution of power flow under contingent conditions. It should be noted that many times when a group of severe contingency events are identified with preventive actions applied, this will ensure the security of many other contingency events that may or may not be in the list of severe contingencies. Certainly, it is ideal to evaluate the impact of all severe contingencies (or root-cause contingencies), but this may not be feasible for high order ones due to the complexity. Hence, identifying a reasonably large subset of severe high order contingencies can greatly help to enhance the system security, especially considering the trade-off between time and resolution.

The contribution of this thesis can be summarized as:

- An efficient searching algorithm based on PSO and tabu search is proposed to identify a set of most severe high order contingency events.

- A reordering approach as a preprocessing of the proposed searching algorithm is applied to sharply increase the solution quality and efficiency of the proposed algorithm.
- A comparison study of the running time and accuracy when different parameters of the PSO-based algorithm are applied.

The rest of this thesis is organized as follows:

- Chapter 2 describes the literature review of the technologies utilized in this research. It gives an overview about particle swarm optimization (PSO) technique. Calculation of DC power flow is discussed. Brute force enumeration method is outlined.
- Chapter 3 describes about the simulation model developed based on PSO and tabu search. It proposes changes made to the original algorithm and the test system to increase the efficiency.
- Chapter 4 presents analysis of results obtained from various runs from the algorithm.
- Chapter 5 gives conclusions and future work.

1.7. Chapter Summary

In summary, this chapter provides an insight into the background and definition of cascading failure in Section 1.1. Section 1.2 describes the scenario of occurrence of cascading failures. Section 1.3 gives a brief description about all the causes that could lead to cascading failures. Section 1.4 explains about the effect of loading on power lines. Section 1.5 describes the motivation of this thesis work. And section 1.6 gives the thesis outline.

Chapter 2 Literature Review

The chapter gives an overview of the terms used in this thesis. Section 2.1 describes the analysis of risks caused by cascading failures. Section 2.2 describes about the brute force enumeration technique. Section 2.3 describes the particle swarm optimization technique. Section 2.4 explains about the tabu search method. Section 2.5 illustrates the DC power flow technique.

2.1. Risk Analysis of Cascading Failures

Voltage collapse or overloading may be resulted due to unexpected contingencies such as line outages. This may even result in severe blackouts. To prevent these contingencies it is essential to estimate the effect of contingencies on the stability margin [8]. Cascading failures cannot be totally eliminated but the counter measures are to reduce its severity and frequency.

Analysis of cascading failures can be very similar to traditional contingency analysis, which is very useful to understand the condition of power systems in advance to take any preventive measures for security control. Contingency analysis can be divided into static and dynamic one. Static contingency analysis investigates the final steady-state of the power systems after the contingency. It ignores the transition from the normal steady state to the post-contingency steady state. As a comparison, dynamic contingency analysis explores the dynamics of power systems moving from pre- to post-contingency states. Hence, static contingency analysis is not as accurate as the dynamic contingency analysis, but runs much faster since it does not consider the very complicate time-domain simulation that is typical for dynamic contingency analysis.

Therefore, it is sensible to apply static contingency analysis first to rank or select a subset of all possible contingencies based on some severity indices; and then use dynamic contingency analysis to run detailed simulation to evaluate the impact of the contingency.

Starting from these basic techniques of contingency analysis, cascading failures can be modeled, and simulated by a broader range of different techniques. Due to the complicated nature of cascading failures, many compromises are made for simulating their models. Usually, failures are assumed to be simultaneous and related. And only high risk and initial failures are considered to simplify the simulation process.

2.2. Contingency Ranking Schemes

Simulations can be turned to reproduce the features of blackouts in order to simulate or predict the events before occurrence of blackouts. Identification of blackouts is difficult due to high probability of rare, unusual and huge number of failures. The phenomena of blackout occurrence are complicated making the analysis, obtaining data and simulating it very difficult in a short duration. Cascading failures can be modeled and simulated by a broader range of different techniques. Due to the complicated nature of cascading failures, many compromises are made for simulating their models. Usually, failures are assumed to be simultaneous and related. And only high risk and initial failures are considered to simplify the simulation process. The discussion below classifies analytical approaches into several categories based on some important features of each approach.

Reference [5] describes a reliability analysis tool called TRELSS (Transmission reliability evaluation for large-scale systems) for screening N-k contingencies and simulating cascading process. It is also used to evaluate system impacts and reliability based on their severity of system problems like overloads, voltage instability and network separations in [5].

Reference [6] presents another simulation method for identifying cascading failures using graph search algorithm based on substation topology search. It points out that substation topology variation because of switching operation as a response to a single contingency may increase the probability of N-k contingency due to hidden failures. This method uses the probabilistic analysis of protection system failure or substation configuration obtained from topology processing data.

Reference [18] presents the ORNL-PSERC-Alaska (OPA) model to study the complex behaviors of dynamics of series of blackouts. In this model, a power grid, which is constantly upgrading as a complex system satisfies an increasing load demand. Linear programming generator dispatch is used to solve DC load flow model. Self-Organized Criticality is used to restore the system to stable state and increase efficiency.

Reference [10] presents the CASCADE model, which is a probabilistic load-dependent cascading failure model. It captures the salient features of large power system blackouts. It shows that there is a power-law region at a critical loading point associated with the saturating quasi-binomial distribution of the number of failed components.

It is a natural approach to apply branching process to simulate the cascading failures [19-21]. Reference [19] describes the Galton Watson branching process, in which it is assumed that there is variable time between stages or fixed time between stages. At each stage the mean number of failures is increased by a factor of λ. A larger number of stages yield λ closer to 1 or criticality. At critical point, the blackout data increases exponentially. Markov Chain branching process is a similar branching method but the failures in each stage are assumed to be at a constant rate. Reference [20] proposes a Poisson Branching process to approximate the CASCADE model. Different from other usual approaches that minimize the risk of the first few cascading failures, this approach attempts to reduce the propagation of the failure. Reference [21] presents an estimator based on branching processes to evaluate the propagation of cascading failures. This estimator is also tested on results from OPA model.

Manchester model [22-24] uses AC load flow and state sampling Monte Carlo method for simulation. Adjustments are made by automatic control centers and operators to reduce the risk of failures. This method is used to show the evidence of existence of criticality in cascading failure blackouts [22].

Reference [25] proposes DC Fuse model to simulate the cascading events in power systems. This model investigates DC load flow to determine the power law behaviors in power system disturbances. A simple mesh network, which represents the network of power transmission systems, is used. In every branch, a fuse that depicts the relay system in actual power system is present. [25].

Reference [26] uses a method to implement a two-stage screening and analysis algorithm to identify multiple contingencies. Minimum change in network to move the power flow feasibility boundary is proposed in the screening process. It uses a spectral graph theory that is cast as an optimization problem. In the analysis stage, the lines that are identified in optimization program are used to identify the combination that may lead to cascading failures.

A related previous work, Reference [27], presents a screening schemes using graph partitioning to find the undesirable partitions that cause severe power imbalance, which is an indication of cascading failures. Some meta-heuristic optimization methods also known as intelligent searches [28-30] are developed to deal with most credible N-k contingencies near global minimum. Reference [28] presents a random search algorithm based on power system heuristics for a fast selection of significant blackout paths such that the most important vulnerable locations can be identified. Reference [29] developed an approach to overcome structural issues like hidden failures and failure sequence. The state space is searched for event trees that are more vulnerable and connected to healthy event trees. A genetic algorithm is used to identify the worms or sequences of state transition leading to a significant loss of load. Reference [30] presents a heuristic search using tabu search (TS) to select the most severe contingencies. It also compares TS approach with other intelligent approaches such as genetic algorithm (GA) or simulated annealing (SA) and claims that TS is a better approach for contingency selection.

Reference [31] presents a cascading collapse model to identify topological and component differences that can be applied for allocation of maintenance resources. The ordinal comparison, which is based on alignment probability, provides the theoretic basis for this model. Also, the graphic search to find the propagation of the disturbance is another basis of this model.

References [32-35] describes a methodology used in multi-level (high-order) contingency analysis and implemented in PSS®E. The methodology employs both deterministic and probabilistic reliability approaches and involves three major components, automatic contingency ranking and multi-Level contingency analysis (up to N-3), tripping action simulations and corrective action optimization and probabilistic index computation.

2.3. Brute Force Enumerative Method

Brute force method is an enumerative technique used to find the optimum solution. Enumeration is defined as a sequence listing of all the solutions satisfying the optimum condition. Finding an optimum solution using brute force algorithm is the same as doing a linear search or checking element by element to find the required solution. This technique is one of the oldest approaches used for problem solving. It is generally used if the problem size is limited or in situations where speed is of less importance. This involves inspection of each data configurations in the search space. Though it gives the most accurate solution and conceptually simple to implement, it is time consuming. The main disadvantage of this algorithm is that it cannot be used for real time situations where the data is relatively large.

2.4. Particle Swarm Optimization

Dr. Kennedy and Dr. Eberhart originally developed particle swarm optimization (PSO) in 1995 [11]. It is a population based stochastic optimization technique modeled based on swarm intelligence. The idea of this optimizer was inspired by social behavior of bird flocking. The birds travel through the whole feasible search space to find the best flowers based depending upon the objective.

The swarm consists of few randomly selected particles. The coordinates of the particle are based on the velocity and position vector associated with it [14]. The position vector (X_i) of particle i in N-dimensional search space is defined as $X_i = [x_{i1}, x_{i2}, \ldots., x_{iN}]$ and the velocity (V_i) vector of that particle is given as $V_i = [v_{i1}, v_{i2}, \ldots., v_{iN}]$. The particles interact with other particles to optimize the search experience. The iterative process of finding the best solution is initiated. The most optimum solution is calculated based on the fitness function. In each time step, fitness value is calculated by each particle. In this iterative process, the particle remembers its best position encountered till then and this position is called the p_{best} or personal best position. All the particles interact and have the ability to see if other neighboring particles are able to succeed in finding the best solution. So the best position encountered by all the particles till then is called global best or g_{best}. Fitness evaluated is compared with the population's overall previous best. If the current value is better than g_{best}, then g_{best} is reset to the current particle's array value. Each particle is accelerated towards the combination of its p_{best} and the g_{best} locations in each iteration. The PSO technique is illustrated in Figure 2.1.

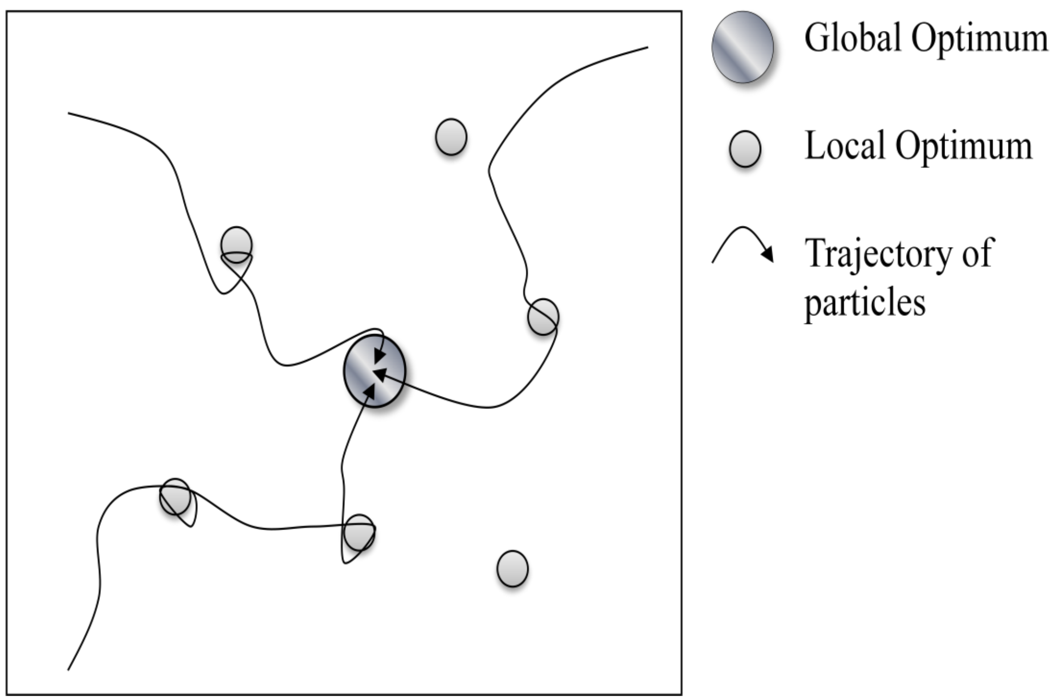

Figure 2.1 Illustration of PSO technique

The velocity and the position in every iteration get updated based on the l_{best} and g_{best} positions [14]. The expressions for updating particle position and velocity are Equations (2.1) and (2.2), respectively.

(2.1)

(2.2)

where

d = dimensions of search space;

X_{id+1} = new location of the particle i;

X_{id} = previous location of the particle i;

V_{id+1} = new velocity of the particle i;

V_{id} = previous velocity of the particle i;

w = inertial weight;

r_1, r_2 = two random numbers in [0, 1] uniformly distributed;

c_1 = cognition component;

c_2 = social component.

d is the number of variables, *i* represents the particle in the swarm, *V* is the velocity vector, *X* is the position vector, p_{best} is the local or personal best of each particle, g_{best} is global best or particle with best fitness in the neighborhood, r_1 and r_2 are random numbers usually generated between 0 and 1 and k represent the iterations number.

The Equation (2.2) has three parts. The first part reflects the memory behavior of particle and is called as inertial velocity of particle. It keeps control between the extents to which the search area is explored by particles. c_1 and c_2 are the learning factors or positive acceleration constants. The second one is cognition part that represents the movement of particle where c_1 is the cognitive acceleration factor. The third part is social behavior part that represents the particle's behavior depending upon all other particles in the population and *c2* is the social learning factor. V_{max} can be defined appropriately to prevent premature convergence of particles or exploding.

These parameters help in guiding the motion of particle by controlling how much the particle behavior and the social behavior of the neighborhood affect the particle. The accuracy and the speed of convergence to the optimal solution generally depend on these input parameters used. Also the ability to find the best solution highly depends on the initial parameters chosen. The initial particle positions also play an important role in finding an efficient solution [12]. The particles are usually attracted both to its own best solution and also the best solution of all particles. As the optimum is neared, most of the particles tend to converge or come closer.

Stopping criteria are needed to terminate the execution of optimization algorithm. Typical convergence conditions include improvement-based criteria like reaching a certain fitness value, or improvement of best objective value or average objective value of population samples. There are also movement-based criteria like reaching a certain number of iterations, movement of particles with respect to a fixed position or with respect to objective function value. More stopping criteria include distribution-based criteria, which are based on the standard deviation of positions, the maximum distance from every particle, and the difference between the best and the worst objective functions below a threshold value.

2.5. Tabu Search

The tabu search is used to solve combinatorial optimization problems. It contains a tabu list which is used to record the data during the search. This increases the efficiency and ease of accessibility when necessary. This is an expandable list whose size keeps on increasing. It is used to guide any process that provides an evaluation function for measuring the objective. The main motivation of tabu search is to have a large number of iterations, and in every iteration there is a single random pair exchange in the sequence. The heuristic makes an exchange only when the exchange improves the objective function. Also it is checked to avoid any repetitions of same exchanges that have occurred in previous steps. The tabu list can be ordered either in ascending or descending depending on the purpose. It can also be used to store the top values necessary for comparisons or evaluations. This technique has been proven to be very useful for escaping isolation phenomena as well as escaping local optima during the course of search for global optima.

2.6. DC Power Flow Analysis

There are various causes of failures and many ways in which they can be propagated. The failures could occur due to overloads, hidden failures, or oscillatory instability. In this research we are mainly focusing on the failures caused by line overloading. The load flow of the power grid is estimated based on the DC power flow approximations. The solution of power flow is used to model the steady state behavior of three-phase, balanced electric power network. The redistribution of power flow after a failure can be calculated. This gives an idea on how the system should be modified to keep secure operation under contingencies and to serve as much load as possible.

This section deals with the basics of power flows and formulation of DC power flow needed for finding voltage phase angles. Real power flow is calculated based on the obtained values. The formulations and assumptions for finding power flow equation for an N-bus system are explained in this section.

Relation between active power transported over a transmission line between nodes s and r and the complex voltages at both nodes is [13]:

(2.3)

Where

V_s = voltage at sending node

V_r = voltage at receiving node

θ_{sr} = phase angle between the voltages

X_{line} = line impedance.

The DC power flow is based on a few sensible approximations to simplify the power flow calculation. It ignores the reactive power balance equations and assumes the voltage magnitude as one per unit for all the components. Line losses and tap dependence in transformer reactance is ignored. Voltage angle differences are small. So $\sin(\theta_{sr}) \approx \theta_{sr}$.

Therefore, by ignoring Q – V relationship from normal power flow, we have line flow given by

(2.4)

And the injection power at Bus s can be written as

(2.5)

Using the Equations (2.4) and Equations (2.5),

(2.6)

Where

P = vector of real bus injection;

B' = bus susceptance matrix; and

θ = a vector of bus voltage angle.

That makes

(2.7)

Equation (2.7) is linear and will have a single solution. The B' matrix is about half the size of the full AC power flow Jacobian matrix and independent of the system state. This makes DC power flow easy to solve without having any iterations. These reasons make the DC power flow 7 to 10 times faster than the AC power flow while the error obtained using DC power flow is about 10~20% compared with AC power flow. The complexity of calculation increases with using AC power flow and has convergence problems in the cases when a line trips. Since finding the contingencies in the network has to be done within very less time, DC power flow is chosen in this thesis.

2.7. Chapter Summary

In this chapter the background of various techniques used in this thesis is discussed. The methodology based on particle optimization technique as well as tabu search is analyzed. The DC power flow approximations and algorithms are presented. Based on this analysis, an algorithm for screening high order contingencies is introduced in next chapter.

Chapter 3 Methodology for Selecting High Order Contingencies

This chapter describes the method incorporated in this thesis for selecting a group of critical, high-order contingencies. The simulation tool used in this work is MATLAB. The MATPOWER package in MATLAB is used to perform DC power flow analysis, which assists in identification of overloads in power line [16]. The main goal of MATPOWER is to provide a simulation tool within MATLAB that was easy to use and modify. Section 3.1 describes the objective of this research. Section 3.2 illustrates the formulation of algorithm to serve the required purpose. Section 3.3 describes about the fitness function required for the optimization problem. Section 3.4 gives an overview about the brute force enumeration method used for ranking the criticality of each N-1 line. Section 3.5 illustrates about the tabu search algorithm which is used along with PSO technique. Section 3.6 shows how PSO algorithm is utilized for solving the proposed problem. Section 3.7 refers to the modifications done to the original PSO algorithm. And Section 3.8 describes about the stopping criteria used in PSO algorithm.

3.1. Thesis Objective

Studies related to vulnerability of collapse are essential to maintain the power system in normal operating conditions. The system that is secure even if one component is removed is referred as stable under N-1 contingency. Such system can still operate in normal conditions even with the loss of one device. But in many cases the same system may not sustain the loss of any two or three devices. For stable operation of power system screening of N-2 and N-3 contingency is also essential. The given simulation model provides to be efficient tool for analysis of cascading failures and identifies most critical lines under N-k contingencies. N-2 and N-3 contingency events are addressed.

The objective of this work is to make use of Particle Swarm Optimization (PSO) technique. It utilizes an intelligent strategy to search the feasible solution space and identify severe high order contingency in a power system. The original PSO algorithm tends to find the best solution only. This proposed method combines the unique features of PSO and tabu search, to select a set of severe high order contingencies such that a number of top candidates can be identified. This fits the need of high order contingency screening, which can be eventually the input to many other more complicated security analyses. The method developed in this thesis work utilizes line overloading based on DC power flow.

The objective of this work is also to test and validate the developed program in MATLAB. The testing is done on the IEEE 118-bus system for ranking N-2 contingencies and on the IEEE 30-bus system for ranking N-3 contingencies. The distribution of load is calculated using DC power flow under contingency condition. DC power flow is performed with the help of package MATPOWER in MATLAB. Several cases were tested varying the parameters of PSO and an observation regarding the range of input parameters for efficiently finding the most severe contingencies in the power system is discussed. To demonstrate its robustness, the algorithm is compared to the traditional brute-force enumerative approach.

3.2. Problem Formulation

The PSO technique is applied to assist with searching the space of possible high order contingency events. It uses an objective function to weigh the optimum location as the particle searches in the test system. Certainly, there may be various ways to define the fitness function. In this research, the fitness function is defined based on overloading of the power lines. The dimensions of search space depend on the type of contingency event. If we are running PSO to rank N-2 failures, we use two dimensional search space where x and y axes represent the branches which are out. And if we are running PSO to rank N-3 failures, three dimensional search space is utilized where x, y, and z-axes represent the three branches which are out.

PSO is used to identify the critical branch combinations which could cause system overload, eventually leading to failure, when these branches are out. The original PSO is used to find only best optimal location. Hence only p_{best} and g_{best} are saved temporarily during the search. This does not meet the need of contingency selection in which a subset of all the N-2 or N-3 contingency events is desired. Therefore this technique has been modified and combined with tabu search so that a list of all the top candidates that require primary attention is maintained. The number of “top candidates” can be 100 or 1000 depending on the actual system, among all visited contingency events.

Therefore, the principle of the proposed idea can be summarized as follows:

- The PSO algorithm is used to guide all particles to traverse through possible good candidate locations (here, “good” really means a severe N-2 contingency with a high impact on line flows and system security).
- The tabu search is applied to keep track of all “good” candidates.
- When PSO stops, it means that particles have visited a sufficient number of “good” candidate locations.

The algorithm is implemented using MATLAB R2009a. Choice of MATLAB is due to its simplicity in running power flow problems. The MATLAB program that was developed specifically works with this test system. However, with some modification to the algorithm the program can be used to analyze any other system with similar topology or with any other dimension. It could be modified to suit the screening of N-k contingencies for k >= 1.

3.3. Fitness Function

Objective function for the present problem is to find the most severe contingencies that cause highest stress on the system. Here the system stress is expressed as the fitness value which determines the particle having best value in the swarm and also the best position of each particle over time. The fitness function based on overloading is defined as the root mean square ratio of line flow of overloaded lines to its line limits. This is given by

For N-k Contingency,

for N-k Contingency, (3.1)

where {OL}= the set of overloaded lines after removing branches x and y in case of N-2 contingencies or branches x, y, and z in case of N-3 contingencies. P_i^{max} is the maximum line flow limit, and P_i is the value of line flow under the N-k contingency.

Equation (3.1) is calculated based on high order contingency power flow. Therefore it is essentially an indication of how severe the contingency could be. It should be noted that here no possible control action is considered; although in reality some certain control actions may be applied. Hence, the post-contingency power flow can be viewed as a conservative evaluation of the fitness function (i.e., the impact) of the high order contingency. However, as long as the impact from every possible severe contingency is evaluated from the same conservative viewpoint, the fitness function defined in Equation (3.1) should be a fair representation of severity. PSO requires only fitness function to measure the solution quality instead of complex mathematical equations. This simplifies the computation complexity.

3.4. Brute Force Enumeration for Ranking Failures

Brute force enumerative approach is also applied to find the most critical lines in the test system in comparison to PSO for benchmarking purpose. In this method, all the combinations of N-k contingencies in the power system are verified for overloads using DC power flow and fitness function evaluations. All the fitness functions are recorded before picking out the top critical branch combinations in the power system. This method is used to analyze the movement of particles in PSO and compare the results obtained from PSO.

3.5. Tabu Search with PSO

The main idea behind a Tabu search heuristic is to have an array or list of already visited positions in the k-dimensional search space. As we iterate and move around the search space, the tabu acts as a memory and remembers all the locations encountered or traversed so far. So in each iteration, the new position is verified if it is present in the list. If the position is already visited, the particle tries to moves to its immediate neighboring position that is not visited. This increases the scope of search and also improves the efficiency of finding all the positions with high fitness value. This technique has proved to be very useful during the course of the search to escape the local maxima and to cover many locations before converging.

Tabu list gives more idea about all the branches with high impact on line flow and the operator can take necessary actions to correct the loading on the critical branches while keeping in mind all the other critical lines and he could take measures not to affect them or increase their severity. After every iteration of PSO, the list is updated as long as there are some new locations visited by particles that are better than any existing items in the tabu list. Hence, when the algorithm converges, we have a list of top candidate contingency events. Various other modifications are made to the original PSO technique to increase its efficiency and keep a balance between the speeds of convergence.

3.6. Adapting PSO to Proposed Problem

The generic PSO technique is applied to assist with searching the space of possible high order contingency events. PSO is mainly applied to unconstrained problems. It is one of the evolutionary computational techniques. The basic idea is described below with an example of N-2 contingencies that are easier to visualize and explain in a three-dimensional diagram. The searching space consists of x and y as independent variables, and z as the fitness function. Here, x and y stand for the branch IDs that are subject to contingency. For instance, if we have (x, y, ff) = (5, 201, 24.23), then it means that after Branch 5 and Branch 9 are removed from the system, the fitness function (i.e., the impact) is 24.23.

Initially, swarm consists of few particles and particles are randomly generated in the search space. The search space is bounded. The boundaries of search space depend on the number of branches in the test system. The evaluation factor for each particle is calculated using the objective function. The branches corresponding to each particle position are removed for calculation of fitness function. The fitness value of the objective function is calculated using Equation (3.1). The line flow of each overloaded line is obtained from DC power flow implementation carried out when the branches are removed. The optimization maximizes the objective function. MATPOWER is utilized for running the power flow.

The learning factors, inertial weight and initial velocity are initialized. The initial velocity is assumed to be zero in this implementation. In every iteration, each particle flies in the search space according to the velocity vector calculated based on momentum, the influence of best solution, and the best solution of its neighbors. It tries to find optimal or near optimal solution. The new velocity and position of the particle is chosen according to Equations (2.1) and (2.2). While the swarm is being updated, the boundary of search space is kept into consideration. If the new particle position violates the boundary in any dimension, the position is reset at its proper limits. The tabu list is updated with all the particles positions visited till then.

The new fitness of each particle is again calculated. The particles have memory and each particle keeps track of the previous "personal" best position, p_{best} and corresponding fitness. Another value g_{best} is the best value encountered by all the particles till then. Fitness evaluated is compared with the population's overall previous best. If the current value is better than g_{best}, then it is reset to the current particle's array value. All the particles are accelerated towards the combination of its p_{best} and the g_{best} locations in each iteration. The branches that are removed are reset back into service after every iteration. The time counter is updated in every step. If one of the stopping criteria is achieved, the iterative process comes to a stop else the whole process is repeated again. Particles move to a new position. The new location corresponds to a set of different branches that will be removed. The motion of particles can be easily demonstrated in the Figure 3.1, where X_{id}^{k} represents the present position of particle in d dimensional plane and X_{id}^{k+1} represents the position after velocity is updated. The new position depends on the velocity factor due to g_{best} and also p_{best}.

Based on Equations (2.1) and (2.2), it is easy to conclude that the convergence speed of the proposed algorithm depends on the choice of c_1 and c_2, the cognition and social coefficients, respectively.

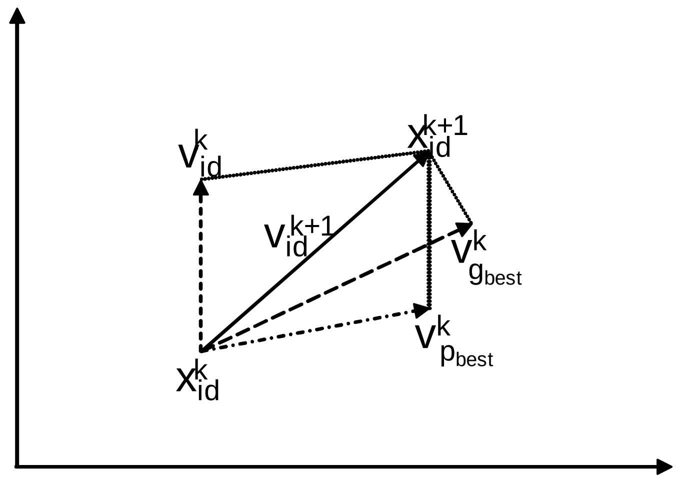

Figure 3.1. Illustration of particle motion in two-dimensional plane [14]

If c_1 is more dominant, the impact from the global best will have less impact on the future location of particles. Hence, the algorithm will traverse more spaces and take longer time to converge. In contrast, if c_2 is more dominant, then the particles are quickly attracted to the global best [14]. Hence, convergence may be faster; however, this means that the searching algorithm may miss some potential good candidates. The parameters r_1 and r_2 are randomly generated numbers between 0 and 1 to simulate the randomness of the movement of particles.

There are a few points regarding implementation that are worthwhile to mention:

- Since (x,y) represents the N-2 contingency losing Branch x and Branch y simultaneously, the order of x and y does not affect the fitness function defined in Equation (3.1). Hence, the location (x, y) implies no difference than (y, x). Therefore, we can only utilize half of the entire searching space for brute force algorithm by automatically converting (y, x) to (x, y) if y>x. This means only the lower triangle shown is needed to reduce the computing time.

- If we have x=y for a particle after a new iteration calculated using Equation (2.1) and (2.2), we simply keep the previous location as the new one since only N-2 contingency is considered.
- If we have x or y out of bound, then we can set it to the nearest boundary point.

As previously stated, when there is an initial outage of a line, the distribution of load flow is changed. A line is prone to fail if the load of the line exceeds its capacity. This may lead to more line outages because of tripping. Major blackouts are generally caused by such step-to-step process. So in this research, when the branches corresponding to particle position are removed, power flow analysis is done to determine the new distribution of load. This is repeated for every particle or for every combination of branches that are removed. According to the new power flow distribution, fitness value is calculated for each particle. This fitness value is used to evaluate the optimal location or the most critical N-k contingencies in the power system.

The overall flow chart of the proposed algorithm is illustrated in Figure 3.2. As previously mentioned, although the above discussion is based on N-2 contingencies, it can be easily applied to higher order contingencies.

3.7. Modified PSO Algorithm

A few changes are made to original PSO algorithm to decrease the computational time and at the same time give more efficient results. The original test system is modified by adding new branches to prevent islanding under contingencies which otherwise result in unusually high fitness values that may cause bias to particles while implementing particle swarm optimization. This process also helps to utilize a system with evenly distributed loading.

The order of branches of the modified test system is changed. Enumeration results with one branch removed (i.e., N-1 contingency) is performed and the fitness value is calculated for each case. These fitness values are reordered in increasing order and similarly the corresponding branches are also reordered. When this reordered system is used in testing PSO algorithm, the particles have more probability to traverse many critical branch combinations in one run and also hence this helps in reaching the most optimal solution faster.

The particles move in the search space such that they will not revisit the old position. Repetitions in the tabu list while running PSO are eliminated. In every iteration of PSO, the particle updates its position according to Equation (2.1). After the new position is obtained, it verifies the tabu list if the particle is already visited. The new position is modified if already visited. All the neighboring positions are checked for any unvisited positions by comparing the tabu list. The new position if visited is updated to the unvisited position. This helps increase the search area. The PSO searches more locations before it converges to optimal position.

The tabu list containing all the N-k contingencies traversed in the path by particles from PSO is compared to the list of most critical branch combinations obtained from the enumeration method. The percentage of matches of the candidates in tabu list is compared to that of original top list obtained from the enumeration method. The percentage of matches in tabu list obtained from PSO also found in the top list of contingencies is recorded.

Total number of N-k branch combinations visited by particles is obtained from PSO. The percentage of these particles when compared to total branch combinations is evaluated. Speedup ratio is calculated. It is the ratio between the percentage match of particles from tabu list with top list from enumeration to the percentage of total branch combinations traversed by PSO.

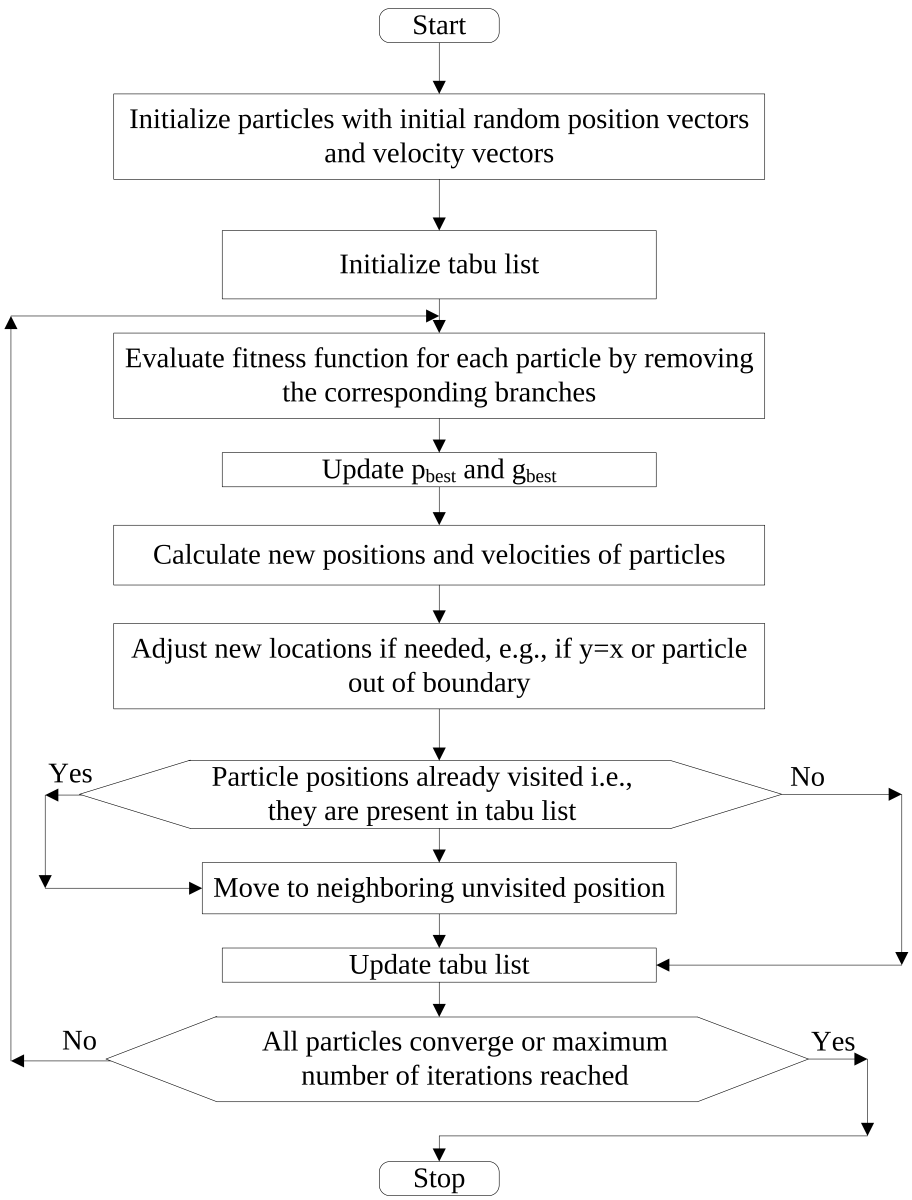

Figure 3.2 Flow chart of the proposed method.

The speedup ratio determines how much the particle searches before finding the list of critical branch combinations**.** For instance, assume we identify the true top 100 N-2 contingency events from enumeration of all 10,000 possible N-2 events. If the PSO algorithm can identify 60 of these top 100 N-2 contingency events by searching only 500 of total 10000 N-2 events. Then, the percentage of matched top events is 60/100=60%, and the percentage of the total searched events is 500/10,000=5%. This means that the algorithm searches 5% of the overall searching space and identifies 60% of the top events. Hence, the speedup ratio is equal to 60%/5%=12.

3.8. Stopping Criteria for PSO

In general the particles converge as the solution is approaching the optimal solution. The distance between the global best of all the particles and the local best of individual particles is decreased as PSO progresses. As all the particles converge, it is an indication that the global best of the particle is closer to optimal solution. The distance from every population member to the best individual is observed. Most of the particles generally converge when optimum solution is found leaving some of them still searching but they do not contribute for finding optimum best. So instead of choosing the whole population to converge, it is considered that most of the particles come closer. So only the best p% particles distance to global best is calculated. A quick sort algorithm is implemented for picking the best p% particles of total population. The percentage p should not be set too low for reliable detection of convergence. The iterative process of optimization then comes to a stop if most of the particles come closer than a tolerance value [15]. This is one of the stopping criteria chosen.

In some cases the solution is converged very fast without traveling much of the search space. Such premature convergence should be reduced by varying the input parameters in PSO. In other cases the particles may be trapped in local minima and the solution might not converge at all. Such cases that could not be converged need another stopping criterion. So maximum number of iterations the PSO can run is also considered.

3.9. Chapter Summary

An overall approach to identify and select a set of high order contingencies is provided. This approach involves an algorithm which uses particle swarm optimization technique with a tabu list to store top visited candidate contingencies to find a set of the critical k branch combinations as the event in an N-k contingency. Stopping criteria of the proposed algorithm are discussed.

Chapter 4 Results and Discussions

The test systems used to validate the potential of the proposed algorithm are modified versions of the IEEE 30-bus system for N-3 contingencies and the IEEE 118-bus test system for N-2 contingencies. It is implemented in MATLAB and the method combines the unique features of particle swarm optimization along with tabu search to select severe high order contingencies. The original PSO algorithm gives an intelligent strategy to search the feasible solution space, but tends to find the best solution only. The proposed method combines the original PSO with tabu search such that a number of top candidates will be identified. This fits the need of high order contingency screening, which can be eventually the input to many other more complicate security analyses. Section 4.1 describes the results of the brute force enumeration method implemented on IEEE 118-bus system to rank N-2 contingencies. Modifications are done to original test system. Enumeration method is performed on this modified test system to study the variation of fitness function over the whole system. Enumeration of N-1 contingencies is performed to rank the critical lines. The results are used to reorder the branches in ascending order of criticality as the pre-processing input to the proposed algorithm. Section 4.2 describes the N-2 contingency screening of the reordered system. It also discusses the variation of output with learning factors, inertial weight, initial particle positions, and stopping criteria on the output of PSO algorithm. Section 4.3 discusses the ranking of N-3 contingencies using both enumeration method and PSO algorithm along with tabu search.

4.1. Ranking N-2 Contingencies in IEEE 118-bus Test System

4.1.1. Modification of Original IEEE 118-bus Test System

Brute force enumeration is a traditional method for finding the contingencies. This method has been implemented on the IEEE 118-bus system to compare the efficiency with the present algorithm. The IEEE 118-bus test system is shown in Figure 4.1. In the original system data, there is no effective line flow limits. Hence, some modifications are necessary. Here, the base case power flow is performed. And the line limits is approximated to be 150% times of base case line flow.

The fitness function of all the branch combinations possible is obtained when any two branches are removed from the system. This process takes a lot of time for implementation but gives the most accurate results. This method is not practical due to the resources implemented, but can be used for benchmarking purpose.

In the original 118-bus system it is observed in few cases of N-2 contingency, there exists a small island because the connection of a generator or load is through a single line.

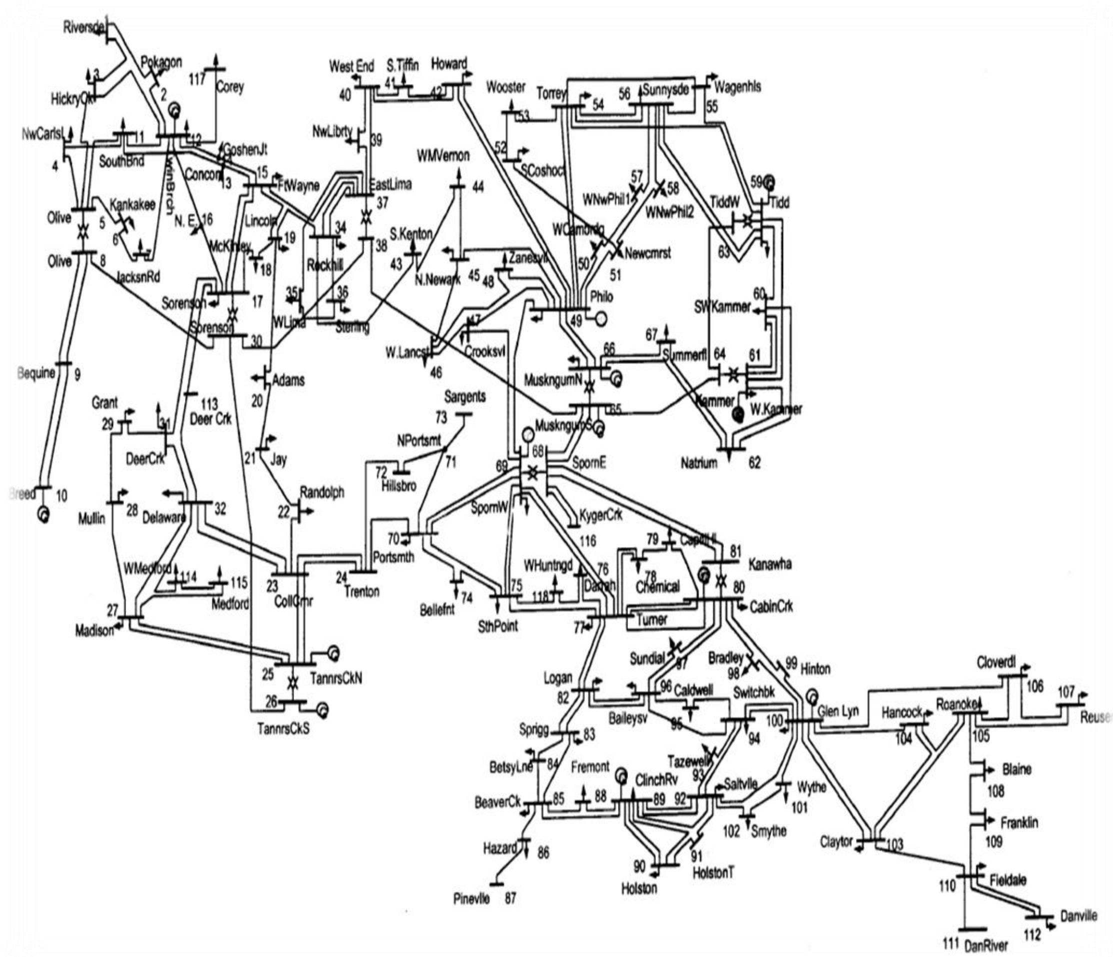

Figure 4.1 IEEE 118-bus test system

Theoretically, it is needed to have some load shedding. Since the objective of the high order contingency is to evaluate the line loading under contingency and it is hard to incorporate un-served load with line loading as the objective. A few extra lines, each in parallel to a single line connecting a generator or load, are added to avoid island under N-2 contingencies such that the algorithm can perform DC power flow and compare the fitness function in a comparable basis.

4.1.2. Enumeration on Modified IEEE 118-bus Test System

The test system with new branches added to IEEE 118-bus test system has 126 bus and 208 branches. This system is referred to as modified 118-bus test system. So the total number of N-2 branch combinations to be searched for enumeration method is $208 \times 207/2 = 21{,}528$. Figure 4.2 shows the brute force enumeration of modified 118-bus test system.

4.1.3. Implementation of PSO on Modified 118-bus Test System

The procedure described in Section 3.6 is implemented in MATLAB. Here, the particles are randomly generated in bounded test system where the two outage branches correspond to the x and y axis.

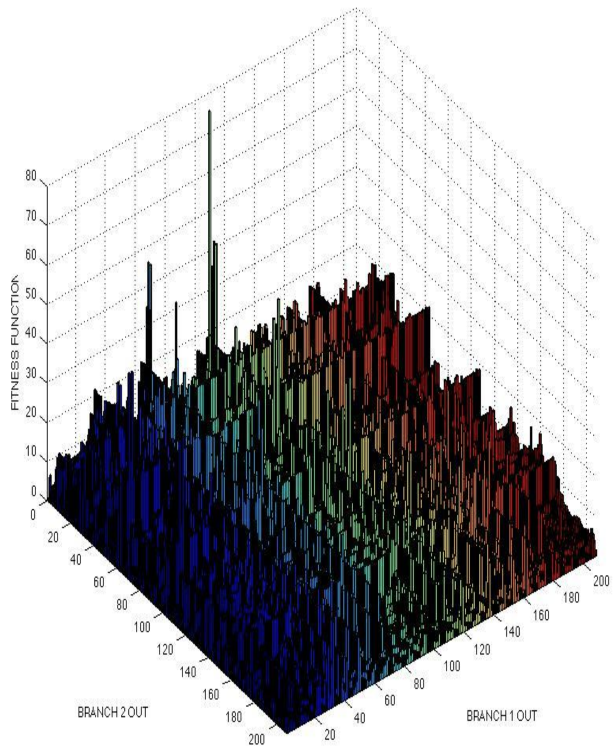

Figure 4.2 Plot of enumeration method on modified 118-bus system

The particles move around the search space trying to find a better global best value as the PSO progresses till the optimum is found. The plot for varying global best with iterations for PSO simulation on modified 118-bus system is shown in Figure 4.3.

A series of experiments has been performed with different variation of PSO parameters. In each experiment, the swarm consists consisting of 15 particles are randomly distributed in the search space. The boundaries of the particles are fixed to be (1, 208) in both x and y-axis where 208 are total number of branches in the new test system. In evolutionary programming, the global and local exploration capabilities are controlled by variances of the fitness function calculated based on loading. The input parameters are varied for the same initial particle position. Many runs are made and the variation of the capability to find the global best position is observed. When at least 10 particles out of the total 15 converge, the algorithm comes to a stop. The tolerance for stopping criteria is set to be 10, i.e., the distance between at least 10 particles out of 15 is 10 or lower when the PSO stops. If the particles do not converge, another stopping criterion should be considered. PSO is terminated when a maximum iterations of 150 is reached.

If the PSO algorithm fails to find global best position with the maximum number of iterations, it is ruled that the particles failed to find the global optimum in this run or the particles are taking longer to find the optimum.

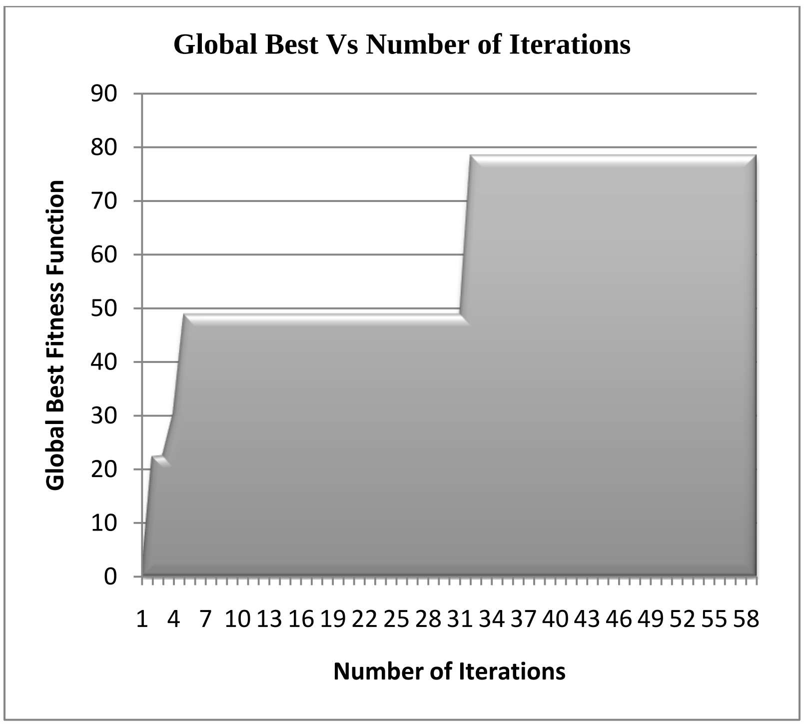

Figure 4.3 Variation of Global best with iterations in a PSO run

In some cases, even if the particles do not converge, the tabu list could be useful to find some critical N-2 contingencies leading to a reasonable speedup factor. A series of random numbers are generated in a pre-processing for r_1 and r_2 and used for all the runs so that all of them could be compared in the same basis.

The convergence of particles can be illustrated for a test case. The stages of output showing converging particles of PSO when $c_1 = 2$, $c_2 = 1.9$ and $w = 0.8$ is shown in Figure 4.4. The particles are randomly generated in the initial stage and are found to be coming closer as the particles find the global best location. When at least 70% of total particles i.e., at least 10 particles of the total 15 come closer by a tolerance of 10, the algorithm is assumed to be converged.

Results from ten PSO runs are randomly selected from a total of 100 simulations done using variable input parameters c_1, c_2 and w. All other variables like initial particle positions and random variables are kept constant. This is noted in Table 4.1. In this table, each row corresponds to a case with different PSO parameters. 10 simulation runs are selected to offset the possible odds due to the needed random numbers, r_1 and r_2 in Equation (2.2). The average of these 10 runs is reported in Table 4.2.

From the results it could be observed that in almost all the cases, PSO is not converging even after the maximum number of iterations is reached. The time taken for one run and number of N-2 branch combinations traversed is large. But still the percentage of N-2 contingencies that are matched with the original top contingency events is very less.

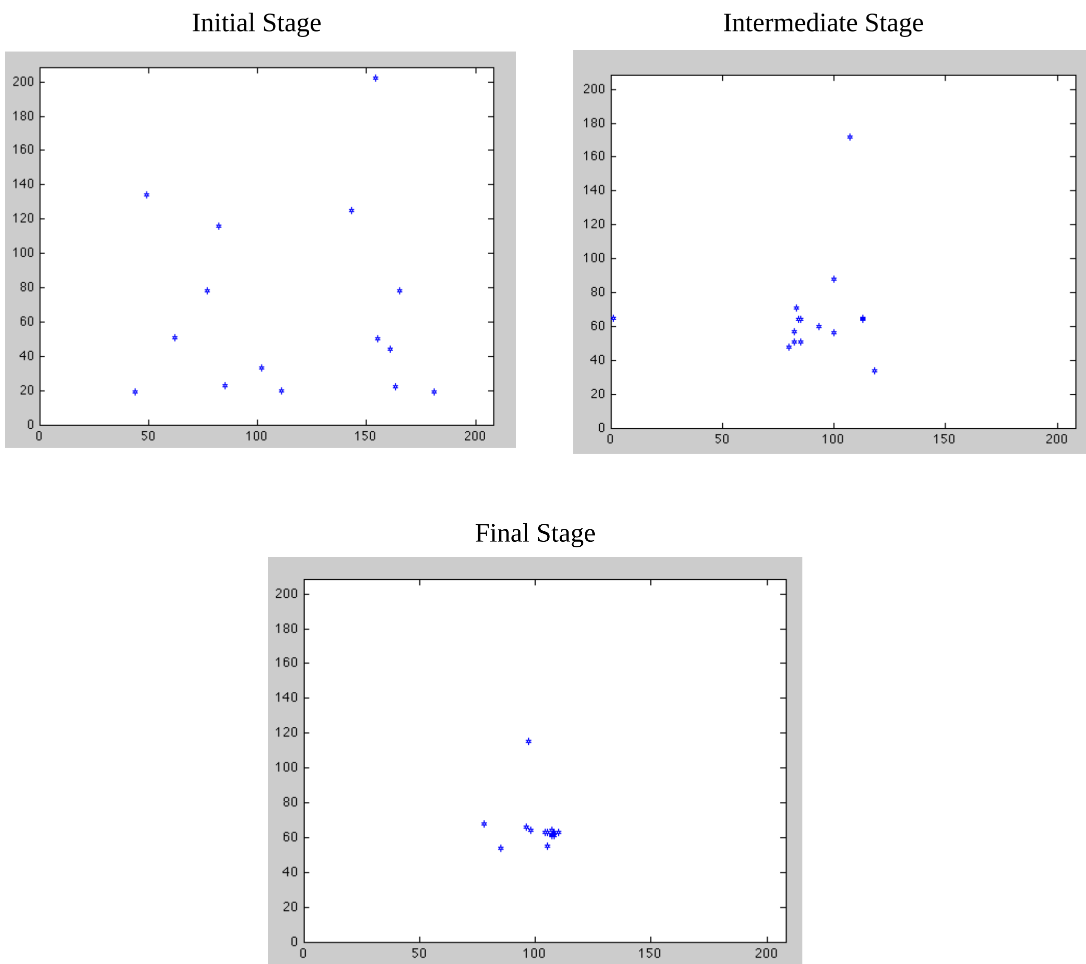

Figure 4.4 Visualization of particle motion in PSO for finding N-2 contingencies

Table 4.1 PSO results of ten random runs on modified 118-bus test system

c_1	c_2	w	N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Time (sec)
1.9	2	0.8	1399	6.50	21.2	3.26	150	93.42
1	2	1	2241	10.41	20	1.92	150	87.29
1.9	1.8	1.1	1962	9.11	8.4	0.92	150	90.73
2	1.9	0.8	1388	6.45	16.8	2.61	150	88.70
2	1.9	0.9	1798	8.35	15.6	1.87	150	88.08
6	3	1	1809	8.40	8.4	1.00	150	90.13
1.8	1.8	0.8	1542	7.16	19.2	2.68	150	94.41
1.8	1.8	1.1	2124	9.87	11.6	1.18	150	99.95
1.9	1.9	0.8	1577	7.33	16	2.18	149	89.749
1.9	1.9	0.9	1888	8.77	19.2	2.19	150	97.05

It should be noted that here the “top” contingency events or “top” combinations are defined as the top 250 combinations obtained from enumeration method. Certainly, other number can be used instead of 250 based on the operators’ experience and judgment. It depends on the size of the system. The speedup ratio is the ratio of the percentage of matched N-2 combinations with top 250 from enumeration method to the percentage of N-2 branch combinations searched. The speed up ratio is found to be in-between 1 to 3 for most of the cases which means that it is only 1 to 3 times more efficient than random search, which should have a speedup ratio of 1. The average number of branch combinations searched, speedup ratio, and time recorded based on 100 recorded runs from different PSO runs is tabulated in Table 4.2.

Table 4.2 Average of 10 random PSO runs on modified 118-bus system for N-2 contingencies

N-2 branch combinations searched	**Percentage (%) of branch combinations searched**	**Percentage (%) of matches with original top 250 combinations**	**Speedup ratio**	**Runs**	**Time (sec)**
1399	6.50	21.2	3.26	150	93.42

4.2. Selecting N-2 Contingencies in Reordered and Modified 118-bus Test System

4.2.1. Selecting N-1 Contingencies using Brute Force Enumeration

Enumeration for N-1 contingencies is implemented to rank all the lines in the modified test system based on criticality. This is very essential to make an analysis of other higher order contingencies in less duration. Since it is N-1 contingency, an enumeration is affordably evidenced by many real-time practical EMS systems. Figure 4.5 shows the plot of enumeration method and how the fitness value changes when there is one branch failure.

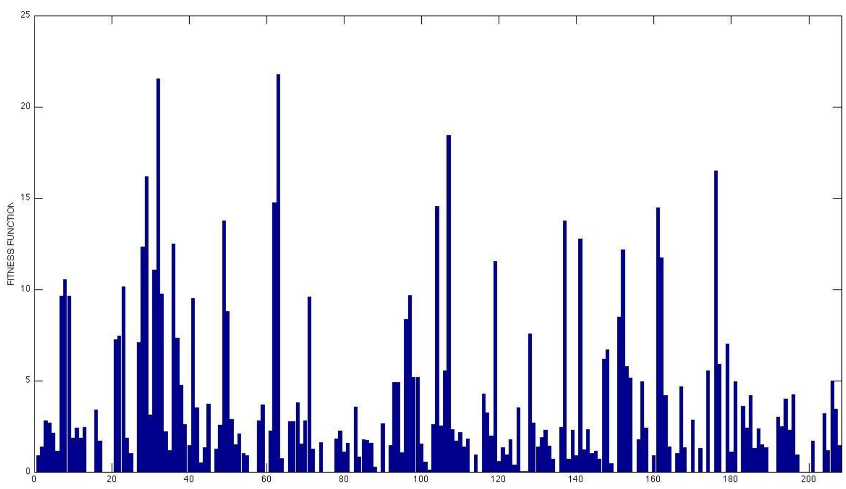

Figure 4.5 Enumeration results of N-1 contingencies of the modified 118-bus system

Next, the modified test system is further changed to increase the global search capability of PSO. The branches are then reordered either in ascending or descending order of criticality of branches. Criticality is measured depending on the fitness functions obtained from the enumeration method. Figure 4.6 shows the reordering of the branches in enumeration method. The x-axis of the plot corresponds to the new reordered branch numbers. The fitness function now varies in ascending order.

This new ordering of branches is implemented accordingly in the modified 118-bus test system. This greatly increases the efficiency of the algorithm as shown later.

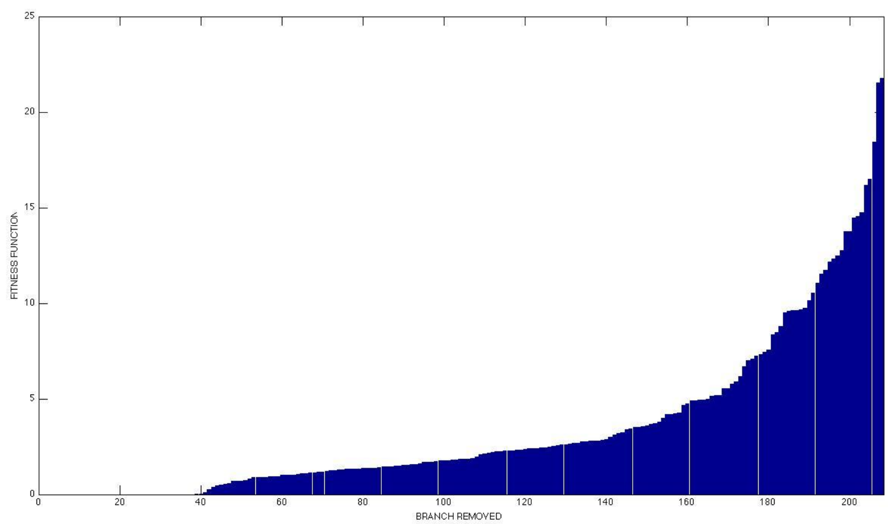

Figure 4.6 Reordered enumeration results of N-1 contingencies of the modified 118-bus system

Changing the order of branches in this way increases the percentage of critical branch combinations searched before converging and has a very high probability of finding many critical N-2 contingency events with high fitness values. The same algorithm for PSO is again implemented on this reordered system and the variation of results from the PSO on unordered system is observed. Before implementing PSO, enumeration method is conducted to observe the variation of fitness function over the whole searching space under N-2 contingencies.

4.2.2. Enumeration On Modified 118-bus System for N-2 Contingencies

Enumeration method for N-2 contingencies is implemented on the modified 118-bus system again with the branches reordered based on N-1 criticality. The total time taken to run the enumeration method for screening N-2 contingency events in modified 118-bus system is around 9 hours. This makes implementation of enumeration method impractical for screening contingencies. The enumeration plot is shown in Figure 4.7. All the N-2 contingencies consisting of the outage of two N-1 critical lines are observed to have a relatively high value of fitness function, which means relatively severe N-2 contingencies. Also, the plot is very evenly distributed in ascending order of N-1 critical lines. Since it is a smooth plot the particles are more attracted towards the corner with high fitness values sooner. This reduces the total duration for each run of PSO and gives more efficient results.

Implementing PSO on the present system can have more scope to move through more critical branches in its path before convergence. Also the tabu could have more number of critical branches in less time. Since time is also a constraint, reordering branches is implemented to get a tabu of most critical N-2 branch combinations.

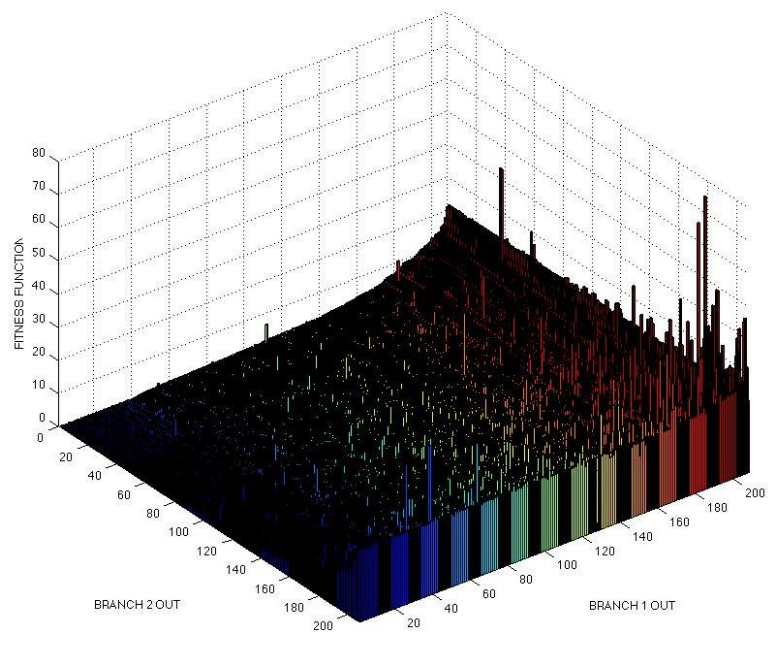

Figure 4.7 Plot of enumeration method for modified and reordered 118-bus system

4.2.3. N-2 Contingency Selection Using PSO for Reordered 118-bus Test System

Ten different experiments are conducted using the proposed PSO-based algorithm and the reordered 118-bus test system. The same values of c_1, c_2, and w used for making experimental results with the unordered system are considered for simulation done on reordered system also. The difference in the convergence properties found when compared to the unordered system is that the global search capability of particles is reduced. The particles are pulled towards the one corner with higher fitness values. All the particles then start moving around in that area. This results in the particles moving around the areas where there are more critical combinations. This gives more probability to find all the critical contingencies. The percentage of particles matched in tabu list with that of top 250 from enumeration method is increased within shorter duration of time. The simulations are run to prove the benefits of reordering.

The initial particles position and the stopping criteria are also the same as the previous experiment. This process of the proposed algorithm is iterated until sufficient solution quality or the maximum number of iterations is reached. The results obtained from random ten PSO runs with different input parameters are tabulated in Table 4.3. Observing the results in Table 4.3 obtained from the proposed algorithm after reordering and comparing them to the results from Table 4.1, it is noted that the convergence properties are more visible after reordering. Particles are converging much faster and the speedup ratio is also greatly increased. This means that larger chance of finding more number of critical N-2 branch combinations within lesser time and lesser number of iterations.

The average number of N-2 combinations, percentage match, speedup, iterations, and time are as shown in Table 4.4. Comparing it with average results obtained from PSO of unordered system, the total number of searched N-2 branch combinations is much less but the percentage match of branch combinations in tabu list with that of the original top 250 from enumeration method is much more, i.e. 62.16.

The speedup ratio has also significantly increased even though the total number of iterations is much less comparatively. The total time is also less. The time should be preferable low because it is always desired for short-term system operators to have a quick scan of potential risky contingencies and take preventive actions as soon as possible. The results show that the proposed algorithm with reordering can meet the motivation of this research work very well such as not only to find the most critical branches but to identify a set of severe N-k contingencies. Reordering gives significant improvement in PSO output over the unordered system

Table 4.3 PSO results for N-2 contingencies of ten random runs using the modified 118-bus system

c_1	c_2	w	N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Time (sec)
1.9	2	0.8	443	2.06	66.8	32.46	71	36.22
1	2	1	424	1.97	66	33.51	89	45.97
1.9	1.8	1.1	402	1.87	71.2	38.13	123	51.65
2	1.9	0.8	475	2.21	59.2	26.83	59	29.09
2	1.9	0.9	454	2.11	66.8	31.68	68	29.79
6	3	1	192	0.89	46	51.58	24	10.11
1.8	1.8	0.8	569	2.64	58.8	22.25	81	34.66
1.8	1.8	1.1	415	1.93	70.4	36.52	150	67.88
1.9	1.9	0.8	559	2.60	52.8	20.33	90	50.11
1.9	1.9	0.9	395	1.83	63.6	34.66	79	46.34

.

Table 4.4 Comparison of average output of 10 PSO runs for N-2 contingencies

	N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Time (sec)
Unordered System	1399	6.50	21.2	3.26	150	93.42
Reordered System	433	2.01	62.16	32.79	83.4	40.18

Figure 4.8 shows the plot between accuracy and efficiency with the number of iterations based on 100 experimental runs with varying input parameters. As the number of iterations increases beyond a limit, efficiency of algorithm decreases as computational time increases. Total number of iterations is generally chosen as a tradeoff between accuracy and efficiency.

The result variation of the PSO runs with different input parameters is discussed in the following sections. The experiments are conducted by varying only one parameter after another.

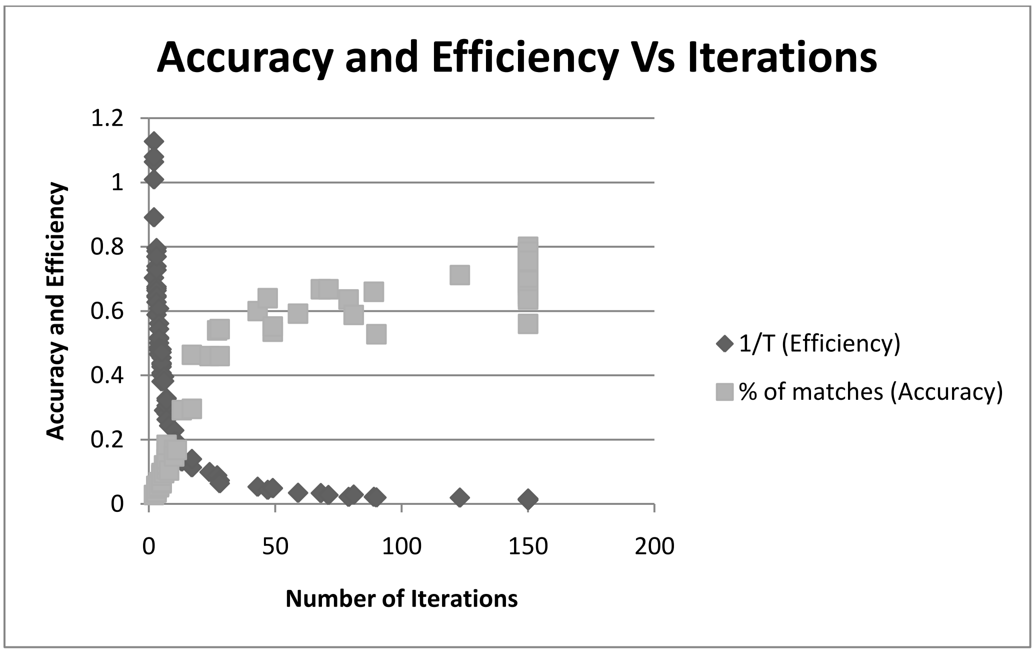

Figure 4.8 Plot between accuracy and efficiency vs number of iterations

All other parameters values are kept constant. The value of basic PSO run used for experimenting is c_1 = 2, c_2 = 1.9 and w = 0.8. The maximum number of iterations are 150 and 10 is tolerance between particles during convergence. 10 particles are assumed to come closer out of total of 15 particles for stopping PSO run.

4.2.3.1. Varying Learning Factors

Based on a number of random runs varying the values of c_1 and c_2, the typical range of c_1 and c_2 that could give more efficient results for ranking N-2 contingencies is observed. The simulations run with c_1 and c_2 varying from 1.8 to 2 gives more accurate results. These observations are in particular with the initial conditions having 15 initial random particles, tolerance for ten particles to come together is 10 and maximum number of iterations is 150. All the runs are performed on the same initial particle positions to avoid the result variation due to initial positions. The observed results are tabulated in Table 4.5. It is observed that the speedup ratio is higher in the cases where c_1 is greater than c_2 and also where $c_1 = c_2$. Also, a series of random numbers r_1 and r_2 between 0 and 1 are generated as pre-processing and fixed for later use such that all runs are based on the same series of random numbers.

From the results observed in Table 4.5, it can be easily identified that c_1 and c_2 are around 1.8 to 2 for most of the cases that give good results. The values of c_1 and c_2

chosen around 2 give results with high speed-up ratio. In this experiment of runs, some other cases where c_1 is greater than c_2 are found to give good results as well. These results include the values of c_1 in the range of 4 to 6 and c_2 to be 2 to 4.

4.2.3.2. Varying Inertia Weight

The inertial weight is an important factor in determining how the particles travel in the search space. It keeps a balance between exploration and exploitation in the search space. Inertial weight acts as a memory of the particle. It remembers the velocity of particle in the present iteration and controls its velocity in the next iteration.

The variation of the PSO search capability with several representative *w* values is recorded in Table 4.6 where the input parameters are c_1 = 1.9 and c_2 = 1.9. As it is observed from the table, as *w* increases from 0.6 to 1.8, PSO converges very fast in only 4 Iterations. The larger *w* results in faster convergence. The global search capability increases and performs search over wide area. It has more exploration capabilities over the whole search area. As *w* decreases, the local search capabilities are increased and convergence speed is decreased. When *w* is around 0.6, the PSO algorithm takes a very long time for convergence. It does not converge for even 150 iterations. So the value of *w* to be chosen for better performance is decided by selection a run which implements for a reasonable number of iterations before finding optimum solution.

Table 4.5 Observations made by varying learning factors

c_1	c_2	N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Time (sec)
1.9	2	442	2.05	70.4	34.29	150	76.39
1.8	2	520	2.42	73.2	30.30	150	76.07
1	2	424	1.97	66	33.51	89	45.97
2	3	308	1.43	56	39.14	150	73.25
2	1.8	476	2.21	71.2	32.20	150	72.34
4	1	669	3.11	80	25.74	150	74.69
4	3	276	1.28	53.6	41.81	49	20.45
5	4	467	2.17	74	34.11	150	62.80
1.9	1.9	130	0.60	29.2	48.36	13	7.60
2	2	84	0.39	18.4	47.16	7	3.80

Table 4.6 Observations made by varying inertia weight

w	N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Time (sec)
0.6	782	3.63	64.8	17.83	150	97.37
0.8	559	2.60	52.8	20.33	90	50.11
0.9	395	1.83	63.6	34.66	79	46.34
1	130	0.60	29.2	48.36	13	7.60
1.2	67	0.31	11.2	35.98	6	2.72
1.4	62	0.28	11.2	38.88	6	2.72
1.6	50	0.23	8.4	36.16	4	1.77
1.8	48	0.22	7.2	32.29	4	1.80

The PSO simulation must have a good local search capability near the corner of search space having high fitness values and at the same time the run should not take much time. With the results from the runs corresponding to around 30 recorded values of *w*, it is observed that the PSO gives a greater search capability with reasonable number of iterations for *w* ranging from 0.8 till 1.

4.2.3.3. Varying Stopping Criteria

Runs are conducted by varying the tolerance and the number of particles that converge by coming close. The results are tabulated and observations are made on how the particles converge. All the runs are made for c_1 = 2, c_2 = 1.9, and w = 0.8, same initial particle generation and random numbers.

a. Varying Tolerance Between Particles

The tolerance is modified in each run and the results are recorded in Table 4.7. It is observed that as tolerance increases the convergence becomes faster and so is the percentage match with the top 250 results of enumeration method. Speedup ratio is observed to be decreasing as tolerance increases. This is because with a very small value of tolerance, the particles move around through different locations in search space for more duration and in some instances the particles could get struck up local maxima and they could even end up not converging. If the tolerance is increased, there is more probability that the PSO run stops even if the particles do not converge. So the tolerance must be chosen not too high or too low such that the particles get more scope to search within less time.

b. Varying the Number of Particles Converged

During PSO implementation, the randomly generated particles slowly come closer as they find the optimum solution. But in some cases few particles do not converge and are struck up moving around its local best. The total number of particles considered in this run is 15. Results are recorded varying the number of particles that come closer for convergence.

Table 4.7 Observations made by varying tolerance between particles

Tolerance	N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Time (sec)
5	682	3.16	61.6	19.44	150	67.74
10	475	2.21	59.2	26.83	59	29.09
15	218	1.01	46.4	45.82	18	7.52
20	93	0.43	19.6	45.37	7	2.74

If the particles coming closer are chosen to be very small, there are more chances of premature convergence and the particles have less scope to move around the search space before convergence. The time for the execution is very short with a reduced accuracy. As the number of particles coming closer is increased, the running time increases and the particles get a chance to move around in a broader region before finding optimal location. It could be observed from Table 4.8 that consideration of 13 particles coming closer out of total 15 particles may not lead to convergence as in most of the cases. Few particles do not converge as they are stuck in local maxima. So the selected number of particles must be a tradeoff between both these categories. In the present simulation 10 particles are chosen as a tradeoff between running time and accuracy.

c. Another Stopping Criteria – Local Bests Of Particles Converge

Observations of the results obtained from PSO are made by varying the stopping criterion. Instead of the general criterion proposed with the algorithm where most of the particles come closer for convergence, a new stopping criterion involving all the local bests of particles come closer is chosen.

Table 4.8 Observations made by varying number of particles converging

Particles converged	N-2 branch combina-tions searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speed-up ratio	Runs	Time (sec)	Tolerance
13	682	3.16	61.6	19.44	150	91.00	10
12	627	2.91	60.8	20.88	112	66.65	10
10	475	2.21	59.2	26.83	59	39.71	10
9	470	2.183	59.2	27.12	58	34.38	10
7	153	0.710	36.4	51.22	12	6.86	10
6	79	0.366	14.4	39.24	6	3.42	10

Table 4.9 records the average results obtained from 100 random PSO runs considering that local bests of particles come closer for convergence. It is compared to the average of 100 PSO runs with same input parameters where the particles come closer for convergence. It is observed that considering distance between local bests of particles as stopping criteria gives slightly higher speedup ratio but the convergence occurs fast. The percentage of combinations searched and percentage of match found compared to top 250 critical branch combinations from enumeration N-2 method is also less. Since we need to search larger area before convergence, the particles coming closer is considered as better stopping criteria for this purpose.

4.2.3.4. Varying Initial Particle Positions

The impact of the initial particle positions are studied here. All the higher fitness function values are towards one corner in the searching space. If the random particles are more towards the corner with higher fitness values result in premature convergence. Particles towards the corner of search space with lower fitness values may not converge to the optimum position but may be trapped in the particles local best positions. So the best particles positions would be randomly distributed over all the search space. Some observations made with different particle initial positions but with same initial conditions are displayed in the Table 4.10.

Table 4.9 Comparison of average output of 100 PSO runs on modified 118-bus system for N-2 contingencies with different stopping criteria

Stopping Criteria	N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Time (sec)
Local bests of particles come closer	167.07	0.78	27.00	36.48	23.76	13.20
Particles come Closer	272.36	1.27	36.65	31.80	59.50	28.76

4.2.3.5. Varying Number of Particles

Selection of particles provides tradeoff between time and global search capability. More number of particles would give more accurate tabu list with many critical branches. But considering the time taken for all the particles to move in the search space and to converge, the particles used for searching are sufficient to provide a good list of critical branches within reasonable amount of time. The number of particles also depends on the size of the search space. Larger the area more particles are necessary to travel towards most of the critical contingencies.

Table 4.10 Observations made by varying initial particle position

	N-2 branch combinations searched	**Percentage (%) of branch combinations searched**	**Percentage (%) of matches with original top 250 combinations**	**Speedup ratio**	**Runs**	**Time (sec)**
Randomly distributed particles	395	1.83	58.4	31.83	72	30.25
Some towards low ff corner and others towards high ff corner	849	3.94	50.4	12.78	150	66.75
More towards high ff corner	110	0.51	23.6	46.19	9	6.55
Towards high ff corner	186	0.86	31.6	36.57	18	19.38
More towards low ff corner	437	2.02	0.40	0.19	57	36.36
Towards low ff corner	96	0.44	2	4.48	7	4.85

As observed from Table 4.11, selecting 15 particles is sufficient to provide a reasonable speedup ratio of 26.83 within 39.71 seconds search time. So the number of particles chosen in this simulation is 15. Table 4.11 shows the variations of the simulations with initial number of particles and their positions. In a similar manner some offline studies can be performed on any system before implementing PSO algorithm. This may give a good idea on the total number of particles sufficient to give good results within reasonable amount of time. This can be used for real time applications in the same system.

Table 4.11 Observations made by varying number of particles

N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Number of Initial Particles	Time (sec)
1585	7.36	78.4	10.65	150	25	134.56
889	4.12	73.6	17.82	150	20	113.73
475	2.21	59.2	26.83	59	15	39.71
114	0.53	23.6	44.57	23	10	10.55
32	0.15	6	40.37	9	5	2.88
9	0.04	0.8	19.14	3	3	0.90

4.3. Ranking Critical N-3 Contingencies

The same process of screening high order contingencies implemented for N-2 contingencies can also be implemented to third order contingencies. The test system used for testing N-3 contingencies is IEEE 30-bus test system. Figure 4.9 shows the original test system. Enumeration method is implemented for N-3 contingencies and the IEEE 30-bus test system is modified in a similar manner by adding new branches and buses to prevent islanding. The modified IEEE 30-bus test system has 75 buses and 136 branches.

The total number of N-3 branch combinations in the whole test system are (136×135×134)/(3×2×1) = 410,040. Enumeration method is performed to rank N-3 contingency events and the running time is approximately more than 24 hours. This is impractical to use in real time situations. This result from enumeration method is used to measure the accuracy of PSO algorithm. The ranking of criticality of branches is obtained from enumeration N-1 method on the modified 30-bus system. All the branches in the test system are reordered and this increases efficiency of PSO algorithm.

Particle searches through a three dimensional search space where x, y, and z coordinates represent the branches that are removed from the service. Some stages of output in one of the PSO case with c_1 = 1.8, c_2 = 1.9, and w = 0.9 is visualized in Figure 4.10. The particles are randomly initialized in initial stage and they are found to be converging as they reach global best location.

IEEE 30 – BUS TEST SYSTEM

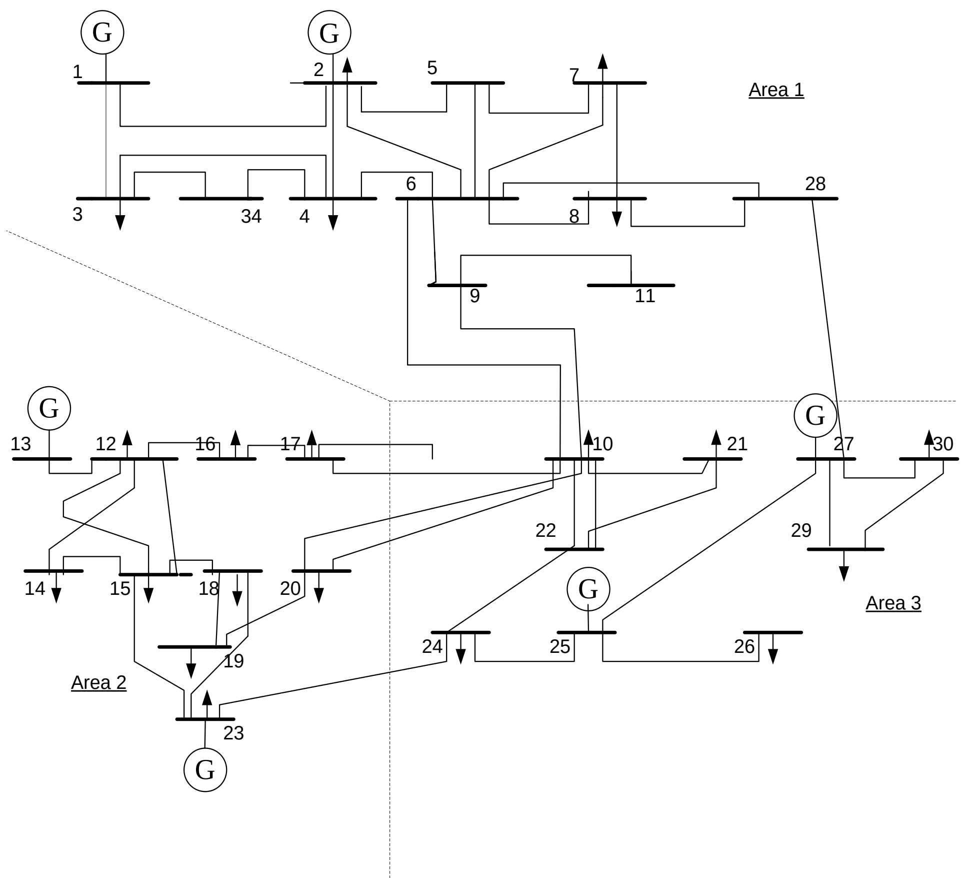

Figure 4.9 IEEE 30 – Bus Test System

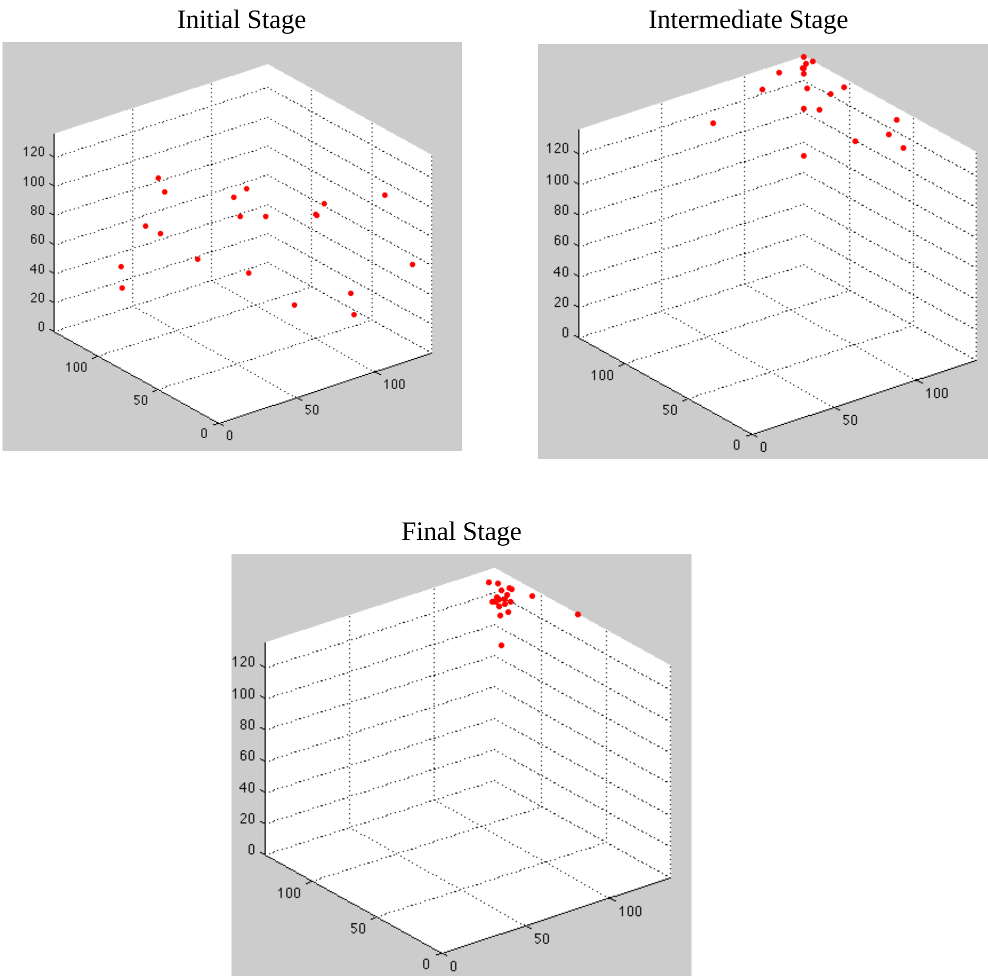

Figure 4.10 Visualization of stages of PSO for screening N-3 contingencies

The statistical results reveal about the input parameters that give efficient list of critical branch combinations. Similar to the previous N-2 contingency test, c_1 and c_2 in the range of 1.8 to 2 appear to have good convergence rate and probability of finding better solution. The variation of input parameters is similarly done for observing the change in output of PSO for screening N-3 contingencies. The speedup ratio for finding N-2 contingencies is better than the ones obtained by N-3 contingencies. This could be due to the increased searching space**.** The algorithm has more computational time and more CPU time compared to the results from Table 4.3 which are implemented for screening N-2 contingencies over a smaller searching space. Depending on the system size the initial random positions are chosen. In this experiment, an initial population size of 20 particles is chosen. Based on 100 experimental runs with 150 maximum iterations in each run, the efficient tolerance value while converging is selected to be 15 and the number of particles coming closer when optimum is found is set as 14. The value of inertial weight around 0.9 and 1 has more search capability. The results are observed for various input parameters and some of the cases with best realization are recorded in Table 4.12.

The results in the Table 4.12 and Table 4.13 reflect a low percentage of match with top 250 contingency events, although the speed up ratio is still very good. However, more researches are necessary to improve the percentage match because the ultimate goal is to identify reasonably more contingencies.

Table 4.12 PSO results obtained for ten random runs for N-3 contingencies

c_1	c_2	w	N-2 branch combina -tions searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speed -up ratio	Iterations	Time (sec)
1.8	1.9	0.9	922	0.22	3.94	17.52	69	51.16
1.8	1.9	1	1261	0.3	7.42	24.12	132	96
1.9	2	1	1135	0.27	6.59	23.8	150	129.43
1	2	1	976	0.23	5.66	23.77	150	125.51
2	1.9	1	956	0.23	4.68	20.08	150	113.59
1.1	1	1	1299	0.31	7.04	22.22	150	131.19
4	1	1	1196	0.29	6.41	21.97	150	131.48
1.8	1.8	0.9	512	0.12	1.42	11.37	66	59.43
1.9	1.9	0.9	896	0.21	4.57	20.91	150	116.63
1.9	1.9	1	1307	0.31	7.27	22.8	150	130.12

The average of 100 PSO runs with varying input parameters is showed in Table 4.13.

Table 4.13 Average PSO parameters based on 100 runs on modified and reordered 30-bus system for N-3 contingencies

	N-2 branch combinations searched	Percentage (%) of branch combinations searched	Percentage (%) of matches with original top 250 combinations	Speedup ratio	Runs	Time (sec)
Reordered System	1046	0.249	5.5	20.856	131.7	10582

Likely, this means more fine-tuning in the future work for the PSO parameters such that a slower search will be achieved to explore more contingency events.

4.4. Chapter Summary

In this chapter the results obtained proved the efficiency of algorithm to screen high order contingencies. The efficiency of the algorithm has been explained with the help of observations made. In the next chapter conclusion is drawn from the thesis and illustrated along with the scope of possible future improvements.

Chapter 5 Conclusions and Future Work

5.1. Conclusion

The original PSO algorithm gives an intelligent strategy to search the feasible solution space to find the most critical high order contingency. PSO combined with tabu search proposed in this thesis is very useful to record a list of critical high order contingencies when the particles traverse in a space, where the (x, y) coordinates represent outage of Line x and Line y. The tabu list along with PSO is very effective in keeping a record of many critical contingency events. This makes the proposed algorithm different from conventional PSO algorithm which seeks the best solution only, while the proposed algorithm seeks a set of top solutions, i.e., critical contingency events.

Reordering branches of test system based on severity of N-1 contingencies is observed to be very useful to increase the convergence properties and efficiency of the algorithm. Global search capability of the algorithm is reduced and local search capability is increased. Reordering of the test systems collects many critical contingencies in a small area in the whole test system. PSO concentrates in searching this location more which increases the number of critical branch combinations searched. Therefore, the speedup ratio is found to increase significantly.

The proposed PSO-based algorithm displays good performance in terms of solution quality, computational costs, and convergence stability. PSO has proved to be very efficient and takes less time and less CPU usage compared to using brute force enumeration method. The impact on results obtained from PSO with variations in different input parameters is studied. Variation of inertia weight, learning factors, and number of particles is tested and the range of values more suitable for this specific algorithm is suggested. PSO is found to be advantageous due to its simplicity of implementation and capability of parallel search.

The proposed algorithm is tested for N-2 and N-3 contingencies using two test systems modified from the IEEE 118-bus and 30-bus systems. It can be extended to other higher order contingencies also but visualization could be difficult because of the increases of the problem dimensions corresponding to the order of contingencies.

In summary, the contribution of this thesis can be given as:

- An efficient searching algorithm based on PSO and tabu search is proposed to identify a set of most severe high order contingency events.
- A reordering approach as a preprocessing of the proposed searching algorithm is applied to sharply increase the solution quality and efficiency of the proposed algorithm.
- A comparison study of the running time and accuracy when different parameters of the PSO-based algorithm are applied.

5.2. Future work

Future work includes the consideration of possible corrective actions under high order contingency events such as re-dispatching generations, calling reserves, load shedding, etc. Also, a full AC-based model may be applied such that voltage magnitudes need to be considered and the corrective action may include increasing generation excitation, switching on reactive compensators, and adjusting load tap changers. This shall give more accurate results. Also, more PSO parameter tuning may be studied for N-3 and higher order contingency events to improve the percentage match because the ultimate goal is to identify more critical contingencies.

5.3. Chapter Summary

This chapter presents the overview and the contribution of the thesis. Possible future work to broaden the scope of usage of the proposed algorithm is discussed.

REFERENCES

[1] M. Anghel, K.A. Werley, A.E. Motter, "Stochastic model for power grid dynamics", *40th Hawaii International Conference System Sciences,* Hawaii, Jan 2007.

[2] S. Tamronglak, S. H. Horowitz, A. G. Phadke, and J. S. Thorp, "Anatomy of Power System Blackouts: Preventive Relaying Strategies," *IEEE Trans. on Power Delivery,* vol. 11, no. 2, April 1996.

[3] IEEE PES CAMS Taks Force, "Initial review of methods for cascading failure analysis in electric power transmission systems," *IEEE PES General Meeting 2008*.

[4] Thomas Nye, Chen-Ching Liu, and Michel Hofmann "Adaptation of relay operations in real time," *15th PSCC*, Leige, 22nd August 2005.

[5] R.C. Hardiman, M.T. Kumbale, and Y.V. Makarov, "An advanced tool for analyzing multiple cascading failures," *8th International Conference on Probability Methods Applied to Power Systems (PMAPS)*, Ames Iowa, September 2004.

[6] Q.Chen and J.D.McCalley, "Identififying high risk n-k contingencies for online security assessment", *IEEE Trans. Power Systems*, vol. 20, no. 2, pp. 823 – 834, May 2005.

[7] Bei Gou, Hui Zheng, Weibaio Wu, Xingbin Yu, "The statistical law of power system blackouts", *38th North American Power Symposium*, 2006, pp. 495-501

[8] Scott Greene, Ian Dobson, Fernando L. Alvarado, “Contingency Ranking for Voltage Collapse via Sensitivities from a Single Nose Curve”, *IEEE Trans. Power Systems*, vol. 14, pp. 232 – 240, 1999

[9] I. Dobson, B.A. Carreras, V.E. Lynch, D.E. Newman, “Complex systems analysis of series of blackouts: cascading failures, critical points, and self-organization”, *Chaos.*,vol. 17, June 2007

[10] Ian Dobson, Benjamin A. Carreras, David E. Newman, “A loading-dependent model of probabilistic cascading failure”, *Probability in Engineering and Informational Sciences*, 2005, pp. 15 – 32.

[11] J. Kennedy and R. C. Eberhart, “A new optimizer using paerticle swarm theory”, *Proceedings of the Sixth International Symposium on Micro Machine and Human Science.*, 1995, pp. 39-43

[12] Carlisle, A. and Dozier, G. Tracking changing extrema with particle swarm optimizer. Technical Report CSSE01-08. 2001. Auburn University.

[13] A.J. Wood, B.F. Wollenberg, Power System Generation, Operation and Control, Wiley, New York, 1984.

[14] H. Yoshida, Y. Fukuyama, S. Takayama and Y. Nakanishi., “A particle swarm optimization for reactive power and voltage control in electric power systems considering voltage security assessment*.”, Proceedings of the IEEE SMC '99 Conf*., Vol. 6, pp. 497 -502, 1999.

[15] K. Zielinski, R. Laur, "Stopping Criteria for a constrained single-objective particle swarm optimization algorithm", *Informatica*, pp. 51- 59, March, 2007.

[16] R. D. Zimmerman, C. E. Murillo-Sanchez, and D. Gan, MATPOWER 3.0.0 User's Manual. [Online]. Available: http://www.pserc.cornell.edu/matpower.

[17] R.C. Hardiman, M.T. Kumbale, and Y.V. Makarov, "An advanced tool for analyzing multiple cascading failures," 8th International Conference on Probability Methods Applied to Power Systems (PMAPS), Ames Iowa, September 2004.

[18] B.A. Carreras, V.E. Lynch, I. Dobson, D.E. Newman, "Critical points and transitions in an electric power transmission model for cascading failure blackouts," Chaos, vol .12, no. 4, Dec. 2002, pp. 985-994. ……………..

[19] I. Dobson, B.A. Carreras, D.E. Newman, "Branching process models for the exponentially increasing portions of cascading failure blackouts," 38th Hawaii International Conference on System Sciences, January 2005, Hawaii.

[20] I. Dobson, B.A. Carreras, D.E. Newman, "A branching process approximation to cascading load-dependent system failure," 37th Hawaii International Conference on System Sciences, Hawaii, January 2004.

[21] I. Dobson, K.R. Wierzbicki, B.A. Carreras, V.E. Lynch, and D.E. Newman, "An estimator of propagation of cascading failure", 39th Hawaii International Conference on System Sciences, January 2006, Kauai, Hawaii.

[22] D.P. Nedic, I. Dobson, D.S. Kirschen, B.A. Carreras, V.E. Lynch, “Criticality in a cascading failure blackout model”, International Journal of Electrical Power and Energy Systems, vol. 28, 2006, pp. 627-633.

[23] Rios M.A., Kirschen D.S., Jayaweera D, Nedic D.P., Allan R.N. “Value of security: modeling time-dependent phenomena and weather conditions,” IEEE Trans on Power Syst, 2002, 17(3), pp. 543-548.

[24] Kirschen, D.S., Jayaweera, D., Nedic, D.P., and Allan, R.N., “A probabilistic indicator of system stress,” IEEE Trans on Power Systems, vol. 19, no. 3, Aug. 2004, pp. 1650 – 1657.

[25] J. Chen, J.S. Thorp, M. Parashar, “Analysis of electric power system disturbance data,” 34th Hawaii International Conference on System Sciences, Maui, Hawaii, Jan 2001.

[26] V. Donde, V. Lopez, B. Lesieutre, A. Pinar, C. Yang, and J. Meza, “Identification of severe multiple contingencies in electric power networks,” 37th North American Power Symposium, Ames, Iowa, 2005

[27] B.C. Lesieutre, S.Roy, V.Donde, and A.Pinar, “Power system extreme event screening using graph partitioning,” 38th North American Power Symposium, Sept. 2006.

[28] H. Wang, J. S. Thorp, “Optimal locations for protection system enhancement: A simulation of cascading outages,” IEEE Trans. Power Delivery, vol. 16, no. 4, October 2001, pp. 528-533.

[29] S. J. Ranade, R. Kolluru, and J. Mitra, “Identification of chains of events leading to catastrophic failures of power system,” International Symposium on Circuits and Systems, pp. 4187-4190, 2005.

[30] Hiroyuki Mori and Yuichiro Goto, “A Tabu Search Based Approach to (N-k) Static Contingency Selection in Power Systems,” IEEE International Conference on Systems, Man, and Cybernetics, 2001, pp. 1954-1959.

[31] Pepyne, D.L.; Panayiotou, C.G.; Cassandras, C.G.; and Ho, Y.-C, “Vulnerability assessment and allocation of protection resources in power systems,” American Control Conference, 2001.

[32] Xiaokang Xu, Feng Dong and Michael J. S. Edmonds, “Application of Risk Assessment in Transmission Planning Studies”, presented and in Proceedings of the 17th Conference of the Electric Power Supply Industry, 27-31 October 2008, Macau SAR, China.

[33] Xiaokang Xu, Feng Dong and Michael J. S. Edmonds, “Risk Assessment of Power Systems with PSS®E”, Siemens PTI eNewsletter, Power Technology, Issue 101, February 2008. (Also appeared in the Energy Central Website: www.energycentral.com).

[34] Xiaokang Xu and Michael J. S, Edmonds, "Probabilistic Reliability Methods and Tools for Transmission Planning and System Analysis", in Proceedings of the 9th International Conference on Probabilistic Methods Applied to Power Systems (PMAPS 2006), June 11-15, 2006, Stockholm, Sweden.

[35] Feng Dong and Baldwin P. Lam, "Bulk Electric System Reliability", Siemens PTI eNewsletter. Power Technology, September 2006.

APPENDICES

Appendix A

Test System Case File Corresponding to Modified 30-Bus Test System

```
%% system MVA base
baseMVA = 100;


%% bus data
%   bus_i   type   Pd   Qd   Gs   Bs   area   Vm   Va   baseKV   zone
    Vmax   Vmin
bus = [
  1  3  0     0  0  0  1  1  0  135 1  1.05    0.95;
  2  2  21.7   12.7   0  0  1  1  0  135 1  1.1 0.95;
  3  1  2.4 1.2 0  0  1  1  0  135 1  1.05    0.95;
  4  1  7.6 1.6 0  0  1  1  0  135 1  1.05    0.95;
  5  1  0  0  0  0.19   1  1  0  135 1  1.05   0.95;
  6  1  0  0  0  0  1  1  0  135 1  1.05     0.95;
  7  1  22.8   10.9   0  0  1  1  0  135 1 1.05  0.95;
  8  1  30  30 0  0  1  1  0  135 1  1.05    0.95;
  9  1  0  0  0  0  1  1  0  135 1  1.05     0.95;
  10  1  5.8 2  0  0  3  1  0  135 1  1.05    0.95;
  11  1  0  0  0  0  1  1  0  135 1  1.05     0.95;
  12  1  11.2   7.5 0  0  2  1  0  135 1  1.05   0.95;
  13  2  0  0  0  0  2  1  0  135 1  1.1      0.95;
  14  1  6.2 1.6 0  0  2  1  0  135 1  1.05    0.95;
  15  1  8.2 2.5 0  0  2  1  0  135 1  1.05    0.95;
  16  1  3.5 1.8 0  0  2  1  0  135 1  1.05    0.95;
  17  1  9  5.8 0  0  2  1  0  135 1  1.05    0.95;
  18  1  3.2 0.9 0  0  2  1  0  135 1  1.05    0.95;
  19  1  9.5 3.4 0  0  2  1  0  135 1  1.05    0.95;
  20  1  2.2   0.7 0  0  2  1  0  135 1  1.05   0.95;
  21  1  17.5   11.2 0  0  3  1  0  135 1  1.05  0.95;
  22  2  0    0  0  0  3  1  0  135 1  1.1    0.95;
  23  2  3.2   1.6  0  0  2  1  0  135 1  1.1   0.95;
  24  1  8.7   6.7  0  0.04  3  1  0 135 1  1.05  0.95;
  25  1  0    0  0  0  3  1  0  135 1  1.05   0.95;
  26  1  3.5   2.3 0  0  3  1  0  135 1  1.05   0.95;
  27  2  0    0  0  0  3  1  0  135 1  1.1     0.95;
  28  1  0    0  0  0  1  1  0  135 1  1.05    0.95;
  29  1  2.4   0.9 0  0  3  1  0  135 1  1.05   0.95;
  30  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  31  1  2.4   0.9 0  0  3  1  0  135 1  1.05   0.95;
  32  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  33  1  2.4   0.9 0  0  3  1  0  135 1  1.05   0.95;
  34  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  35  1  2.4   0.9 0  0  3  1  0  135 1  1.05   0.95;
  36  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  37  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  38  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  39  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  40  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  41  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  42  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  43  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  44  1  10.6   1.9 0  0  3  1  0  135 1  1.05   0.95;
  45  1  2.4   0.9 0  0  3  1  0  135 1  1.05   0.95;
```

```
46 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
47 1 2.4 0.9 0 0 3 1 0 135 1 1.05 0.95;
48 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
49 1 2.4 0.9 0 0 3 1 0 135 1 1.05 0.95;
50 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
51 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
52 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
53 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
54 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
55 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
56 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
57 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
58 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
59 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
60 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
61 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
62 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
63 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
64 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
65 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
66 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
67 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
68 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
69 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
70 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
71 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
72 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
73 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
74 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
75 1 10.6 1.9 0 0 3 1 0 135 1 1.05 0.95;
];
```

```
%% generator data
%	bus	Pg	Qg	Qmax	Qmin	Vg	mBase	status	Pmax	Pmin
gen = [
	1	23.54	0	150	-20	1	100	1	80	0;
	2	60.97	0	60	-20	1	100	1	80	0;
	22	21.59	0	62.5	-15	1	100	1	50	0;
	27	26.91	0	48.7	-15	1	100	1	55	0;
	23	19.2	0	40	-10	1	100	1	30	0;
	13	37	0	44.7	-15	1	100	1	40	0;
];
```

```
branch = [
	2	4	0.06	0.17	0.02	65	65	65 0 0 1;
	6	7	0.03	0.08	0	130	130	130 0 0 1;
	9	11	0	0.21	0	65	65	65 0 0 1;
	14	15	0.22	0.2	0	16	16	16 0 0 1;
	15	18	0.11	0.22	0	16	16	16 0 0 1;
	10	22	0.07	0.15	0	32	32	32 0 0 1;
	23	24	0.13	0.27	0	16	16	16 0 0 1;
	29	30	0.24	0.45	0	16	16	16 0 0 1;
	8	28	0.06	0.2	0.02	32	32	32 0 0 1;
```

31	20	0.02	0.06	0.01	32	32	32 0 0 1;
15	32	0.32	0.6	0	16	16	16 0 0 1;
1	33	0.06	0.2	0.02	32	32	32 0 0 1;
33	2	0.02	0.06	0.01	32	32	32 0 0 1;
3	34	0.32	0.6	0	16	16	16 0 0 1;
11	36	0.32	0.6	0	16	16	16 0 0 1;
37	15	0.02	0.06	0.01	32	32	32 0 0 1;
18	39	0.06	0.2	0.02	32	32	32 0 0 1;
5	6	0.06	0.2	0.02	32	32	32 0 0 1;
7	8	0.02	0.06	0.01	32	32	32 0 0 1;
42	17	0.02	0.06	0.01	32	32	32 0 0 1;
44	11	0.02	0.06	0.01	32	32	32 0 0 1;
12	14	0.12	0.26	0	32	32	32 0 0 1;
22	24	0.12	0.18	0	16	16	16 0 0 1;
43	11	0.02	0.06	0.01	32	32	32 0 0 1;
22	41	0.06	0.2	0.02	32	32	32 0 0 1;
21	40	0.06	0.2	0.02	32	32	32 0 0 1;
10	42	0.06	0.2	0.02	32	32	32 0 0 1;
34	4	0.24	0.45	0	16	16	16 0 0 1;
28	44	0.06	0.2	0.02	32	32	32 0 0 1;
16	17	0.08	0.19	0	16	16	16 0 0 1;
18	19	0.06	0.13	0	16	16	16 0 0 1;
10	31	0.06	0.2	0.02	32	32	32 0 0 1;
6	10	0	0.56	0	32	32	32 0 0 1;
24	25	0.19	0.33	0	16	16	16 0 0 1;
32	18	0.24	0.45	0	16	16	16 0 0 1;
2	5	0.05	0.2	0.02	130	130	130 0 0 1;
19	20	0.03	0.07	0	32	32	32 0 0 1;
41	17	0.02	0.06	0.01	32	32	32 0 0 1;
1	3	0.05	0.19	0.02	130	130	130 0 0 1;
12	16	0.09	0.2	0	32	32	32 0 0 1;
18	38	0.06	0.2	0.02	32	32	32 0 0 1;
35	11	0.02	0.06	0.01	32	32	32 0 0 1;
2	6	0.06	0.18	0.02	65	65	65 0 0 1;
5	7	0.05	0.12	0.01	70	70	70 0 0 1;
28	27	0	0.4	0	65	65	65 0 0 1;

10	20	0.09	0.21	0	32	32	32 0 0 1;
3	4	0.01	0.04	0	130	130	130 0 0 1;
6	9	0	0.21	0	65	65	65 0 0 1;
39	17	0.02	0.06	0.01	32	32	32 0 0 1;
6	35	0.06	0.2	0.02	32	32	32 0 0 1;
36	28	0.24	0.45	0	16	16	16 0 0 1;
6	43	0.06	0.2	0.02	32	32	32 0 0 1;
6	8	0.01	0.04	0	32	32	32 0 0 1;
23	24	0.13	0.27	0	16	16	16 0 0 1;
12	13	0	0.14	0	65	65	65 0 0 1;
21	22	0.01	0.02	0	32	32	32 0 0 1;
9	10	0	0.11	0	65	65	65 0 0 1;
4	6	0.01	0.04	0	90	90	90 0 0 1;
1	2	0.02	0.06	0.03	130	130	130 0 0 1;
15	23	0.1	0.2	0	16	16	16 0 0 1;
25	27	0.11	0.21	0	16	16	16 0 0 1;
27	29	0.22	0.42	0	16	16	16 0 0 1;
12	37	0.06	0.2	0.02	32	32	32 0 0 1;
40	29	0.02	0.06	0.01	32	32	32 0 0 1;
10	21	0.03	0.07	0	32	32	32 0 0 1;
4	12	0	0.26	0	65	65	65 0 0 1;
27	30	0.32	0.6	0	16	16	16 0 0 1;
6	28	0.02	0.06	0.01	32	32	32 0 0 1;
10	17	0.03	0.08	0	32	32	32 0 0 1;
38	12	0.02	0.06	0.01	32	32	32 0 0 1;
12	15	0.07	0.13	0	32	32	32 0 0 1;
1	45	0.13	0.27	0	16	16	16 0 0 1;
45	3	0	0.14	0	65	65	65 0 0 1;
1	46	0.01	0.02	0	32	32	32 0 0 1;
46	2	0	0.11	0	65	65	65 0 0 1;
1	47	0.01	0.04	0	90	90	90 0 0 1;
47	33	0.02	0.06	0.03	130	130	130 0 0 1;
33	48	0.1	0.2	0	16	16	16 0 0 1;
48	2	0.11	0.21	0	16	16	16 0 0 1;
2	49	0.22	0.42	0	16	16	16 0 0 1;
49	3	0.06	0.2	0.02	32	32	32 0 0 1;
3	50	0.02	0.06	0.01	32	32	32 0 0 1;

50	4	0.03	0.07	0	32	32	32 0 0 1;
10	51	0	0.26	0	65	65	65 0 0 1;
51	22	0.32	0.6	0	16	16	16 0 0 1;
10	52	0.02	0.06	0.01	32	32	32 0 0 1;
52	21	0.03	0.08	0	32	32	32 0 0 1;
21	53	0.02	0.06	0.01	32	32	32 0 0 1;
53	22	0.07	0.13	0	32	32	32 0 0 1;
27	54	0	0.26	0	65	65	65 0 0 1;
54	29	0.32	0.6	0	16	16	16 0 0 1;
29	55	0.02	0.06	0.01	32	32	32 0 0 1;
55	30	0.03	0.08	0	32	32	32 0 0 1;
30	56	0.02	0.06	0.01	32	32	32 0 0 1;
56	27	0.07	0.13	0	32	32	32 0 0 1;
56	57	0.07	0.13	0	32	32	32 0 0 1;
57	27	0.07	0.13	0	32	32	32 0 0 1;
27	58	0.07	0.13	0	32	32	32 0 0 1;
58	28	0.07	0.13	0	32	32	32 0 0 1;
2	33	0.07	0.13	0	32	32	32 0 0 1;
4	59	0	0.26	0	65	65	65 0 0 1;
59	12	0.32	0.6	0	16	16	16 0 0 1;
4	60	0.02	0.06	0.01	32	32	32 0 0 1;
60	6	0.03	0.08	0	32	32	32 0 0 1;
2	61	0.02	0.06	0.01	32	32	32 0 0 1;
61	6	0.07	0.13	0	32	32	32 0 0 1;
2	62	0	0.26	0	65	65	65 0 0 1;
62	5	0.32	0.6	0	16	16	16 0 0 1;
21	63	0.02	0.06	0.01	32	32	32 0 0 1;
63	40	0.03	0.08	0	32	32	32 0 0 1;
40	64	0.02	0.06	0.01	32	32	32 0 0 1;
64	29	0.07	0.13	0	32	32	32 0 0 1;
29	65	0.07	0.13	0	32	32	32 0 0 1;
65	30	0.07	0.13	0	32	32	32 0 0 1;
30	66	0.07	0.13	0	32	32	32 0 0 1;
66	27	0.07	0.13	0	32	32	32 0 0 1;
27	67	0.07	0.13	0	32	32	32 0 0 1;
67	28	0.03	0.07	0	32	32	32 0 0 1;
2	68	0.07	0.13	0	32	32	32 0 0 1;

```
68   4   0.03   0.07   0      32   32   32 0 0 1;
 1  69      0   0.26   0      65   65   65 0 0 1;
69   2   0.32    0.6   0      16   16   16 0 0 1;
 1  33      0   0.26   0      65   65   65 0 0 1;
33   2   0.32    0.6   0      16   16   16 0 0 1;
 6  70      0   0.26   0      65   65   65 0 0 1;
70  28   0.32    0.6   0      16   16   16 0 0 1;
 8  71   0.02   0.06   0.01   32   32   32 0 0 1;
71  28   0.07   0.13   0      32   32   32 0 0 1;
28  72      0   0.26   0      65   65   65 0 0 1;
72  44   0.32    0 .6     0     16   16     16 0 0 1;
44  73   0.02   0.06   0.01   32   32   32 0 0 1;
73  11   0.03   0.08   0      32   32   32 0 0 1;
 6  74   0.02   0.06   0.01        32   32 0 0 1;
                       32
74   8   0.03   0.08   0      32   32   32 0 0 1;
 7  75   0.02   0.06   0.01   32   32   32 0 0 1;
75   8   0.03   0.08   0      32   32   32 0 0 1;
];

%%-----  OPF Data  -----%%
%% area data
areas = [
	1	8;
	2	23;
	3	26;
];

%% generator cost data
%	1	startup	shutdown	n	x1	y1	...	xn	yn
%	2	startup	shutdown	n	c(n-1)	...	c0
gencost = [
	2	0	0	3	0.02	2	0;
	2	0	0	3	0.0175	1.75	0;
	2	0	0	3	0.0625	1	0;
	2	0	0	3	0.00834	3.25	0;
	2	0	0	3	0.025	3	0;
	2	0	0	3	0.025	3	0;
];
```

APPENDIX B

Test System Case File Corresponding to Modified 118-Bus Test System

%% system MVA base
baseMVA = 100;

%% bus data
% bus_i type Pd Qd Gs Bs area Vm Va baseKV zone Vmax Vmin
bus = [

1	2	51	27	0	0	1	0.955	10.67	138	1	1.06	0.94;
2	1	20	9	0	0	1	0.971	11.22	138	1	1.06	0.94;
3	1	39	10	0	0	1	0.968	11.56	138	1	1.06	0.94;
4	2	39	12	0	0	1	0.998	15.28	138	1	1.06	0.94;
5	1	0	0	0	-40	1	1.002	15.73	138	1	1.06	0.94;
6	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
7	1	19	2	0	0	1	0.989	12.56	138	1	1.06	0.94;
8	2	28	0	0	0	1	1.015	20.77	345	1	1.06	0.94;
9	1	0	0	0	0	1	1.043	28.02	345	1	1.06	0.94;
10	2	0	0	0	0	1	1.05	35.61	345	1	1.06	0.94;
11	1	70	23	0	0	1	0.985	12.72	138	1	1.06	0.94;
12	2	47	10	0	0	1	0.99	12.2	138	1	1.06	0.94;
13	1	34	16	0	0	1	0.968	11.35	138	1	1.06	0.94;
14	1	14	1	0	0	1	0.984	11.5	138	1	1.06	0.94;
15	2	90	30	0	0	1	0.97	11.23	138	1	1.06	0.94;
16	1	25	10	0	0	1	0.984	11.91	138	1	1.06	0.94;
17	1	11	3	0	0	1	0.995	13.74	138	1	1.06	0.94;
18	2	60	34	0	0	1	0.973	11.53	138	1	1.06	0.94;
19	2	45	25	0	0	1	0.963	11.05	138	1	1.06	0.94;
20	1	18	3	0	0	1	0.958	11.93	138	1	1.06	0.94;
21	1	14	8	0	0	1	0.959	13.52	138	1	1.06	0.94;
22	1	10	5	0	0	1	0.97	16.08	138	1	1.06	0.94;
23	1	7	3	0	0	1	1	21	138	1	1.06	0.94;
24	2	13	0	0	0	1	0.992	20.89	138	1	1.06	0.94;
25	2	0	0	0	0	1	1.05	27.93	138	1	1.06	0.94;
26	2	0	0	0	0	1	1.015	29.71	345	1	1.06	0.94;
27	2	71	13	0	0	1	0.968	15.35	138	1	1.06	0.94;

28	1	17	7	0	0	1	0.962	13.62	138	1	1.06	0.94;
29	1	24	4	0	0	1	0.963	12.63	138	1	1.06	0.94;
30	1	0	0	0	0	1	0.968	18.79	345	1	1.06	0.94;
31	2	43	27	0	0	1	0.967	12.75	138	1	1.06	0.94;
32	2	59	23	0	0	1	0.964	14.8	138	1	1.06	0.94;
33	1	23	9	0	0	1	0.972	10.63	138	1	1.06	0.94;
34	2	59	26	0	14	1	0.986	11.3	138	1	1.06	0.94;
35	1	33	9	0	0	1	0.981	10.87	138	1	1.06	0.94;
36	2	31	17	0	0	1	0.98	10.87	138	1	1.06	0.94;
37	1	0	0	0	-25	1	0.992	11.77	138	1	1.06	0.94;
38	1	0	0	0	0	1	0.962	16.91	345	1	1.06	0.94;
39	1	27	11	0	0	1	0.97	8.41	138	1	1.06	0.94;
40	2	66	23	0	0	1	0.97	7.35	138	1	1.06	0.94;
41	1	37	10	0	0	1	0.967	6.92	138	1	1.06	0.94;
42	2	96	23	0	0	1	0.985	8.53	138	1	1.06	0.94;
43	1	18	7	0	0	1	0.978	11.28	138	1	1.06	0.94;
44	1	16	8	0	10	1	0.985	13.82	138	1	1.06	0.94;
45	1	53	22	0	10	1	0.987	15.67	138	1	1.06	0.94;
46	2	28	10	0	10	1	1.005	18.49	138	1	1.06	0.94;
47	1	34	0	0	0	1	1.017	20.73	138	1	1.06	0.94;
48	1	20	11	0	15	1	1.021	19.93	138	1	1.06	0.94;
49	2	87	30	0	0	1	1.025	20.94	138	1	1.06	0.94;
50	1	17	4	0	0	1	1.001	18.9	138	1	1.06	0.94;
51	1	17	8	0	0	1	0.967	16.28	138	1	1.06	0.94;
52	1	18	5	0	0	1	0.957	15.32	138	1	1.06	0.94;
53	1	23	11	0	0	1	0.946	14.35	138	1	1.06	0.94;
54	2	113	32	0	0	1	0.955	15.26	138	1	1.06	0.94;
55	2	63	22	0	0	1	0.952	14.97	138	1	1.06	0.94;
56	2	84	18	0	0	1	0.954	15.16	138	1	1.06	0.94;
57	1	12	3	0	0	1	0.971	16.36	138	1	1.06	0.94;
58	1	12	3	0	0	1	0.959	15.51	138	1	1.06	0.94;
59	2	277	113	0	0	1	0.985	19.37	138	1	1.06	0.94;
60	1	78	3	0	0	1	0.993	23.15	138	1	1.06	0.94;
61	2	0	0	0	0	1	0.995	24.04	138	1	1.06	0.94;
62	2	77	14	0	0	1	0.998	23.43	138	1	1.06	0.94;
63	1	0	0	0	0	1	0.969	22.75	345	1	1.06	0.94;
64	1	0	0	0	0	1	0.984	24.52	345	1	1.06	0.94;

65	2	0	0	0	0	1	1.005	27.65	345	1	1.06	0.94;
66	2	39	18	0	0	1	1.05	27.48	138	1	1.06	0.94;
67	1	28	7	0	0	1	1.02	24.84	138	1	1.06	0.94;
68	1	0	0	0	0	1	1.003	27.55	345	1	1.06	0.94;
69	3	0	0	0	0	1	1.035	30	138	1	1.06	0.94;
70	2	66	20	0	0	1	0.984	22.58	138	1	1.06	0.94;
71	1	0	0	0	0	1	0.987	22.15	138	1	1.06	0.94;
72	2	12	0	0	0	1	0.98	20.98	138	1	1.06	0.94;
73	2	6	0	0	0	1	0.991	21.94	138	1	1.06	0.94;
74	2	68	27	0	12	1	0.958	21.64	138	1	1.06	0.94;
75	1	47	11	0	0	1	0.967	22.91	138	1	1.06	0.94;
76	2	68	36	0	0	1	0.943	21.77	138	1	1.06	0.94;
77	2	61	28	0	0	1	1.006	26.72	138	1	1.06	0.94;
78	1	71	26	0	0	1	1.003	26.42	138	1	1.06	0.94;
79	1	39	32	0	20	1	1.009	26.72	138	1	1.06	0.94;
80	2	130	26	0	0	1	1.04	28.96	138	1	1.06	0.94;
81	1	0	0	0	0	1	0.997	28.1	345	1	1.06	0.94;
82	1	54	27	0	20	1	0.989	27.24	138	1	1.06	0.94;
83	1	20	10	0	10	1	0.985	28.42	138	1	1.06	0.94;
84	1	11	7	0	0	1	0.98	30.95	138	1	1.06	0.94;
85	2	24	15	0	0	1	0.985	32.51	138	1	1.06	0.94;
86	1	21	10	0	0	1	0.987	31.14	138	1	1.06	0.94;
87	2	0	0	0	0	1	1.015	31.4	161	1	1.06	0.94;
88	1	48	10	0	0	1	0.987	35.64	138	1	1.06	0.94;
89	2	0	0	0	0	1	1.005	39.69	138	1	1.06	0.94;
90	2	163	42	0	0	1	0.985	33.29	138	1	1.06	0.94;
91	2	10	0	0	0	1	0.98	33.31	138	1	1.06	0.94;
92	2	65	10	0	0	1	0.993	33.8	138	1	1.06	0.94;
93	1	12	7	0	0	1	0.987	30.79	138	1	1.06	0.94;
94	1	30	16	0	0	1	0.991	28.64	138	1	1.06	0.94;
95	1	42	31	0	0	1	0.981	27.67	138	1	1.06	0.94;
96	1	38	15	0	0	1	0.993	27.51	138	1	1.06	0.94;
97	1	15	9	0	0	1	1.011	27.88	138	1	1.06	0.94;
98	1	34	8	0	0	1	1.024	27.4	138	1	1.06	0.94;
99	2	42	0	0	0	1	1.01	27.04	138	1	1.06	0.94;
100	2	37	18	0	0	1	1.017	28.03	138	1	1.06	0.94;
101	1	22	15	0	0	1	0.993	29.61	138	1	1.06	0.94;

```
	102	1	5	3	0	0	1	0.991	32.3	138	1	1.06	0.94;
	103	2	23	16	0	0	1	1.001	24.44	138	1	1.06	0.94;
	104	2	38	25	0	0	1	0.971	21.69	138	1	1.06	0.94;
	105	2	31	26	0	20	1	0.965	20.57	138	1	1.06	0.94;
	106	1	43	16	0	0	1	0.962	20.32	138	1	1.06	0.94;
	107	2	50	12	0	6	1	0.952	17.53	138	1	1.06	0.94;
	108	1	2	1	0	0	1	0.967	19.38	138	1	1.06	0.94;
	109	1	8	3	0	0	1	0.967	18.93	138	1	1.06	0.94;
	110	2	39	30	0	6	1	0.973	18.09	138	1	1.06	0.94;
	111	2	0	0	0	0	1	0.98	19.74	138	1	1.06	0.94;
	112	2	68	13	0	0	1	0.975	14.99	138	1	1.06	0.94;
	113	2	6	0	0	0	1	0.993	13.74	138	1	1.06	0.94;
	114	1	8	3	0	0	1	0.96	14.46	138	1	1.06	0.94;
	115	1	22	7	0	0	1	0.96	14.46	138	1	1.06	0.94;
	116	2	184	0	0	0	1	1.005	27.12	138	1	1.06	0.94;
	117	1	20	8	0	0	1	0.974	10.67	138	1	1.06	0.94;
	118	1	33	15	0	0	1	0.949	21.92	138	1	1.06	0.94;
	119	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
	120	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
	121	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
	122	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
	123	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
	124	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
	125	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
	126	2	52	22	0	0	1	0.99	13	138	1	1.06	0.94;
];

%% generator data
%	bus	Pg	Qg	Qmax	Qmin	Vg	mBase	status	Pmax	Pmin
gen = [
	1	0	0	15	-5	0.955	100	1	100	0;
	4	0	0	300	-300	0.998	100	1	100	0;
	6	0	0	50	-13	0.99	100	1	100	0;
	8	0	0	300	-300	1.015	100	1	100	0;
	10	450	0	200	-147	1.05	100	1	550	0;
	12	85	0	120	-35	0.99	100	1	185	0;
	15	0	0	30	-10	0.97	100	1	100	0;
	18	0	0	50	-16	0.973	100	1	100	0;
	19	0	0	24	-8	0.962	100	1	100	0;
	24	0	0	300	-300	0.992	100	1	100	0;
	25	220	0	140	-47	1.05	100	1	320	0;
	26	314	0	1000	-1000	1.015	100	1	414	0;
	27	0	0	300	-300	0.968	100	1	100	0;
	31	7	0	300	-300	0.967	100	1	107	0;
	32	0	0	42	-14	0.963	100	1	100	0;
```

34	0	0	24	-8	0.984	100	1	100	0;
36	0	0	24	-8	0.98	100	1	100	0;
40	0	0	300	-300	0.97	100	1	100	0;
42	0	0	300	-300	0.985	100	1	100	0;
46	19	0	100	-100	1.005	100	1	119	0;
49	204	0	210	-85	1.025	100	1	304	0;
54	48	0	300	-300	0.955	100	1	148	0;
55	0	0	23	-8	0.952	100	1	100	0;
56	0	0	15	-8	0.954	100	1	100	0;
59	155	0	180	-60	0.985	100	1	255	0;
61	160	0	300	-100	0.995	100	1	260	0;
62	0	0	20	-20	0.998	100	1	100	0;
65	391	0	200	-67	1.005	100	1	491	0;
66	392	0	200	-67	1.05	100	1	492	0;
69	516.4	0	300	-300	1.035	100	1	805.2	0;
70	0	0	32	-10	0.984	100	1	100	0;
72	0	0	100	-100	0.98	100	1	100	0;
73	0	0	100	-100	0.991	100	1	100	0;
74	0	0	9	-6	0.958	100	1	100	0;
76	0	0	23	-8	0.943	100	1	100	0;
77	0	0	70	-20	1.006	100	1	100	0;
80	477	0	280	-165	1.04	100	1	577	0;
85	0	0	23	-8	0.985	100	1	100	0;
87	4	0	1000	-100	1.015	100	1	104	0;
89	607	0	300	-210	1.005	100	1	707	0;
90	0	0	300	-300	0.985	100	1	100	0;
91	0	0	100	-100	0.98	100	1	100	0;
92	0	0	9	-3	0.99	100	1	100	0;
99	0	0	100	-100	1.01	100	1	100	0;
100	252	0	155	-50	1.017	100	1	352	0;
103	40	0	40	-15	1.01	100	1	140	0;
104	0	0	23	-8	0.971	100	1	100	0;
105	0	0	23	-8	0.965	100	1	100	0;
107	0	0	200	-200	0.952	100	1	100	0;
110	0	0	23	-8	0.973	100	1	100	0;
111	36	0	1000	-100	0.98	100	1	136	0;
112	0	0	1000	-100	0.975	100	1	100	0;
113	0	0	200	-100	0.993	100	1	100	0;
116	0	0	1000	-1000	1.005	100	1	100	0;

];

% branch data

%	fbus	tbus	r	x	b	rateA	rateB	rateC	ratio	angle	status
branch = [											
	1	2	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	1	3	0.0129	0.0424	0.01082	9900	0	0	0	0	1;
	4	5	0.00176	0.00798	0.0021	9900	0	0	0	0	1;
	3	5	0.0241	0.108	0.0284	9900	0	0	0	0	1;
	5	6	0.0119	0.054	0.01426	9900	0	0	0	0	1;
	6	7	0.00459	0.0208	0.0055	9900	0	0	0	0	1;
	8	9	0.00244	0.0305	1.162	9900	0	0	0	0	1;
	8	5	0	0.0267	0	9900	0	0 0.985	0		1;
	9	10	0.00258	0.0322	1.23	9900	0	0	0	0	1;
	4	11	0.0209	0.0688	0.01748	9900	0	0	0	0	1;
	5	11	0.0203	0.0682	0.01738	9900	0	0	0	0	1;
	11	12	0.00595	0.0196	0.00502	9900	0	0	0	0	1;
	2	12	0.0187	0.0616	0.01572	9900	0	0	0	0	1;
	3	12	0.0484	0.16	0.0406	9900	0	0	0	0	1;
	7	12	0.00862	0.034	0.00874	9900	0	0	0	0	1;
	11	13	0.02225	0.0731	0.01876	9900	0	0	0	0	1;

12	14	0.0215	0.0707	0.01816	9900	0	0	0	0	1;
13	15	0.0744	0.2444	0.06268	9900	0	0	0	0	1;
14	15	0.0595	0.195	0.0502	9900	0	0	0	0	1;
12	16	0.0212	0.0834	0.0214	9900	0	0	0	0	1;
15	17	0.0132	0.0437	0.0444	9900	0	0	0	0	1;
16	17	0.0454	0.1801	0.0466	9900	0	0	0	0	1;
17	18	0.0123	0.0505	0.01298	9900	0	0	0	0	1;
18	19	0.01119	0.0493	0.01142	9900	0	0	0	0	1;
19	20	0.0252	0.117	0.0298	9900	0	0	0	0	1;
15	19	0.012	0.0394	0.0101	9900	0	0	0	0	1;
20	21	0.0183	0.0849	0.0216	9900	0	0	0	0	1;
21	22	0.0209	0.097	0.0246	9900	0	0	0	0	1;
22	23	0.0342	0.159	0.0404	9900	0	0	0	0	1;
23	24	0.0135	0.0492	0.0498	9900	0	0	0	0	1;
23	25	0.0156	0.08	0.0864	9900	0	0	0	0	1;
26	25	0	0.0382	0	9900	0	0	0.96	0	1;
25	27	0.0318	0.163	0.1764	9900	0	0	0	0	1;
27	28	0.01913	0.0855	0.0216	9900	0	0	0	0	1;
28	29	0.0237	0.0943	0.0238	9900	0	0	0	0	1;
30	17	0	0.0388	0	9900	0	0	0.96	0	1;
8	30	0.00431	0.0504	0.514	9900	0	0	0	0	1;
26	30	0.00799	0.086	0.908	9900	0	0	0	0	1;
17	31	0.0474	0.1563	0.0399	9900	0	0	0	0	1;
29	31	0.0108	0.0331	0.0083	9900	0	0	0	0	1;
23	32	0.0317	0.1153	0.1173	9900	0	0	0	0	1;
31	32	0.0298	0.0985	0.0251	9900	0	0	0	0	1;
27	32	0.0229	0.0755	0.01926	9900	0	0	0	0	1;
15	33	0.038	0.1244	0.03194	9900	0	0	0	0	1;
19	34	0.0752	0.247	0.0632	9900	0	0	0	0	1;
35	36	0.00224	0.0102	0.00268	9900	0	0	0	0	1;
35	37	0.011	0.0497	0.01318	9900	0	0	0	0	1;
33	37	0.0415	0.142	0.0366	9900	0	0	0	0	1;
34	36	0.00871	0.0268	0.00568	9900	0	0	0	0	1;
34	37	0.00256	0.0094	0.00984	9900	0	0	0	0	1;
38	37	0	0.0375	0	9900	0	0	0.935	0	1;
37	39	0.0321	0.106	0.027	9900	0	0	0	0	1;
37	40	0.0593	0.168	0.042	9900	0	0	0	0	1;
30	38	0.00464	0.054	0.422	9900	0	0	0	0	1;
39	40	0.0184	0.0605	0.01552	9900	0	0	0	0	1;
40	41	0.0145	0.0487	0.01222	9900	0	0	0	0	1;
40	42	0.0555	0.183	0.0466	9900	0	0	0	0	1;
41	42	0.041	0.135	0.0344	9900	0	0	0	0	1;
43	44	0.0608	0.2454	0.06068	9900	0	0	0	0	1;
34	43	0.0413	0.1681	0.04226	9900	0	0	0	0	1;
44	45	0.0224	0.0901	0.0224	9900	0	0	0	0	1;
45	46	0.04	0.1356	0.0332	9900	0	0	0	0	1;
46	47	0.038	0.127	0.0316	9900	0	0	0	0	1;
46	48	0.0601	0.189	0.0472	9900	0	0	0	0	1;
47	49	0.0191	0.0625	0.01604	9900	0	0	0	0	1;
42	49	0.0715	0.323	0.086	9900	0	0	0	0	1;
42	49	0.0715	0.323	0.086	9900	0	0	0	0	1;
45	49	0.0684	0.186	0.0444	9900	0	0	0	0	1;
48	49	0.0179	0.0505	0.01258	9900	0	0	0	0	1;
49	50	0.0267	0.0752	0.01874	9900	0	0	0	0	1;
49	51	0.0486	0.137	0.0342	9900	0	0	0	0	1;
51	52	0.0203	0.0588	0.01396	9900	0	0	0	0	1;
52	53	0.0405	0.1635	0.04058	9900	0	0	0	0	1;
53	54	0.0263	0.122	0.031	9900	0	0	0	0	1;
49	54	0.073	0.289	0.0738	9900	0	0	0	0	1;
49	54	0.0869	0.291	0.073	9900	0	0	0	0	1;
54	55	0.0169	0.0707	0.0202	9900	0	0	0	0	1;

54	56	0.00275	0.00955	0.00732	9900	0	0	0	0	1;
55	56	0.00488	0.0151	0.00374	9900	0	0	0	0	1;
56	57	0.0343	0.0966	0.0242	9900	0	0	0	0	1;
50	57	0.0474	0.134	0.0332	9900	0	0	0	0	1;
56	58	0.0343	0.0966	0.0242	9900	0	0	0	0	1;
51	58	0.0255	0.0719	0.01788	9900	0	0	0	0	1;
54	59	0.0503	0.2293	0.0598	9900	0	0	0	0	1;
56	59	0.0825	0.251	0.0569	9900	0	0	0	0	1;
56	59	0.0803	0.239	0.0536	9900	0	0	0	0	1;
55	59	0.04739	0.2158	0.05646	9900	0	0	0	0	1;
59	60	0.0317	0.145	0.0376	9900	0	0	0	0	1;
59	61	0.0328	0.15	0.0388	9900	0	0	0	0	1;
60	61	0.00264	0.0135	0.01456	9900	0	0	0	0	1;
60	62	0.0123	0.0561	0.01468	9900	0	0	0	0	1;
61	62	0.00824	0.0376	0.0098	9900	0	0	0	0	1;
63	59	0	0.0386	0	9900	0	0	0.96	0	1;
63	64	0.00172	0.02	0.216	9900	0	0	0	0	1;
64	61	0	0.0268	0	9900	0	0	0.985	0	1;
38	65	0.00901	0.0986	1.046	9900	0	0	0	0	1;
64	65	0.00269	0.0302	0.38	9900	0	0	0	0	1;
49	66	0.018	0.0919	0.0248	9900	0	0	0	0	1;
49	66	0.018	0.0919	0.0248	9900	0	0	0	0	1;
62	66	0.0482	0.218	0.0578	9900	0	0	0	0	1;
62	67	0.0258	0.117	0.031	9900	0	0	0	0	1;
65	66	0	0.037	0	9900	0	0	0.935	0	1;
66	67	0.0224	0.1015	0.02682	9900	0	0	0	0	1;
65	68	0.00138	0.016	0.638	9900	0	0	0	0	1;
47	69	0.0844	0.2778	0.07092	9900	0	0	0	0	1;
49	69	0.0985	0.324	0.0828	9900	0	0	0	0	1;
68	69	0	0.037	0	9900	0	0	0.935	0	1;
69	70	0.03	0.127	0.122	9900	0	0	0	0	1;
24	70	0.00221	0.4115	0.10198	9900	0	0	0	0	1;
70	71	0.00882	0.0355	0.00878	9900	0	0	0	0	1;
24	72	0.0488	0.196	0.0488	9900	0	0	0	0	1;
71	72	0.0446	0.18	0.04444	9900	0	0	0	0	1;
71	73	0.00866	0.0454	0.01178	9900	0	0	0	0	1;
70	74	0.0401	0.1323	0.03368	9900	0	0	0	0	1;
70	75	0.0428	0.141	0.036	9900	0	0	0	0	1;
69	75	0.0405	0.122	0.124	9900	0	0	0	0	1;
74	75	0.0123	0.0406	0.01034	9900	0	0	0	0	1;
76	77	0.0444	0.148	0.0368	9900	0	0	0	0	1;
69	77	0.0309	0.101	0.1038	9900	0	0	0	0	1;
75	77	0.0601	0.1999	0.04978	9900	0	0	0	0	1;
77	78	0.00376	0.0124	0.01264	9900	0	0	0	0	1;
78	79	0.00546	0.0244	0.00648	9900	0	0	0	0	1;
77	80	0.017	0.0485	0.0472	9900	0	0	0	0	1;
77	80	0.0294	0.105	0.0228	9900	0	0	0	0	1;
79	80	0.0156	0.0704	0.0187	9900	0	0	0	0	1;
68	81	0.00175	0.0202	0.808	9900	0	0	0	0	1;
81	80	0	0.037	0	9900	0	0	0.935	0	1;
77	82	0.0298	0.0853	0.08174	9900	0	0	0	0	1;
82	83	0.0112	0.03665	0.03796	9900	0	0	0	0	1;
83	84	0.0625	0.132	0.0258	9900	0	0	0	0	1;
83	85	0.043	0.148	0.0348	9900	0	0	0	0	1;
84	85	0.0302	0.0641	0.01234	9900	0	0	0	0	1;
85	86	0.035	0.123	0.0276	9900	0	0	0	0	1;
86	87	0.02828	0.2074	0.0445	9900	0	0	0	0	1;
85	88	0.02	0.102	0.0276	9900	0	0	0	0	1;
85	89	0.0239	0.173	0.047	9900	0	0	0	0	1;
88	89	0.0139	0.0712	0.01934	9900	0	0	0	0	1;
89	90	0.0518	0.188	0.0528	9900	0	0	0	0	1;

89	90	0.0238	0.0997	0.106	9900	0	0	0	0	1;
90	91	0.0254	0.0836	0.0214	9900	0	0	0	0	1;
89	92	0.0099	0.0505	0.0548	9900	0	0	0	0	1;
89	92	0.0393	0.1581	0.0414	9900	0	0	0	0	1;
91	92	0.0387	0.1272	0.03268	9900	0	0	0	0	1;
92	93	0.0258	0.0848	0.0218	9900	0	0	0	0	1;
92	94	0.0481	0.158	0.0406	9900	0	0	0	0	1;
93	94	0.0223	0.0732	0.01876	9900	0	0	0	0	1;
94	95	0.0132	0.0434	0.0111	9900	0	0	0	0	1;
80	96	0.0356	0.182	0.0494	9900	0	0	0	0	1;
82	96	0.0162	0.053	0.0544	9900	0	0	0	0	1;
94	96	0.0269	0.0869	0.023	9900	0	0	0	0	1;
80	97	0.0183	0.0934	0.0254	9900	0	0	0	0	1;
80	98	0.0238	0.108	0.0286	9900	0	0	0	0	1;
80	99	0.0454	0.206	0.0546	9900	0	0	0	0	1;
92	100	0.0648	0.295	0.0472	9900	0	0	0	0	1;
94	100	0.0178	0.058	0.0604	9900	0	0	0	0	1;
95	96	0.0171	0.0547	0.01474	9900	0	0	0	0	1;
96	97	0.0173	0.0885	0.024	9900	0	0	0	0	1;
98	100	0.0397	0.179	0.0476	9900	0	0	0	0	1;
99	100	0.018	0.0813	0.0216	9900	0	0	0	0	1;
100	101	0.0277	0.1262	0.0328	9900	0	0	0	0	1;
92	102	0.0123	0.0559	0.01464	9900	0	0	0	0	1;
101	102	0.0246	0.112	0.0294	9900	0	0	0	0	1;
100	103	0.016	0.0525	0.0536	9900	0	0	0	0	1;
100	104	0.0451	0.204	0.0541	9900	0	0	0	0	1;
103	104	0.0466	0.1584	0.0407	9900	0	0	0	0	1;
103	105	0.0535	0.1625	0.0408	9900	0	0	0	0	1;
100	106	0.0605	0.229	0.062	9900	0	0	0	0	1;
104	105	0.00994	0.0378	0.00986	9900	0	0	0	0	1;
105	106	0.014	0.0547	0.01434	9900	0	0	0	0	1;
105	107	0.053	0.183	0.0472	9900	0	0	0	0	1;
105	108	0.0261	0.0703	0.01844	9900	0	0	0	0	1;
106	107	0.053	0.183	0.0472	9900	0	0	0	0	1;
108	109	0.0105	0.0288	0.0076	9900	0	0	0	0	1;
103	110	0.03906	0.1813	0.0461	9900	0	0	0	0	1;
109	110	0.0278	0.0762	0.0202	9900	0	0	0	0	1;
110	111	0.022	0.0755	0.02	9900	0	0	0	0	1;
110	112	0.0247	0.064	0.062	9900	0	0	0	0	1;
17	113	0.00913	0.0301	0.00768	9900	0	0	0	0	1;
32	113	0.0615	0.203	0.0518	9900	0	0	0	0	1;
32	114	0.0135	0.0612	0.01628	9900	0	0	0	0	1;
27	115	0.0164	0.0741	0.01972	9900	0	0	0	0	1;
114	115	0.0023	0.0104	0.00276	9900	0	0	0	0	1;
68	116	0.00034	0.00405	0.164	9900	0	0	0	0	1;
12	117	0.0329	0.14	0.0358	9900	0	0	0	0	1;
75	118	0.0145	0.0481	0.01198	9900	0	0	0	0	1;
76	118	0.0164	0.0544	0.01356	9900	0	0	0	0	1;
69	119	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
119	75	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
34	120	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
120	37	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
44	121	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
121	45	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
25	120	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
120	27	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
23	121	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
121	32	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
34	38	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
37	38	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
8	122	0.0303	0.0999	0.0254	9900	0	0	0	0	1;

```
	122	5	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	 30	123	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	123	17	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	74	124	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	124	75	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	 85	125	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	125	89	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	88	126	0.0303	0.0999	0.0254	9900	0	0	0	0	1;
	98	99	0.0303	0.0999	0.0254	9900	0	0	0	0	1;

  ];


%% area data
areas = [
	1	1;
];


%% generator cost data
%	1	startup	shutdown	n	x1	y1	...	xn	yn
%	2	startup	shutdown	n	c(n-1)	...	c0
gencost = [
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.022222	20	0;
	2	0	0	3	0.117647	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.045454	20	0;
	2	0	0	3	0.031847	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	1.42857	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.526316	20	0;
	2	0	0	3	0.049019	20	0;
	2	0	0	3	0.208333	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.064516	20	0;
	2	0	0	3	0.0625	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.025575	20	0;
	2	0	0	3	0.025510	20	0;
	2	0	0	3	0.019364	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.020964	20	0;
```

```
	2	0	0	3	0.01	40	0;
	2	0	0	3	2.5	20	0;
	2	0	0	3	0.016474	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.039682	20	0;
	2	0	0	3	0.25	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.277778	20	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
	2	0	0	3	0.01	40	0;
];
```

Vita

Ashwini Chegu was born on 26th January, 1985, in Vijayawada, India. She obtained her B.E degree in Electrical and Electronics Engineering (EEE) with distinction in May 2006. She joined University of Tennessee, Knoxville to pursue her Master's degree in Electrical Engineering. She worked on her thesis supervised by Dr. Fangxing Li. She developed a blackout model for higher order contingency analysis for prevention of blackouts using particle swarm optimization (PSO) and tabu search. During her masters she worked as a Graduate Research Assistant with Dr. Fangxing Li and was involved in developing a capability to visualize the live status of electric transmission system infrastructure using KML which could identify most critical lines that could lead to blackout. This project is called "Visualization of Energy Resouses on Google Earth (VERDE)".She also worked as a Graduate Research Assistant in GLORIAD (Global Ring Network for Advanced Applications Development) and was involved in web development and programming. She also worked with network monitoring tools and measurement systems. She graduated with Master of Science Degree in Electrical Engineering from Univerity of Tennessee in August 2010.