A Study of Power Distribution System Reconfiguration based on Reliability Indices

A Thesis Presented to the

Faculty of the Graduate School at the University of Missouri-Columbia

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Zhe Chen

Professor Robert O'Connell, Thesis Supervisor

December 2016

The undersigned, appointed by the dean of the Graduate School, have examined the thesis entitled

A Study of Power Distribution System Reconfiguration based on Reliability Indices

Presented by Zhe Chen,

A candidate for the degree of Master of Science hereby certify that, in their opinion, it is worthy of acceptance.

Professor Robert O'Connell

Professor Michael Devaney

Professor Bahaa Kazem Ansaf

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisor Prof. O'Connell for the continuous support of my Master's study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Master's study.

Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Devaney, and Prof. Ansaf, for their insightful comments and encouragement, but also for the hard question which incented me to widen my research from various perspectives.

Moreover, I would like to express my appreciation to Ameren Missouri, for their supports and advices.

Last but not least, I would like to thank my friends and family for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Network reconfiguration is one of the feasible methods for reducing the distribution network loss. The reconfiguration of a distribution system is a process that alters the feeder topological structure by changing the open/ closed status of the sectionalizing (normally closed) and tie (normally open) switches in the system, with the objective of improving performance. Reliability indices, such as System Average Interruption Frequency Index (SAIFI) and Energy Not Supplied (ENS) indices, are important measures of system reliability.

Very few published articles include detailed methods for calculation of the reliability indices. This thesis includes a detailed explanation of the reliability index calculation process and the algorithm for programming it. This algorithm is used within the Fast, Non-dominated sorting genetic algorithm (FNSGA) to determine switch sets for multi-objective optimization purposes.

The objectives of the problem in normal operation are to improve the reliability level of the system, and obtain small power loss with a relatively high voltage profile. This thesis adopts FNSGA for the purpose of solving the Distribution System Reconfiguration (DSR) problem by satisfying reliability objectives simultaneously with a relatively small number of generations, small population sizes, and relatively short computation time.

In the thesis, a 5-bus sample test system was introduced to illustrate the calculation of the reliability indices and the standard IEEE 16-bus and 32-bus test system were used to evaluate the algorithm.

Chapter One

Introduction

Electric power systems are extremely complex, but the typical electric power system can be described in terms of three components: generation, transmission, and distribution. The distribution system (DS) is the part of the power system infrastructure that serves as a link between the high-voltage highly meshed transmission system and the end-users of the electric energy. Statistically, the majority of service interruptions to customers come from the distribution systems [1] [2]. Typically, the DS is designed as a meshed system. However, it is operated in radial configuration in order to exploit the advantages this topology offers: easier coordination of protective measures, lower fault currents, easier voltage and power flow control and, definitely, lower cost [3].

Over the past few decades, distribution systems have been evolving, yet the primary emphasis has been on providing an economic and reliable supply of electric energy to the customers. It has been reported that 80% of the customer service interruptions are due to failures in the distribution network. Reliability evaluation of the distribution system has, therefore, become very important. For economic reasons, minimization of the losses in the distribution system should also be considered [4].

The most commonly used measures which can improve the performance of a distribution system are:

i. Changing the operating scheme of the parallel connected power transformers (Transformer substitution)

ii. Variation and control of the voltage by using on-load tap-changers for power transformers

iii. Variation and control of the reactive power flow through the system (voltage load balancing)

iv. Rehabilitation of the system by adding parallel circuits or replacing the existing line with one having greater cross sectional area

v. Network reconfiguration, as known as distribution system reconfiguration (DSR)

Since 1975, Merlin and Back [5] introduced the idea of distribution system reconfiguration with the purpose of obtaining a radial topology leading to the lowest power losses. Starting from this work, DSR has become the most viable alternative for loss reduction in both planning and real time operation due to its efficiency, low cost, and simplicity as compared to the other mentioned methods. Then, from 1988-1992, more authors introduced the idea of DSR obtaining a radial topology leading to the lowest power losses [6-10]. In 1990, A. Makinen [11] first described a computer-aided method for the reliability analysis of a distribution system. In recent years, reconfiguration has been applied to the distribution network to achieve significant and immediate improvement in reliability [4] [11-15].

This thesis focuses on optimization through the reconfiguration of power distribution systems. The reconfiguration problem is one of the multi-criteria optimization types, where the solution is chosen after the evaluation of some indices. This thesis presents a method based on material described in “A Fast Non-Dominated Sorting Guided Genetic Algorithm (FNSGA)” [16]. In Chapter 2, the problem is formulated into a multi-objective optimization problem. In Chapter 3, the detailed calculation of the reliability indices is introduced, and a 5-bus sample system is used to illustrate the calculation. The FNSGA is applied to the standard IEEE 16-bus and 32-bus systems in Chapter 4. Then conclusions and future works are discussed in Chapter 5.

Chapter Two

Non-dominating sorting GA and Problem Formulation

2.1 Introduction

The reconfiguration problem is one of the multi-criteria optimization types, where the solution is chosen after the evaluation of some indices (e.g., active power losses, reliability indices, branch load limits, voltage drop limits, etc.), which represent multiple objectives. Ever since Merlin and Back [5], a lot of researchers have proposed diverse methods and algorithms to solve the reconfiguration problem as a single objective problem. Usually, active power losses are adopted as the main objective [5-10]. However, according to today's demand for electric power, single objective optimization is not enough to satisfy the requirements for distribution systems. As a result, there is usually more than one objective that needs to be taken into the optimization consideration.

On the other hand, some authors have studied the DSR problem using aggregation functions, which convert a multi-objective problem into a single equivalent objective one that assumes a weighted (normalized) sum of the selected objective functions [17-20]. The major issue in this kind of problem consists in the incompatibility of different objectives. In order to create a global function, all objectives must be converted to the same measurement unit; a frequently used method is to convert them into costs, which is usually tricky and problem dependent. Also, there's hardly any direct connection between reliability and costs, which makes a good normalization function hard to obtain in my research. To eliminate the complexity and subjectivity of the above methods, the Pareto optimality concept was adopted. In this thesis, instead of searching for a single solution that satisfy all the objectives, a set of non-dominated solutions is generated. The concept of non-dominated sorting was based on the non-dominated sorting genetic algorithm-II (NSGA-II) [21]. Kumar [22] and Xiaochao [23] first adopted NSGA-II to the distribution problem. The detailed explanation of each problem mentioned above will be given in the following sections.

2.2 Objectives for Optimization

The DSR problem is to determine the optimum status of all the switches in the system. As is mentioned above, the problem is formulated as a multi-objectives optimization problem to optimize system performance. These objectives are expressed as fitness functions to be implemented in the Algorithm.

2.2.1 Reliability indices

Reliability indices are important measures of the distribution system reliability, and they were first developed in 1998 by IEEE specifically for distribution system evaluation. There are many reliability indices, classified into different groups based on usage, and they are introduced in [24-26]. In this thesis, only the most representative four indices were considered. Moreover, SAIFI is the most popular index among all the reliability indices in both today's research and industrial evaluation use.

a. System Average Interruption Frequency Index (SAIFI)

The System Average Interruption Frequency Index indicates how often the average customer experiences a sustained interruption over a predefined period of time. The SAIFI index is the most popular one used to indicate sustained interruption of the system, an expression for it is given in equation (2.1):

$$SAIFI = \frac{\sum Total\ number\ of\ customer\ interrupted}{total\ number\ of\ customer\ served} \tag{2.1}$$

b. Energy Not Supplied (ENS)

The term Energy Not Supplied represents the total unserved energy of the distribution system. ENS is one of the load-based indices of the reliability of the energy supply, an expression for it is given in equation (2.2):

$$ENS = \sum energy\ not\ supplied\ at\ each\ load\ point \tag{2.2}$$

Some papers use Average Energy Not Supplied (AENS) for calculation [25] [26], in which the calculation for AENS is to divide ENS by Total number of Customers served. The concepts of ENS and AENS are similar, and the ENS simplifies the calculation process. As a result, ENS is used in this thesis for reliability evaluation.

c. System Average Interruption Duration Index (SAIDI)

The term SAIDI represents the total duration of interruptions for the average customer during a predefined period of time, an expression for it is given in equation (2.3):

$$SAIDI = \frac{\sum customer\ minutes\ of\ interruption}{total\ number\ of\ customers\ served} \tag{2.3}$$

d. Customer Average Interruption Duration Index (CAIDI)

The term CAIDI represents the average time required to restore service, an expression is for it is given in equation (2.4):

$$CAIDI = \frac{\sum customer\ minutes\ of\ interruption}{total\ number\ of\ customers\ interrupted} \quad (2.4)$$

2.2.2 Power Losses

Power losses represent the most important criterion and cannot be ignored in reconfiguration problems. In order to evaluate this criterion, it is necessary to perform the load flow analysis. Basically, power flow algorithms are iterative and are nonlinear. To calculate the losses, the current flow through all the branches of the system must be known by solving the power flow equations. Because power loss is the only factor that we need in this thesis, the Power flow analysis, based on Newton-Raphson method, is simple and fast. Some adjustments were adopted to ensure better performance when considering more than one objectives. The expression for power loss is given in equation (2.5).

$$PL = \sum R_b * i^2 \quad (2.5)$$

2.2.3 Minimum Voltage

The purpose of voltage profile optimization is a to enhance the power quality [16]. Power quality is extremely important not only for the utility companies' financial profits, but also to maintain a stable electric power supply for the customers. This can be achieved by choosing the system topology with the highest minimum voltage values.

Equation (2.6) compares the voltages at all the buses in a certain topology and saves the smallest one as V_{min}. The higher the V_{min} is, the better power quality a topology has.

$$V_{min} = \min(V_j), \qquad j \in N \tag{2.6}$$

where V_j= voltage at bus j   N= bus number

2.3 Approaches for DSR problem

Many reports have been published on the DSR problem using different well-known methods, which can be classified into the following three categories:

2.3.1 Heuristic

This includes Branch Exchange [6] [7], Branch and Bound [1] [8]. These methods are simple and easy to formulate. However, the main drawback of these approaches is, that they need long computation times, and they may result in some combinatorial explosion problem when dealing with multi-objective problems [8].

2.3.2 Meta-Heuristic

The most popular methods include Tabu Search (TS) [27] [28], Simulated Annealing (SA) [29-31], etc. TS is a method that searches for the optimal solution of a current solution. For large scale and multi-dimensional systems, however, it cannot guarantee the optimal solution in any single run, which is problematic. SA made up for the weakness of TS, because SA gives good convergence while TS cannot. The weakness of

SA is that the same set of solutions may be found several times in a single run, which is time-consuming and inefficient.

2.3.3 Modern Heuristic Artificial Intelligence

a. Other Techniques

Artificial Intelligence became very popular ever since it was introduced, which refers to intelligence exhibited by machines. The term "artificial intelligence" is applied when a machine mimics "cognitive" functions that humans associate with other human minds, such as "learning" and "problem solving" [32]. Artificial Intelligence can also be considered Meta-Heuristic approach mathematically, and mostly used methods includes: Artificial Neural Network(ANN) [33] [34], Expert System(ES) [35] [36], Evolutionary Programming(EP) [37], Genetic Algorithm(GA).

The ANN is the fastest method for DSR [33], but its accuracy depends greatly on proper training process. The ES suffered from one major disadvantage: the rules for a certain problem are usually different from those for another, which makes this method problem dependent [35]. The EP is very similar to the GA, in which phenotype evolution is considered. In [37], the problem was formulated as a multi-objective evolutionary programming method for loss reduction, etc. It was based on fuzzy logic to evaluate the objectives in order to solve the DSR problem. The goal of [37] has some importance for this thesis; however, the EP approach requires long computational times and convergence to optimal solutions is not guaranteed, which is not viable in my problem.

b. Genetic Algorithm

The genetic algorithm (GA) is a metaheuristic approach inspired by the process of natural selection, which can be used for solving both constrained and unconstrained optimization problems.

It consists of the following steps: initialization, genetic operators, and terminations or delections. GAs differ from other traditional search approaches in that they start with a random population of so called chromosomes, and each chromosome consists of a set of genes representing a solution to the problem. Each population evolves in a repeated process called generation, and every chromosome in the population is evaluated before the next generation is created. A new chromosome is generated from two randomly selected chromosomes in the current generation by using a set of genetic operators that includes crossover and mutation. Finally, a selection process is applied to choose the best fitted chromosomes from a combination of parents and offspring. Elitism is achieved by entering these selected chromosomes into the next generation to compete as the new generation. The process is repeated until convergence is achieved, or a termination condition has been reached.

GA is different from other optimization and search methods in the following significant ways [9] [16]:

i. They search from a population of points rather than a single point, which speeds the process.
ii. They are efficient in multi-objective optimization problems.
iii. They explore the whole search space, and determine a best solution.

iv. GA uses a non-step-wise search technique, and thus the probability of getting the global solution is higher.

These properties make GAs much more efficient, more powerful and less data-dependent than many other meta-heuristic techniques.

The multi-objective techniques, such as Non-Dominated Sorting Genetic Algorithm II (NSGA-II) [21] solve most of the issues mentioned in other techniques, but there still remain the problems of long computational times needed to classify all solutions into several ranks, tuning the crossover and mutation rates, and the elitism process can also be complicated and time consuming. There has been much work published for applying GAs to the DSR problem and the algorithm that is adopted in this thesis is also based on GA. The author, A. Eldurssi, based the algorithm on NSGA-II; he adopted the non-dominated decision making process, but introduced a new, faster genetic algorithm called FNSGA [16].

2.4 Definition of the Pareto Optimality Problem

2.4.1 Pareto optimal Sets

In the multi-objective optimization problem, instead of looking for a single point, the algorithm searches for a set of points named the Pareto-optimal set, which is the set of optimal solutions of a problem with more than one objective function. The concept of Pareto optimality was descripted by Deb [38] and Carrano et al. [39]. In this paper, the

Pareto-optimal solutions sets are found with a multi-objective genetic algorithm that uses a unique encoding scheme and some problem-specific mutation and crossover operators. After obtaining the Pareto set, the user has to choose the best solution based on the decision criterion.

2.4.2 Definition of Domination

A solution P_1 is said to dominate another solution P_2 in an M objective functions problem, if both of the following conditions are true:

1. The solution P_1 is not worse than solution P_2 in all objectives
2. The solution P_1 is strictly better than solution P_2 in at least one objective

If either of the above conditions is not true, the solution P_1 does not dominate the solution P_2. If both of the conditions are true, it is customary to write any of the following:

- P_2 is dominated by P_1.
- P_1 is non-dominated by P_2 or;
- P_1 is non-inferior to P_2.

This concept can be illustrated with the aid of an example, if we consider the minimization of objectives F_1 and F_2, as is shown in Fig. 3.2. Comparing solution A and B, solution A has a smaller value of F_1 than B, while solution B has a smaller value of F_2 than A. Hence, neither of them can be said to be better than the other with respect to both objectives, so they are both non-dominated with respect to each other. Thus, all solutions that lie on the curve shown are called non-dominated solutions (Pareto-optimal solutions) and they form a Pareto-optimal front [38]. The direction of the optimal front varies from case to case. For example, if we consider a max-max optimization problem, the curve will be drawn from bottom-right to top-left. All the 4 possible Pareto optimal curves are shown in Figure 2.2.

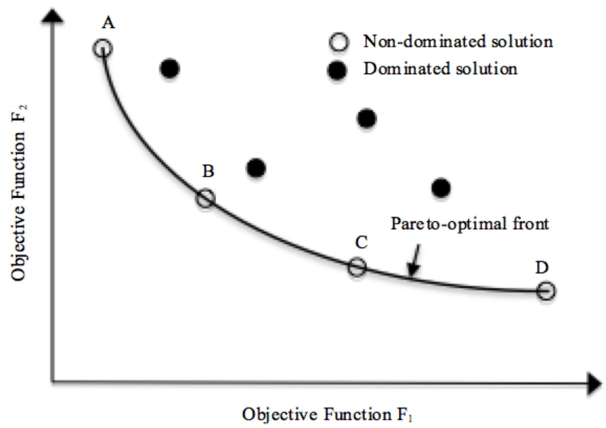

Figure 2.1 - Pareto-optimal front

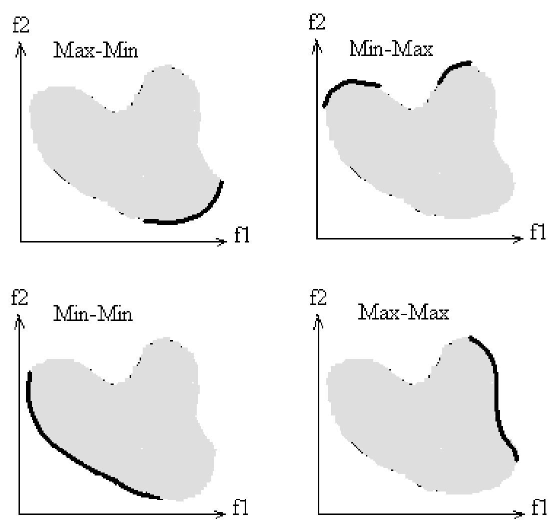

Figure 2.2 – 4 possible Pareto-optimal fronts

2.5 Introduction to FNSGA

In this thesis, the algorithm that is introduced by the author in [16] is adopted. FNSGA is designed for both small and large distribution system optimization problems. However, in this thesis, a few changes and adjustments are made due to the scale of the systems (small and medium size systems) and the reduced objectives (3 objectives compared to 5 or more objectives). In the author's original paper, different population size and crossover rate are evaluated, and he chose the population size of 20 and 26 for 16-bus and 32- bus systems, respectively. The crossover rate, however, didn't show much influence on the convergence of the systems. As a result, the crossover rates are set to 1 for less complexity of the calculation process.

2.5.1 Coding

There is a great variety of techniques used for coding topologies (chromosomes or individuals) for GA in DSR. In this thesis, real number coding is adapted because it's not complicated or time-consuming, while most other techniques requires long strings of numbers. Real number codification is presented in [13], and it takes into account the idea of fundamental loops vectors of a network. A fundamental loop is defined as one that does not contain any other loop within itself. Also, fundamental loops are formulated starting from the source(s) side. The number of fundamental loops L of the meshed system is equal to the number of tie (open) switches, and is given by the relation: $L = N_b - N + 1$, where N_b and N are the numbers of branches and busses in the system, respectively.

2.5.2 Crossover

The crossover operator allows the exchange of genetic information between two randomly selected parents from the current generation to produce new two children for the next generation. Many types of crossovers are applied by different authors in different situations: one-point crossover, two-point crossover, and uniform crossover [13] [16]. In this thesis, a three fundamental-loops case, the 16-bus test system to be described, a one-point crossover is sufficient for full convergence. Moreover, for larger systems like 32-bus system, one-point crossover is also sufficient at the cost of few more generations. Again, we observe that types of crossover will not affect the convergence of the algorithm.

2.5.3 Mutation

This operator randomly changes one bit in the string, and it is applied with a probability that has been set up by the operator. The mutation type used in this thesis is called "directed mutation." For example, if the mutation process indicates to change the Ch1 in the bits 2, then only feasible options for bit 2 is considered. As for the mutation rate, most GAs use random mutation with a very small mutation rate (0.01 to 0.03) [32]. The author in [40] uses 0.03 for the 39-bus distribution system, and 0.02 for the larger 69-bus system. A slightly larger mutation rate, 0.03, is applied to the test systems of this thesis for better performance. The crossover and mutation operators example are shown in the figure below.

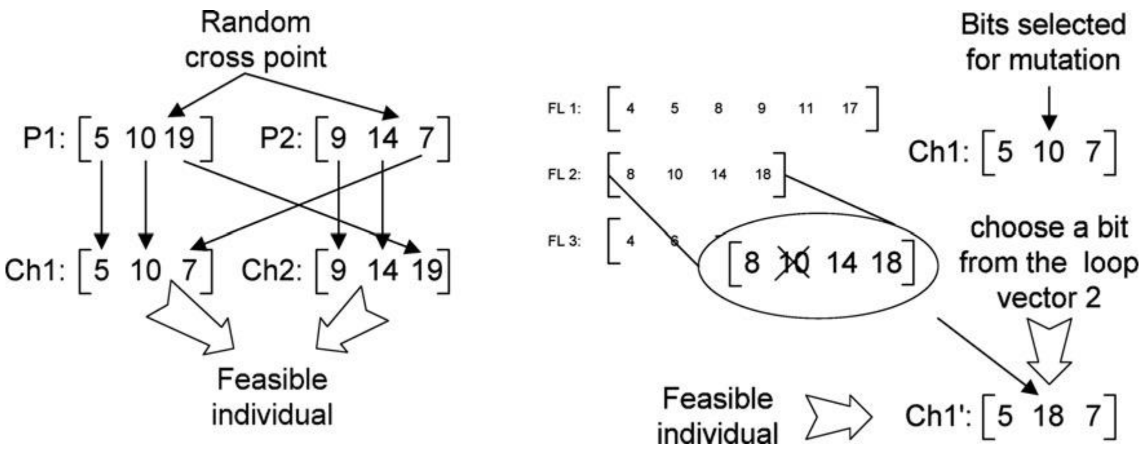

Figure 2.2 - Crossover and Mutation operator

2.5.4 Radial topology

As is mentioned above, distribution systems are normally operated as radial networks for reliability and network efficiency. To create radial topologies, one should select switches from the group of fundamental loop elements that are to be disconnected (one per loop). After the switches are selected to form a chromosome, an approach based on the branch-bus incidence matrix BBIM was introduced to determine whether the system is radial (valid tree) [16].

2.5.5 Selection and Elitism

The goal of the selection operator is to assure more chances to produce best chromosomes in a population. As is explained above, selection of the best solution is based on the non-dominated set. In order to keep the optimal solutions, the non-dominated sets should be selected to form the population of the next generation. However, the number of non-dominated sets, which are selected for use in the next generation, may not exceed half the population size to ensure a good chance to produce more optimal solutions. Because too strong a fitness selection bias can lead to sub-optimal solutions, the second half of the next generation is produced randomly.

2.5.6 Convergence and stopping criteria

This algorithm stops on either

1. The number of generations reaches its limit, which is set by the operator. (Generation limit)

   Or

2. No changes occur in the non-dominated solution set for four successive iterations. (Fully Convergence)

2.5.7 Evaluation of Equal Importance Objectives

After one of the above-mentioned convergence criteria is satisfied, a set of non-dominated switches will be generated. If there is a preference objective from the operator, or there're some practical reports that describe the methods in choosing from such objectives. Otherwise, a simple approach to identify the best solution is implemented by summing all the normalized values of the objective functions [42]. The minimized objective functions (reliability indices and power loss) are normalized using.

$$obj_{min}(norm) = \frac{obj-obj_{min}}{obj_{max}-obj_{min}} \tag{2.7}$$

and the maximized objective function (voltage) is normalized using

$$obj_{max}(norm) = \frac{obj_{max} - obj}{obj_{max} - obj_{min}} \tag{2.8}$$

where obj_{min} and obj_{max} are the minimum and maximum values for that objective function in the non-dominated set respectively. The best solution is defined as the one with the smallest sum of normalized objective functions. The application of (2.7) and (2.8) to the DSR problem is a important feature when multi-objective optimization problem.

Chapter Three

Reliability Indices

The basic reliability indices that are calculated in this thesis were introduced in Chapter 2. In the systems we studied, buses are defined as load nodes, and lines are the transmission lines connecting one bus to another bus. Although the original data does not provide the numbers of customers served at each bus, the loads at each bus can be used to determine the number of customers. To calculate reliability indices described in chapter 2 are converted to: [24]

$$SAIFI = \frac{\sum_{i=1}^{Nc} \lambda_i * MVA_i}{\sum_{i=1}^{Nc} MVA_i} \tag{3.1}$$

$$SAIDI = \frac{\sum_{i=1}^{Nc} U_i * MVA_i}{\sum_{i=1}^{Nc} MVA_i} \tag{3.2}$$

$$CAIDI = \frac{SAIDI}{SAIFI} \tag{3.3}$$

$$U_i = \lambda_i * r_i \tag{3.4}$$

and

$$ENS = \sum_{i=1}^{Nc} MW_i * U_i \tag{3.5}$$

Here λ_i, U_i and r_i refer to the failure rate, unavailability, and restoration time of each of the N_c system buses. Also, MVA_i and MW_i refer to the apparent power and active power values of the loads at each bus.

The detailed calculation of SAIFI, SAIDI, CAIDI, and ENS are introduced in this Chapter, as well as the algorithm for programming it. A 5-bus system, as is shown in figure 3.1, is also used to illustrate the process.

3.1 Calculation Methodology and Examples

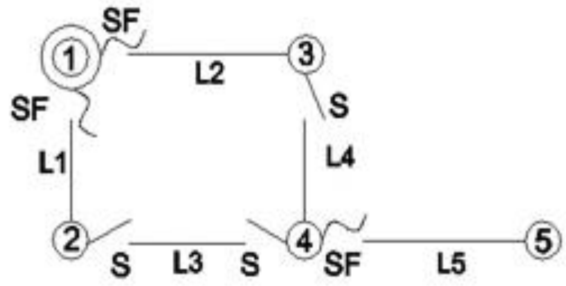

Figure 3.1- Original meshed model of 5-bus system

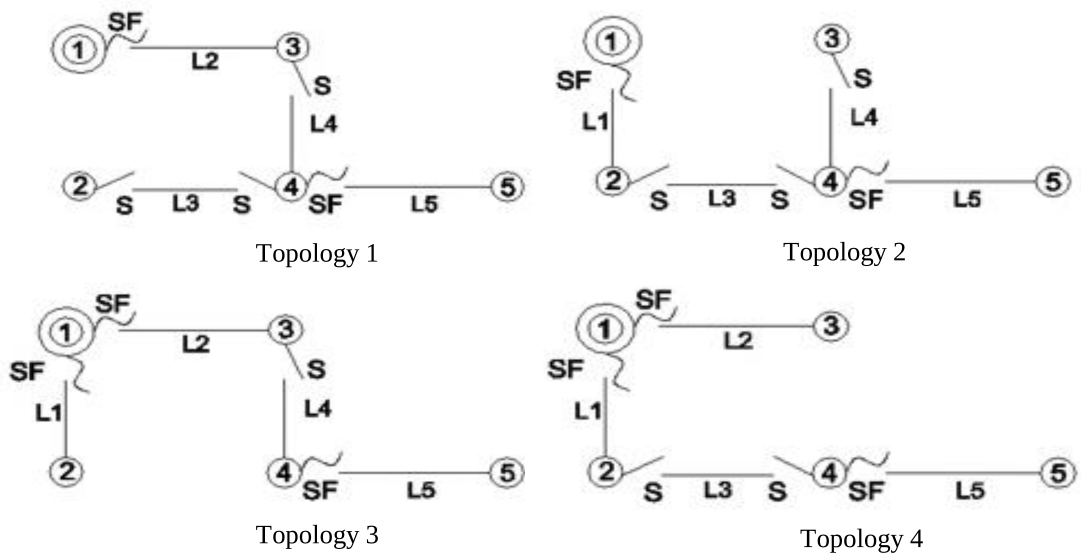

Figure 3.2- Possible radial topologies of the 5-bus system

Line	Node	$\lambda\ (f/year)$	P (MW[a])	Q (MVAr[a])	S (MVA[a])
L1	1-2	0.2	3.4	1.4	3.68
L2	1-3	0.3	10.0	4.0	10.77
L3	2-4	0.2	6.7	2.7	7.22
L5	4-5	0.1	10.0	4.0	10.77
L4	3-4	0.2	6.7	2.7	7.22

[a]Power at the end of the line, refers to bus load.

Table 3.1- Original data for the 5-bus system

Figure 3.2 shows all the four possible radial topologies for the 5-bus meshed system provided in figure 3.1. In this chapter, topology 1 of the given 5-bus system is used to illustrate the procedure of reliability index calculation. Analysis of failure rate and unavailability are calculated as follows:

3.1.1 Failure rate

In topology 1, one fuse, labeled SF is connected to the source (bus 1), and another is connected to bus 4. A fuse represents the entrance of a section; therefore, topology 1 is divided into two sections: section 1, which contains lines L2, L3, and L4, connecting the source to buses 2, 3, and 4; and section 2, which contains line L5, connecting bus 4 to bus 5. Based on the distance to the source, we define section 1 as the primary section, and section 2 as the secondary section. It should be noted that failure on any of lines L2, L3, or L4 affects all buses 2, 3, 4, and 5, but the failure of line L5 only affects bus 5. This is because the failure of a component in a secondary section does not affect buses in a primary section.

3.1.2 Unavailability

In topology 1, line L1 is open. According to equation (3.4), the unavailability U_i of a certain line is the product of the failure rate λ_i of the line and its repair or its maneuver time r_i.

Since the failure rate of each line is fixed, it is necessary to determine whether a given line needs to be repaired or maneuvered in different situations. For a certain bus and every line connected to that bus (in the same or higher sections), if we can open a switch to isolate a line from this bus, then only maneuver time is applied for that line. Otherwise, that line needs to wait for repair, and repair time must be used in the calculation.

For example, in topology 1, consider bus No. 4. For line L3, we can open the switch that is near bus No. 4 to isolate line L3 from bus No. 4, so line L3 only needs to be maneuvered in this case. Therefore, $U_{L3} = \lambda_3 * r_3$, where λ_3 = 0.2 f/yr, and r_3 = 0.5 hr (maneuver time), so U_{L3}=0.1. On the other hand, for line L4, there is no nearly switch that we can operate to isolate that line, so line L4 has to wait for repair. Therefore $U_{L4} = \lambda_4 * r_4$, where λ_4 = 0.2 f/yr, and r_4 = 1hr (repair time) , so U_{L4}=0.2.

After we have the unavailability of each line, we need to sum the unavailabilities of the lines that are connected to each bus. Thus, in the present example, the unavailability of bus No. 4 will be $U_4 = U_{L2} + U_{L3} + U_{L4}$ = 0.3*0.5 + 0.2*0.5 +0.2*1.0 = 0.45.

Then we can use the unavailability we have for each bus to calculate SAIDI, ENS, and CAIDI. Note that if the bus is in a secondary section, the primary (higher level section) needs to be taken into consideration as well as the lines in the secondary section.

Calculation of topology 1:

Step 1. Create a table for the failure rate analysis. See Table 3.2.

The first row refers to the buses in the 5-bus system, which are No. 2-5 (bus No. 1 is the power source), and the first column refers to the closed lines in the system (line L1 is open in topology 1). Except for the last row, every number in Table 3.2 refers to a given failure rate, and the failure rate for a certain row should be the same. If there’s a non-zero failure rate in a certain cell, the failure of the corresponding line of that unit will interrupt the flow of the power to the corresponding bus of that unit. For example, cell (2,1) = 0.2, which means that failure of line 3 will affect bus 2. With cell (1,1) and cell (3,1), that will add up to the 0.7 in cell (5,1), which is the total failure rate for bus 2.

Bus / Line Failed	2	3	4	5
L2	0.3	0.3	0.3	0.3
L3	0.2	0.2	0.2	0.2
L4	0.2	0.2	0.2	0.2
L5				0.1
Total	0.7	0.7	0.7	0.8

Table 3.2 - Failure rate analysis results for topology 1 of the 5-bus system

Step 2. Create a table for the unavailability analysis.

Table 3.3 indicates the results for every line unavailability in topology 1. The whole table, except for the last row, is the product of Table 3.2 and the corresponding repair or maneuver time of each cell. The last row is the sum of each column, represents the unavailability of the nodes shown in the first row.

Bus / Unavailability Of lines	2	3	4	5
L2	0.3*0.5	0.3*1	0.3*0.5	0.3*0.5
L3	0.2*0.5	0.2*0.5	0.2*0.5	0.2*0.5
L4	0.2*0.5	0.2*0.5	0.2*1	0.2*1
L5				0.1*1
Total	0.35	0.5	0.45	0.55

Table 3.3 - Unavailability analysis results for topology 1 of the 5-bus system

Step 3. Use equations (3.1) through (3.5) and data in Table 3.1 to calculate the reliability indices.

Using eq. 3.1,

$$SAIFI_{topology1} = \frac{\sum_{i=1}^{Nc} \lambda_i * MVA_i}{\sum_{i=1}^{Nc} MVA_i}$$

$$= \frac{0.7 * 3.68 + 0.7 * 10.77 + 0.7 * 7.22 + 0.8 * 10.77}{3.68 + 10.77 + 7.22 + 10.77} = 0.73$$

The numerator of the equation refers to the sum of every bus's total failure rate times its apparent power, while the denominator corresponds to the sum of the apparent powers at all the buses. For example, the first element 0.7*3.68 corresponds to the total failure rate of bus 2 from Table 3.2 times the apparent power of bus 2 from Table 3.1, and so on.

Using eq. 3.2,

$$SAIDI_{topology1} = \frac{\sum_{i=1}^{Nc} U_i * MVA_i}{\sum_{i=1}^{Nc} MVA_i}$$

$$= \frac{0.35 * 3.68 + 0.5 * 10.77 + 0.45 * 7.22 + 0.55 * 10.77}{3.68 + 10.77 + 7.22 + 10.77}$$

$$= 0.49$$

The numerator of the equation refers to the sum of every bus's unavailability times its apparent power, while the denominator corresponds to the sum of the apparent powers. For example, the first element 0.35*3.68 corresponds to the unavailability of bus 2 from Table 3.3 times the apparent power of bus 2 from Table 3.1, and so on.

Using eq. 3.3,

$$CAIDI_{topology1} = \frac{SAIDI}{SAIFI} = \frac{0.49}{0.73} = 0.67$$

The numerator of the equation refers to the SAIDI of topology 1, while the denominator corresponds to the SAIFI of topology 1.

Using eq. 3.5,

$$ENS_{topology1} = \sum_{i=1}^{Nc} MW_i * U_i = 0.35 * 3.4 + 0.5 * 10.0 + 0.45 * 6.7 + 0.55 * 10.0$$

$$= 14.705$$

The ENS of topology 1 refers to the sum of every bus's unavailability times its real power. For example, the first element 0.35*3.4 corresponds to the unavailability of bus 2 from Table 3.3 times the real power of bus 2 from Table 3.1, and so on.

3.2 Computation Details

In this research, all the programs were implemented in Matlab software. Based on the calculation process, we discovered a rule for calculating the failure rate for each bus: by determining the route from one bus to the source and adding up the failure rates of all lines in the route and the failure rates of other lines inside the section(s) that the route passes, the correct value of failure rate can be obtained. Referring to [41], we borrowed the notion of Primary Route (PR) and Secondary Route (SR) to apply in our program. The Primary Route (PR) is defined as the route connecting the target bus to the source, while the Secondary Route (SR) is defined as the route connecting the bus(es) in the PR to other bus(es) that are not part of the PR and not connected to the PR through a fuse. For example, for bus 5 in topology 1 of the 5-bus system, its PR is 5 – 4 – 3 – 1, and its SR is 4 – 2. After obtaining all the route(s) and the correct failure rate of one bus, the program starts analyzing the unavailability of every line in the route(s). After obtaining both the failure rate and unavailability of each bus, then the reliability indices SAIFI, SAIDI, CAIDI and ENS are calculated. Topology 1 of the 5-bus system is chosen as the example to illustrate the complete methodology.

Step 1: Present the system topology in a matrix form.

Referring to [41], the first step in doing the calculation by program is to present the system topology as a matrix, which is called the "Intelligence Matrix" (IM). The IM of topology 1 of the 5-bus system is shown in Figure 3.3.

i \ j	1	2	3	4	5
1			-1		
2				1	
3	2			1	
4		1	2		-1
5				2	

Figure 3.3 - Intelligence Matrix of topology1 of the 5-bus system

In this matrix, the first column and the first row both represent the addresses of the buses, while all the other cells (i, j) represent the first component (lines, fuses, and switches) encountered in moving from bus i to bus j. "-1" represents a fuse, "1" represents a switch, and "2" represents a line. For instance, in this topology, both a fuse and a line are between the source and bus 3. Since the fuse is the first item encountered in moving from the source to bus 3, "-1" is put in cell IM_{13}, while "2" is put in cell IM_{31} because the line is encountered first in moving from bus 3 to the source. The IM contains all the information necessary for SAIFI calculation.

With this setup of the IM, we can carry out the next step, which is the determination of the PR and SR(s) for each bus.

Step 2: Determine the PR and SR.

Figures 3.4 and 3.5 illustrate how to determine the PR and SR for bus 4 in topology 1 of the 5-bus system.

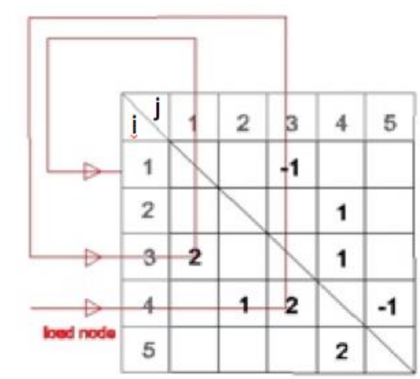

Figure 3.4 - Searching for bus 4's Primary Route for topology 1 of the 5-bus system

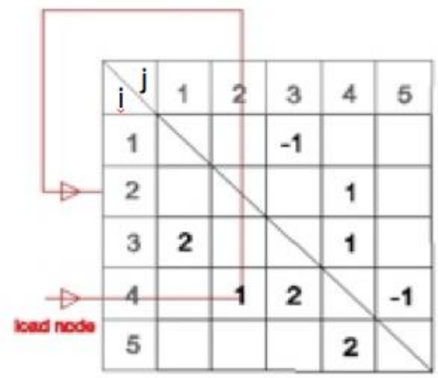

Figure 3.5 - Searching for bus 4's Secondary Route(s) for topology 1 of the 5-bus system

The process begins at row 4 on the left side of the IM in Figure 4. Searching through this row, any positive number that appears in this row indicates the components that would affect bus 4's failure rate (a switch or a line), and thus should be included in the routes for bus 4, whereas the component represented by the negative number should not be included. Following this rule, a "1" is found in cell IM_{42}, meaning the sectionalizing switch between bus 4 and bus 2 would contribute to bus 4's failure rate. After putting bus 2 into the route, we start searching the second row. However, only one positive number is found, a "1" in cell IM_{24}， which is a repeated path. Since there's no way to go from this bus (bus 2) to any other bus, and the route doesn't end with the source (bus 1), this route is identified as a SR of bus 4.

Next, we go back to the left side of IM row 4, and this time we continue with "2" in cell IM_{43}. The failure of the line between bus 4 and bus 3 will affect bus 4's failure rate. Then we do the same thing through the third row, and "2" in cell IM_{31} lead us to the source (the "1" in cell IM_{34} refers to the repeated path so it should be left out). As the search ends up with the source, this route is recognized as the primary route.

When performing the above-described search, two things need to be noticed. First, in order to record all branches during the searching process, multiple counters are needed. For example, in the above-mentioned search for the routes of bus 4, there are two positive numbers in the IM row 4. Hence we need two counters to store the addresses of each column associated with them. Second, due to the number sequence of the buses, it is hard for the program to always search the PR with first priority. Instead, the program will search the route with the smaller sequence number when two or more positive numbers appear in the same row, and then finish searching the other routes for other branches. Table 3.4 and 3.5 illustrate how this mechanism works for searching routes of bus 4, again in topology 1.

4	3	1

Table 3.4 – Recording Primary Route for bus 4, topology 1

4	2	…

Table 3.5 – Recording Secondary Route for bus 4, topology 1

Step 3: Determine the unavailability time in each line included in the route(s) for each bus.

i \ j	1	2	3	4	5
1	0	0	0	0	0
2	0	0	0	0	0
3	0.5	0	0	0	0
4	0	0.5	1	0	0
5	0	0	0	0	0

Table 3.6 - Unavailable time (hour) for each line in UM for bus 4, topology 1

j / i	1	2	3	4	5
1	0	0	0	0	0
2	0	0	0	0	0
3	0.15	0	0	0	0
4	0	0.1	0.2	0	0
5	0	0	0	0	0

Table 3.7 - Unavailability for each line in UM for bus 4, topology 1

In order to determine the unavailability time for each line included in the route(s), an Unavailability Matrix (UM) is created. The UM is intended to store the unavailability of each line involved in analyzing the unavailability of a certain bus. Tables 3.6 and 3.7 help illustrate how the UM works with the analysis of unavailability of bus 4, topology 1. First, we initialize UM, making a zero matrix with the same dimension as the IM. Then the program starts from bus 4, searching the nearest "1"- sectionalizing switch to bus 4 in each route. In PR, there's no "1" in cell IM43, so line L4 needs repair and we put "1" in cell IM43, which refers to the repair time of 1 hour. Then the search moves to bus 3, searching cell IM34 and cell IM31 to find either of them is "1". Since cell IM34 is "1", the nearest sectionalizing switch in the PR is found. Then the unavailability time of all the lines between the source and this switch are determined as the maneuver time of 0.5 hour. Thus we put "0.5" in cell UM31. In SR, starting from bus 4, we find "1" in cell IM42, and this is the nearest sectionalizing switch in the SR. Thus, we put "0.5" in cell UM42. After we obtain all the unavailable times shown in Table 3.6 of the involved lines, we multiply every non- zero component in UM with the corresponding line failure rate from Table 3.1 to obtain he unavailability of each line shown in Table 3.7, and then add them up together to obtain the unavailability of the given bus. For example, Table 3.7 shows the unavailability of bus 4 in topology 1 is 0.45, which is used for further reliability indices calculation.

3.3 5-Bus System Results

By running the program described, we can calculate all the four reliability indices introduced above. The complete Matlab code used to obtain the results shown in Table 3.8, is provided in Appendix 1, and the code containing the data shown in Table 3.1, is also provided in Appendix 3.

After we have the results shown Table 3.8, we can sort out the non-dominated solutions for the above 5-bus system. The three objectives that we want to maximize are: power loss, SAIFI and minimum voltage. However, without the power flow information we were not able to find out any voltage profile in this sample system. Also, the non-dominated sorting process will not be shown since topology 4 is an obviously dominated solution among all, if we use ENS and SAIFI to evaluate the system. As a result, ENS and CAIDI is used for the evaluation of this 5-bus test system.

Topology \ Index	ENS	SAIFI	SAIDI	CAIDI	Normalized Sum
1	14.71	0.73	0.49	0.67	3.286
2	12.04	0.63	0.40	0.63	1.848
3	11.53	0.50	0.38	0.77	2.278
4	8.36	0.40	0.28	0.69	0.429

Table 3.8 – The reliability indices for 5-bus system

Figure 3.6 shows the sorting process between indices ENS and SAIDI. Since this is a min-min optimization case, we should draw a curve from the top-left side to the bottom-right side (the direction of the curve varies from case to case as is mentioned in the previous chapter). As is shown above, topology 2 and 4 are on the curve. In result, considering minimizing ENS and SAIDI, topologies 2 and 4 are the non-dominated solutions.

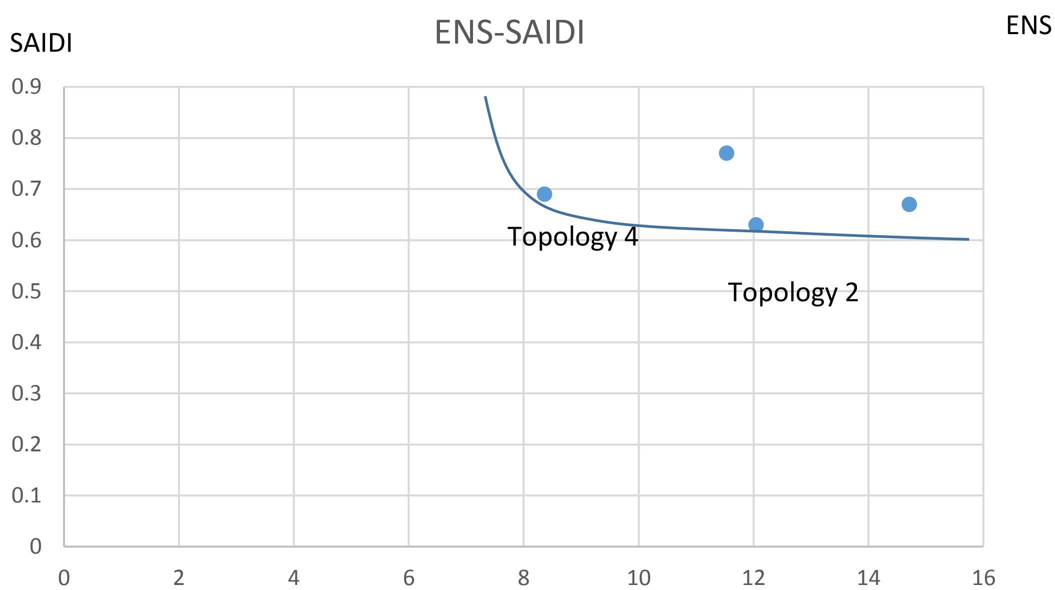

Figure 3.6 - 5-bus system ENS-SAIDI non-dominated sorting

Alternatively, in Table 3.8, the last column shows the result for sum of normalized objectives, which is introduced in chapter 2. If there is no preferred objective, the best solution is selected based on the sum of the normalized values of these objective functions. In this case, topology 4, with the smallest sum of normalized objectives (0.429), would be considered the best of the four solutions in Table 3.8

Chapter 4

The FNSGA for Application

4.1 Revised FNSGA

In the original FNSGA paper, the author uses the program to generate non-dominated solutions that considered five different objectives simultaneously. However, to use the FNSGA to solve reliability problems, calculation for the reliability indices which is introduced above need to be included in the final program. The original flow diagram for FNSGA is shown in figure 4.1.

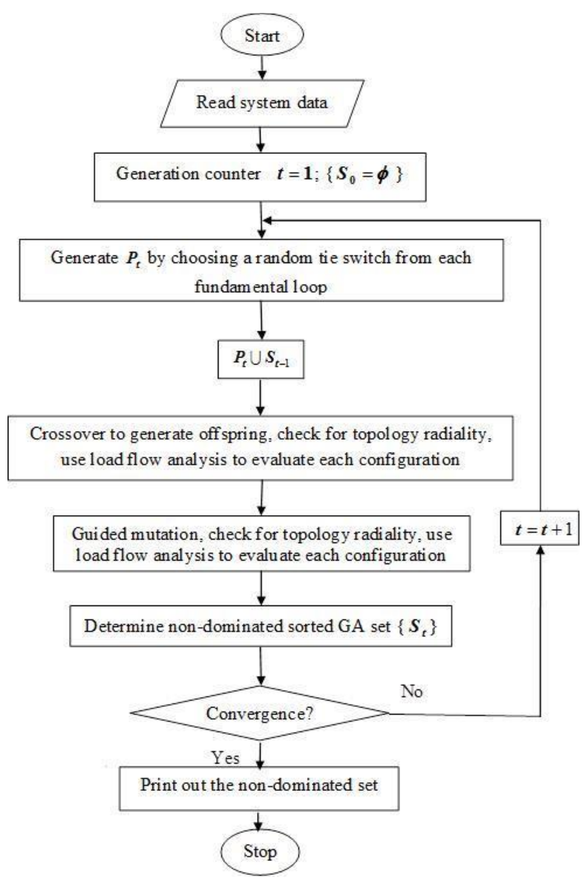

Figure 4.1 – Original FNSGA flow diagram

In the original FNSGA, the calculation for the five objectives are involved in the load flow analysis. Thus, every time the program runs load flow analysis, each configuration is evaluated and the objective results are saved. To evaluate every possible configuration with the objectives I need for the non-dominated sorting, the reliability indices calculation need to be performed every time the FNSGA runs load flow analysis – two times in the original program. Along with the calculation for reliability indices, the non-dominated sorting is also adjusted to use the objectives that we want in the optimization. The flow chart for adjusted FNSGA is shown in Figure 4.2. The computation detail for the reliability calculation is introduced in Chapter 3, and the program for reliability indices calculation will be shown in the Appendix 2.

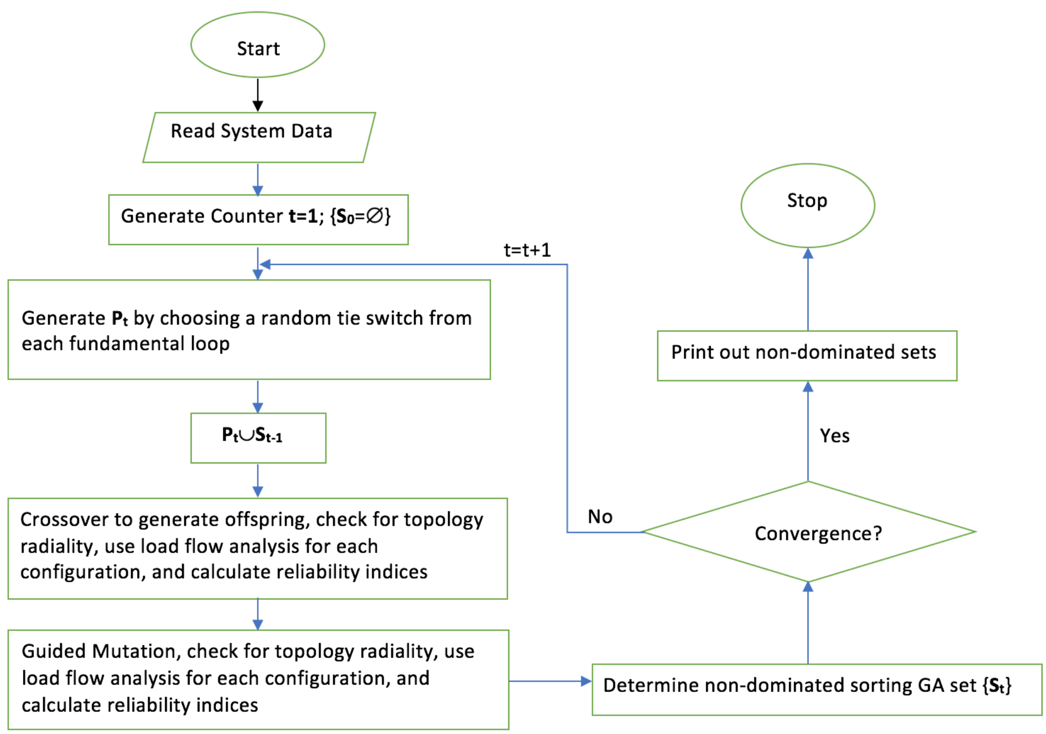

Figure 4.2 – Revised FNSGA flow diagram

4.2 16-Bus System

After our algorithm was verified with the 5-bus system, it was adapted and applied to the IEEE standard 16-bus system. The meshed model of the 16-bus system is shown in Figure 4.3, and the complete data set is shown in Table 4.1. The system load is assumed to be balanced and constant with a total of 28.47 MW and 5.9 MVAR. The base MVA and kV are 100 and 23, respectively. The complete Matlab code used to obtain the results shown in Tables 4.2 and 4.3 are provided in Appendix 2, and the code for the data shown in Table 4.1, is provided in Appendix 4.

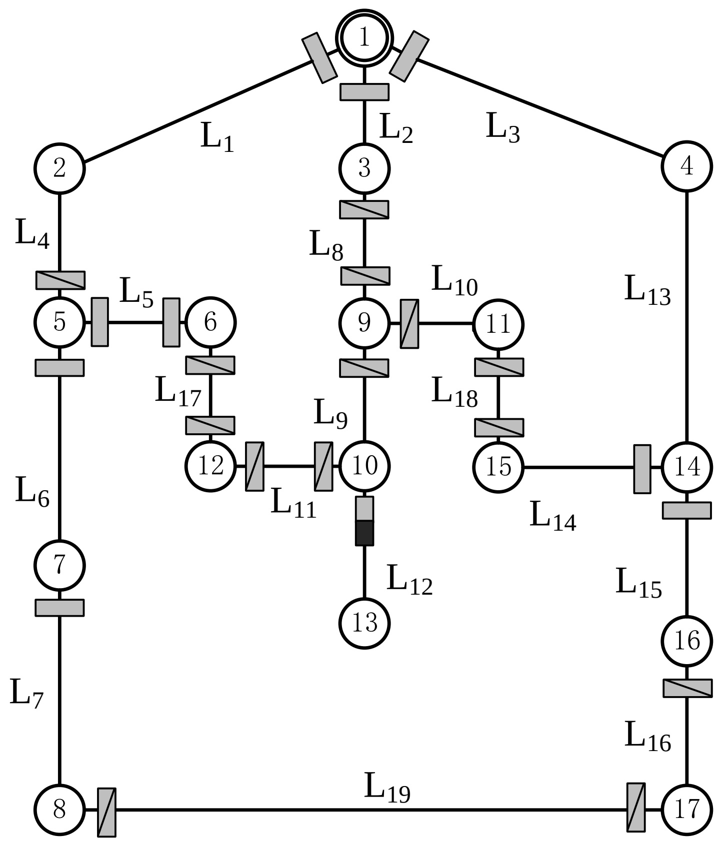

Figure 4.3 - Original meshed model of 16-bus system

Line no.	n1	n2	R, p.u.	X, p.u.	Imax, p.u.	EPM	$\lambda\,(f/year)$	Repair time, h	Manoeuver time, h	End bus load P, MW	End bus load Q, MVAr
1	1	2	0	0.0001	0.20	1	0	1	0.5	0.0	0.0
2	1	3	0	0.0001	0.20	1	0	1	0.5	0.0	0.0
3	1	4	0	0.0001	0.20	1	0	1	0.5	0.0	0.0
4	2	5	0.075	0.10	0.20	22	3.50	1	0.5	2.0	1.6
5	5	6	0.080	0.11	0.10	333	3.00	1	0.5	3.0	0.4
6	5	7	0.090	0.18	0.10	3	1.50	1	0.5	2.0	-0.4
7	7	8	0.040	0.04	0.10	3	3.50	1	0.5	1.5	1.2
8	3	9	0.110	0.11	0.20	222	1.10	1	0.5	4.0	2.7
9	9	10	0.080	0.11	0.12	2	2.80	1	0.5	5.0	1.8
10	9	11	0.110	0.11	0.10	2	1.10	1	0.5	1.0	0.9
11	10	12	0.110	0.11	0.10	222	0.80	1	0.5	0.6	-0.5
12	10	13	0.080	0.11	0.10	1	2.00	1	0.5	4.5	-1.7
13	4	14	0.110	0.11	0.10	0	0.50	1	0.5	1.0	0.9
14	14	15	0.090	0.12	0.10	3	1.00	1	0.5	1.0	-1.1
15	14	16	0.080	0.11	0.10	3	1.50	1	0.5	1.0	0.9
16	16	17	0.040	0.04	0.10	2	4.40	1	0.5	2.1	-0.8
17	6	12	0.040	0.04	0.10	222	4.00	1	0.5	0.0	0.0
18	11	15	0.040	0.04	0.10	222	5.00	1	0.5	0.0	0.0
19	8	17	0.090	0.12	0.10	222	1	1	0.5	0.0	0.0

Table 4.1- Original data for the 16-bus system

Solutions	Topology	SAIFI	SAIDI	CAIDI	ENS	Normalized sum
1(Initial)	17-18-19	5.74	3.83	0.67	112.43	2.023
2	11-18-19	5.82	4.07	0.70	120.17	3.795
3	11-18-16	5.91	3.97	0.67	117.35	3.335
4	11-18-7	5.87	3.95	0.67	117.14	3.161
5	17-18-7	5.79	3.91	0.68	115.53	2.719
6	17-18-16	5.83	3.93	0.67	115.74	2.868
7	10-11-19	5.61	3.94	0.30	115.44	1.468
8	7-10-11	5.66	3.82	0.51	112.42	1.419
9	10-17-19	5.47	3.87	0.29	113.10	0.674
10	10-11-16	5.70	3.84	0.36	112.63	1.227
11	7-10-17	5.52	3.75	0.32	110.08	0.187
12	10-16-17	5.56	3.77	0.33	110.29	0.385

Table 4.2 – 16-bus system results 1

In the system's initial topology, lines L17, L18, and L19 are disconnected, and the given values from [13] of reliability indices are SAIFI = 5.740/yr, SAIDI = 3.828h/yr, CAIDI = 0.667h/f, and ENS = 112.43 MWh/yr, respectively.

With this system, our goal is to find the topologies with relatively better reliability indices compared with those of the initial topology. In order for the 16-bus system to be radial, three lines must be disconnected (one in each fundamental loop). Using the method for reliability calculation that is introduced in chapter 3, the topologies with best reliability performances are listed in Table 4.2. The 9th solution in Table 4.2 is the optimum in terms of both SAIFI and CAIDI, while the 11th solution is the optimum one considering the other two indices. If no preference is provided by the developer, we can simply choose the one with the best normalized sum based on the last column on Table 4.2.

Besides reliability, in order to meet today's industrial demands, active power loss and power quality also need to be considered for the DSR problem evaluation. Thus, we conducted simulations with power loss as one objective, minimum voltage as another objective representing power quality, and the most commonly used reliability index, SAIFI, to measure system's reliability. Table 4.3 shows the results of using non-dominated sorting, considering these three objectives. The computing CPU time was near 1s, and the algorithm needs only three generations to achieve convergence. Note that we use the reciprocal of minimum voltage instead of minimum voltage itself for optimization, since it is easier to analyze the results when each objective is a minimum optimization problem. The algorithm produced two non-dominated solutions (defined as the non-dominated set) listed as solutions No. 2 and No. 3 in the Table. The run was repeated 100 times, and each run produced the same results. Solution No. 2 is the optimum in terms of index SAIFI only, while solution No. 3 is the optimum considering power loss and minimum voltage. Also, solution No. 3 has a better normalized sum compared to the other solutions according to equations 2.7 and 2.8. The three-dimensional (three objectives) Pareto-optimal solutions are shown in figure 4.4

Solutions	Topologies	Power loss	SAIFI	1/ (Min. Voltage)	Min. Voltage	Normalized Sum
1(Initial)	17-18-19	0.511	5.74	1.032	0.969	2.657
2(Non-Dominated)	10-17-19	0.484	5.470438	1.0299	0.971	0.735
3(Non-Dominated)	10-11-19	0.466	5.611979	1.0288	0.972	0.357
4	11-18-19	0.493	5.823559	1.032	0.969	2.474
5	11-14-19	0.512	5.868308	1.032	0.969	3.000
6	10-16-17	0.508	5.560265	1.031	0.970	2.105

Table 4.3 – 16-bus system results 2

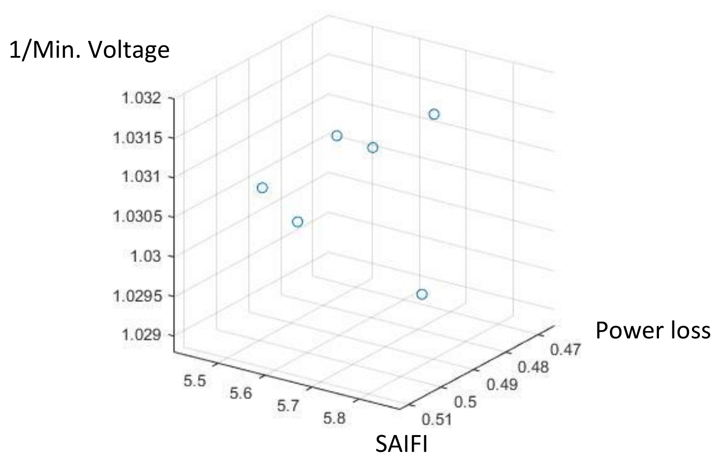

Figure 4.4 – 16-bus Pareto-optimal Solutions shown in three-dimensional coordinate

4.3 32-bus System

32 bus system is a more sophisticated system compared to the previous evaluated 16-bus system. This system consists of 33 buses (including the slack bus, and the reason why described as 33-bus system by some author [26]) nodes, 37 branches, and five fundamental loops, as shown in Figure 4.5.

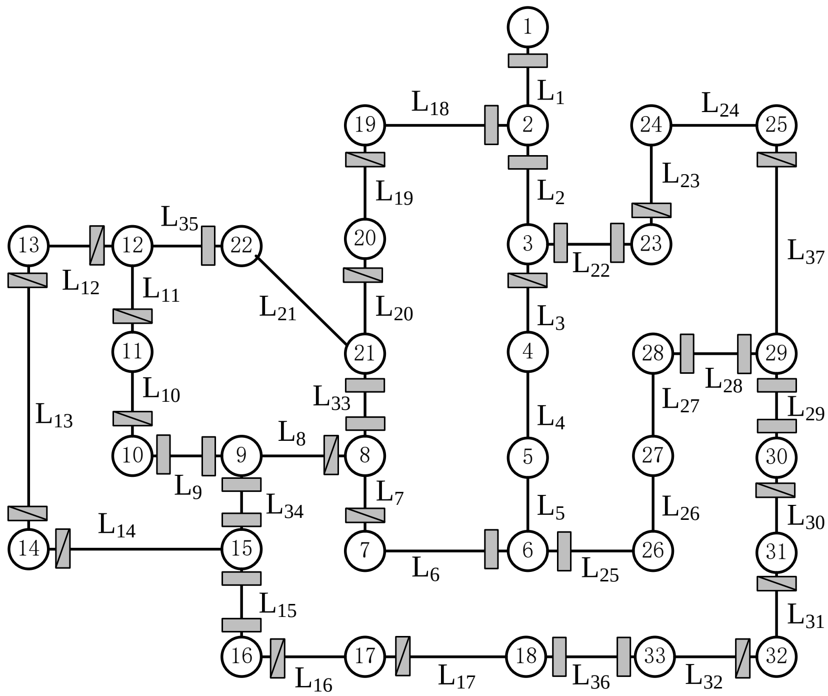

Figure 4.5 - Original meshed model of 32-bus system

In the system's initial topology, lines L33, L34, L35, L36, and L37 are disconnected from the system. The power losses and reliability indices in the initial topology are 0.2028 MW, 3.177 f/yr, 1.819 h/yr, 6.695 Mwh/yr and 0.5728 h/f, respectively. The complete 32-bus system data used for calculation is shown in table 4.4.

Line no.	n1	n2	R, p.u.	X, p.u.	Imax, p.u.	EPM	$\lambda\ (f\ /year)$	Repair time, h	Maneuver time, h	End bus load	
										kW	kVAr
1	1	2	0.0058	0.0029	0.5	1	0.05	1	0.5	100	60
2	2	3	0.0308	0.0157	0.4	3	0.30	1	0.5	90	40
3	3	4	0.0228	0.0116	0.2	2	0.22	1	0.5	120	80
4	4	5	0.0238	0.0121	0.2	0	0.23	1	0.5	60	30
5	5	6	0.0511	0.0441	0.2	0	0.51	1	0.5	60	20
6	6	7	0.0117	0.0386	0.2	3	0.11	1	0.5	200	100
7	7	8	0.0444	0.0147	0.2	2	0.44	1	0.5	200	100
8	8	9	0.064	0.0462	0.2	2	0.64	1	0.5	60	20
9	9	10	0.0651	0.0462	0.2	333	0.65	1	0.5	60	20
10	10	11	0.0123	0.0041	0.2	2	0.12	1	0.5	45	30
11	11	12	0.0234	0.0077	0.2	2	0.23	1	0.5	60	35
12	12	13	0.0916	0.0721	0.2	2	0.91	1	0.5	60	35
13	13	14	0.0338	0.0445	0.2	222	0.33	1	0.5	120	80
14	14	15	0.0369	0.0328	0.2	2	0.36	1	0.5	60	10
15	15	16	0.0466	0.0340	0.2	333	0.46	1	0.5	60	20
16	16	17	0.0804	0.1074	0.2	2	0.80	1	0.5	60	20
17	17	18	0.0457	0.0358	0.2	2	0.45	1	0.5	90	40
18	2	19	0.0102	0.0098	0.2	3	0.10	1	0.5	90	40
19	19	20	0.0939	0.0846	0.2	2	0.93	1	0.5	90	40
20	20	21	0.0255	0.0298	0.2	2	0.25	1	0.5	90	40
21	21	22	0.0442	0.0585	0.2	2	0.44	1	0.5	90	40
22	3	23	0.0282	0.0192	0.3	333	0.8	1	0.5	90	50
23	23	24	0.0560	0.0442	0.3	2	0.56	1	0.5	420	200
24	24	25	0.0559	0.0437	0.3	0	0.55	1	0.5	420	200
25	6	26	0.0127	0.0065	0.2	3	0.12	1	0.5	60	25
26	26	27	0.0177	0.0090	0.2	0	0.17	1	0.5	60	25
27	27	28	0.0661	0.0583	0.2	2	0.66	1	0.5	60	20
28	28	29	0.0502	0.0437	0.2	333	0.50	1	0.5	120	70
29	29	30	0.0317	0.0161	0.2	333	0.31	1	0.5	200	600
30	30	31	0.0608	0.0601	0.2	2	0.60	1	0.5	150	70
31	31	32	0.0194	0.0226	0.2	2	0.19	1	0.5	210	100
32	32	33	0.0213	0.0331	0.2	2	0.21	1	0.5	60	40
33	8	21	0.1248	0.1248	0.2	333	1.24	1	0.5		
34	9	15	0.1248	0.1248	0.2	333	1.24	1	0.5		
35	12	22	0.1248	0.1248	0.2	33	1.24	1	0.5		
36	18	33	0.0312	0.0312	0.2	333	0.31	1	0.5		
37	25	29	0.0312	0.0312	0.2	2	0.31	1	0.5		

Table 4.4 - Original data for the 32-bus system

Again, like 16-bus test system, we use non-dominated sorting considering three objectives: power loss, SAIFI, and minimum voltage. Also, note that the reciprocal of minimum voltage was used instead of minimum voltage itself for optimization. These results were obtained after ten generations and with an initial population of 30 chromosomes. The CPU time was near 8s to achieve the three objectives, and the process was repeated 100 times to ensure every possible optimal solution was found.

Solutions	Topologies	Power loss	SAIFI	1/ (Min. Voltage)	Min. Voltage	Normalized Sum
1(Initial)	33-34-35-36-37	0.2028	3.177	1.095	0.913	3.000
2	7-9-14-32-37	0.1396	3.136	1.066	0.938	0.915
3	7-9-14-28-32	0.1399	3.131	1.063	0.941	0.805
4	7-10-14-32-37	0.1402	3.123	1.066	0.938	0.868
5	7-10-14-28-32	0.1406	3.118	1.063	0.941	0.759
6	7-11-14-32-37	0.1413	3.110	1.065	0.939	0.798
7	7-11-14-36-37	0.1435	3.133	1.064	0.941	0.902
8	7-9-14-17-37	0.1476	3.078	1.064	0.941	0.727
9	7-10-14-17-37	0.1479	3.065	1.066	0.938	0.738
10	7-11-14-17-37	0.1484	3.052	1.064	0.934	0.627
11	7-10-14-16-37	0.1523	3.045	1.066	0.938	0.721
12	7-11-14-16-37	0.1527	3.032	1.064	0.934	0.608
13	9-14-32-33-37	0.1600	3.026	1.066	0.938	0.760
14	10-14-32-33-37	0.1637	2.997	1.064	0.940	0.630
15	11-14-32-33-37	0.1676	2.970	1.066	0.938	0.636
16	11-14-17-33-37	0.1727	2.955	1.065	0.939	0.621
17	11-14-16-33-37	0.1759	2.952	1.064	0.940	0.627
18	11-13-16-33-37	0.1874	2.947	1.066	0.938	0.850

Table 4.5 - 32-bus system results

The revised FNSGA produced a set of 17 non-dominated solutions, which are sorted in Table 4.5 according to least power loss. Solution No. 2 is the optimum in terms of power loss, solution No. 18 is the optimum considering index SAIFI, and solutions No. 3 and No. 5 in the table are optimum when maximizing the minimum voltage is considered as the main objective. Also, if the operators have no preferred objectives, the best one will be selected based on the sum of the normalized values of these objective functions, according to equations 2.7 and 2.8. In this case, solution No. 12, would be considered the best solution among the eighteen solutions in table 4.5. The three-objectives Pareto-optimal solutions for 32-bus system are shown in figure 4.6.

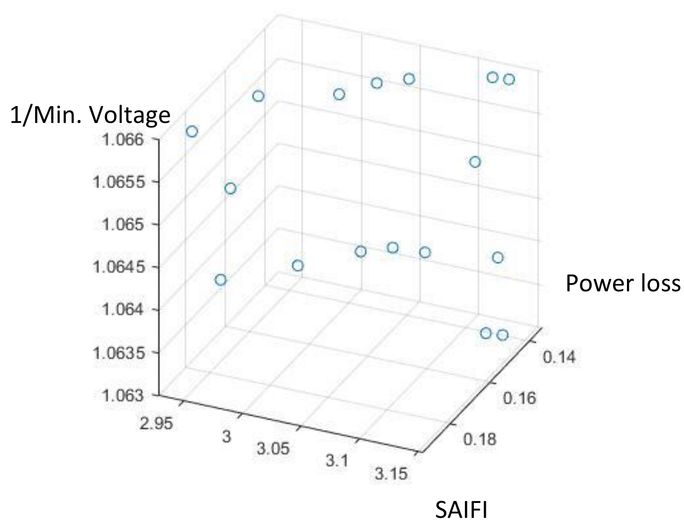

Figure 4.6 - 32-bus Pareto-optimal Solutions shown in three-dimensional coordinate

4.4 Comparison

In order to check whether this algorithm produces the right results on 16-bus and 32-bus system, the objectives were adjusted to make parallel comparison to the results in other reports.

4.4.1 16-bus system

The 16-bus system is used to evaluate algorithm performance in the following report: [13] and [16]. The author in [13] evaluated four cases which considered power loss and four reliability indices, respectively. Meanwhile, the author in [16] considered power loss and minimum voltage along with some other indices that are not used in this thesis. In this thesis, a two-objectives non-dominated optimization with power loss and the most important reliability index SAIFI was used to compare to the results in [13], which is shown in Table 4.6 and 4.7. Another two-objective non-dominated optimization, shown in Table 4.8 and 4.9, with power loss and minimum voltage to compare to the results in [16].

Solutions	Topologies	Power Loss (MW)	SAIFI (f/yr)
1	10-11-19	0.4661	5.6120
2	10-17-19	0.4839	5.4704

Table 4.6 - 16-bus results for comparison **1**

Solutions	Topologies	Power Loss (MW)	SAIFI (f/yr)
1	10-11-19	0.466	5.612
2	10-17-19	0.484	5.470

Table 4.7 - 16-bus results from [13]

Solutions	Topologies	Power Loss (MW)	Minimum Voltage (p.u.)
1	10-11-19	0.4661	5.6120
2	10-17-19	0.4842	5.4704

Table 4.8 - 16-bus results for comparison **2**

Solutions	Topologies	Power Loss (kW)	Minimum Voltage (p.u.)
1	10-11-19	466.13	5.6120
2	10-17-19	483.87	5.4704

Table 4.9 - 16-bus results from [16]

Despite of the differences in the objectives in each paper, the results are exactly the same as other authors results. Thus, we can conclude that the calculation and the programming methods illustrated in this thesis is correct.

4.4.2 32-bus system

The 32-bus system (sometime called 33-bus system) are used in the following reports: [13], [16], and [26]. The author in [13] used power loss and reliability indices, which is the same as he had in 16-bus system. The author in [16] also used the optimization method he used in the 16-bus system, which involved power loss, minimum voltage and some other indices. The author in [26] adopted a three-objectives optimization using power loss, SAIFI and ENS. Therefore, three different non-dominated optimization analysis with different objectives were carried out in order to compare with the above mentioned three papers, respectively.

Solutions	Topologies	Power loss (MW)	SAIFI (f/yr)
1	7-9-14-32-37	0.1396	3.1362
2	7-10-14-32-37	0.1402	3.1234
3	7-11-14-32-37	0.1413	3.1100
4	7-9-14-17-37	0.1476	3.0781
5	7-10-14-17-37	0.1479	3.0648
6	7-11-14-17-37	0.1484	3.0516
7	7-10-14-16-37	0.1523	3.0450
8	7-11-14-16-37	0.1527	3.0322
9	9-14-32-33-37	0.1600	3.0263
10	10-14-32-33-37	0.1637	2.9968
11	11-14-32-33-37	0.1676	2.9700
12	11-14-17-33-37	0.1727	2.9550
13	11-14-16-33-37	0.1759	2.9521
14	11-13-16-33-37	0.1874	2.9469

Table 4.10 - 32-bus results for comparison **1**

Solutions	Topologies	Power loss (MW)	SAIFI (f/yr)
1	7-9-14-32-37	0.1396	3.136
2	7-10-14-32-37	0.1402	3.123
3	7-11-14-32-37	0.1413	3.110
4	7-9-14-17-37	0.1476	3.078
5	7-10-14-17-37	0.1479	3.065
6	7-11-14-17-37	0.1484	3.052
7	7-10-14-16-37	0.1523	3.045
8	7-11-14-16-37	0.1527	3.032
9	9-14-32-33-37	0.1600	3.026
10	10-14-32-33-37	0.1637	2.997
11	11-14-32-33-37	0.1676	2.970
12	11-14-17-33-37	0.1727	2.955
13	11-14-16-33-37	0.1759	2.952
14	11-13-16-33-37	0.1874	2.947

Table 4.11 - 32-bus results from [13]

Solutions	Topologies	Power loss (MW)	Minimum Voltage (p.u.)
1	7-9-14-32-37	0.1396	0.938
2	7-9-14-28-32	0.1399	0.941
3	7-10-14-32-37	0.1406	0.941

Table 4.12 - 32-bus results for comparison **2**

Solutions	Topologies	Power loss (kW)	Minimum Voltage (p.u.)
1	7-9-37-14-32	139.49	0.938
2	7-9-28-14-32	139.89	0.941
3	7-10-28-14-32	140.66	0.941

Table 4.13 - 32-bus results from [16]

Solutions	Topologies	Power loss (MW)	SAIFI (f/yr)	ENS (MWh/yr)
1	7-9-14-32-37	0.1396	3.1362	6.645
2	7-10-14-32-37	0.1402	3.1234	6.616
3	7-11-14-32-37	0.1413	3.1100	6.566
4	7-11-14-36-37	0.1435	3.1121	6.546
5	7-9-14-17-37	0.1476	3.0781	6.478
6	7-10-14-17-37	0.1479	3.0648	6.449
7	7-11-14-17-37	0.1484	3.0516	6.399
8	7-10-14-16-37	0.1523	3.0450	6.401
9	7-11-14-16-37	0.1527	3.0322	6.352
10	9-14-32-33-37	0.1600	3.0263	6.387
11	10-14-32-33-37	0.1637	2.9968	6.366
12	11-14-32-33-37	0.1676	2.9700	6.319
13	10-14-17-33-37	0.1692	2.9810	6.314
14	10-14-16-33-37	0.1724	2.9760	6.301
15	11-14-17-33-37	0.1727	2.9550	6.238
16	11-14-16-33-37	0.1759	2.9521	6.225
17	11-13-16-33-37	0.1874	2.9469	6.235

Table 4.14 - 32-bus results for comparison **3**

Solutions	Topologies	Power loss (kW)	SAIFI (f/yr)	ENS (MWh/C/yr)
1	7-9-14-32-37	139.551	1.1048	0.4422
2	7-9-14-28-32	139.978	1.0327	0.4118
3	7-9-14-28-36	141.916	1.0173	0.4056
4	7-10-14-28-36	142.4292	1.0162	0.4054
5	7-9-14-17-28	146.289	1.0042	0.3998
6	7-10-14-17-28	146.513	1.0031	0.3995
7	9-14-28-33-36	146.666	1.0021	0.3999
8	10-14-28-33-36	148.608	0.9982	0.3991
9	7-9-14-16-28	150.203	1.0003	0.3984
10	7-10-14-16-28	150.248	0.9992	0.3982
11	9-14-17-28-33	150.977	0.9910	0.3952
12	10-14-17-28-33	152.590	0.9871	0.3943
13	10-14-16-28-33	156.100	0.9847	0.3936
14	10-13-16-28-33	161.580	0.9841	0.3936

Table 4.15 - 32-bus results from [26]

The table 4.10 and table 4.11 indicates that there are no differences between the algorithm used in this thesis and the approach used in [13]. When comparing with results in [16], which is shown in table 4.12 and table 4.13, there shows some minor differences between the power loss results. However, the differences did not affect the optimal solutions generated by the algorithm; thus, they still produced the same sets of switches.

Different from the previous two authors, author in [26] used a different set of original data to calculate all the indices, so there exist only few overlaps between his work and mine. Nevertheless, the switches sets that are produced in [26] showed good performance if evaluated in my algorithm, and vice versa. As a result, I can conclude that the results between [26] and this thesis are very similar to each other and the difference lies only in the original data.

Chapter 5

Conclusion and Future Work

Conclusion

The Distribution System Reconfiguration (DSR) problem is defined as the search for the best topology that satisfies the problem objectives and constraints. This is achieved by changing the status of the tie (open) switches and the sectionalized (closed) switches in the system.

In normal operation, the objectives considered in this thesis were the minimization of the active power losses, the minimization of commonly used reliability indices (SAIFI, SAIDI, CAIDI and ENS), and the minimum voltage. The FNSGA algorithm that is used and adapted in this thesis solves the DSR problem by satisfying all the objectives simultaneously within a shorter computation time compared to other techniques like NSGA, or NSGA-II.

The first item in this thesis is a clear explanation of the calculation of the most commonly used reliability indices, and a description of the method of programming DSR problems. Second, with the help of FNSGA, the program can now automatically generate all the valid switches sets and determine the non-dominated sets. Then, the operator can choose from the solution sets and decide which solution best suits their need.

Next, the algorithm was tested in a 5-bus sample system, and the most widely studied test systems (16-bus and 32-bus). Results show good performance of this algorithm when applied to the different systems to generate sets of optimal solutions.

Compared to other recently published similar paper, the adjusted FNSGA showed good performance in archiving all objectives with relatively small population size and number of generations.

Future Work

The following issues may be resolved in the future as an extension of the current research.

1. Reliability Indices

In this thesis, only one reliability index (SAIFI mostly) is taken into the multi-objective optimal calculation with two other indices. Moreover, we cannot simply consider more than one reliability index in one optimization run. However, if we want to take more than one index into consideration, one way is to find a weight for every reliability index. In that way, we can calculate a weighted total before we use that total as one objective for the optimization process. Moreover, all the indices values can be normalized to a 0-1 scale to better illustrate the comparison.

2. Further Applications

The algorithm in this thesis has only been applied to relatively small systems (16-bus and 32-bus). As an extension of this work, the algorithm need to be tested on larger systems starting with the 69-bus system. To do this, further adjustment of the algorithm may be needed due to the increasing complexity of the system.

3. Algorithm

The FNSGA that is used and adapted in this thesis is very powerful, but it does many things that are not necessary for this project. The adjusted algorithm consumes more than three times the calculation time in the 16-bus system compared to the original algorithm. In order for the simulation to work better, a new program needs to be developed. The new code should contain only the part of the power flow used for loss calculation and have the calculation for all the indices in only one cycle.

References

[1] R. Billinton, R. N. Allan, Reliability Evaluation of Power Systems, 2nd ed. New York: Plenum, 1996.

[2] H. L. Willis, Power Distribution Planning Reference Book, 2nd ed. New York: Mercel Dekkar, 2004.

[3] T. A. Short, Electric Power Distribution Handbook. BocaRaton, FL, USA: CRC Press, 2004.

[4] B. Amanulla, S. Chakrabarti and S. N. Singh, “Reconfiguration of power distribution systems considering reliability and power loss” IEEE Trans. Power Deliv., vol. 27, no. 2, pp.918 – 926, April 2012.

[5] Merlin, H. Back, “Search for a minimal-loss operating spanning tree configuration in an urban power distribution system” In Proceedings of the 1975 Fifth Power Systems Computer Conference (PSCC), Cambridge, UK, 1–5, pp. 1–18, September 1975.

[6] S. Civanlar, J. J. Grainger, H. Yin and S. S. H. Lee, “Distribution feeder reconfiguration for loss reduction” IEEE Trans. Power Deliv., vol. 3, no.3, pp. 1217–1223, 1988.

[7] M. E. Baran, F. F. Wu, “Network reconfiguration in distribution systems for loss reduction and load balancing” IEEE Trans. Power Deliv., vol. 4, no.2, pp. 1401–1407, 1989.

[8] D. Shirmohammadi, H. W. Hong, “Reconfiguration of electric distribution networks for resistive line losses reduction” IEEE Trans. Power Syst., vol. 4, no. 2, pp. 1492–1498, Apr. 1989.

[9] K. Nara, A. Shiose, M. Kitagawa and T. Ishihara, “Implementation of genetic algorithm for distribution systems loss minimum re-configuration” IEEE Trans. Power Systems, vol. 7, no.3, pp. 1044–1051, 1992.

[10] Y. Hsu, Y. Jwo-Hwu, “Planning of distribution feeder reconfiguration with protective device coordination” IEEE Trans. Power Deliv., vol. 8, No. 3, pp. 1340-1347, July 1993.

[11] A. Makinen, J. Partanen and E. Lakervi," A Practical Approach for Estimating Future Outage Costs in Power Distribution Networks" IEEE Trans. on Power Del., vol. 5, no. 1, pp. 287-294, Jan 1990.

[12] W. Li, P. Wang, Z. Li, and Y. Liu, “Reliability evaluation of complex radial distribution systems considering restoration sequence and network constraints” IEEE Trans. Power Del., vol. 19, no. 2, pp. 753–758, Apr. 2004.

[13] J. E. Mendoza, M. E. Lopez, and E. A. Lopez, “Micro-genetic multi-objective reconfiguration algorithm considering power losses and reliability indices for medium voltage distribution network” IET IET Gener. Transm. Distrib., vol. 3, iss. 9, pp. 825-840, Sep. 2009.

[14] S. Elsaiah, M. Benidris and J. Mitra, “Reliability improvement of power distribution system through feeder reconfiguration” IEEE Probabilistic Methods Applied to Power Systems (PMAPS) international Conf., July 2014.

[15] J. López, M. Lavorato, J. F. Franco, M. J. Rider, "Robust optimisation applied to the reconfiguration of distribution systems with reliability constraints" IET Gener. Transm. Distrib., vol. 10, iss. 4, pp. 917-927, 2016.

[16] M. Eldurssi, R. M. O'Connell, "A Fast non-dominated sorting guided genetic algorithm for multi-objective power distribution system reconfiguration problem" IEEE Trans. Power Syst., vol. 30, no. 2, pp. 593-601, March 2015

[17] Roytelman, V. Melnik, S.S.H. Lee and R.L. Lugtu, "Multi-objective feeder reconfiguration by distribution management system" IEEE Trans. Power Syst., vol. 11, pp. 661–667, 1996.

[18] M. Tsai, F. Hsu, "Application of grey correlation analysis in evolutionary programming for distribution system feeder reconfiguration" IEEE Trans. Power Syst., vol. 25, no. 2, 1126–1133, 2010.

[19] A. C. Santos, A. C. B. Delbem, J. B. A. London and N. G. Bretas, "Node-depth encoding and multiobjective evolutionary algorithm applied to large-scale distribution system reconfiguration" IEEE Trans. Power Syst., vol. 25, no. 2, pp. 1254–1265, 2010.

[20] M. A. N. Guimaraes, C. A. Castro and R. Romero, "Distribution systems operation optimisation through reconfiguration and capacitor allocation by a dedicated genetic algorithm" IET Gener. Transm. Distrib. vol. 4, iss.11, pp. 1213–1222, 2010.

[21] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II" IEEE Trans. Evolutionary Computation, vol. 6, no. 2, April 2002.

[22] Y. Kumar, B. Das, and J. Sharma, "Multiobjective, multiconstraint service restoration of electric power distribution system with priority customers" IEEE Trans. Power Deliv., vol. 23, no. 1, pp. 261-270, Jan. 2008.

[23] H. Xianchao, G. A. Taylor, "Service restoration of distribution systems based on NSGA-II" in Proc. 45th Int. Universities Power Engineering Conf. (UPEC), pp. 1-6, Aug. 2010.

[24] IEEE Guide for Electric Power Distribution Reliability Indices, IEEE Power & Energy Society, IEEE Standard-1366, May 2012.

[25] D. Murali, A. Sekhar, "Capacitance based reliability indices of a real time radial distribution feeder" IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE), Vol. 7, Iss. 2, pp. 10-21, July 2013.

[26] N. Paterakis, A. Mazza, etc. "Multi-objective reconfiguration of radial distribution systems using reliability indices" IEEE Trans. Power Syst., vol. 31, no. 2, pp. 1048-1062, March 2016.

[27] K. Nara, Y. Mishima and A. Gjyo, T. Ito, and H. Kaneda, "Loss minimum reconfiguration of distribution system by tabu search" in Proc. 2002 IEEE Conf. on Transmission and Distribution, Asia, pp. 232-236, 2002.

[28] M. A. N. Guimaraes, J. F. C. Loernzeti, and C. A. Castro, "Reconfiguration of distribution systems for voltage stability margin enhancement using tabu search" in Proc. 2004 Int. Conf. Power System Tech. (POWERCON), Singapore, vol. 2, pp. 1556-1561, 2004.

[29] H. D. Chiang, J. J Rene, "Optimal network reconfiguration in distribution system, part1: a new formulation and solution methodology", IEEE Trans. Power Deliv., vol. 5, no. 4, pp. 1902-1908, Nov. 1990.

[30] H. D. Chiang, J. J. Rene, "Optimal network reconfiguration in distribution system, part 2: solution algorithms and numerical results" IEEE Trans. Power Deliv., vol. 5, no. 3, pp. 1568-1574, Jul. 1990.

[31] Skoonpong, S. Sirisumrannukul, "Network reconfiguration for reliability worth enhancement in distribution systems by simulated annealing" in Proc. 2008 IEEE Int. Conf. Electronics, Computer Telecommunications, and Information, pp. 937-940, 2008.

[32] S. J. Russell, P. Norvig, Artificial Intelligence: A Modern Approach 3rd edition, Upper Saddle River, New Jersey: Prentice Hall 2009.

[33] D. Bouchard, A. Chikhani, V. I. John, and M. M. A. Salama, "Applications of hopfield neural networks to distribution feeder reconfiguration" in Proc. 2nd Int. Conf. Application of Neural Networks to Power Systems, 1993, pp. 311-316.

[34] M. A. Kashem, G. B. Jasmon, A. Mohamed, and M. Moghavvemi, "Artificial neural network approach to network reconfiguration for loss minimization in distribution networks" Int. J. Elect. Power Energy Syst., vol. 20, no. 4, pp. 247–258, May 1998.

[35] V. V. K. Reddy, M. Sydulu, "A heuristic-expert based approach for reconfiguration of distribution systems" in Proc. 2007 IEEE Power Engineering Society General Meeting, pp. 1-4, 2007.

[36] K. Butler-Purry, S. K. Srivastava, "Expert system-based reconfiguration of shipboard power distribution systems" in Proc. 2008 IEEE Power and Energy Society General Meeting, pp.1-5, 2008

[37] Y. T. Hsiao, "Multi-objective evolution programming method for feeder reconfiguration" IEEE Trans. Power Syst., vol. 19, no. 1, pp. 594-599, Feb. 2004.

[38] K. Deb, Multi-objective Optimization using E volutionary Algorithms. West Sussex: Wiley, 2002.

[39] E. G. Carrano, L. A. E. Soares,Electric, and etc. "Distribution network multi-objective design using a problem-specific genetic algorithm" IEEE Trans. Power Deliv., vol. 21, no. 2, pp. 995-1005, April 2006.

[40] J. Moshtagh, S. Ghasemi "Optimal distribution system reconfiguration using non-dominated sorting genetic algorithm (NSGA-II)" Journal of Operation and Automation in Power Engineering, vol. 1, no. 1, pp. 12-21, March 2013.

[41] E. López, J. Campos, C.Tardon, F. Salgado, J. Tardon, Rodrigo López "Solution to the reliability problem of radial electric distribution systems through artificial intelligence" International Journal of Computers, Communications & Control, vol. I, pp. 61-71, 2006.

[42] K. Miettinen, Nonlinear multi-objective optimization, Norwell, MA: Kluwer Academic Publishers, 1999.

Appendix 1: Program Code for 5-bus System Reliability Analysis

```
clc;clear all;

data5bus;

i =input('   line to break(choose from 1 to 4) =  ');  %obtain the modified
topology:

T(line(i,2),line(i,3))=0;

T(line(i,3),line(i,2))=0;

%----------------main program-----------------

for k=1:5  %find f of node2 to  node5

  %---------------find path1----------------

  h=2; i=2; fp=k;pr = 0; sr = 0; %"i"- count for primary route;

  pr(1)= fp; lp = fp;            %"h"- count for secondary route;

  path1out = false;              %"fp"- former point;%"lp"- current load point;

  while path1out == false        %"pr"- primary route;

      count = 0;                 %"sr"- secondary route;

      for j =1:5

          if T(lp, j)>0 && j~=fp      % count the number of paths

              count=count+1;

          end

      end

      if count ==1               % if path number = 1

          flag = 0;

          for j =1:5

           if T(lp, j)>0  && j~=fp

               flag =1; fp=lp; lp=j;

               pr(i)=lp; i=i+1;

           end

           if flag==1

               break

           end

          end

      end

      if count ==2               % if path number = 2

              flag1 = 0;

              for j =1:5
```

```
                if T(lp, j)>0 && j~=fp
                    flag1= flag1+1;
                    if flag1==1           %select first available point into
path1
                        pr(i)=j;i=i+1;
                    end
                    if flag1==2
                        sr(1)=lp;sr(h)=j; %select second availble point into
path2,
                        fp=lp; lp=pr(i-1);% and store as the begining of branch
                    end
                end
                if flag1==2
                    break
                end
               end
      end
      if count ==0
          path1out = true;
      end
  end
%----------------find path2------------------
if sr~=0
 fp2 = sr(1);lp2 = sr(2);    %"fp2"- former point;
 path2out=false;             %"lp2"- current load point;
 while path2out ==false
    count2 = 0;flag2 = 0;
    for j =1:5
        if T(lp2, j)>0 && j~=fp2
            count2 =count2+1;                 % count the number of paths,
            flag2 =1; fp2=lp2; lp2=j;         % as in 5bus system there won't
            sr(h+1)=lp2; h=h+1;               % be two braches in path2, so
        end                                   % count2 can only be 1 or 0
        if flag2==1
            break
        end
```

```
    end
    if count2==0
       path2out=true;
    end
 end
end
disp(pr);
disp(sr);
lpr= length(pr); lsr= length(sr); % length of each path
%--------------unavailability matrix----------------
%unavailable time matrix
UT=zeros(5,5);
%------------for primary route-------------
check=0;   %determine where is the nearest sectionalizer;
for n=1:lpr-1       %check = 1 - the nearest sectionalizer finded;
  if check==0       %chech = 0 - not yet;
    if n == 1
        if T(pr(n),pr(n+1))==2     % 2 - sectionalizer, 0.5h, otherwise 1h;
             UT(pr(n),pr(n+1))=0.5;  %analyze unavailble time for the
        else UT(pr(n),pr(n+1))=1;    %nearest line;
        end
    else if T(pr(n),pr(n+1))==2 || T(pr(n),pr(n-1))==2
            UT(pr(n),pr(n+1))=0.5;   %analyze unavailble time for other
        else UT(pr(n),pr(n+1))=1;    %lines;
        end
    end
    if UT(pr(n),pr(n+1))== 0.5
        check =1;
    end
  end
  if check==1
     UT(pr(n),pr(n+1))=0.5;
  end
end
disp('UT');disp(UT);
%-----------for secondary route------------
```

```
if sr~=0
   %step1 find the complete secondary route
  sro=0;check2=0; %sro= routes before the path2;
  for n = 1:lpr
    if pr(n)~= sr(1)
        sro(n) = pr(n);
    else check2=1;
    end
    if check2 == 1
        break
    end
  end
  if sro~=0
     src=[sro,sr];
  else src=sr;
  end
  lsrc=length(src); %src- complete secondary route;
  disp('src');disp(src);
  %step2 analyze unavalable time for secondary route
  check3=0;   %determine where is the nearest sectionalizer;

  for n=1:lsrc-1       %check = 1 - the nearest sectionalizer finded;
   if check3==0       %chech = 0 - not yet;
    if n == 1
        if T(src(n),src(n+1))==2
             UT(src(n),src(n+1))=0.5;  %analyze unavailble time for the
        else UT(src(n),src(n+1))=1;    %nearest line
        end
    else if T(src(n),src(n+1))==2 || T(src(n),src(n-1))==2
            UT(src(n),src(n+1))=0.5;   %analyze unavailble time for other
        else UT(src(n),src(n+1))=1;
        end
    end
    if UT(src(n),src(n+1))== 0.5
        check3 =1;
    end
```

```
    end
    if check3==1
      UT(src(n),src(n+1))=0.5;
    end
  end
end
disp(UT);
% --------------calculate f_rate----------------
D=zeros(5,5);         %initialize for position counting
for n=1:lpr-1;
    D(pr(n),pr(n+1))=1;
end                             % marking related position
for n=1:lsr-1;
    D(sr(n),sr(n+1))=1;
end
D=D.*frate;  %modified failure rate matrix
fr=sum(D);
fr=sum(fr);
F_rate(k)=fr;
% --------------unavailability-----------------
UT=UT.*D;
un=sum(UT);
un=sum(un);
UN(k)=un;
end
%-----------------result--------------------
disp('unavailability');disp(UN);
SAIFI = F_rate*load'/total;
disp('SAIFI');disp(SAIFI);
SAIUI = sum(UN*load')/total;
disp('SAIUI');disp(SAIUI);
ENS = sum(UN*MW');
disp('ENS');disp(ENS);
SAIDI = SAIUI/SAIFI;
disp('SAIDI');disp(SAIDI);
```

Appendix 2: Program Code for 16&32 bus System Reliability Analysis

```
clc;clear all;

bus16data;

%obtain the modified topology:

disp('choose three different lines to break following the
instruction:')

i1 =input('line to break from FL1(choose from 4,5,8,9,11,17) = ');
%line1

T(line(i1,2),line(i1,3))=0;

T(line(i1,3),line(i1,2))=0;

if i1 ~= 5             % debugging for two adjunct tie switches(line 5)

    if i1 == 4          % if l4 is cut off, the route should be able to
pass from

       T(5,6)=1;        % node 5 to 6, but not 6 to 5, otherwise vice
versa.

    else T(6,5)=1;

    end

end

i2 =input('line to break from FL2(choose from 8,10,14,18) = '); %line2

T(line(i2,2),line(i2,3))=0;

T(line(i2,3),line(i2,2))=0;

i3 =input('line to break from FL3(choose from 4,6,7,15,16,19) = ');
%line3

T(line(i3,2),line(i3,3))=0;

T(line(i3,3),line(i3,2))=0;

if i3 == 6

    T(7,8)=1;

end

if i3 == 4 && i1 == (9||11||17)

   T(6,5)=1;

end

if i1 == 5 && i3 == 4 %bugs

    T(5,7)=1;

    T(7,8)=1;
```

```
end
%----------------main program-----------------
for k=5:17  %find f of node5 to  node17
 %---------------find path1----------------
  i=2; h=2; g=2;fp=k;             % "g"- count for third route
  pr = 0; sr = 0; tr = 0;         % "i"- count for primary route;
  pr(1)= fp; lp = fp;             % "h"- count for secondary route;
  times= 0;                       %"times"- the times that count==2;
  path1out = false;               % "fp"- former point;% "lp"- current
load point;
  while path1out == false         % "pr"- primary route;
      count = 0;                  % "sr"- secondary route;
      for j =1:17                  % "tr"- third route
          if T(lp, j)>0 && j~=fp
              count=count+1;      % count the number of paths
          end
      end
      if count ==1               % if path number = 1
          flag = 0;
          for j =1:17
           if T(lp, j)>0 && j~=fp
               flag =1; fp=lp; lp=j;
               pr(i)=lp; i=i+1;
           end
           if flag==1
               break
           end
          end
      end
      if count ==2               % if path number = 2
              times= times+1;
              flag1 = 0;
              for j =1:17
               if T(lp, j)>0 && j~=fp
```

```
                    flag1= flag1+1;
                    if flag1==1           %select first available point 
into path1
                        pr(i)=j;i=i+1;
                    end
                    if flag1==2 && times ==1
                        sr(1)=lp;sr(h)=j;  %select second availble point 
into path2,
                        h=h+1;
                        fp=lp; lp=pr(i-1); % and store as the begining 
of branch
                    end
                    if flag1==2 && times ==2
                        tr(1)=lp;tr(g)=j;  %select second availble point 
into path3,
                        g=g+1;
                        fp=lp; lp=pr(i-1); % and store as the begining 
of branch
                    end
                end
                if flag1==2
                    break
                end
            end
      end
      if count ==3
            flag1_2 =0;
            for j =1:17
             if T(lp, j)>0 && j~=fp
                    flag1_2= flag1_2+1;
                    if flag1_2==1            %select first available 
point into path1
                        pr(i)=j;i=i+1;
                    end
                    if flag1_2==2
```

```
                       sr(1)=lp;sr(h)=j; %select second availble point
into path2,
                       h=h+1;            % and store as the begining of
branch
                   end
                   if flag1_2==3
                       tr(1)=lp;tr(g)=j;
                       g=g+1;            %select third availble point
into path3,
                       fp=lp;lp=pr(i-1); % and store as the begining of
branch
                   end
               end
               if flag1_2==3
                   break
               end
              end
      end
      if count ==0
          path1out = true;
      end
  end
%----------------find path2------------------
if sr~=0
 fp2 = sr(1);lp2 = sr(2);   %"fp2"- former point;
 path2out=false;            %"lp2"- current load point;
 while path2out ==false
    count2 = 0;flag2 = 0;
    for j =1:17
        if T(lp2, j)>0 && j~=fp2
            count2 =count2+1;              % count the number of
paths,
            flag2 =1; fp2=lp2; lp2=j;      % as in 16bus system there
won't
            sr(h)=lp2; h=h+1;              % be two braches in path2,
so
```

```
        end                                 % count2 can only be 1 or 0
        if flag2==1
            break
        end
    end
    if count2==0
       path2out=true;
    end
 end
end
%----------------find path3------------------
if tr~=0
 fp3 = tr(1);lp3 = tr(2);    %"fp3"- former point;
 path3out=false;             %"lp3"- current load point;
 while path3out ==false
    count3 = 0;flag3 = 0;
    for j =1:17
        if T(lp3, j)>0 && j~=fp3
            count3 =count3+1;                % count the number of
paths,
            flag3 =1; fp3=lp3; lp3=j;        % as in 16bus system there
won't
            tr(g)=lp3; g=g+1;                % be two braches in path3,
so
        end                                 % count2 can only be 1 or 0
        if flag3==1
            break
        end
    end
    if count3==0
       path3out=true;
    end
 end
end
lpr= length(pr); lsr= length(sr);ltr= length(tr);% length of each path
```

```
%--------------unavailability matrix----------------
%unavailable time matrix
UT=zeros(17,17); %ri matrix
%------------for primary route-------------
check=0;   %determine where is the nearest sectionalizer;
for n=1:lpr-1        %check = 1 - the nearest sectionalizer finded;
  if check==0        %check = 0 - not yet;
    if n == 1
        if T(pr(n),pr(n+1))==2     % 2 - sectionalizer, 0.5h, otherwise
1h;
             UT(pr(n),pr(n+1))=0.5;  %analyze unavailble time for the
        else UT(pr(n),pr(n+1))=1;    %nearest line;
        end
    else if T(pr(n),pr(n+1))==2 || T(pr(n),pr(n-1))==2
            UT(pr(n),pr(n+1))=0.5;   %analyze unavailble time for other
        else UT(pr(n),pr(n+1))=1;    %lines;
        end
    end
    if UT(pr(n),pr(n+1))== 0.5
        check =1;
    end
  end
  if check==1
     UT(pr(n),pr(n+1))=0.5;
  end
end
%-----------for secondary route------------
if sr~=0
   %step1 find the complete secondary route
  sro=0;check2=0; %sro= routes before the path2;
  for n = 1:lpr
    if pr(n)~= sr(1)
        sro(n) = pr(n);
    else check2=1;
```

```
    end
    if check2 == 1
        break
    end
  end
  if sro~=0
     src=[sro,sr];
  else src=sr;
  end
  lsrc=length(src); %src- complete secondary route;
  %step2 analyze unavailable time for secondary route
  check2_2=0;   %determine where is the nearest sectionalizer;
  for n=1:lsrc-1       %check = 1 - the nearest sectionalizer finded;
   if check2_2==0       %chech = 0 - not yet;
    if n == 1
        if T(src(n),src(n+1))==2
             UT(src(n),src(n+1))=0.5;  %analyze unavailble time for the
        else UT(src(n),src(n+1))=1;    %nearest line
        end
    else if T(src(n),src(n+1))==2 || T(src(n),src(n-1))==2
            UT(src(n),src(n+1))=0.5;   %analyze unavailble time for
other
        else UT(src(n),src(n+1))=1;
        end
    end
    if UT(src(n),src(n+1))== 0.5
        check2_2 =1;
    end
   end
   if check2_2==1
     UT(src(n),src(n+1))=0.5;
   end
  end
end
```

```
%-----------for third route------------
if tr~=0
   %step1 find the complete third route
  tro=0;check3=0; %tro= routes before the path3;
  for n = 1:ltr
    if pr(n)~= tr(1)
        tro(n) = pr(n);
    else check3=1;
    end
    if check3 == 1
        break
    end
  end
  if tro~=0
     trc=[tro,tr];
  else trc=tr;
  end
  ltrc=length(trc); %trc- complete third route;
  %step2 analyze unavailable time for third route
  check3_2=0;   %determine where is the nearest sectionalizer;
  for n=1:ltrc-1       %check = 1 - the nearest sectionalizer finded;
   if check3_2==0       %chech = 0 - not yet;
    if n == 1
        if T(trc(n),trc(n+1))==2
             UT(trc(n),trc(n+1))=0.5;  %analyze unavailble time for the
        else UT(trc(n),trc(n+1))=1;    %nearest line
        end
    else if T(trc(n),trc(n+1))==2 || T(trc(n),trc(n-1))==2
            UT(trc(n),trc(n+1))=0.5;   %analyze unavailble time for
other
        else UT(trc(n),trc(n+1))=1;
        end
    end
    if UT(trc(n),trc(n+1))== 0.5
```

```
        check3_2 =1;
    end
   end
   if check3_2==1
     UT(trc(n),trc(n+1))=0.5;
   end
  end
end
disp(UT);
% --------------calculate f_rate----------------
D=zeros(17,17);         %initialize for position counting
lpr= length(pr); lsr= length(sr);
ltr= length(tr);                % length of each path
for n=1:lpr-1;
    D(pr(n),pr(n+1))=1;
end                             % marking related position
for n=1:lsr-1;
    D(sr(n),sr(n+1))=1;
end
for n=1:ltr-1;
    D(tr(n),tr(n+1))=1;
end
D=D.*f;  %modified failure rate matrix
fr=sum(D);
fr=sum(fr);
F(k)=fr;
% --------------unavailability----------------
UT=UT.*D;
un=sum(UT);
un=sum(un);
UN(k)=un;
end
SAIFI = F*load'/total;
```

```
disp('SAIFI');disp(SAIFI);
disp('unavailability');disp(UN);
SAIDI = sum(UN*load')/total;
disp('SAIDI');disp(SAIDI);
CAIDI = SAIDI/SAIFI;
disp('CAIDI');disp(CAIDI);
ENS = sum(UN*MW');
disp('ENS');disp(ENS);
data = [SAIFI, SAIDI, CAIDI, ENS];
```

Appendix 3: Data for 5-bus System Reliability Analysis

```
%-----------------data------------------
%node   j  1   2   3   4   5
IM =     [0  -1  -1   0   0    % 1 - line;
           1   0   0   2   0    % 2 - sectionalizing switch
           1   0   0   2   0    % -1- tie switch
           0   2   1   0   -1
           0   0   0   1   0];

frate =  [0  0.2 0.3 0   0  %this matrix presents the failure rate of
each line
          0.2  0   0  0.2  0
          0.3  0   0  0.2  0
           0  0.2 0.2  0   0.1
           0   0   0  0.1  0];

line= [ 1 1 2; %line info
         2 1 3;  %column 1: line number;
         3 2 4;  %column 2: start bus; column 3: end bus
         4 3 4;
         5 4 5];

MW= [0 3.4 10 6.7 10]; %real power
```

```
MVAr= [0 1.4 4.0 2.7 4.0]; %reactive power

load = [ 0 3.68 10.77 7.22 10.77];  %apparent power

total= sum(load);
```

Appendix 4: Data for 16-bus System* Reliability Analysis

* programming data for 32-bus system is similar to 16-bus system using the data provided in chapter four, did not listed due to formatting issues

```
%--------------16 bus data-----------------
line=[1 1 2 0;
      2 1 3 0;
      3 1 4 0;
      4 2 5 3.5;
      5 5 6 3;
      6 5 7 1.5;  %line info
      7 7 8 3.5;  %column 1: line number;
      8 3 9 1.1;  %column 2: start bus; column 3: end bus
      9 9 10 2.8; %column 4: failure rate
     10 9 11 1.1;
     11 10 12 0.8;
     12 10 13 2;
     13 4 14 0.5;
     14 14 15 1;
     15 14 16 1.5;
     16 16 17 4.4;
     17 6 12 4;
     18 11 15 5;
     19 8 17 1];

%complete topology:1 - line; 2 - sectionalizer; -1 - tie switch(fuse);
% n 1  2  3   4   5  6  7   8  9  10 11 12 13 14 15 16 17
```

```
T =[0,-1,-1,-1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; %1
     1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; %2
     1, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 0; %3
     1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0; %4
     0, 2, 0, 0, 0,-1,-1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; %5
     0, 0, 0, 0,-1, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0; %6
     0, 0, 0, 0, 1, 0, 0,-1, 0, 0, 0, 0, 0, 0, 0, 0, 0; %7
     0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2; %8
     0, 0, 2, 0, 0, 0, 0, 0, 0, 2, 2, 0, 0, 0, 0, 0, 0; %9
     0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 2,-1, 0, 0, 0, 0; %10
     0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 2, 0, 0; %11
     0, 0, 0, 0, 0, 2, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0; %12
     0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0; %13
     0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,-1,-1, 0; %14
     0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 1, 0, 0, 0; %15
     0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 2; %16
    0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 0, 0, 1, 0;];%17

% falure rate matrix
f = [0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0;
      0,   0,   0,   0,3.5,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0;
      0,   0,   0,   0,   0,   0,   0,   0,1.1,   0,   0,   0,   0,   0,   0,   0,   0;
      0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,0.5,   0,   0,   0;
      0,3.5,   0,   0,   0,3.0,1.5,   0,   0,   0,   0,    0,   0,   0,   0,   0,
0;
      0,   0,   0,   0,3.0,   0,   0,   0,   0,   0,   0,4.0,   0,   0,   0,   0,   0;
      0,   0,   0,   0,1.5,   0,   0,3.5,   0,   0,   0,   0,   0,   0,   0,   0,   0;
      0,   0,   0,   0,   0,   0,3.5,   0,   0,   0,   0,   0,   0,   0,   0,   0,1.0;
      0,   0,1.1,   0,   0,   0,   0,   0,   0,2.8,1.1,   0,   0,   0,   0,   0,   0;
      0,   0,   0,   0,   0,   0,   0,   0,2.8,   0,   0,0.8,2.0,   0,   0,   0,   0;
      0,   0,   0,   0,   0,   0,   0,   0,1.1,   0,   0,   0,   0,   0,5.0,   0,   0;
      0,   0,   0,   0,   0,4.0,   0,   0,   0,0.8,   0,   0,   0,   0,   0,   0,   0;
      0,   0,   0,   0,   0,   0,   0,   0,   0,2.0,   0,   0,   0,   0,   0,   0,   0;
      0,   0,   0,0.5,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,1.0,1.5,   0;
```

```
     0,   0,   0,   0,   0,   0,   0,   0,   0,   0,5.0,   0,   0,1.0,   0,   0,   0;
     0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,1.5,   0,   0,4.4;
    0,   0,   0,   0,   0,   0,   0,1.0,   0,   0,   0,   0,   0,   0,   0,4.4,
0;];


%%Power Matrix

MW =[0 0 0 0 2 3 2 1.5 4 5 1 0.6 4.5 1 1 1 2.1];


Apparent Power =[ 0, 0, 0, 0, 2.561, 3.027, 2.040, 1.921, 4.826, 5.314,
1.345, 0.781, 4.810,1.345,1.487,1.345,2.247];


total= sum(Apparent Power);

F=zeros(1,17); %failure rate for each node
```