

Master of Science Thesis

Towards Real-Time Power Restoration Using a Hybrid Genetic Algorithm

Allan Kwashigah Dziwornu

Network Architectures and Services (NAS) Group
Department of Intelligent Systems
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Copyright

All rights reserved. No section of the material protected by this copyright may be reproduced or utilised in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without permission from the author, Delft University of Technology.

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF EMBEDDED SYSTEMS

The undersigned hereby certify that they have read and recommend to
the Faculty of Electrical Engineering, Mathematics and Computer Science for
acceptance a thesis entitled
Towards Real-Time Power Restoration Using a Hybrid Genetic Algorithm
by
Allan Kwashigah Dziwornu
in partial fulfilment of the requirements for the degree of
MASTER OF SCIENCE.

Dated: 10th July 2015

Committee

Chairperson:	Rob Kooij, EWI, TUDelft
Supervisor:	Martijn Warnier, TPM, TUDelft
Member:	Michel Oey, TPM, TUDelft
Member:	Yakup Koç, TPM, TUDelft

Acknowledgements

First of all I would like to thank the Almighty God for seeing me through this project till its completion. It took quite a while to get it all done but I finally made it.

My second thanks goes to Prof Rob Kooij.

The next thanks goes to my daily supervisor, Dr Martijn Warnier.

Special thanks to my biweekly supervisor, Yakup Koç.

The warmest one goes to my parents, brothers, and sister who have always supported me during this time. It is an honour to be born in such a great family.

Abstract

In this thesis a Hybrid Genetic Algorithm has been proposed for the restoration of power in a power grid after a blackout has occurred. The algorithm uses a weighting factor to give importance to the objective functions used in modelling the problem. Minimisation of the out-of-service area of the grid, minimisation of the number of switches (switching operations), and minimisation of the amount of losses were the three main objective functions considered in this project. The hybridisation of conventional GA was done by using a local search algorithm. The algorithm was tested on single-fault and multi-fault blackout systems. Based on the simulation results, the performance of the hybrid GA has been found to be better than the conventional GA when restoring power to a radial power grid. The results were validated for three different IEEE networks namely IEEE14, IEEE30, and remodelled IEEE118, which are benchmark networks used in the power restoration field.

Table of Contents

Abstract.....	3
Chapter 1.....	7
Introduction.....	7
1.1 Electricity Grid Background.....	7
1.2 Motivation for the Research.....	8
1.3 Research Question.....	10
1.4 Objective of the Project.....	10
1.5 Scope of This Project.....	11
1.6 Outline of This Thesis.....	11
Chapter 2.....	12
Introduction.....	12
2.1 Power Restoration.....	12
2.2 Power System Operating State.....	13
2.3 Overview of Blackout.....	14
2.4 Overview of Blackout Restoration.....	19
2.5 Categorisation of Power System Restoration.....	21
2.5.1 Subsystem Restoration.....	21
2.5.2 Manual Restoration.....	21
2.5.3 Computer-Aided Restoration.....	22
2.5.4 Cooperative Restoration.....	22
2.5.5 Fully Automated Restoration.....	22
2.5.6 Algorithmic Restoration.....	22
Chapter 3.....	24
The Genetic Algorithm (GA).....	24
3.1 Literature Review on Genetic Algorithms.....	24
3.2 Basics of Genetic Algorithms.....	28
3.3 Different Types of Operators Used in Genetic Algorithms.....	29
3.3.1 Selection/Reproduction.....	29
3.3.1.1 Tournament Selection.....	30
3.3.2 Crossover.....	31
3.3.3 Mutation.....	32
3.3.4 Fitness Evaluation.....	32
	4

3.3.5 Control Parameters of Genetic Algorithms	33
3.3.6 Stopping Criteria for Genetic Algorithms	34
3.4 Application of Genetic Algorithms	34
3.4.1 General Uses of GAs	34
3.4.2 GA Applications for Power Restoration.....	35
3.5 Advantages of GAs	35
3.6 Disadvantages of GAs.....	36
Chapter 4.....	38
Power Flow Problem.....	38
4.1 Power Flow Analysis	38
4.2 Classification of Buses.....	39
4.3 Basic Power Flow Problem.....	40
4.4 One Method of Solving the Load Flow Problem	42
4.4.1 Using AC Power Flow (PF) Analysis.....	42
4.5 Overview of MATPOWER.....	43
Chapter 5.....	45
Power System Modelling	45
5.1 Modelling of Blackout (Encoding)	45
5.2 Modelling of Blackout Restoration.....	49
5.3 Power Restoration Using a Genetic Algorithm.....	50
5.3.1 Power Restoration in MATLAB Using a GA	50
5.3.2 Selection of Parents, Crossover, and New Mating Pool.....	51
5.3.3 Mutation.....	54
5.4 Evaluation of the Fitness Function.....	54
5.5 Weight Values	55
5.6 Data Normalisation of the Objective Function	55
5.7 Stopping Criteria	56
Chapter 6.....	58
Hybridisation of the Proposed GA with Local Search.....	58
6.1 Formulation of the Proposed Local Search.....	58
6.2 Addition of Radiality Check	62
6.2.1 Implementation.....	64
Chapter 7.....	67

Simulation Results and Discussion	67
7.1 Parameters for Fine Tuning the Hybrid GA.....	70
7.1.1 Rules for Parameter Selection in This Project.....	71
7.2 Assessment of the Proposed Hybrid Genetic Algorithm	72
7.2.1 Steps in Testing the Hybrid Genetic Algorithm	73
7.3 Two Types of Blackout Modelling	74
7.3.1 How a Single-Fault Blackout Was Modelled in This Project	75
7.3.2 How a Multi-Fault Blackout Was Modelled in This Project.....	76
7.4 Metrics for the Evaluation of the Proposed GA.....	77
7.5 Robustness or Scalability of the Proposed Hybrid GA	78
7.6 Quality of the Solution (Accuracy)	78
7.6.1 Single-Fault Blackout.....	79
7.6.2 Multi-Fault Blackout	82
7.6.3 Computational Effort of the Solution	87
7.7 Discussion, Guidelines, and Rules for Using the Algorithms.....	91
Chapter 8.....	93
Conclusion and Future Work	93
Appendices.....	106
Appendix 1	106
Appendix 2	108
Appendix 3	111

Chapter 1

Introduction

The goal of this project is to use a hybrid genetic algorithm to restore power to a power grid after a blackout by minimising the lost load and number of switches within a reasonable amount of time for power operators as well as the amount of losses. Electricity has become an indispensable commodity of this twenty-first century, especially in the developed world. Consumers in sectors such as health and welfare, finance, supplies of food and water, electronics and commercial enterprise require electricity. Electricity is needed when cooling a house during the hot summer weather or when warming a house during the winter periods. Gadgets that are produced for households and factories all rely on electricity to operate. Even though new gadgets are being produced constantly, the power system has remained the same and has been operating dependably for many decades thanks to careful management of the resource. The careful management and control of the electrical power grid in developed countries has significantly lowered the incidence of power failure, but power blackouts still occur in the power network. Even though blackouts are rare, there have been cases reported in the literature of the United State experiencing over 400 blackouts within 1984 and 1999 [1]. Brazil experienced a five-hour blackout in 2009 as reported in [1]. When a major portion or all of the electrical power grid fails, it is called a blackout. This can be due to the closed breakers linking the grid system malfunctioning or even due to a natural disaster like a tornado breaking part of the grid. In the Third World, in countries such as Ghana, blackout is a common phenomenon. There is consistent load shedding to help save the ageing power grid. But for the majority of the First World, even short power failures are rare; but if a blackout should occur and last for hours or days, it would significantly impact the activities and economics of daily life. Because of the impact a blackout can have on the economy, this thesis proposes a hybrid algorithm that will help in restoring power to the grid in a reasonable amount of time as well as at a reduced cost.

1.1 Electricity Grid Background

Electrical voltages in the range of approximately 12,000 to 30,000 volts are produced in generators which are powered by fuel sources such as nuclear, coal, etc. [2]. Only a few public electric utility companies sometimes own such generators. Electricity from these generators must be raised to higher voltage levels, helping to transport the power over

standard transmission lines. As stated in [2], transmission lines are operated at voltages in the level of “230,000 to 765,000 volts” in order to limit electricity loss due to conductor heating. This permits power to be moved economically over lengthy distances. The lines are connected at switching stations and substations in order to form a network which is known as a power grid. From the generators, electrical flows pass along the network of lines using the path of least resistance. According to [2], the physical laws for electrical flow are usually compared to the manner of water flowing through a network of canals. Once the electrical flow nears a load centre, the voltage level is stepped down so that it can be distributed to customers. The system which handles bulk power transfer is largely an alternating current (AC) system, rather than a direct current (DC) system. AC systems are easier and more economical in converting voltages from one level of the grid to another level. Some customers, such as larger commercial and industrial users, accept intermediate voltage level service from 12,000 to 115,000 volts, but residential customers typically accept 120 to 240 volt electrical service.

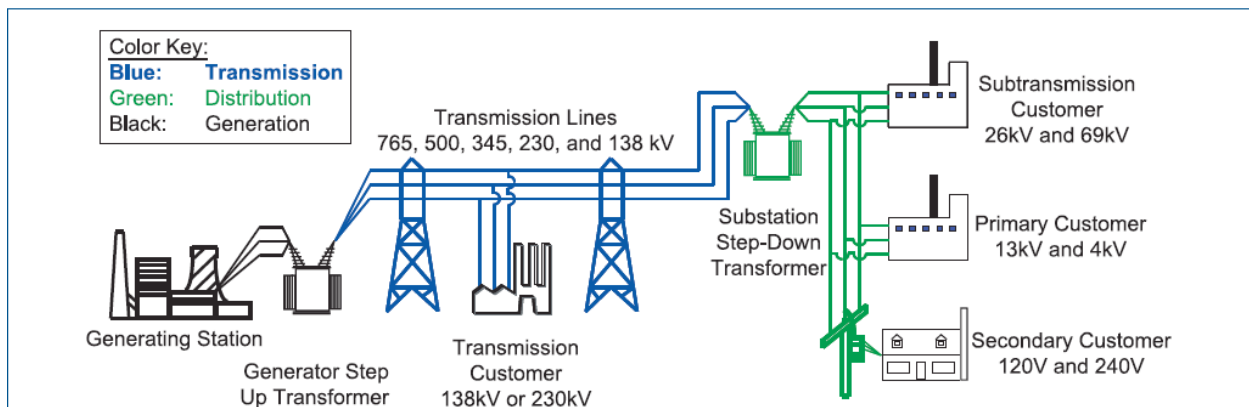


Figure 1.1: Structure of electrical power generation (by Energy.gov [2]).

1.2 Motivation for the Research

Several major power failures have been experienced around the globe during the past few decades. A major Italian blackout was reported on 28 September 2003 and Libya experienced a blackout on 8 November of the same year. Also in 2003, 14 August, there was a major US-Canadian blackout [3]. As power systems become more complex, they are more exposed to blackouts. They are also more likely to be operating close to their security boundaries because the number of people dependent on electric power is increasing while the grid structure is still the same. The major causes of blackouts include human error; natural

disasters like earthquakes, heavy rainfalls, and strong winds; the lack of evolving regulatory rules; power system deregulation; and reduced investment in the power system industry.

When a significant portion or the entire power system fails it causes a small or major blackout. Usually, a major failure of the system is created when a series of cascading failures [4] cause the system to trip a transmission line or generation units. Such failures are labelled “contingencies”. A typical pattern of a single-fault blackout begins with a strong contingency which will create a variation in the flow of power and “bus-bar” voltage. This voltage variation may cause generation units or transmission lines to fail. This event causes a gap between the demand for power and the generation of the power [5]. As the spread of problems grows, it can cause a major blackout. Developing a reliable algorithm which restores power [6] as well as tools such as MATCASC that can analyse cascading line overloads [7] is crucial.

The current methods used in restoring power following a failure of the system require many professional experiences and these methods are dependent on specific power plants. The time which is necessary to restore power can vary between convenient and unfavourable based on where the blackout has occurred. With the existing situation, this thesis proposes a hybrid genetic algorithm (GA) model that tries to provide a solution to the problem. The hybrid GA that has been proposed in this thesis is aimed to help operators in a real scenario of power blackout restoration. Furthermore it is up to date with the current power system models that are found in the literature review. The result of using the proposed GA is that there is minimum impact for the customers involved or the community as a whole.

The proposed hybrid algorithm in this thesis was implemented using the MATPOWER tool [8] with the standard IEEE14 bus system [9] as one of the main testing benchmarks. It was also run on IEEE30 and remodelled IEEE118 and the results show how resilient the algorithm is. Larger grids beyond IEEE118 were not tested due to the time constraint of the project and the amount of time required to perform extensive tests on them. The results were then compared to conventional GAs by using the same parameter settings and type of system. Some of the limitations of a conventional GA for power restoration in a power grid motivate the need to have a robust hybrid genetic algorithm that can adapt to several power grids by consistently producing accurate results.

1.3 Research Question

The complexity of restoring a power grid following a single-fault or multi-fault blackout makes it a time-consuming and delicate task. It is considered to be an NP-hard problem by the scientific community because *methods that can be used in finding efficient solutions seldom exist* [10]. In other words it can be hard to solve in polynomial time.

When there is a power blackout, a significant portion of the system loses voltage due to the fault. Voltage can also be lost due to the cascading failure of the grid [11]. In order to recover from such a failure, there must be accurate coordination between the start-up and regulation of the generators and the connection of power lines and electrical loads so that the risk of another blackout is eliminated.

The main research question of this thesis is

How can power be efficiently restored in a power grid after a power blackout by minimising the number of switching operations, minimising the out-of-service area, minimising the losses, and restoring a high percentage of the load, thereby leading to a reduction of cost?

To effectively answer this question, the following sub-research questions were posed.

What is the current state of the existing power grid?

What is a power blackout and what are its causes?

What are the steps that need to be taken in order to restore power to customers when there is a blackout on the power grid?

How can a hybrid genetic algorithm approach be used to restore power in a power grid?

1.4 Objective of the Project

The research aims to achieve the following objectives:

- a) To design a flexible hybrid GA model for optimum power restoration which can be used in practical power systems.
- b) To reduce overall cost of the restoration in a power grid by
 - minimising switching operation costs by determining the optimal configuration of the switches during the restoration period
 - reducing the losses in the distribution grid
 - minimising the out-of-service area

1.5 Scope of This Project

In this thesis, the hybrid GA was extensively tested on IEEE14 bus systems [9] which can be seen in the results section. The IEEE test system was modelled so that the main generator is always turned ON for the power grid to operate. The system was run using AC power flow analysis in MATPOWER. MATPOWER is considered one of the most efficient and robust tools used in simulating and finding solutions for power flow models. Its other uses are in optimal power flow situations.

The constraints considered are

1. Voltage constraint: Voltage on each bus should be between V_{min} and V_{max} for a given bus.
2. Line overload constraint

While taking the constraint into considering, during some run, strings that violate the constraints are discarded. In some other cases, a penalty of increasing the cost when constraint is violated is imposed instead. Each of the case helped to fine tune the experiment.

1.6 Outline of This Thesis

Chapter 2 presents the literature review of the thesis. This chapter presents existing methods that have been used over the years to restore power to a system.

An overview of genetic algorithms and power system modelling is presented in Chapter 3. A closer look at how a genetic algorithm is applied to power restoration is discussed.

Chapter 4 presents the power flow analysis used in the work.

Chapter 5 discusses the power system modelling during power restoration.

Chapter 6 discusses the hybridisation of the GA. The hybridisation was done by introducing local search in the search phase as well as a novel radiality check algorithm.

Chapter 7 presents the results obtained from the simulation in MATPOWER. It also shows the performance, scalability, and quality of the solutions.

Chapter 8 presents the limitations of this work.

Chapter 9 gives the conclusion and future work for this project.

Chapter 2

Introduction

This chapter presents a detailed overview of how power restoration is done in a power grid. First, the power restoration as a search problem is explained. Second, a detailed overview of the operating state of a typical power grid is presented. Any disturbance to the normal operation of the grid is communicated to the grid operators. These are sent as alert signals. Third, an overview of power blackout as well as its consequences as stated in the literature review is presented. Some companies like Ford [5] had to rebuild a furnace which stopped working due to a power blackout. Lastly, different restoration procedures and approaches are presented to help better understand the restoration process.

2.1 Power Restoration

The power restoration problem is considered a search problem and is sometimes called a solution for many problems. It can be explained by finding a series of actions that can be applied to the power grid in order to get a desirable goal. One of the goals can be to restore power to priority customers in the network. Each action that is performed on the power grid changes the state of it. For example, when we change the state of one switch from ON to OFF, it changes that particular section of the grid. The same happens when a switch is turned ON. When the correct restoration action is not taken, it can cause enormous damage to the power grid. Furthermore, as the grid is composed of several components and each component affects its state, the main aim in restoration is to find a sequence of actions to restore the grid from its blackout state to its normal working condition.

A well-defined search problem should have the following characteristics according to the following [12].

- Starting or initial state: This state in a power restoration problem is when a blackout has occurred, as this is when restoration needs to be done. When the grid is functioning properly there is no need to do blackout restoration.
- Operator function: The set of states reachable from the initial or main component (usually the main generator) to the other buses or small generating units.
- State space: All states reachable from the initial one by any sequence of actions. This can be the turning of switches from ON to OFF or OFF to ON. When there is a

connection between, for example, bus 1 and bus 2, then bus 2 has a switch and the switch is off. To be able to reach bus 2, the switch needs to be turned ON.

- Path to take: This is the path taken from one bus through lines to a switch and from the switch to the next line, then to the next bus.
- Cost of taking a certain path: This is where the cost is assigned as switching operations incur a cost. In the end, the total cost will be the sum of the individual cost along the path which leads to the blackout restoration of the grid.
- Testing whether the goal has been reached: At each step, a test is made to determine whether the goal has been reached, that is, whether the restoration steps that were taken have restored enough load to the grid during a partial restoration and whether full restoration happened, meaning that all the isolated buses have been powered.

In Chapter 1, a simple diagram of the power grid was shown. Furthermore, based on what was stated above it is known that the grid consists of several components. Whenever there is a problem, there is the need to search for the problem area and also to find the faulty components or the cause of the problem. The emergency state of the grid can be caused by a faulty switch, a faulty generator, a line that has been cut, or a faulty bus. Locating this area requires that a search be done either manually by the operators or by an implemented algorithm. In the power grid we consider the source as the generators of the network which send power to the various buses through the lines that are connecting them. Once the faulty area has been located, a series of steps are taken to restore the lost load or power to the blackout area. One of the main goals in the restoration process is to have a state of the grid where the minimum requirements are satisfied, such as supplying power to high-priority customers and taking into consideration that a good percentage of the load that is lost has been recovered. Numerous papers to date have suggested blackout restoration algorithms. In this work, the focus is on heuristics-based search methods and more specifically on a genetic algorithm for power restoration.

2.2 Power System Operating State

As the power grid is a structured system, it has certain operating conditions. These operating conditions can be classified as five different states: normal, alert, emergency, in extremis, and restorative [13].

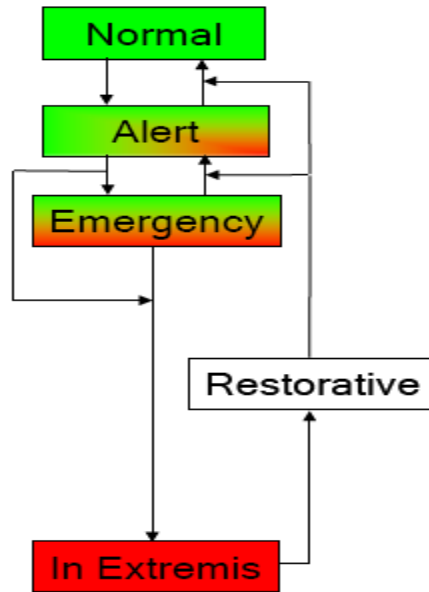


Figure 2.1: Operating conditions of a power grid by Zima [13].

The normal operational condition means that “each of the variables in the system” [14] lie within the normal range. No overloading of the equipment is present. The system is able to handle a contingency without loss of security. Constraints of both inequality and equality are met. The alert state of operation occurs when “the security of the system” is lowered while constraints of inequality and equality are still in place. The system can continue to operate within the range of acceptable limits, but a contingency could cause a state of emergency or even an in extremis condition. When the system goes to an emergency state, inequality constraints are violated if the operation of the system occurs at a reduced frequency with voltages outside of the normal range. Alternatively a portion of the equipment may be overloaded. When the system has reached an emergency state due to a severe disturbance, it may be brought back to the alert state by applying emergency actions. Without successful application of emergency control measures, the system may go into an in extremis state, which causes both the equality and inequality constraints to be violated. Cascading outages could cause a single-fault blackout or a complete failure of the system [14].

2.3 Overview of Blackout

Bulk power supply systems of today are a proven source for electrical power. A combination of circumstances can result in the remote likelihood of an outage which is system-wide.

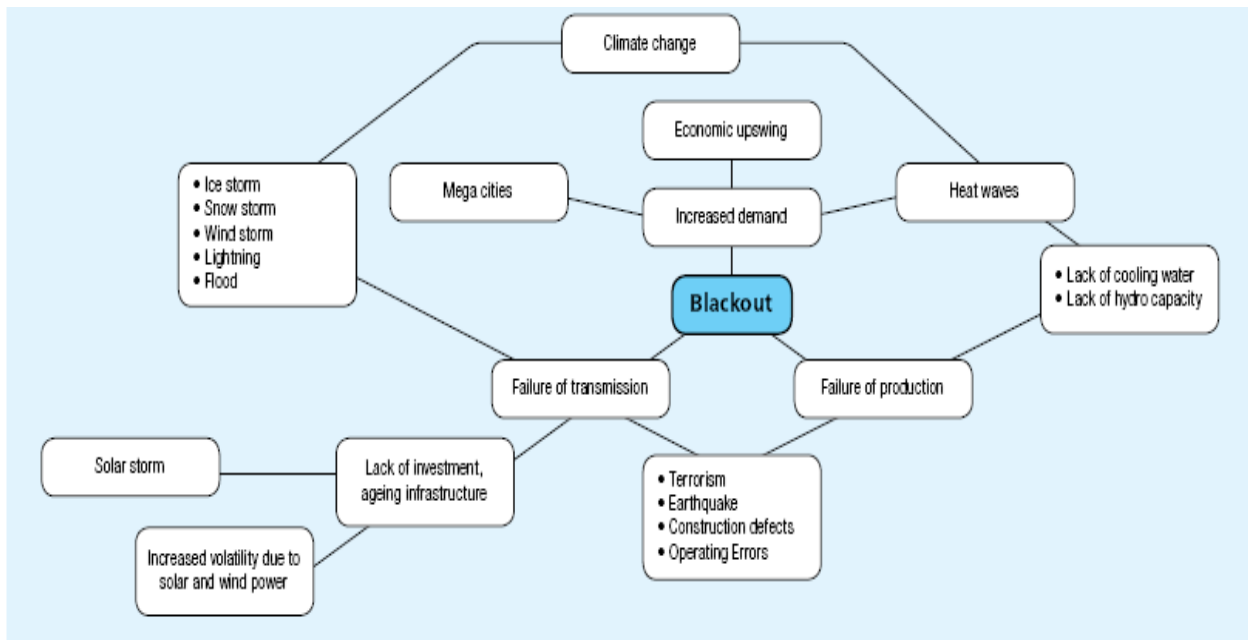


Figure 2.3: Causes of power blackout by Bruch et al. [5].

Figure 2.3 shows some of the causes of power blackout. The figure contains a list of natural disasters that causes blackout and the increase in demand of consumers during peak periods which leads to overloading of the power lines. Usually power failures are not created by just one event but rather by a combination of multiple power deficiencies. In the literature most researchers have stated that there are rarely situations in which a power grid will totally collapse as a result of one single problem or cause. In most cases it is a combination of several different or similar causes. Here are a few of the conditions which increase the risk of a power blackout occurring.

• **High demand for power which causes the grid to be excessively used.**

Electrical utility companies are not building new lines to take care of the increased demand for electricity by consumers [15]. As a result, many lines are reaching their physical limits, especially when cold weather adds demands on the system. When the temperatures are outside the moderate ranges, either hot or cold, it leads to an increase in demand for electrical power. When the power demands are high, it pushes the grid to work near or over its limit. When the demand exceeds the ability of the lines to carry the load, a switch must be closed so that the line is protected from additional damage, causing a blackout. Such bottlenecks in the lines make the entire grid less responsive to the additional demand or to the excess capacity in other regions. It becomes more difficult for one area to help out another area [16].

• **Switching off part of the grid to shed loads or for the purpose of maintenance.**

Because the power system is ageing, some parts must undergo regular maintenance so that the entire grid remains in top working conditions. When this occurs, it becomes necessary for some parts of the power grid to be taken out of operation or completely switched off [17] [18].

• **Defects caused by the ageing of material used in the grid construction.**

Much of the infrastructure which serves First World countries like the United Kingdom and US is ageing. For example, the average age of power plants is over three decades. Most of the facilities have a life expectancy of forty years [18]. The components of distribution and transmission grids are also ageing. Power transformers average more than four decades in age. Seventy percent of lines today are more than a quarter-century old. As the components of the systems reach retirement age, they are replaced with newer components. These components are linked to automated systems or communications. During the linking process, there may be a break in the power supply.

Even though there haven't been many blackouts recently, the five most severe blackouts recorded in history are listed below.

i. **New Zealand – 20 February 1998**

A line failed, causing a chain reaction. The result was a power outage for 70,000 people. This was the longest period recorded for this type of fault [19] [5].

ii. **Brazil – 11 March 1999**

In Bauru, São Paulo state, there was a lightning strike at an electric substation which tripped the circuits. The result was a blackout which affected ninety-seven million people covering over 70 percent of the area. It lasted for five hours [20] [5] [21].

iii. **Northeast United States and Central Canada – 14 August 2003**

“A shunt fault was created when power lines were pushed together by trees” [2]. There were also as a series of technical difficulties and other circumstances that worsened the situation. The result of this was a power blackout lasting four days and affecting fifty million people. The losses for this outage were estimated at US \$6 billion. Insurance only covered about half of these [20] [21] [2].

iv. **Italy – 28 September 2003 [22]**

The entire country of Italy experienced an eighteen-hour electricity blackout when a technical failure caused the nation to be separated from the other European nations.

v. **Indonesia (Java Island) – 18 August 2005 [2] [15] [5]**

A seven-hour power outage (for around 100 million people in Java and Bali) began when there was a power failure in the utility system which serves Madura, Java, and Bali. The resultant cascading failure caused at least two units of the grid at the East Java Paiton plant to shut down and six units at West Java Suralaya plant to shut down [5].

vi. **India – 30-31 July 2012**

On this day, “there was a grid disturbance at the Northern part of India which caused a blackout to 8 states in India” [23]. It is stated in [24], over 300 million people were affected in this incident. The metro had to start an hour late and major hospitals were affected as well as other public transports like the train service experienced a delay in operation. Hospitals which were able to operate during the black had to use their backup generators.

Furthermore, the immediate cost of any blackout includes production loss (companies and individuals), facilities depending on power being unable to work or becoming idle, electronic data on computers being lost or damaged, spoiled food as there are big refrigerators that are used to store perishables, and equipment being damaged. In addition to these direct costs, there are indirect costs associated with blackouts. Some of the most common ones include loss of water, legal expenses, accidental injuries, and looting. Generally, indirect costs are up to five times those of direct costs. The previous paragraph stated the biggest blackouts that have happened so far. The next paragraph presents four companies that were hard hit by blackout in the United States. These are recorded as the most expensive cost ever to be incurred by companies as a result of blackout.

As stated in the US blackout, approximately \$6 billion USD was lost. Three companies that were hard hit by the blackout are listed below.

- i. Daimler Chrysler: Had production losses in nearly half (fourteen) of its thirty-one plants [1]. Six of the plants had paint shops where there were 10,000 vehicles in the process of being painted when the outage hit. All these vehicles had to be discarded or scrapped [25] [5] [26].
- ii. Ford Motor Company: At the Brook Park, Ohio, Ford casting plant, molten metal cooled and solidified in a furnace. The firm estimated that removing the metal and rebuilding the furnace resulted in a week of lost production [26].
- iii. Marathon Oil Corporation [26]: Emergency shutdown procedures were triggered due to the blackout at the Marathon Ashland refinery located sixteen kilometres south of Detroit, Michigan. The authors stated that a carbon monoxide boiler could not be shut down properly when the blackout occurred. Therefore, there was a minor explosion

which released steam and hydrocarbons. The police took precautionary measures and evacuated hundreds of local residents forcing them to find alternative shelter. The evacuation extended in a one-mile wide strip around the plant.

In general, various media, including newspapers and electronics work to get information out to the general public during a blackout. In most instances, the media had to use generator backup to provide electricity for their production purposes. The blackout happened at the same time that critical cyberthreats began to emerge. Business networks which were not patched or were only partially patched saw significant impact [2]. During the outage, most wireless services were overwhelmed. These included cellular and personal communication services [17].

The blackout created storage and shipping problems for dairy producers and commercial establishments. Many retailers had to discard stock because of spoilage and numerous dairy producers were forced to send milk to Manitoba so that it could be processed.

- iv. A lengthy blackout created difficulties for a healthcare industry because of the inability to procure the chemicals needed to purify and treat drinking water [26] [5].
- v. Because a blackout occurred at the close of a working day across the service areas, there were disadvantageous effects on the grids used for distribution. The effect was highly felt during rush hour because traffic lights were no longer working and electronic signs on the highway were also switched off. There were travel delays at the airport, bus, and rail terminals.

Figure 2.4 shows the summary of the impact of blackout on various industries. One or more of these industries will experience a loss when there is a blackout in the area in which they are connected to the power grid.

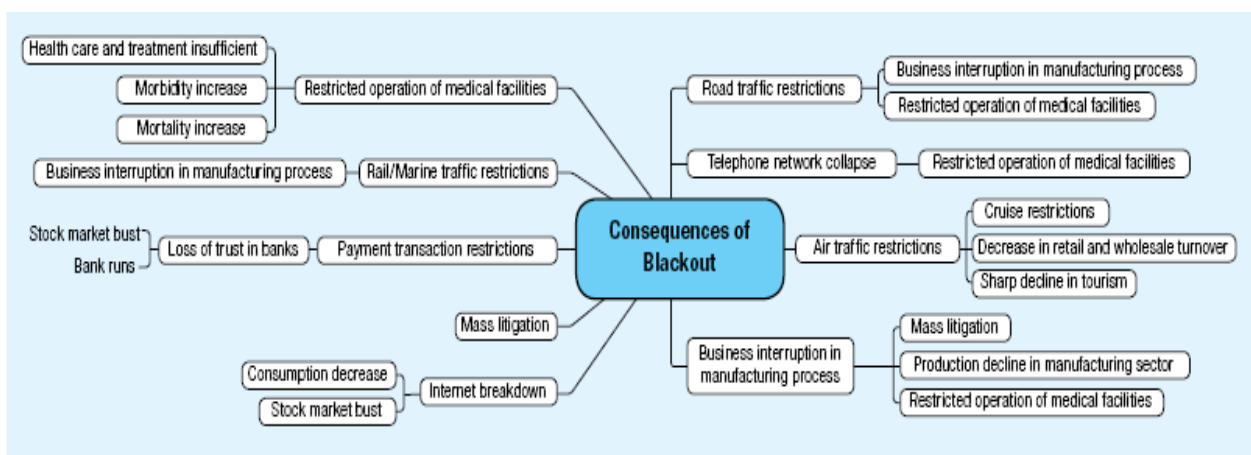


Figure 2.4: Impact of blackout on various industries by Bruch et al. [5].

2.4 Overview of Blackout Restoration

There are several definitions of blackout power system restoration but in this thesis it is defined as the process of restoring generators to their normal operating state, transmission lines repair, and the restoration of network loads to an optimum condition in a favourable amount of time possible without damage to elements of the power grid [19]. Operators are under tremendous pressure during power restoration to restore the power supply quickly so that actions which might cause severe damage to the plants are limited. Operators must also ensure that appropriate staff is informed. If there are questions about operations in the aftermath of the outage, the validity of their decisions may need to be justified. It is common to encounter problems during the restoration process. This is due to the abnormal state of the system, differences between the forecasts of the control and outstation staff, and differences between training session drills and the actual event.

Some of these problems occur are when there have been cascading failures in which emergency conditions of the power grid are unknown, thereby the same failures cause another blackout in the short-run. Firstly, it can be the consequence of an error caused by the operator [20]. Secondly, when there is voltage overload caused by conditions such as overexcitation of the transformers, self-excitation of the generator, or under-excitation of the generator, a problem can occur. Thirdly, insufficient knowledge of the system is another problem. In the power industry, having good knowledge of the circumstances of the grid failure is very important (overloading, equipment failure, or possibly human error) [27] [28]. Fourthly, too fast of a restoration can be bad because the operators may attempt to pick up demand too quickly. Then the generator demand balance can't be achieved. The frequency could fall and the subsystem could collapse, leading to a delay in the overall restoration process. Fifthly, during some restoration operations, some equipment may not be available for operation, therefore causing the restoration process to be difficult. Also, according to [17], there are also times when "there is competition among distribution utilities to have blocks of load" reconnected that sometimes far exceed (high demand) what is possible during the restoration process. This can create new difficulties [17] [28]. As a result of these difficulties, a few considerations have to be taken into consideration, such as path management, where the status of the path that is found must be examined, i.e., faults, line capacity, and transients, due to the switching operations of the network. As restoration is a search problem, finding the shortest path with minimum switching actions between the distribution unit and the receiving units is the best result unless such a path is not possible or feasible. Also during restoration,

the loads are to be restored by taking into consideration factors such as the priority of the customers as well as the system security of the grid. The switching actions are to be coordinated to minimise problems such as overvoltage and voltage transients. Energisation from the low-voltage side to high-voltage side of transformers is prohibited. Line and transformer capacities should not be exceeded during restoration [20].

Overview of Existing Blackout Restoration Concepts and Approaches

Several blackout restoration concepts have been proposed in the literature. According to Omara (2009), the general concept of power restoration is as follows [29].

- i. Sectionalise power system to islands [6].

During this process the blackout location is sectionalised into multiple subsystems and the restoration is done concurrently across several islands. The problems of sectionalising the system can be fairly complex for a large-scale power system when constraints such as black-starts, generation load balances, and voltage stability are taken into consideration. Planning the sections requires each of them to have 1) adequate black-start capacity to offer power starts to the additional generating locations within the island, as well as to supply critical load power, and 2) adequate generation to carry the majority of the load requirements within the island [30].

- ii. Restore system islands.

Within each of the islands, beginning with a sizeable thermal power generation station, the framework of the system is energized. The system then builds paths to other generation stations and on to the large load centres. At each step, loads are energised. Some of the critical loads are restored in order to stabilise the blackout units. The completion of this step ensures that the network has adequate power and stability to handle transients due to additional load pickup, as well as adding large power generation locations [6] [19].

- iii. Synchronise islands.

This step in the restoration process is when the various subsystems are interconnected. Over time, the rest of the loads are incorporated into the system and the system is able to transition to the alarm state or preferably, to the normal state [31].

- iv. Normalise the system.

The system begins its normal operation after the above three criteria have been met.

The organisation and implementation of a plan for restoration is the key factor for success in any restoration process [19]. There have been multiple strategies suggested, including the bottom-up restoration which begins with restoration of the main network. Once that is

completed and transmission and substations are energised, the generators are resynchronised and the loads are picked up. Under this strategy, there is a parallel restoration of several areas. The design permits quick load pickup. During this process, the loss of one island doesn't cause failure of the whole system. Another major strategy is the top-down system. When there is a multiple-fault blackout condition and electricity cannot be brought in from adjacent power units, this strategy is appropriate. First of all to start the restoration process, the system is sectionalised to some islands. In each island, the existence of at least one black-start generator is necessary. The next step is to restore the lines, loads, and generators of each island, followed by resynchronisation of all the islands. The reconnection of the balance of the components is the final step [31]. The benefits of this strategy include the ability to restore more than one section of the system concurrently, quick restoration of critical power to generating stations as well as power and light to substations, and the fact that it does not require synchronisation due to one island. Other papers also proposed a combined method of using both the bottom-up strategy and the top-down strategy. The combined method uses the advantages of both systems.

2.5 Categorisation of Power System Restoration

This subsection presents ways in which the power system restoration problem can be categorised. Various research conducted in this field has different categories of classifying the power system. In this thesis, the following categories have been chosen as the main ones.

2.5.1 Subsystem Restoration

This category focuses on situations where there is only a minor part of the system which has no power, for example, in a restoration problem which uses a generator start-up sequence operation as stated in [28].

2.5.2 Manual Restoration

This category assumes that operators have complete information about the system available. In the event of a power outage, the location of the blackout is determined by checking each station. Once the problem area is determined, the problem is manually repaired by the operators based on their knowledge and training [28].

2.5.3 Computer-Aided Restoration

In a computer-aided restoration, a tool is developed with the aim of helping operators during the blackout restoration process. Operators in the control room go through a specific action sequence to restore power. The computer-software tool presents the steps required and it shows the results in the form of a log once the operation has been performed [32] [33]. The operator also has a restoration manual which is studied as part of his training and is also familiarised with [33].

2.5.4 Cooperative Restoration

Cooperative restoration methods combine the efforts of the computer and the operator. Each is responsible for performing tasks where efficiency is greatest. The restoration is able to proceed with the ability to plan future steps and execute them [19]. The operators or operator teams can work in parallel lines with the computer responsible for checking the proposed steps of each operator to ensure consistency between the lines [6].

2.5.5 Fully Automated Restoration

When the system operates under fully automated restoration, the operator is only in a role of supervisor or observer. This strategy has not yet been proposed, but implementation is doubtful any time soon. It would require extensive and time-intensive searches [6] [34].

The objective of any restoration process is to return the power system from a disturbed status and loss of loads to a status of stability. All customers should be supplied with power. Then the voltage profile and the economically efficient generation status are adjusted. The goal is achieved through the correct sequence of restorative tasks, including switching, setting, queries, or commands to the appropriate power stations and outside control centres.

Due to a certain extent of indistinct system response, having a steady orientation of the sequence at the actual power system status is crucial. Thus, an interactive proceeding is required, and the particular operational actions cannot be firmly predefined. Based on these principles and prevailing conditions, the concept of the restoration system was derived [35].

2.5.6 Algorithmic Restoration

Multiple different algorithmic systems have been proposed over time. These include expert systems, tree search (heuristic search), petri-nets, discrete particle-swarm optimisation, combined Tabu Search (TS), genetic algorithms, and a multi-agent system [36] [37] [38]. An algorithmic approach is essential for improving consistency and efficiency during the

operation. Enhanced skills are acquired in the application of the process. Efficiency is gained when there is analysis readily available without forgetting the process of specification.

The focus of this thesis is on the algorithmic category where there is no need for manually preparing guidelines for the operators of the power unit. The use of an algorithm helps in the development and identification of the various processes, not to mention the major decisions and variables that may come in handy to solve a given problem.

To conclude, this chapter presented the different types of restoration approaches currently being used in blackout restoration. Even though there are other ways of categorisation, these six were chosen because they provide a broad view of power system restoration. The monetary losses of companies were also given, which shows the economic impact blackout can have on industries and communities as a whole. In the next chapter, the literature review of genetic algorithms was presented. The application of genetic algorithms in the power restoration field is also discussed at the end of this chapter. The proposed hybrid genetic algorithm is then presented.

Chapter 3

The Genetic Algorithm (GA)

In this chapter, an overview of the GA for this project and its important role in the field of power restoration research is presented. When using a genetic algorithm to solve a problem in any study, there is a need to find a balance between exploration and exploitation while finding the best solutions for success in various environments [39]. In the literature, genetic algorithms have been proven to give a robust search process in the most complicated search fields. The validity of the GA process has been demonstrated in multiple dissertations and research papers. It has been successful in optimisation and application problems such as those found in power restoration applications. The GA has been established as a reliable algorithm in complex search problems and is now being used in more far-ranging applications, including engineering, science, and business [36]. The reason for the increased number of applications lies in the simplicity of GA computations and its strength in finding improvements to the solution that is produced.

But there is a limitation when using a GA because it makes assumptions that are restricted to the search space required. In such a scenario a solution is not guaranteed as the algorithm will not converge. In other cases, finding a solution which is close to optimal is acceptable (e.g., in this project when there is no full restoration [i.e., not all the buses were able to be powered], we take into account the amount of load restored) given that the computations (restoration) can be completed fairly quickly. In this instance, one approach would be to use a probabilistic algorithm which would reach solutions approaching the real optimum. Genetic algorithms are in the same category as probabilistic algorithms, where random candidates are initially generated and these candidates have to go through the process of evolution for an efficient solution to be obtained. As the project is based on how to find an optimal way of restoring power during a power blackout using a proposed hybrid genetic algorithm, this chapter will lay the foundation for this thesis.

3.1 Literature Review on Genetic Algorithms

One of the widely used heuristics-based search algorithms is the Tabu Search. The TS algorithm works by searching through a determined solution space in order to avoid cyclic searching or being trapped in local optima. Details of the TS algorithm are out of the scope of

this thesis, hence readers who are interested in this algorithm can find references in [40], [41], [42], [43], and others.

GA is another widely used heuristic search algorithm. This thesis focuses on using GA for power restoration. The algorithm makes use of adaptive search methods which use a genetic process similar to that of biological organisms. A GA begins by randomly generating several solutions to any problem and encodes them as strings of symbols which are usually made of 1 and 0. The generated string which encodes the individual solution is called a “chromosome”. (See the literature review for an explanation of this term.) The solution set consisting of several chromosomes is called the “population”. Some papers use the term “allele”, which refers to the position of the symbol within the string that was generated. The term “gene” is sometimes used to refer to the symbol or value which the allele contains. The generation of the initial population can be a random set. It may also be created from a number of known solutions. The population could also combine both. The GA steps involve replacing the initial population at the beginning of each of the generation steps with another population which is considered to be better in terms the fitness function. In each case, there is hope that the new population includes a solution to the problem in the question. As this follows the evolution process, there is the reproduction process, which involves combining chromosomes of the previous population in order to create new chromosomes. There are three operations for each reproduction step: selection (several forms are available), crossover, and mutation. This reproduction process continues till the necessary number of “chromosomes” is found in the newest population that has been generated.

The idea of using GAs to solve difficult optimisation combinatorial problems resides in the fact that GAs perform a search through the space on which the problem is defined in several directions [44]. A random initialisation is performed on the population. Each individual becomes an encoded solution of the problem. In order to evaluate its fitness function, the individual information has to be decoded. It captures “the objective function and the constraints of the problem; in fact, it plays the role of the environment” [45] [46]. The search for the optimal solution starts from several points (e.g., initial population of individuals or solutions). Thanks to the selective adaptation of individuals to the environment through a number of generations, the search is guided by the fittest among them [47] [48]. Although GAs explore a search space randomly, this exploration is usually done only through the regions that are chosen considering the objective function. This makes GAs capable of finding the global optimal solution. Because GAs are capable of finding globally optimal

solutions when it comes to large-scale optimisation problems and according to the authors in [44], GAs have been found to be efficient in finding solutions in the power restoration domain [49]. Another characteristic of a GA is that the constraints or even the objective function can be easily changed and after those changes the algorithm can be applied to solve another problem.

Irving et al. (2002) proposed a new systematic supply GA which utilises a scheme for encoding integer permutations. Each integer is a representation of a single controllable switch. Thus the integers are numbered from one to any higher number of positions, and they form integer permutation encoding. Determining a solution for the chromosome depends upon the order of the integers. The authors used graph theory to provide a way for to create “a mapping” leading from the list of permutations to the end status of each individual switch. The GA adheres to the constraints and objectives that are needed for the specific supply restoration scheme. The results of the simulation indicate that the proposed algorithm is effective for partial or full supply restoration [50].

Luan et al. (2002) proposed a GA where an “integer permutation encoding scheme” is used. The switches are represented by chromosomes and the final state of the chromosome determines which switches are to be turned off or on. The authors used a graph theory method to guarantee that the chromosome is connected to unique radial network topology during the evolution process. A “zero” gene was used to help find the point where load shedding could be done in a situation when the demand for power by customers is beyond the supply of the power grid. Load shedding is commonly used in Third World countries like Ghana. Real-life power systems were used to test the robustness of their algorithm [42].

In [51] an improved genetic algorithm for blackout restoration in power distribution systems was proposed taking into consideration the priority customers. Priority customers are customers who need power the most and this is dependent on the place where the blackout occurred. In a situation where a hospital and households lose power, the restoration is done first to the hospital and then to the households. The authors found that there were high-quality solutions produced at every next generation. In the paper, they also used a string to represent the status of the switches in the network. The three main objectives in the paper were to minimise the out-of-service area, reduce the number of used switches in the restoration, and lower the losses of the grid when there is a blackout, taking into consideration constraints like voltage and current. The proposed method tries to obtain a solution for the problem of service

restoration by utilising a remote-control switch. The solution, as demonstrated in the simulated results, requires less time to restore power to the grid.

Gomez et al. (2009) proposed a GA which uses a new type of crossover technique and also a new form of initialising the algorithm in order to find a feasible solution [39]. In the paper, the proposed algorithm utilises a Power Flow Program (PFP) to compute the fitness function of problems that are considered unfeasible. A new crossover and mutation procedure was developed to only find feasible solutions in the search space [39]. According to the authors, this is an efficient way of reducing time because only feasible solutions are generated. Once the PFP has completed computing the feasible states, the result is then fed to the GA for processing. Then based on the fitness function an evaluation is made which helps in the restoration plan when there is power blackout [39].

The GA proposed in [45] was able to lower the out-of-service area. Repairs are implemented on the failed section at the same time. The authors proposed a solution which divides the problem into two sections, both handled by ranked genetic algorithms. The constraints of each search space are reduced by a ranking procedure; therefore only local knowledge of the problem is required. The solution process is reduced to a manageable level because of the split. In solving the problem, the most desirable network structure in regard to the switch position is determined. Those genes which may be eliminated are determined. The next step is to decide on the most effective switching order. This step makes the best use of the switching task that was already used in the first step. The purpose was to limit the required time. Only a single constraint was utilised in the paper, i.e., switching precedence (avoiding overloading during switching). The results avoided overloading while the switching was going on. The result of the simulated approach indicated that the suggested method is fast and flexible. It doesn't require onerous parameterisation for the grid. The authors stated that out-of-service customers can be supplied with power when the proposed technique is used with other algorithms in finding the faulty areas in the grid.

In this project, a GA has been chosen for this work because of the following advantages: 1) It can find several alternative solutions even in complex problem scenarios [44]. This can be achieved because there are different numbers of ways in which a GA can be improved or optimised. 2) The exploitation of the solution space can be varied, thus leading to a solution that fits the problem [52]. 3) It provides a flexible way for the algorithm to be hybridised. 4) It always provides an answer and the answer gets better with time [53].

3.2 Basics of Genetic Algorithms

John Holland was the first to propose the well-known concept of genetic algorithms. The basis of the search algorithm is natural genetics and selection. It applies the “survival of the fittest” rule to a string structure. But in this case, there is a randomised exchange of information to create a possible solution to a problem [36]. Then each new set of chromosomes is produced by using the best parts of the old parents [45]. The GA continues the process of evolution to converge on the best solution at each generation. It is for this reason that GAs are considered to be function optimisers. Genetic algorithms are suitable in many types of problems when traditional computational methods fail. Creating the random population of chromosomes is the initial process when solving a problem using a GA. The chromosome selection is done whereby the ones which are the fittest are always given more changes than the others.

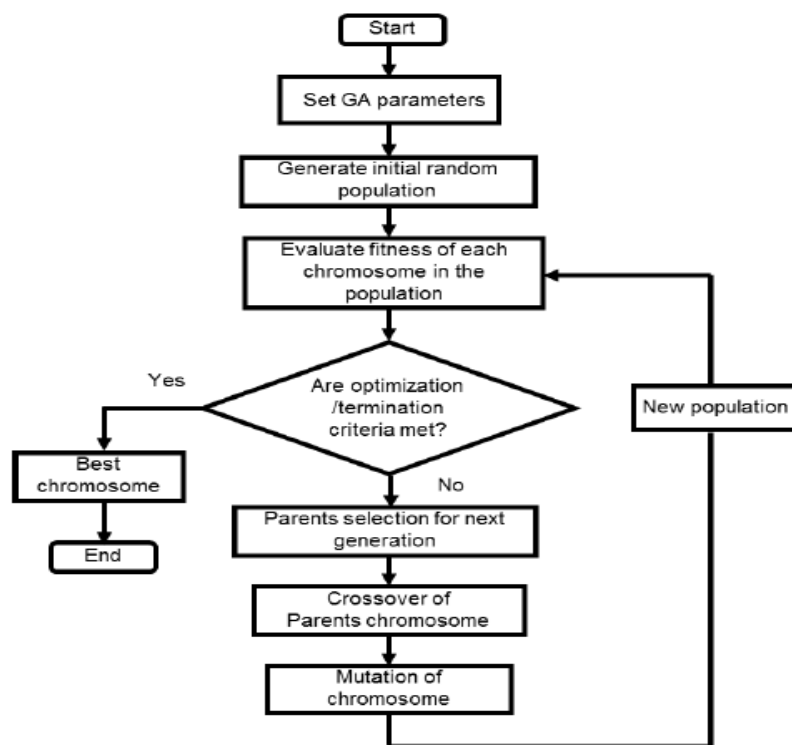


Figure 3.1: Layout of a GA by Razali and Geraghty (taken from [54]).

Figure 3.1 shows the schema of a genetic algorithm and the processes involved. Details of the steps involved when applying a GA to a problem have been explained.

The following shows the differences between a GA and other conventional search methods [46] [45].

- GAs use a population of solutions to conduct a search in a given space. This sometimes helps a GA search through the problem space with optimum points. While other optimisation algorithms depend on a single point during the search phase, a GA uses the different areas present or all the information present as a guide towards the solution to the problem. GAs have parallel computation ability because the fitness value of several individuals of the solution set are calculated at the same time. Lastly, it is efficient for GAs to find globally optimal solutions in a given problem.
- GAs use fitness functions or several objective functions for the search process so specific information is used rather than some auxiliary knowledge of the solution. GAs can therefore find solutions to real-life optimisation problems that are complex to work on. The power restoration problem is such a problem that can be solved by applying GAs. A GA uses probabilistic transition rules during its evolution process, and this is preferable when compared to algorithms that use deterministic rules.
- GAs can find solutions in a complicated area and they provide a flexible platform for hybridisation with other algorithms, which can lead to further accurate solutions. Also, a GA is a robust algorithm for complex problems [55].

3.3 Different Types of Operators Used in Genetic Algorithms

3.3.1 Selection/Reproduction

When an individual is directly copied to the next generation, it is called reproduction [54]. In the process, individual strings are reproduced using values which are related to their objective function. In some of the papers or literature, it is called the selection operator. The selection operator is responsible for reproducing strings according to the relative fitness value. To state it in another way, good solutions propagate to the next generation and poor solutions are eliminated. Typically, reproduction is the initial operator applied to the population. The selection is accomplished by [56] [57] doing the following:

- Identifying the good solutions from the population which typically depends on the cost function of the defined problem. The lower the cost function, the better the solution.
- Producing varied forms of the solution.
- Eliminating bad solutions that are found in the set and replacing them with good ones (most fit ones).

The brief explanation of what happens during reproduction involves the replacement of bad solutions with good solutions that can progress to the next stage of the population. To do so several selection procedures can be used. The roulette wheel, tournament, and ranking selections are the commonly used selection methods in genetic algorithms [57] [39].

In this thesis, tournament selection was used. Readers who are interested in ranking selection or roulette wheel selection can refer to the following papers: [58], [44], and [59].

3.3.1.1 Tournament Selection

This type of selection is one of the efficient and useful ways to select parents for mating [54]. The selection is done by randomly taking two individuals from the population space. Mating is then done between the most “fit” pair. Each pair then creates an offspring which has a mix of the characteristics of the two parents, using the crossover method, which is described in the following section. The two parents then return to the population where they can be reselected [57]. The random selection of pairs and mating of the most “fit” individuals will continue until a new generation which has the same size population is created. Figure 3.2 shows a diagram of how this is usually done. It can be seen that there are several candidates in the population and two which are the most “fit” won the competition in the tournament. They proceed to the next phase. In the next phase, the better one is chosen which then proceeds to the new generation phase.

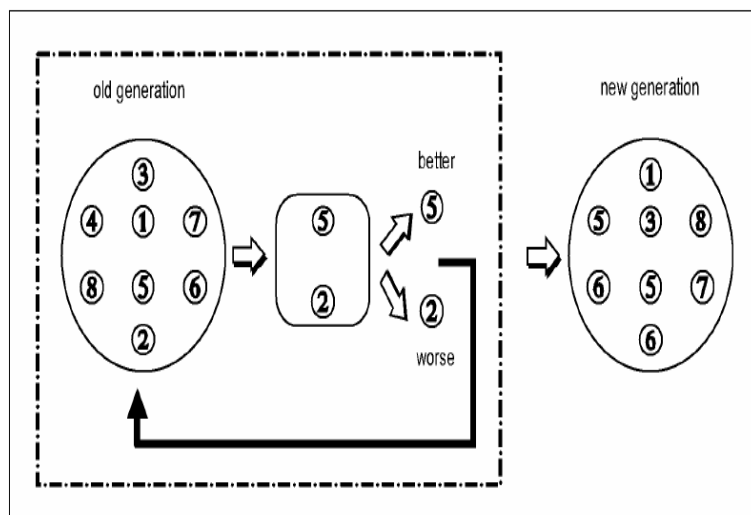


Figure 3.2: Tournament selection by El-werfelli (from [57]).

3.3.2 Crossover

The crossover operator helps the GA explore additional areas within the search space. It is the exchange of data between solutions. Two individuals from the pool of potential mates are randomly selected. Then a point is selected for the crossover process. The data which originates from one of the solutions is transferred to the other solution to the specified junction point. This allows for the generation of a fresh pair of solutions in each subsequent generation. This is comprised of a mixture of the solutions from the previous generation [42] [57] [60].

In the literature, three types of crossover are commonly used.

1. One-point or single-point crossover [57] [61]

In this crossover method a random point along the solutions is picked. The bits which are present on the right side of the specified point are transferred between the solution strings so that a pair of offspring solutions is produced, as shown in Figure 3.3.

Parent 1	1	0	1	0	1
Parent 2	1	0	0	1	0

crossover point is the thick line

Offspring1	1	0	0	1	0
Offspring2	1	0	1	0	1

Figure 3.3: Single-point crossover [57].

2. Two-way/-point crossover [60]

This type of crossover technique makes use of two crossover places which are randomly selected. Bits of information between the two positions are transferred so that two offspring solutions are produced. Figure 3.4 shows how two-point crossover is implemented in GAs.

Parent 1	1	0	1	0	0
Parent 2	1	1	0	1	1

2 crossover points are chosen

Offspring1	1	1	0	0	0
Offspring2	1	0	1	1	1

Figure 3.4: Two-point crossover [61].

3. Uniform crossover

The third type of crossover uniformly selects positions [62]. The bits are transferred at coordinate pairs which have identical probability, as shown in Figure 3.5.

Parent 1	1	0	1	0	0
Parent 2	1	1	0	1	1
Offspring1	1	1	0	1	1
Offspring2	1	0	1	0	0

Figure 3.5: Uniform crossover [60].

3.3.3 Mutation

The procedure requires randomly altering information in order to create a new modified solution. It ensures that search diversification for the population is maintained. Mutation allows the search space to be enlarged because it provides diversity to the population. The technique is applied with care. It requires randomly selecting a solution and bit locations, followed by transferring ones to zeros, or zeros to ones, as shown in Figure 3.6. The process is used to prevent sticking in the local optima when applied carefully with selection and crossover.

Parent 1	1	0	1	0	0
Offspring	1	1	1	0	1

Figure 3.6: Mutation [57].

3.3.4 Fitness Evaluation

Every solution must be checked. The fitness function should be used for evaluating it. This will lead to a value which can determine the respective quality of a solution [57] [60] [63] [64]. Hence, the values of maximum fitness, minimum fitness, and average fitness for all members of a generation are calculated. If a predefined criterion of convergence has not been met so far, then the process moves to genetic operations dealing with selection and reproduction as well as crossover and mutation.

3.3.5 Control Parameters of Genetic Algorithms

A GA is an optimisation algorithm; therefore it requires using some parameters that help to make the algorithm work properly. These parameters are changed based on the problem or the guessed solution. Some parameters are to be kept constant while others are to be consistently changed by either increasing or reducing them. Specifically for GAs, the most common parameters include population size, iterations (maximum generation number), and types of genetic operations that fall under crossover and mutation. In [65], the two most important parameters are the population size and the number of iterations. Researchers can choose any of the listed crossover methods when working with genetic algorithms. The parameters must be carefully selected, because the GA's performance depends upon the value chosen and kind of the operators used. Typically, low population size, high crossover, and low probability of mutation are the parameters frequently used in research. However, it is stated in the literature that the size of the population is directly proportional to the difficulty level of the problem [59]. Generally, the crossover rates vary from 0.5 to 1, and the mutation rate is usually very small [57].

There is not a specific hard rule for determining the best value of operators [66]. Each problem is different and thus requires parameter tuning to find the best solution. Most researchers have reported that they used values which are suitable for their problems [65]. One of the widely used parameters comes from the De Jong function. In his book, De Jong [64] concluded that the generally optimal value for the population size should be between 50 and 100, 0.001 should be used for the mutation rate, and that a crossover value of 0.6 should be used [67] [36]. He recommended that these settings be used as the guidelines for GA problems. On the quest to prove whether these parameters were optimal, some researchers conducted an experiment by using a different test system. It was later concluded that the efficiency of De Jong's parameter settings outside his test suit was not clear. A mutation rate of 0.01 per bit in a 30 solution population plus a crossover rate of 0.95 per pair of solutions was later found to be the optimal setting [68]. The latter settings showed a distinct improvement in performance over the De Jong recommendations [37]. Owing to the fact that there are a variety of problem types, encoding patterns, and performance criteria, selection of the parameters is usually problem- and goal- specific. Some researchers have tried to treat this uncertainty by adapting the values of these operators during a GA's run [46].

3.3.6 Stopping Criteria for Genetic Algorithms

A GA without a convergence criterion will run forever without producing any result. The number of runs that algorithm can make depends on this convergence criterion. The GA is almost always stopped when the criterion is met. Widely used convergence criteria in GAs include the following [46] [39] [47] [57]:

- Using the number of iterations when running an algorithm. Also, stopping the algorithm when the solution does not show any improvement after a set number of iterations. Solutions generated by GAs are expected to keep yielding improved results over time but in some cases this does not happen. Setting this criterion causes the GA to converge. In some cases, after finding a satisfactory average value for the fitness function, the algorithm can be made to stop.

3.4 Application of Genetic Algorithms

3.4.1 General Uses of GAs

Due to the efficiency of the GA, it's widely used in several scientific research domains. There are several different applications of the GA; the following are areas where GAs are extensively being used [65] [69] [70] [71].

- **Modelling of social systems:** Such as how a group of people can cooperate together. This study stems from the analogy of how ants are able to work together towards a common goal. So with the use of GAs, new models of communication can be invented to help in the cooperative process.
- **Machine and robot learning:** In this field, GAs are used for predicting purposes and they are also being used in designing neural network structures.
- **Population genetics models:** This is the biomedical field where the genes of living things are studied and how they evolve is being noted. It involves studying how one population evolves into another.
- **Automated Programming:** In this field computer programs or software are being produced by using GAs. This involves studying the differing evolution of the software and hardware structure.
- **Immune system models:** GAs are being used to study how the immune system of mammals and humans work. The result will be to find an improved immune system that can fight against several diseases.

- **Interactions that happen between evolution and learning:** Studies are being conducted to find how the evolution of species affects other species in the ecosystem.
- **Economic models:** In the stock market, several algorithms that determine the “variations in the stock markets” [53] are mainly modelled by GAs. This helps to find the ups and downs of various stocks.
- **Optimisation:** Optimisation problems such as blackout restoration, timetable scheduling, and other forms of scheduling have been based on GAs.

3.4.2 GA Applications for Power Restoration

In power restoration problems, three different types of GAs can be used to restore a blackout. They are stand-alone, integrated, and fused systems [53].

Stand-alone systems allow the GAs to operate completely alone without incorporating another algorithm for the restoration process. This can be handled in one of three manners. The first option is to obtain data for input and run the GA. The second is to use the GA to assist other methodologies in arriving at a final answer. The process involves running the methodology to find an answer and then applying the GA to find the optimal answer. A second option is to run the GA from the first input numbers to arrive at a partial answer. The methodology is then run from the partial answer, thus helping to reach the final answer. Integrated systems are the other type. Here, GAs and the methodology are run in cycles, one following the other. As an example, input data is given to the GAs and crossover and mutation is done for a number of iterations. At the end of the mutation process, a new solution is produced and this is fed into the methodology. From this data, the methodology runs its algorithm and obtains the next solution. This answer is returned to the GA and the process repeats. The process continues until a final solution is reached. When training a neural network, this is the preferred type of process.

The last type is the fused systems. In these systems the GA runs together with the other methodology. There is no break between the end of one method and the start of the next. One can hardly make a distinction between the processes as they are fused together.

3.5 Advantages of GAs

A GA is considered a novel optimisation technique; therefore it is being used in a wide variety of ways.

Below are some of the advantages of GAs as stated in the literature. GAs are known to be all of the following [61] [46].

1. **Flexible:** The ability of genetic algorithms to hybridise with other heuristic algorithms makes them efficient in solving different types of mathematical problems.
2. **Adaptable:** When using a GA to optimise a problem there isn't much information needed due to the evolutionary nature of the algorithm. The algorithm is able to globally search within the problem space; therefore almost any type of objective requirements and constraints are acceptable.
3. **Robust:** Because of the ability of GAs to evolve, they are quite effective in using probability for global searches. Most heuristic search algorithms only do local searches within the problem area. Studies show that GAs are more efficient and robust in finding the best solutions and reducing the computational time than most heuristic algorithms [46].

3.6 Disadvantages of GAs

There are some disadvantages to the use of GA, even though they have been successfully used in solving complex problems. These disadvantages include [72]:

1. The limitation of using a GA in real-time applications due to the randomness of the solutions which sometimes cause the algorithm to take a long time before it converges. To overcome this disadvantage, it is critical to test GAs in several simulation models and environments before using them in real systems.
2. GAs producing some deceptive fitness functions which can give wrong information about the global optima and GAs getting stuck at the global optimum, making it difficult to solve some optimisation problems. Choosing a very bad fitness function can affect the convergence of the GA. This causes bad chromosome blocks to be generated. In a GA only good chromosomes can progress to the next phase. When there are only bad chromosomes, the algorithm gets stuck or the global best answer may not be found [61]. Hence hybridizing the genetic algorithm helps to solve this problem.

In conclusion, a GA is considered one of the best algorithms for power restoration due to its ability to restore power to the grid quickly and its other numerous advantages. Its disadvantages can be improved by hybridizing the genetic algorithm or using it together with other search algorithms. As can be seen, there is still ongoing research to improve the way this algorithm works in this field. The next chapter of this thesis discusses the basics of power flow analysis

by presenting the most commonly used Newton–Raphson (NR) power flow analysis found in the MATPOWER package [8].

Chapter 4

Power Flow Problem

This chapter presents an overview of the power flow problem and its importance when restoring power during a power blackout. In the field of power systems engineering, power flow is a difficult problem. When the angles and voltage magnitudes for a set of buses are needed, then the same information is available for another bus set [73] [74]. Any computer software that implements a solution procedure for the power flow process is collectively called a PFP. The solution for any power flow analysis includes all the angles and voltages of the buses, allowing us to calculate load levels, real and reactive generation, and real and reactive flows at circuits and buses, respectively [75]. The word “power” is interchangeable with the word “load”. Most researchers prefer to use the former wording, as “load” is not usually associated with “flow” [76].

Originally, the power flow analysis was used only for planning purposes. Engineers found it important to look at various network configurations which might be needed to handle a projected load in the future. After time, it was more of an operational issue [77]. Power grid operators were forced to continually check the status of the network in real time depending upon the circuit flows and voltage magnitudes. At present, the power flow analysis is the basic operation performed on power grids [19]. There are a multitude of highly advanced, commercial-grade power flow software packages which can address power system analysis. The optimised program is able to handle power flow calculations for thousands of buses in a power grid. Power system engineers who are knowledgeable about power flow and its formulation and procedures, which are associated, are widely respected, especially if they are experienced with power flow programs at the commercial level.

Determining the issues which are relevant to power flow requires an analysis of the power grid. The study of network operation will offer a better understanding when issues occur in the grid. The upcoming sections discuss power flow analysis (AC) as well as the use of the MATPOWER tool [8].

4.1 Power Flow Analysis

The procedures or analysis of a power flow (also known as a load flow) attempts to define the voltages and current flow (both real and reactive) within the power grid system when the load conditions are known [78]. Planning ahead to determine a course of action in various

hypothetical scenarios is the purpose of conducting load flow analyses. Questions such as whether operational lines in a power system can handle the necessary loads if one line must be taken down for maintenance are examples of such case studies [77].

A power flow problem is usually represented mathematically as a system which consists of non-linear algebraic equations and hence calls for an iterative technique for obtaining the solution. The Gauss–Seidel (GS) method and Newton–Raphson method are commonly used to get the power flow solution [79].

Figure 4.1 is a simple power grid where several buses are interconnected by lines (branches). These lines can be either transmission lines or power transformers [79]. The buses receive power from the connected buses. Each transmission line has its capacitance and impedance upon which it works. In some grids, to boost the load at certain critical times, static capacitors are used.

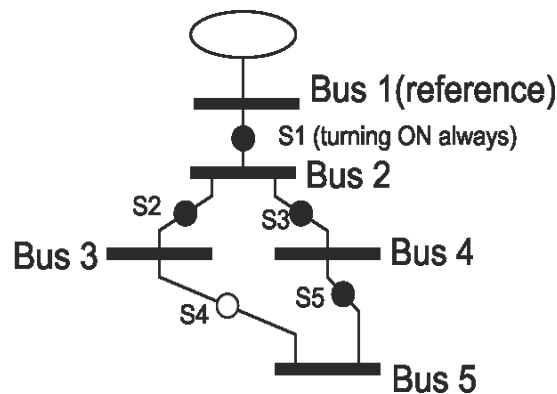


Figure 4.1: Simple power network.

4.2 Classification of Buses

Generally, in a typical power grid it is possible to classify buses in four different categories. They are generator, power-voltage, load, and switch buses. Table 4.1 shows the details of four types of buses, their most commonly used symbols of representation and the uses of these buses [80] [81].

Bus Type	Specified Quantities/ their symbols used in	Comments
Generator Bus	P v Q_{max}, Q_{min}	The voltage magnitude remains fixed so long as the needed Q levels fall between the maximum and minimum range of Q. When the required Q goes beyond the range, the bus is defined as a PQ

		bus. In this instance, Q is specified as the limit of Q which has been exceeded. P is set as zero for a synchronous condenser.
PV Bus	P $ v $	Constant voltage magnitudes at the PV bus are maintained regardless of the level of reactive power (Q) which is needed. For a synchronous condenser, the value of P is zero.
Swing Bus	δ $ v $	The swing bus is set as the baseline. The angle of “other” buses uses the swing bus angle as a measurement tool. If there are losses in the system, they are understood as being due to the swing bus. The reference bus is also responsible for holding the voltage of the system constant through manipulation of the power in the system.
PQ or Load Bus	P Q	A PQ bus or load bus is a non-generator type. Both active and reactive levels of power are defined with this bus. The phase angle and voltage level are not known.

Table 4.1: Types of buses in a power grid [82] [83] [84].

4.3 Basic Power Flow Problem

The calculation of various currents and voltages, which forms a big part of the power system, and its analyses are of critical importance [85]. The measurements are required in order to accurately create components which are part of the power system, including shunt elements, generators, transformers, and lines. These components must be able to handle the stresses of steady state loads with no danger of system damage. In order to maintain the economic efficiency of the system, any losses should be minimised [79] [86]. Any constraints in the system’s operation must be noted. The system must also allow for risks due to unstable

operational modes. To successfully manage all the complex voltages throughout the system, the state of the system must be monitored. By identifying the active and reactive power flows of the grid, the currents flowing through the grid can be calculated. Other relevant quantities can also be calculated from the known quantities. The equation set is formulated using a non-linear formula for the load flow $f(x, u, p) = 0$ [85],

where

f is a non-linear function;

x is a vector which consists of the states of the components of the grid;

u is the control outputs of the vector. These are usually the voltage outputs of the generators;

and p is also a vector that consists of the parameters of the comment section in Table 4.1, such as the line resistance.

Two commonly used methods are applied to solving problems related to load flow. These are the Gauss–Siedel and Newton–Raphson power flow analysis methods, each of which are root (iterative) finding solutions because they are designed to solve equations such as $f(x)=0$, which when solved produces the root of the function $f(x)$.

The term “iterative root” is applied to the process because a series of approximations are required in order to arrive at a solution. The method usually follows a series of steps beginning with a preliminary guess at a solution. This first guess will, except in unusual circumstances, be wrong. The first solution is updated with a new response with the purpose of moving the new solution nearer to the final solution. An important part of the iterative process is in how the update is determined. We must be able to determine that each update is nearer to the correct solution. This process results in a guaranteed solution, given that the procedure is repeated enough times.

Commercial-level load flow programs typically will offer several different procedures for calculating a solution for a power grid. Nearly all such software programs will utilise at least the NR method. The NR approach has been accepted as the industry standard [87]. The NR approach is preferred due to the fact that *convergence properties* are quite good if the initial solution that starts the process is good. Using the NR scheme for calculating load flow typically begins with a good initial guess.

The quality of the initial guess is often due to the fact that a previous procedure has already been conducted and the new set of conditions are very similar to the previous one. Perhaps only one circuit is changed or the load has been changed only slightly. The solution to the previous run may be utilised as the starting solution for the new conditions. Even without a

good starting-point solution, a guess may be very close. The accuracy of the solution comes because all quantities are solved as “per unit”. This standardisation using a per-unit voltage base means that under normal operating conditions the result will usually be near 1.0 per unit. The angles are not known from this information, but 1.0 per unit for the voltages and zero degrees at the angle is a satisfactory starting point [85].

In any “root” finding method, the process needs to converge at a certain amount of time. Determining whether the method will converge and how fast it will converge are the two convergence properties. Using NR means there are two factors which determine whether the method will result in convergence: these are the accuracy of the randomly generated solution and the function that is close to the final solution. If the guess is close and the function “smooths” out as it nears the correct solution, a convergence of NR will be found. Furthermore, the convergence will be quadratic, signifying that the solution’s accuracy is improved by at least two decimal places. For instance, when the result of a solution is 1.73737273, we begin with the first decimal, which is 0.700000000. Then the first iteration will result in 1.737xxxxxx, the second iteration equals 1.73737xxxx, the third iteration equals 1.7373727xx, and the final solution will be the fourth iteration [85] [84].

The NR load flow algorithm is the most powerful one in practice, but it does have a drawback because the Jacobian matrix terms require recalculation each time there is an iteration. This requires that the full range of linear equations be resolved during each iteration. Because the planning or operations study can require thousands of solutions, it is necessary to find methods for accelerating the process.

The GS method is out of the scope of this project and will not be discussed. However, readers may find more information about the GS method in the following references [85] and [88].

4.4 One Method of Solving the Load Flow Problem

4.4.1 Using AC Power Flow (PF) Analysis

The typical power flow study includes both reactive and active load flow measurements. Four variables for each node become the formulation components. These are the voltage angle, magnitude, and the active and reactive power injections. It is not possible to know the power losses ahead of time, since they are dependent upon the profile of the voltage and the active pattern of the power injection. Because other variables are interdependent, the problem becomes non-linear in nature. For the above reasons, the problem is commonly made linear [85]. A solution is reached by using a linearised form of iteration. At each iteration, there is

another estimation of the losses based on interactive variables. Today's tools for analysis of power systems use the NR algorithm as a base.

Usually the form of equations for power flow studies begins with a case of differential algebraic equations at steady state [85] [86]. It is then changed into a non-linear algebraic equation when a power flow analysis is done. This is written as

$$g(x) = 0$$

where g denotes a set consisting of algebraic equations in the grid. The formulations of the AC power flow is the current vector that is injected at the nodes of the network.

$I = [I_1, I_2, I_n]^T$ is a function of voltage vector $V = [V_1, V_2, V_n]^T$ and admittance matrix $Y = Y_{i,j}$:

$$I_i = Y \times V$$

Where $i = 1, 2, \dots, n$

which leads to writing the complex power injections (S) at nodes as:

$$S = V \times I - \text{AC power transmission between two nodes}$$

The voltage amplitude, the load, and the reactive and active powers are some of the main variables required when performing power flow studies. In [85], [89], [90], [91], and [74], readers can find more information on AC power flow analysis.

4.5 Overview of MATPOWER

MATPOWER is a powerful tool consisting of MATLAB "M-files" which allow users (mainly power system engineers or researchers) to obtain solutions for optimal power flow and power flow issues. It is easy to use and modifications can be made to it, too. One good thing about the tool is that it is open source [92]. MATPOWER gives top-tier performance. The tool was part of the features and developments under the PowerWeb project of Cornell University [93]. Its architecture permits the users to include additional variables, limitations, and cost to the standard OPF formulation without losing the specific structure required when pre-compiled solvers are necessary [92]. The package has inbuilt tools for power flow analysis, including the Newton–Raphson method and the Gauss–Seidel method. The MATPOWER program incorporates .m files or .mat files, which list and provide one MATLAB structure. The .m file can have a .txt output therefore, a standard text editor can be used to edit it [8]. It also contains the IEEE bus test systems which make it easy to run simulations in MATLAB. For a more detailed overview of MATPOWER, the reader should read the MATPOWER manual provided by the authors of the program [8].

In summary, this chapter presented the AC power flow analysis used in this project as well as the basic power flow problem. It also provided an overview of the MATPOWER package used in the simulation in MATLAB. A detailed overview of the NR method used for AC power flow analysis and its advantages were explained. The next chapter of this thesis discusses how power blackout was modelled in this project.

Chapter 5

Power System Modelling

This chapter covers the design and implementation of the restoration algorithm that was implemented in this project. Designing an algorithm to restore power when there is a blackout is a broad research area and several solutions do exist. When it comes to using a GA, several solutions spaces are possible. Due to this fact, it is assumed that the state of the power grid is known before applying a genetic algorithm in the solution space. When cascading effects cause failures in the grid, thus causing parts of the grid to be without power, the load needs to be restored [4]. The goal after any blackout is to restore the entire grid to its normal working condition. How to restore the grid is generally a complicated and difficult decision for the power system operator to make. This is due to the fact that different paths of restoring the network involve different restoration times. These restoration times could vary from short and acceptable times to unacceptable times. Moreover, the operator has to deal with many power system constraints at one time. Therefore, specific restoration procedures can be very helpful for the operator. The goal here is to find an optimised restoration path by using reserved switches in the power grid to restore the grid and then to pick up the rest of the loads within an acceptable period. This section will discuss the restoration method used in this thesis. The idea of the restoration is that there are reserved switches that can be used when there is a power blackout. Three objective functions will be taken into consideration: minimising the out-of-service area, lowering the cost by minimising the switching operation, and finally, minimising the losses. Switching in a power grid is expensive; therefore it is advised to have the minimum number of switches turned on during the restoration. “Minimum” means that the number of reserved switches used to restore the grid should be as small as possible.

5.1 Modelling of Blackout (Encoding)

This subsection presents how power blackout is modelled. The power grid is considered a network and therefore it can be represented in the form of a network topology [94]. Consider a working power system with five buses in Figure 5.1. The topology shows that one generator injects power into the whole system (network). The state of this network functions properly by distributing the required power to the subsequent buses and thus to the customers. This distribution complies to and obeys Kirchhoff’s law in the electrical engineering field. The

power flows from Bus 1 via S1 to Bus 2 and Bus 4 via S3 till it reaches Bus 5. Note that the state of the switches is represented in a binary numeral. When the value of the switch is equal to 1, it means the switch is closed and 0 represents an open switch. In Figure 5.1 the coloured circles show a closed switch while the uncoloured circle is an open switch. Note that S4 is known as a reserved switch. A reserved switch is the alternative path taken during the restoration of power to Bus 5.

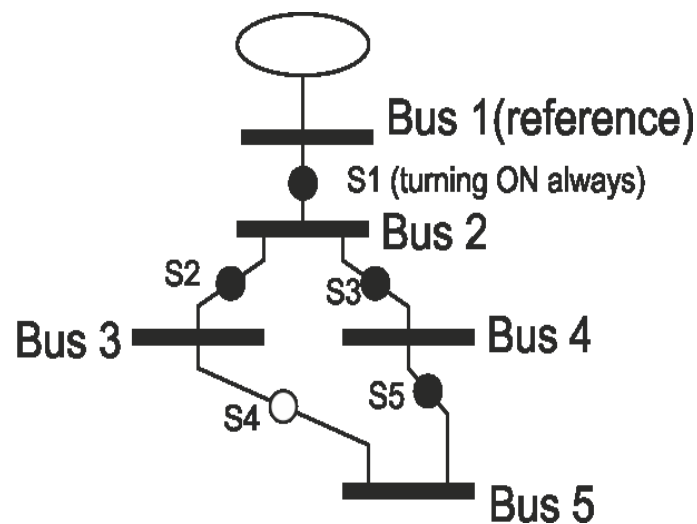


Figure 5.1: A simple power grid with five buses in working condition.

From the above, the network configuration can be determined by two set of parameters:

- the array of switch states and
- the array of bus loads.

Each branch is a connection between two buses, and this is essentially how the topology of the network is represented. Each bus also has its status as 1 being ON and 0 being OFF. This is a one-to-one correspondence with the switches. Thus the first branch in the case file is labelled as the first switch (S1), the second branch becomes the second switch (S2), and that continues till all the branches have been named.

Initial pre-fault switch configuration

The initial pre-fault switch configuration is represented as a string of 0s and 1s, where 1 means a closed switch and 0 an open switch. The length of the string is the same as the number of branches and there is a one-to-one correspondence, so the first element of the string corresponds to the first branch, etc.

Branches on which the fault occurred

The array of the IDs of the faulted branches (if there is more than one fault) represents the branches where the fault occurred. For example, if the array is (2, 5, 7), that means that the second, fifth, and seventh branches are faulted and hence set to 0.

- For a simple network, these arrays are as follows.

	S1	S2	S3	S4	S5
Switch state (binary)	ON (1)	ON (1)	ON (1)	OFF (0)	ON (1)

Table 5.1: Switch state of the working five-bus network.

	Bus1	Bus2	Bus3	Bus4	Bus5
Load value	0 (generator without load)	10 MW	15 MW	20 MW	25 MW
		Total load is $10+15+20+25 = 70$ MW Total power on all buses is 70 MW.			

Table 5.2: Total load value of the working five-bus network.

The “skeleton” diagram in Figure 5.2 shows the topology of the power grid in a normal working condition. As can be seen, S1 is directly linked to the generator source and should always be turned on. Power begins to flow from S1 to Bus 2, then through to S3 before reaching Bus 4. The last is S5 before reaching the final destination, which is Bus 5. S2 and S4 will be not used so all the power injected goes through from S1 to S3 and then to S5.

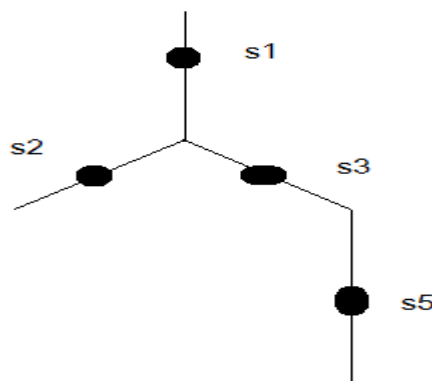


Figure 5.2: Skeleton network.

To model a blackout scenario, a switch or several switches in the network have to be turned off. For example in Figure 5.3, when switch S3 is turned off, the power flow is interrupted and power cannot be sent to the subsequent buses, thus there is no power supply to Bus 5.

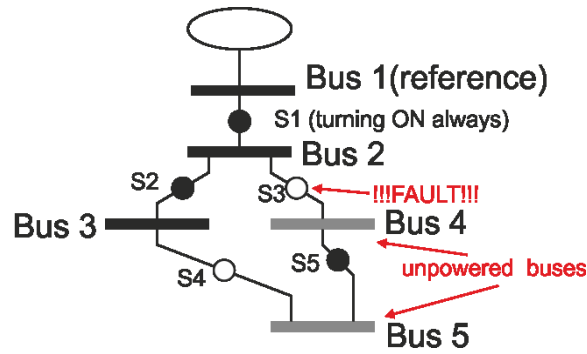


Figure 5.3: Model of blackout in a simple power grid.

In Tables 5.3 and 5.4, new network parameters are presented.

	S1	S2	S3	S4	S5
Switch state (binary)	ON (1)	ON (1)	OFF (0)	OFF (0)	ON (1)

Table 5.3: Switch state of the faulty five-bus network.

	Bus1	Bus2	Bus3	Bus4	Bus5
Load value	0 (generator without load)	10 MW	15 MW	0 MW	0 MW
Total load is 10+15+0+0 = 25 MW					

Table 5.4: Total load of the faulty five-bus network.

The faulty switch changes the network topology and the new skeleton of Figure 5.3 can be represented as shown in Figure 5.4.

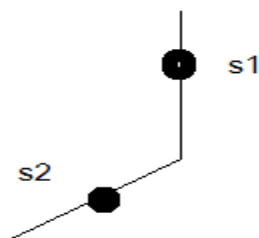


Figure 5.4: Skeleton diagram when there is a blackout.

The rest of the nodes in Figure 5.4 are disconnected from the main power source. As a result, there is no possibility to transfer power to the last node. In such an instance a blackout has occurred and the state of the grid changes from normal to extreme.

5.2 Modelling of Blackout Restoration

The power grid that is being used in most First World countries was built centuries ago. The design of the grid was made so that it would not completely fail. It takes energy to create electrical energy. A power plant cannot start without any energy. For example, coal has to be burnt to help generate energy to start the grid. Also for hydropower plants, dams must be filled with water, which manually turns the valves and in the process generates energy. This subsection describes how a power blackout is restored in this project.

When a blackout has occurred, the task of the operators is to restore power by finding an alternative route to get power to the affected areas or to the customers. Since Figure 5.1 had a reserved switch whose main task is to be used to restore power when there is a blackout, the new network topology when switch S4 is used is shown below.

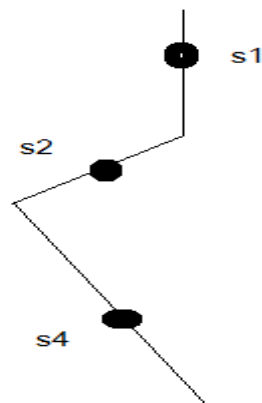


Figure 5.5: Alternative route with the reserved switch for power restoration.

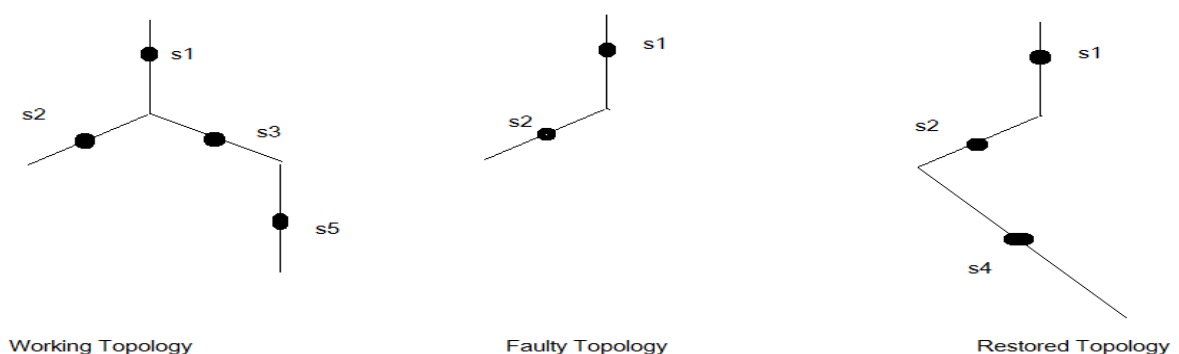


Figure 5.6: Schema of the restoration process.

Figure 5.6 shows a depiction of the restoration during a power blackout. The working topology is a representation of the power grid in its normal functioning state, the one which follows next is when there is fault, and the one on the far right is the restored network after

the reserve network has been used. The abovementioned descriptions are the representation of a small network with only one reserved switch. In a typical network topology there are several switches, known as reserved switches, which provide alternative routes when a fault occurs on a different switch. In any restoration planning, the constraints of the network must be adhered to.

5.3 Power Restoration Using a Genetic Algorithm

The goal of this research is to restore power to a power grid using a genetic algorithm. This subsection provides details of how the algorithm was used. In a genetic algorithm, a population of possible solutions for restoration of the network experiencing a blackout is created as a start point. As several solutions are possible, the fitness function is calculated for these possible solutions. In this project the best fitness value is one with a low number. Between any two good solutions, one can be better than the other, and using the fitness function helps determine that. This is because the objective function is to minimise the result; therefore the lower the fitness function, the better the solution. Once these possible solutions are chosen, two parents (two topologies) are chosen from this solution set and crossed in a process known as a crossover. The purpose of the crossover is to have a new set of solutions which is known to have “better” fitness. Later on, mutation is done and a new set of offspring is generated and later put into the new population. When the stopping criteria set for the algorithm is achieved, the algorithm stops and the best solution is then used in the next population.

5.3.1 Power Restoration in MATLAB Using a GA

In restoring power using a GA, it is assumed that the network operators have prior knowledge of the state of the grid. The GA takes as its input the initial array which represents the network, the desired size of the population, and the number of initial seeds. The population size shows the number of solutions to the network. The number of initial seeds is the number of items in the population which will be seeded from the input array that is the topology. The idea is that the solution for the network restoration problem is probably “near” the initial state of the switches. The rest of the population is generated at random in order to widen the search space. The GA starts with some topologies close to the initial pre-faulted network and some randomly generated topologies. At each iteration, it chooses the best ones (the topologies with the smallest cost function) and combines them to produce better solutions. It is not

practical to keep the entire network for each instance in the GA so the topologies are coded as binary strings (an array of 0s and 1s) when string size is the number of branches. Active branches are coded with 1 and inactive branches with 0. For example, the following topology shows five branches:

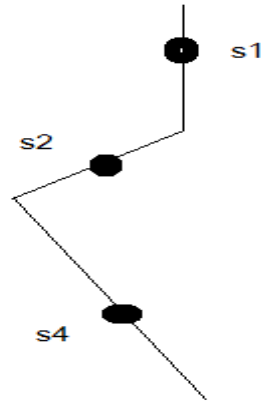


Figure 5.6: Structure of the grid using binary codes.

This network topology is coded as 11010 since branches 1, 2, and 4 are active and branches 3 and 5 are inactive. Each instance (an array representing topology) in the GA is called a genome. The GA is run at multiple iterations. At the start of each iteration, the cost function is calculated for each genome. The genomes with the highest cost function are selected as parents (to be combined to produce genomes for a new population).

5.3.2 Selection of Parents, Crossover, and New Mating Pool

In selecting the parents for the GA, tournament selection is used. Tournament selection is an algorithm for selection of the parents for crossover operation in genetic algorithms. It is more flexible than uniform and weighted selection algorithms.

The algorithm has two parameters:

1. Tournament size, k
2. Selection probability, p

Firstly, k parents are selected from the parent pool at random (uniform distribution). After that a tournament is held in order to select the parent. The tournament is done in the following way [95]:

The overall best genome from the tournament pool is selected with probability p .

The next best genome is chosen with a probability of $(1-p)p$ from the pool.

The third best genome is chosen with a probability of $(1 - P)^2p$ from the pool.

This means that the best genome can win the tournament with probability p . If it does not, the chance is given to second best to do it with the same probability, which leads to a $(1-p)p$ probability that the second genome will win it. The process is repeated until a genome wins the tournament.

Parameters can be adjusted to best suit the given problem. If $k = 1$, tournament selection becomes uniform selection. When $p = 1$, the best genome from the tournament pool is always selected.

Parameter p should always be greater than 0.5 in order to give better genomes a bigger chance to win. When p is larger, better genomes are being “pushed” towards selection more strongly. Also, if k is large, genomes that are not among the best will be seldom selected. A careful combination of these parameters will lead to a selection which will converge but still not get stuck in the local optimum.

The pseudocode can be written in the following way until all children are generated:

```

tournamentPool = Select k genomes from parents
pool at random
parent1 = best genome from tournamentPool with
probability p
second best from tournamentPool will have a
probability of  $(1-p)p$ 
third best from tournamentPool will have a
probability of  $(1-p)^2p$ 
tournamentPool = Select k genomes from parents
pool at random
parent2 = best genome from tournamentPool will
have a probability of p
second best from tournamentPool will have a
probability of  $(1-p)p$ 
third best from tournamentPool with
probability  $(1-p)^2p$ 
child = crossover(parent1,parent2)
perform mutation and evaluate the children
Take the child into the child population
Combine best child and parent to form a new
population

```

Figure 5.7: Pseudocode for tournament selection.

The process can be illustrated with the following example. Let’s say there is a parent pool of ten genomes with the following costs:

1. 10101 3
2. 00101 5
3. 11011 6
4. 01010 9
5. 10110 13

6. 10000 16
7. 00001 18
8. 11110 22
9. 11111 25
10. 00100 28

The tournament size is $k = 5$ and the selection probability is $p = 0.7$. The first five genomes are selected from the population at random. This leads to the following tournament pool:

1. 00101 5
2. 11011 6
3. 10000 16
4. 00001 18
5. 00100 28

A tournament is held between these genomes, where 00101 has a 0.7 chance to win as the best genome in the tournament pool, 11011 has a $0.3 \cdot 0.7 = 0.21$ chance to win, 10000 has a 0.063 chance to win, etc. A random number between 0 and 1 is generated to check if the first genome won and if it is less than p . If so, the first genome wins the tournament; if not, the same check is done on the second, etc. Let's say the random number generator outputs $0.867 > 0.7$. This means that genome 00101 failed to win and we now proceed to the second one. Now the random number outputs $0.701 > 0.7$, which means the second one failed to win it and we proceed to the third one, 10000. Let's say now random number generator outputs $0.543 < 0.7$, which means the third genome won it and we have parent1 = 10000.

The same process is now done for the second parent.

The first five random genomes are selected from the population.

1. 10101 3
2. 11011 6
3. 00001 18
4. 11111 25
5. 00100 28

Now the same tournament process is held with $p = 0.7$.

At first we check for 10101. Let's say the random number generator outputs $0.342 < 0.7$, which means 10101 won the tournament and we have parent2 = 10101. Now, crossover is done between these two parents to produce children for the next generation. When the new child is produced, it is put back into the mating pool to start a new tournament. Finally,

combining parents and newly generated children and having a tournament played among them helps to provide new offspring with better fitness.

5.3.3 Mutation

Mutation is a GA operation where bits are flipped randomly. Each bit has a probability of flipping and this probability is generally low, so $p < 1/n$, where n denotes the length of the string. At first, the iteration mutation probability starts with $1/n$ and decays with each iteration with a factor of 0.9. Initially the solution is far from a good solution and by adding a higher probability of mutation, the search space is spread out. As the iterations continue, it gets closer to the solution and mutation is no longer done with a high probability anymore.

For example, with a string of 10011, when mutation happens on bit 4, the resultant answer will be 10001.

A pseudocode of the genetic algorithm which has been described above is given Appendix 1.

5.4 Evaluation of the Fitness Function

The cost function is calculated for each genome at the given iteration. (The genome represents the network topology we are currently considering.) The final cost function is calculated as a linear combination of three cost functions and three constraints, thus,

$$F = w_1F_1 + w_2F_2 + w_3F_3$$

where F_1 represents the minimisation of the out-of-service area. It is the difference between the total load of the pre-faulted network and the network currently being considered. Total load is the sum of loads on each energised bus. The smaller this function is, the smaller the difference is between pre-faulted load and current load; thus more buses are properly energised. (One could argue that this is the most important part of the cost function as it tells us how much power has been restored.)

F_2 represents the number of switching operations to be performed in order to move from an initial network to the current network. It is the hamming distance between the initial configuration and current configuration. For example, there could be an initial configuration of (11011 – init configuration, 10101 – current configuration).

Then the hamming distance would be three since three bits need to be changed in order to change from initial to current configuration. This needs to be as small as possible since power needs to be restored via a small number of switching operations.

F3 represents the minimisation of losses. The losses are the sum of the losses on each branch since each branch will contribute to the power losses in the network. It also has to be small since the branches with small losses are a better option.

Furthermore, F1 and F3 are calculated with the help of the power flow in MATPOWER.

5.5 Weight Values

The values placed for the weight determine the importance of the objective function as well as the constraints. From the literature it is known that the objective function of any power restoration depends on the network at hand. In some networks, an operator would want to reduce the amount of losses while another operator will put more importance on the switching operations. In this project we assume that the operator has prior knowledge of the objective function required to restore power to the grid. In cases where the operator has no prior knowledge of the network, the objective function that is given the highest weight becomes the most important. When the aim of the restoration is to satisfy the need of the customer, then it is appropriate to put high weights on the minimisation of the out-of-service area. Cost of restoration is also important to the network operators; therefore it is put as the second highest weight. The least important weight for this project was minimisation of losses; therefore it is assigned the least value.

5.6 Data Normalisation of the Objective Function

The algorithm has different objective functions and these functions have different measurement units. Their value lies within different ranges. Objective functions having large, differing ranges of values will result in a significant problem; therefore functions having large values will have a larger effect on the objective function than those with small values. This does not necessarily reflect their relative importance for classification. Therefore data normalisation must be performed first to overcome the differences in units between each objective function. In this work we restrict all feature values to a range of [0, 1]. This scaling guarantees that all features will be normalised to the same range of values.

1. F1 is normalised by TL (initial total load) so now $F1 = (TL - \text{currentLoad})/TL$ and is thus a relative error.
2. The number of switching operations is normalised by N (number of switches) so now it is $\text{HammingDistance}(\text{initialSwitch}, \text{currentSwitch})/N$. So it represents the percentage of switch operations relative to the total number of switches.

3. Losses are normalised by the total generator output power, so $F3 = \text{sumOfLossesOnEachBranch}/\text{totalGeneratorPower}$. Now it represents a percentage of losses relative to the total output power.

5.7 Stopping Criteria

The stopping criteria in this project uses `eps` [96] (a MATLAB function known as “double precision”). The number of iterations are parameters of all numerical and optimisation algorithms and thus are GAs. At each iteration, the best genome of the current population and its cost function are calculated [97]. It is expected that the cost function of the best genome at iteration $i+1$ is smaller than the cost function of the genome at iteration i . The difference of $\text{Cost}(i+1) - \text{Cost}(i)$ is kept. As the algorithm gets closer to the true solution, this difference will become smaller. It will ideally become 0 when the exact solution is reached; however, in practice this is rarely the case. The algorithm stops when this difference is smaller than `eps`. Practical values of `eps` are generally in the range 0.1 or smaller but depend on the application range of the cost function.

Furthermore, the number of iterations is tied to `eps`. It is possible to have a case where the value of `eps` is never reached (so the difference between the costs at two subsequent iterations is never $< \text{eps}$), and as the algorithm cannot run indefinitely, thus it is bound by the number of iteration parameter. Therefore, regardless of the `eps` parameter, the algorithm will always run at most `NumberOfIterations` iterations. This parameter is set to a high value since it is only an upper bound and the algorithm will finish much sooner in most cases. So `NumberOfIterations` is set to a high number like 100 since its high value gives a better probability of reaching the local minima.

To summarise, `eps` is the criteria which asks if the distance between two iterations has decreased below it. This can be represented on a graph where the number of iterations is on the x-axis and objective or function on the y-axis. Let's say we have $y = f(x)$. `Eps` will check every iteration if $\text{abs}(f(x) - f(x-1)) < \text{eps}$, and if it is, it stops the algorithm.

In conclusion, this chapter gave a detailed description of how a GA was modelled in this project for power restoration. It should be noted that selecting the parameters for performing a GA is one of the most difficult tasks, especially with regard to the population size since a too small value or too big value has a negative impact on the solutions that are generated. The fitness function is also another important concept in GAs because it determines how the next chromosome will be reproduced. In this work, the smaller the fitness value, the better the

child produced because the aim is to reduce the cost of the restoration process. There is ongoing research to help find rules on how parameters for GAs can be selected. The next chapter presents the hybridisation technique used in combination with the conventional GA with the aim of improving the performance of the GA.

Chapter 6

Hybridisation of the Proposed GA with Local Search

This chapter presents the second search algorithm that was added to the conventional GA. One way of understanding the working principles behind the use of local search is explained by paraphrasing Di Gaspero [98]. Imagine an individual who has to climb to the top of Mount Afadjato [99] on a day which is foggy. The individual can only see what is close to him but cannot see the top of the mountain because of the foggy atmosphere. To reach the top of the mountain, the person must make decisions based on the slope of the mountain [98].

The mountain in this illustration is the “shape of the objective function” and choices can only be made based on the nearest slope. To reach the summit, the individual must continue to choose the upwards slope but even by making such a choice there is no assurance that he is at the top or midway [98].

6.1 Formulation of the Proposed Local Search

There are two conflicting performance objectives of a GA and they depend on how the best solution is exploited as well as how the solution space is explored to find a good solution. How the exploration and exploitation process can be combined optimally shows how powerful the GA is. One advantage of a local search algorithm is that it has the ability to improve the chances of the individuals so that they can enter into the next generation phase [100]. Any local optimisation algorithm that can be used to find a good balance between the exploration and exploitation process can help to make the genetic algorithm produce highly accurate solutions. One problem with a GA is that it can easily find a global optimum but faces difficulties when finding the local optimum in the solution space. The addition of a local search algorithm can improve the search performance when the combination is done efficiently.

Conventional GAs are known to be very slow in finding a local optimum because they do not make good use of the information available. As a result, a local search algorithm can be implemented to help remove this deficiency which happens in the local search phase of the GA [101]. When adding a local search to a GA, there is always a compromise between the time taken for the solution to converge and the accuracy of the solution [55]. In this project, it was realised that the restoration time increased but the better side was that full restoration was almost always achieved when the local search algorithm was used. In areas where it was

not possible, the algorithm restored the highest amount for a node. Figure 5.8 below shows the incorporation of local search into a standard GA.

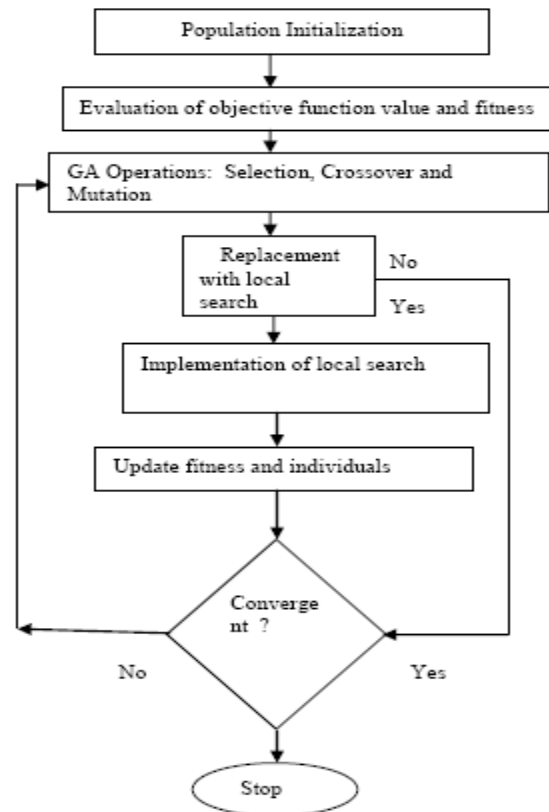
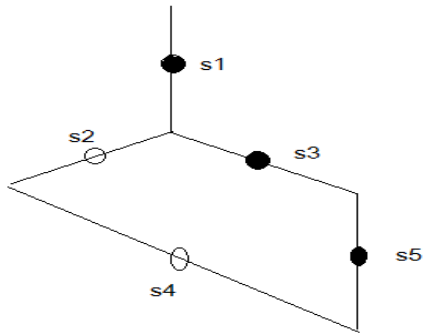


Figure 5.8: Incorporation of local search by Garg and Mittal (from [101]).

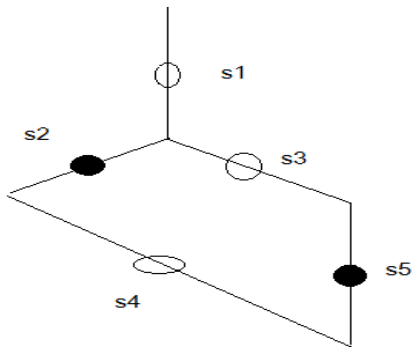
After crossover and mutation is done, a new population is ready to be evaluated and passed to the next iteration. In the standard GA, this new population will be input into the population for the next iteration. To hybridise the proposed GA, one intermediary step is added in order to enhance this population. For each genome of the population, its local neighbourhood is examined to try to find a better solution. Only neighbours up to distance 1 (hamming distance, i.e., the difference in bits) are examined due to high running time. This means that for each genome in the population, each other possible string is examined when it only differs on one branch. Then the initial genome is replaced with the one in its neighbourhood with the lowest cost value. If the population size is N , the number of switches is M , and the time needed to calculate cost is t , then the running time of this method is $O(MNt)$.

Let's say that after recombination and mutation we have the following populations and costs:

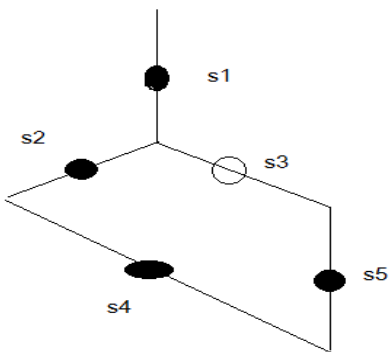
1. 10101 cost = 30



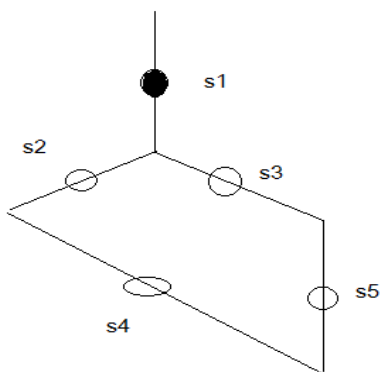
2. 01001 cost = 20



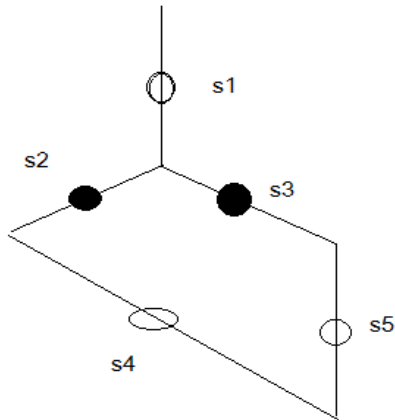
3. 11011 cost = 15



4. 10000 cost = 22

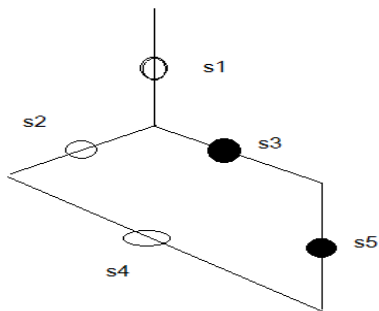


5. 01100 cost = 18

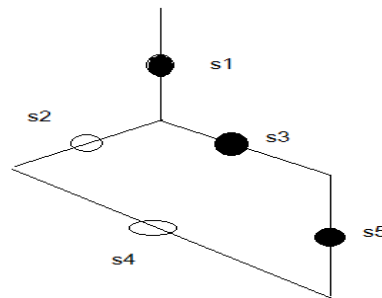


A local search for each of these genomes is run. The first local search is run for 10101. The neighbours of 10101 in the population list are those below because they differ in one bit, which is the same as having a hamming distance of 1:

- 00101
- 11101
- 10001
- 10111
- 10100



00101 (Neighbour 1)



10101 (Original genome)

Figure 5.9: Representation of the chosen genome and its neighbour.

A similar process as shown in Figure 5.9 is done for the rest of the neighbours. A threshold is also used to limit the number of neighbours for the original genome.

Let us now assume that the costs of the neighbours are listed below:

00101	22
11101	35
10001	25
10111	48

10100 37

The lowest among them is 00101, with a cost of 22. Since that cost is lower than the cost of the original genome, which is 30, the original genome 10101 is replaced with 00101.

Now another local search is run for the second genome from the population 01001 and all of its neighbours are examined:

11001 35

00001 23

01101 28

01011 55

01000 64

The lowest among them is 00001 with a value of 23. However it does not have a value lower than the original genome 01001, with a value of 20, so here the original genome is not replaced. It stays in the population since none of its neighbours are better. The same procedure is repeated for genomes 3, 4, and 5 from the population in the example. The introduction of local search thus helps to reduce the cost of the network and also helps to find the optimum genomes for the population.

6.2 Addition of Radiality Check

The layout of a power grid can be defined by two main topologies, namely, radial and mesh [102]. The radial grid is in the form of a tree-shaped topology in which closed loops do not exist. This means that you start on one bus and deliver power to the next without the possibility of finding the original bus unless you turn backwards [102]. It is stated in the literature that it is the simplest and cheapest topology for an electrical distribution grid, but if a line is disconnected, all the lines below the tree shape will also lose power. Such a topology is also not computationally expensive. When it comes to a GA, since there is a random generation of different topologies during the population generation process, having a radial network helps in encoding and decoding. With radiality in place, the final topology is the same as the initial topology but the final has no blackout area. The second topology is the mesh grid topology, where it is possible to find closed loops and the power is delivered through multiple lines, thus forming a mesh structure. If one line is malfunctioning or is cut, it is possible to reroute power from elsewhere while the damaged line is repaired [103]. The advantages are that a mesh topology is more reliable due to the several reroutes possible

[104] and has less losses, but this means a bigger investment is required in maintaining the other lines [105]. Figure 5.10 shows a simple representation of both network topologies.

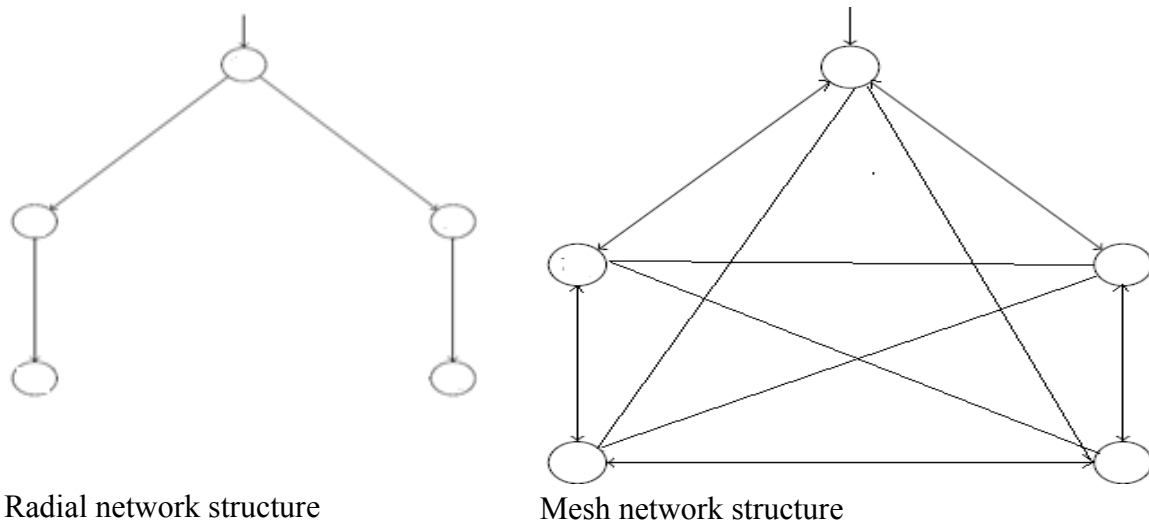


Figure 5.10: Radial and mesh network structure adapted from [106].

Furthermore, all the IEEE bus test systems, which are the commonly used test systems for a power flow analysis, have a mesh topology. Restoring power to a mesh network is expensive and computationally extensive due to the interconnection between the nodes because the flow calculations are difficult. As the main goal of this project is reducing cost, converting a mesh network to a radial network is more appropriate and non-radial networks are penalized by increasing its cost. Also, the conversion helps in the encoding of the network because when the restoration is complete, the string is decoded. After the decoding, the network that the power was restored to should be the same network which initially had the fault. Put in another way, when the final restoration of the restoration is achieved, it should be possible to trace/map it back to the initial network. Radial power networks can be provided with additional branches called “reserved branches”, which can be enabled as an alternative route in case of a failure. In this current project, the initial power network is considered to be radial with reserved branches. Once a new configuration of branches is found, it is essential to maintain that radiality. Hence a check must be done to see that there are no loops in the network, i.e., the network must remain radial. When a GA produces a string which is non-radial, it is penalised by increasing its cost. This penalty serves as a check so all networks maintain a radial structure during and after restoration. During the generation of the population, it is efficient to select only topologies that are radial. The radiality of these parents and offspring should be maintained throughout the process of using the GA. Having a radial network improves the performance of the GA. The main criterion is that *if the number*

of branches connecting the buses is one less than the number of buses, the network is said to be radial. One important condition to note here is that only the buses that are supplied are taken into consideration. For example, if a network has ten buses out of which only six are supplied, i.e., connected to a generator, then the number of switches that are connecting these buses must be only five. In other words, the network must form a minimum spanning tree between the supplied nodes.

In order to understand the terminologies, please refer to the diagram below. S1, S2..., etc. represents the switches/branches which can be ON or OFF. M1, M2..., etc. represents nodes/buses which can be supplied or not supplied. Some of these nodes can be generators. If a node is connected to a generator with a series of switches, then it is considered supplied because load will be sent to it.

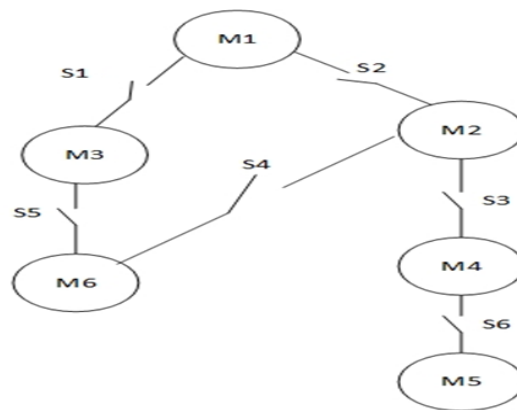


Figure 5.10: Radiality of a network.

6.2.1 Implementation

Consider the network of five buses below.

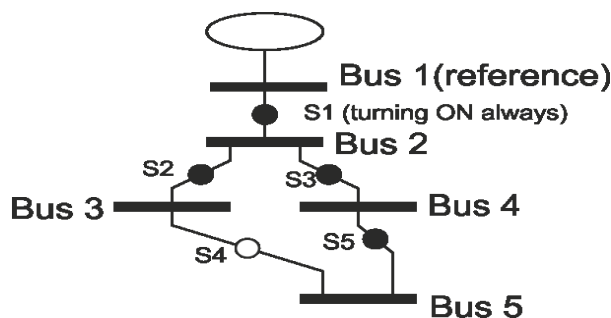


Figure 5.10: Sample network with blackout.

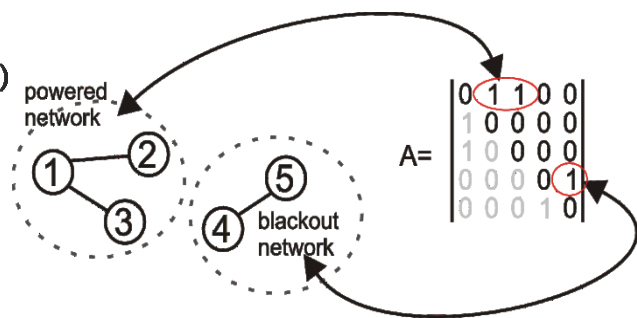


Figure 5.11: Representation of adjacent matrix.

Algorithm

As described above, first the number of supplied buses must be calculated and then the number of branches/switches connecting only these buses must be determined.

To find the number of supplied buses:

Start with the initial bus vector $M = \{M_1, M_2, \dots, M_n\}$. Where m_i is the state of the bus, a 1 indicates the corresponding bus is supplied by the generator and a 0 indicates a blackout. Start by setting all the generator buses to 1.

Consider the network adjacency matrix A . Now perform $C_j = M \times A_j$, where A_j is the column of the network adjacency matrix. If C_j is greater than 0, then the corresponding $M(j)$ is set to 1. M is changed to $M \parallel A_j$.

Once this operation is completed for every column of the adjacency matrix, it is repeated again starting with the newly calculated vector M . This iteration is terminated once two successive vectors M remain the same. At the end of the iteration, vector M indicates the supplied buses, i.e., the count of 1s in the vector gives the number of supplied buses.

Note: Only one diagonal half of the adjacency matrix must be considered. Either the top right of the principal diagonal or bottom left. This is because a connection between, say, Buses 1 and 2, is the same as a connection between Buses 2 and 1. The diagonals of an adjacent matrix is symmetrical hence the upper half is the same as the lower half.

To find the number of branches/switches:

The number of branches is simply obtained by counting the number of 1s in the adjacency matrix. Please see the above note on adjacency matrices.

However, the branches connecting the unsupplied/blackout buses must be ignored. This is accomplished by simply performing $A_j = A_j \times M$.

This gives a new adjacency matrix with the branches involving blackout nodes being removed.

Now we have vector M and the new adjacency matrix. The number of 1s in both are counted. The number of 1s in the newly obtained adjacency matrix must be one less than the number of 1s in vector M . Then the network is radial. An example of the implementation is given in Appendix 2.

In conclusion, Chapter 6 presented the detailed overview of how the GA was hybridised by using a local search. Local search helps to fine tune the selection process and reduces the cost even further since the selection is done twice. During the second process, the fitness function that is used in the next generation is less than or equal to the original. Also, the addition of the local search helps the objective functions to converge quickly. Even though radiality

helps in keeping the structure of network it can have some implications when converting the mesh grid to a radial grid. For example, in some cases, there is a tendency to increase losses every time a connection line between two buses is removed even though the aim was to remove a connection with less losses. Also, when a line is removed, it can cause a voltage drop between the buses. All of this should be taken into consideration when applying any radiality algorithm on a mesh topology. Chapter 7 of this thesis discusses the results of the hybrid GA. An extensive test was performed on the IEEE14 bus system where the objective function of different GAs used in this project is discussed.

Chapter 7

Simulation Results and Discussion

To know the efficiency of any algorithm, solutions to the problem are to be first produced and later analysed. Therefore, this chapter presents the simulation results of the hybrid genetic algorithm for the solution to blackout restoration problem. For the purpose of showing the results, each different type of GA is applied separately. In this work, the GA was tested on these bus systems: IEEE14, IEEE30, and the remodelled IEEE118. The three different power grids have differences in the number of buses, switches, and generators. The higher the number of branches in the network, the more computation power is required by the algorithm to solve the power flow problem. Bus 1 is the black-start bus, which should not be turned off as it is considered the main generator for the distribution grid. A comparison is made between a GA without local and radiality check and a GA which includes local search and radiality. Each test system is different; therefore, there is the need to vary the parameters of the algorithm when running the test for the different bus systems. This means that with the population size and the number of iterations, the penalties are varied for IEEE14, IEEE30, and IEEE118. Without having optimal parameters, the GA will not converge. Two types of blackouts are used to conduct this comparison. The first one is a single-fault blackout and the second one is a multi-fault blackout. After running each test, the objective functions for the different GAs are compared. Also due to the time limitation, only the result of the smallest system is presented in this results section. For the other systems, a summary of the results is provided. In analysing the results, these were the benchmarks in assessing which GA was the best during restoration. One of the important things during a restoration is to find out where there was a full restoration or a partial restoration. In a full restoration, first and foremost, the whole grid is connected, which means power was supplied to the buses that needed them and secondly, the total load that was lost during the restoration is fully regained. This means that when the grid is functioning normally and its load is 189 MW, when a blackout occurs and let's say 40 MW is lost, the grid is operating at 149 MW while the blackout is still in place. It is expected that during any restoration process, the whole grid is reconnected and that the total load is restored. In situations where the total load is not restored, then that is called a partial restoration. In a real-world situation, several factors are possible, such as the location of the affected area or the type of fault that occurred. In the case of this project, it was realised that the type of switch that caused a blackout can be the determining factor along

with the number of available reserved switches for restoration and the type of switch used for the restoration. When the case file is studied carefully, each of the branches (switches) has its unique configuration, thus making each of them different. As there are several reserved switches available, there are situations where the chosen reserved switch does not have the same configuration and working capacity as the fault switch. Thus it is not able to transfer the same amount of load to the affected area. In other cases, the reserved branch is better in terms of its configuration and working capacity. In the power grid, it is assumed that network operators already know the reserved switches that can be used for the restoration. Without knowledge of the reserved switches, it is not possible to restore power through the use of any switch.

After running the proposed hybrid algorithm, it is checked whether a full restoration was achieved (i.e., whether the whole grid was reconnected and there is a full load restoration or a high amount of it restored) or whether only part of the load was restored, i.e., a partial restoration. Once this information is known, the analysis of the objective function follows. The first objective function is the minimisation of the out-of-service area. Therefore after obtaining the results, this first objective function is compared. When the result is the same for the different types of GAs, the next objective function is then taken. This objective function is the minimisation of the switching operation for the network that was restored. Therefore the less number of switches used, the less the cost. If the result continues to be same, then the third objective function, which is the minimisation of losses, is analysed. When doing the comparison, if it is found that the first objective function of one GA has a smaller value compared to the others, it is considered the superior solution.

Selection of Reserved Switches

Figure 7.1 shows the reserved switches in a network containing thirteen buses and ten switches.

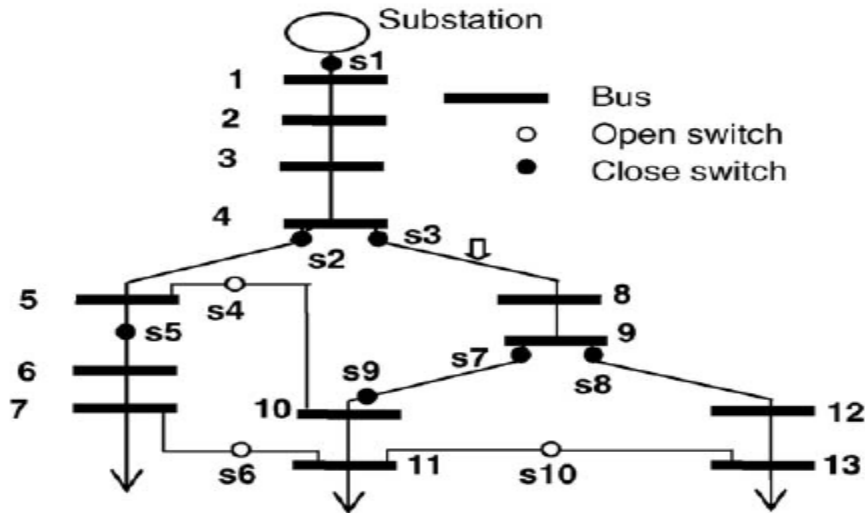


Figure 7.1: Network showing reserved switches S4, S6, and S10 (taken from [107]).

In a power restoration problem, it is assumed that the initial network will have radial topology. In other words, all the buses will be connected by the minimum required number of branches. If there are n buses in the network, there will be exactly $n-1$ branches connecting them. Such a network is also called a spanning tree. Any additional branches apart from the minimum required branches will be termed “reserved branches”. These reserved branches will be used when there is failure in any part of the network.

Description of IEEE14 Bus System

The IEEE14 bus system is mostly used in a power restoration problem, as was found in the literature. In this project, extensive testing was performed on it. This system has twenty branches (switches) and all the branches can be available for power restoration. But since the first branch is always ON, then there are only nineteen switches available for restoration. Thus said, there can be several choices for the reserved switches and several combinations. Testing all the combinations is a time-consuming process and thus is also a limitation of this project. Therefore the test presented here only shows part of test that was run for this project. In the simulation, single-fault and multiple-fault conditions were tested. “Single fault” means only one a single switch was turned off while the multiple uses two or more switches to model the faulty area.

During the test, the different parameters of the GA were varied. The test results show the average of ten executions and ten iterations but the number of iterations can also be increased. The higher the iteration number, the longer it takes for the algorithm to complete

its run. Also, the test was performed on an Intel Core i5 processor with a Windows 8.1 operating system and 8 GB of RAM.

7.1 Parameters for Fine Tuning the Hybrid GA

In any GA, there are a number of generations made in order to solve the problem. Within the generation, the fittest members are chosen and propagated to the next process. Each generation consists of strings of 1s or 0s and in each generation there are individuals that represent some points in a search space of the GA or are a possible solution/candidate that can go through the evolution process. This subsection presents the parameters chosen for this project. In the literature it was found that there is no standard benchmark for selecting the different parameters of a GA. In [108] it is known that the parameters chosen for any genetic algorithm will have an influence on its performance; therefore the parameters need to be carefully investigated. The goal should always be to find the optimal parameter values for the problem. The optimal values for the parameters depend mainly on 1) the problem and 2) and the computational time that will be spent in solving the problem [108]. There is a population size parameter which initially it is best to select a high population for and then later decrease the value. Having a big population size always broadens the solution space by making it stochastic, thus it can lead to the GA not finding a solution. Later it is advisable to reduce the search space so a local optima can be found. The role of the population is to hold possible solutions for power restoration. In the population, each individual is a possible solution that will help restore power to the grid. And within the population, some individuals are better than others when their fitness function is analysed. Others are quickly discarded because they do not provide any acceptable solution. Once the good individuals have been selected, two of them are chosen and they form the parents. There is mating done between the parents to produce a child during crossover. Then mutation is done. Once the stopping criteria are met, the algorithm stops. A more detailed explanation was provided in Chapter 3. A good population size gives a good solution space for the GA. The number of iterations tells us the number of times the algorithm is run before it stops.

	Meaning	Value
Populationsize	The population size	100
numParents	Number of parents to be selected from the ranked population	20

numInitSeed	Number of initial seeds constructed from parents	10
numTransferredParents	Number of parents to be transferred to the next generation	10
numIterations	Number of iterations	100
mutationProb	Initial probability of mutation	$3.0/\text{length}(\text{initSwitchArray})$

Table 7.1: List of parameters that should be tuned when using GA. The numbers in the value section should be changed based on the network.

7.1.1 Rules for Parameter Selection in This Project

As choosing parameters is time consuming and requires a lot of changes, after tuning, these were found to run the algorithm efficiently when applied to the hybrid algorithm proposed in this project.

1. PopulationSize: Size of the population in each iteration. The larger this value is, the better the chances are for finding a good solution. However, very high values may increase running time. One should start with a value of 100–500 and increase it if one is not satisfied with the solution.
2. NumParents: Number of parents to be selected from the population. Empirically the best value lies around 20 to 30 percent of the populationSize, but it can be increased if one wants less optimal genomes to be included in the crossover.
3. NumTransferredParents: Number of best parents to be transferred to the next generation. The value should be around half of the numParents but never greater than numParents as stated in the literature.
4. NumInitSeeds: Number of genomes in the initial population to be created from the initial string (network topology similar to the initial one since the solution lies somewhere near). It should be around 10 percent of the populationSize.
5. NumIterations: Maximum number of iterations that a problem can run if it does not find good solutions. In most cases, the program will converge in a few number of iterations (10–15 or less) and this is the upper bound (so that we are certain the program will not go into an infinite loop). It is kept at 100 and should be increased only if the program fails to find a solution in this number of iterations.

6. MutationProb: The probability of a mutation. It represents the probability of a mutation for a specific genome. If a more randomly spread solutions is desired, this should be increased. It is represented as a portion of the string size, i.e., $3/\text{size}(\text{initString})$. The numerator should never be greater than size (initString) and should be kept low or the solutions will be very random. Lower values will assure faster convergence but higher chances to be stuck in a local optimum.

7.2 Assessment of the Proposed Hybrid Genetic Algorithm

This subsection discusses how the performance of the proposed hybrid algorithm is assessed. This is done by comparing the result of the GA to that of the hybrid GA. The goal of comparing two different genetic algorithms is to show that the hybrid version outperforms the conventional GA in the subsequent sections. In order to show the advantages of combining the GA with a local search and radiality check to the problem of the design of a restoration plan, a comparative case between the restoration algorithm and the traditional GA method is presented. The comparison between the restoration algorithm and the traditional GA uses the IEEE14 bus test system. The algorithm was also run on IEEE30 and IEEE118 bus systems to test its scalability and robustness. The comparison focuses on the three objective functions, the time required to restore the network, and the amount of restored load of the network. Whenever there is a blackout, a certain amount of load is lost. When there is full restoration, it is expected that the areas that lost power will be reconnected and that the load will be restored to its full amount. In some cases, when there is a full restoration of the network, only some amount of the load is restored. Thus in such cases the percentage of the load restored is used for the analysis. This is called partial restoration.

Depending on the type of GA used, there can be several steps involved when running the project. As this is a power restoration problem, the nine steps stated below were found to be useful for this project. There is the need to have the input, which is the topology of the power grid. This tells us how big the grid is. The size of the grid is directly proportional to the complexity of the restoration process. Thus when the grid is small, it is expected that the restoration process will take less time and vice versa. From the structure of the grid, the data of the components, such as the generators, lines, and transformers, are known. These data tell the operator how the power is being transferred from the source to the destination (usually the customer). The grid is tested to be certain that there are no faults and that the load calculations are correct. Once that is done, the parameters of the GA are set. These

parameters take into consideration the rule that was set for Section 7.1.1, which helps to run the GA efficiently. The constraints are also set for the project. The constraints help to show which solution fully satisfied the GA. Furthermore, a fault is then injected into switching off one or several switches. When this happens, a blackout is said to have occurred. The power flow analysis is run and during the process, the fitness function of the individual candidates in the population is calculated. The calculation is done internally by the MATPOWER suite during the simulation process. After completing the run, the program outputs the graph which shows the objective function and the number of iterations it took for the algorithm to converge. The other output results include the buses in the network which were not supplied during the restoration process, the number of reserved switches used in the restoration, and the total load of the network after the restoration process. By analysing these, the best GA was chosen as the restoration method for the grid. This process of running the algorithm is explained in the subsection below.

7.2.1 Steps in Testing the Hybrid Genetic Algorithm

Step 1: Input the IEEE14 bus test system data, which contains the generator, bus, line, transformer, and reactor data. This is the input of the power grid into the system. The information provided to the program is the type of IEEE test bus system (i.e., the topology) used and its respective data, such as generator input and the line data. In summary, this is the input needed for MATPOWER to recognise the type of power grid that is being used in the modelling.

Step 2: Put a number for the population size (pop size), crossover rate, and iterations. These are parameters which are to be optimised when using a GA. A population size is required before running a GA. The iteration count helps in determining when the algorithm needs to stop. A detailed explanation of the parameters can be found in Table 7.1.

Step 3: Initialise the number of reserved branches and the fault. The number of reserved branches tells the model which alternative switches can be used when restoring a network with a blackout. The faulty branches inject a problem into the network, thus changing its state from normal to critical. When there is a critical state it means there is a blackout and there is the need to restore the load back to the system.

Step 4: Set whether to use only a GA, a GA with a local search, a GA with a radiality check, or a genetic algorithm with a local search and radiality check. This is the part where the

selection of which GA to use is made as there are different algorithms called into play during the run-time of the project.

Step 5: Run a Newton–Raphson power flow. The load flow analysis is done by using the inbuilt analysis function of MATPOWER. In this case, an AC power analysis is being used and using the NR flow is one powerful technique for solving equations numerically. The solution time is fast and it converges with a reasonably small tolerance.

Step 6: Calculate the fitness value. This is also known as the objective function of the model. The objective of the thesis was to minimise the fitness function. The smaller the fitness function, the better it is.

Step 7: Check the constraints that are mentioned in Step 6. The constraints are of utmost importance; therefore when one constraint is violated, the solution is discarded.

Step 8: Check the final output of the objective function, the load of the grid during the blackout, and the load that was restored after using the algorithm to restore power to the grid. During this process the program shows the switch string which helps us to know which reserved switches were used during the restoration process.

Step 9: Plot the graph of the objective function against the number of iterations of the model. By comparing the different graphs produced by the algorithm, we are able to assess the correctness of the solution. The values of the objective function are inputted on the project window of MATLAB. The graph helps to show the minimisation of the objective functions and it also helps in the analysis of the results.

Step 10: Select the best GA that solved the problem by satisfying all the required objectives and restored power to the grid either fully or partially.

These ten steps helped in this research to fully test the proposed hybrid GA for blackout restoration. Depending on the project or task at hand, there can be fewer or more steps than what is provided. Steps that remain general for most projects are the setting of the parameters of the GA and using the power flow analysis when the project is modelled in MATLAB.

7.3 Two Types of Blackout Modelling

In this project, two types of power blackout scenarios, namely, single fault and multi fault were used. The input data required for this to be done are 1) the IEEE test system or any grid topology, 2) the number of reserved branches which can be used during the restoration process, 3) the faulty switch which causes a blackout to the power grid, 4) turning ON/OFF

the radiality function by using 1 (on) or 0 (off), and 5) finally, turning ON/OFF the local search function by using 1 (on) or 0 (off).

These are explained in the subsequent subsection.

7.3.1 How a Single-Fault Blackout Was Modelled in This Project

In this project the modelling of a single-fault blackout was done by disconnecting one branch (line) in the power network. Any branch excluding the connection to the main bus of the network can be disconnected. The reason being that when the main generator is cut off from the grid, there cannot be any restoration of power since there is no way electricity can be transferred to another part of the grid. So keeping the main bus connection always ON is a must for the algorithm to work. For the IEEE14 bus, one switch was disconnected. In this case, Switch 9 was selected as the faulty switch. The criterion for the selection of the faulty area is that when the switch is turned off it should interrupt the normal working condition of the grid and that the main generator is always connected to the grid. Therefore it doesn't matter which switch is chosen in this case. What needs to be taken into consideration is that the chosen bus must cut off power to some zone of the grid. Any branch can be turned off to model the blackout scenario, excluding the ones connected to the main generator as was explained above. In this project any candidate switch for a blackout was chosen and the algorithm was applied. Figure 7.1 is the layout for an IEEE14 bus test system that is commonly used in power simulation problems.

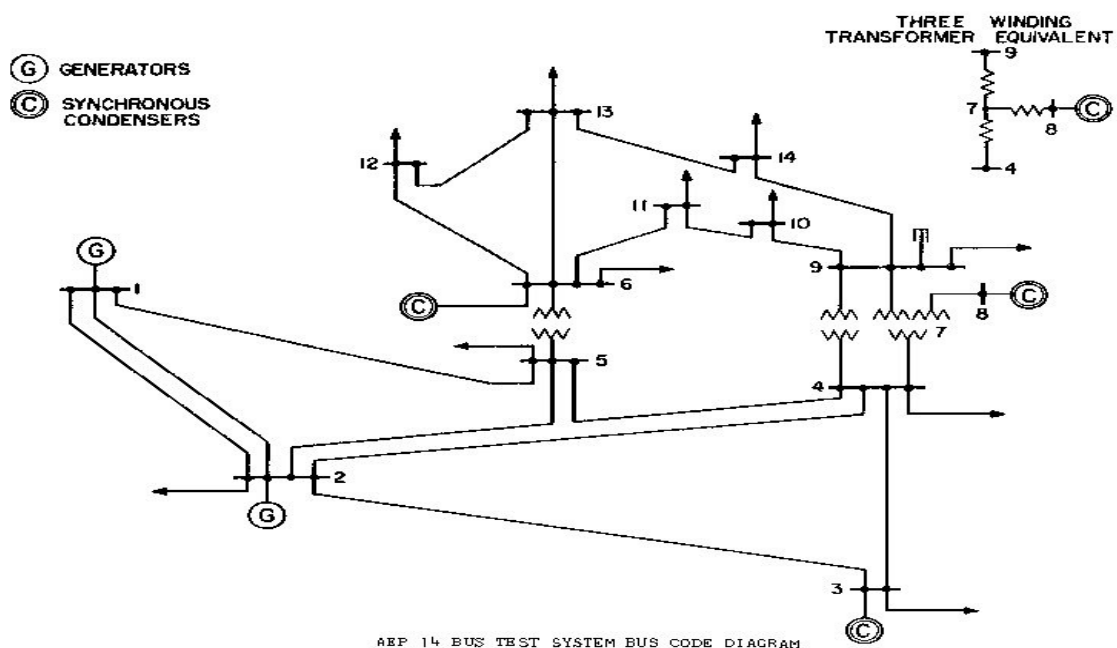


Figure 7.1: IEEE14 test system [9].

The IEEE14 bus was remodelled to have reserved branches (switches), which are alternative paths or switches that can be used during the restoration process. In order to model a blackout, part of the network has to be switched off. Refer to Appendix 3 to see how the remodelling was done. After the remodelling, a fault is injected into the power grid, for example, at the fourth branch, by turning this switch off which is known as switch number 4 (S4). It is possible to inject a fault by turning off any other branch in the grid when running the simulation.

Table 7.2 shows an example of the parameters for a single-fault blackout.

Reserved switches*	[S5, S6, S7, S15, S18, S19, S20]
Faulty switch**	[S4]
Weight***	[400 for F1, 200 for F2, 100 for F3]

Table 7.2: Single-fault blackout parameter settings.

**Refers to the switches or alternative path that can be used for restoring the power grid when there is a faulty switch.*

***Refers to the blackout area.*

****Refers to the weight that is put on the constraint. The effect of the weights is explained in Chapter 3 of this thesis.*

7.3.2 How a Multi-Fault Blackout Was Modelled in This Project

In order to model a multi-fault blackout, two or more switches can be turned off in the network. The greater the number of switches turned off, the higher the computational time to restore the lost load and to put the grid back to its normal working condition as well as the higher the computation of the power flow. For example, multiple faults can be injected into the power grid at the fourth, eleventh, twelfth, thirteenth, eighth, and ninth branches.

An example of the parameters for a multi-fault blackout is given below.

Reserved switches*	[S5, S6, S7, S15, S18, S19, S20]
Faulty switches**	[S4, S11, S12, S13, S8, S9]
Weight***	[700 for F1, 300 for F2, 50 for F3]

Table 7.3: Modelling of a multi-fault blackout.

**Refers to the switches or alternative path that can be used for restoring the power grid when there is a faulty switch.*

***Refers to the blackout area.*

****Refers to the weight that is put on the constraint. The effect of the weights is explained in Chapter 3 of this thesis.*

7.4 Metrics for the Evaluation of the Proposed GA

This subsection presents the metrics for the evaluation of the hybrid GA proposed in this thesis. The performance of the GA is later also presented. The evaluation of performance was done based on the following metrics [109].

1. **Scalability (robustness) of the hybrid GA:** Scalability refers to how the algorithm performs when used on different test systems. In this case, it means the different IEEE bus test systems and any other grid system available. The hybrid algorithm was run on the IEEE14, IEEE30, and IEEE118 bus test systems. In all cases, the proposed algorithm was about to find results for the restoration of the system back to its normal condition. In some cases for the multi-fault blackout, not all the buses of the network were powered but a greater percentage of the lost load was restored.
2. **Quality of the results:** In this metric we take into account how optimal the final result is by considering the objective functions as well as the constraints stated in the first chapter of the thesis. This takes into account the number of switches used in the restoration process. For example, in a restoration process where there are about ten reserved switches that can be used, using all of these switches will make the restoration very costly. An ideal solution would be to use a fewer number of switches.
3. **Computational effort** refers to how fast the computation was. Computation effort is equal to the time the algorithm is run in order to obtain the solution for any particular case. Time is essential in power restoration. Consumers are not used to having blackouts; therefore the need to restore power to the grid in a time-efficient manner is ideal. Secondly, as there are different test systems that the algorithm can be run on, the GA should be able to find a solution even for a large grid. It is expected that the restoration will be faster for a small network than a large network and that the difference will not be extremely big.

Metrics one and two are used to determine the efficiency of most metaheuristic algorithms and metric three deals with the time it takes for the restoration of a blackout. In this project, each of the different types of GAs were run approximately ten times with 10 iterations for the IEEE14, and the average of the solutions is recorded.

7.5 Robustness or Scalability of the Proposed Hybrid GA

To compare several forms of the same algorithm, different test problems are to be taken into consideration. In the electrical engineering field and especially with researchers in the power flow field, the widely used test system is the IEEE bus test system. This has been standardised by power engineers in the power grid community and is known as the benchmark for most papers written on power systems as can be found in the literature. An advantage of using a standard benchmark such as the IEEE bus test system is that one can easily make a comparison between the performance of the hybrid GA and existing GAs, thus helping to understand its performance. From the literature it was understood that the bigger the power grid, the longer it takes to restore power to it when there is a blackout and the more expensive the flow calculations are. Therefore it is expected that if the algorithm works in similar manner, then the biggest IEEE test system should have the longest run or should pose the most difficulties during power restoration.

Test System	Time Run (seconds)
IEEE14	23(+/-2)*
IEEE30	58(+/-5)
IEEE118	244(+/-5)

Table 7.4: Test using different benchmark systems (average of ten executions)**.

* *The mean plus or minus the standard deviation of the time*

***The number of executions is not the same as the number of iterations. It is possible to have 20 executions and 100 iterations for a larger network*

As can be seen from the results in Table 7.4, the algorithm was able to find solutions for different types of grids. The time run, which increases as the grid becomes bigger, was as expected since the time is directly proportional to the size of the power grid. This shows the scalability of the proposed hybrid GA. Also the type of switch that is turned OFF to model the blackout also affects the rate at which the restoration can be done.

7.6 Quality of the Solution (Accuracy)

This subsection describes the quality of the solution. To assess the results produced by the hybrid GA, the constraints set for the grid have to be met. Also the aim of the project is to minimise cost (objective function); therefore, the lower the cost, the better the solution is for the electricity company. The total cost is a sum total of the three objective functions

(weighted sum) and as a result, each individual function is checked and compared with the others.

7.6.1 Single-Fault Blackout

This subsection will discuss the results obtained when a single-fault blackout was modelled in the project. The results will include a comparison of the 1) minimisation of the out-of-service area as well as the minimisation of the switches and losses, 2) the weighted sum of the objective function, 3) the completion time for the restoration, and 4) the load that was restored to the network after the restoration procedure.

Henceforth, the following abbreviations will be used in representing the tested algorithm:

- The traditional genetic algorithm will be represented as GA.
- The GA with a radiality check will be represented as GA + RAD.
- The GA with a local search will be represented as GA + LS.
- The GA with a local search and radiality check will be represented as GA + LS + RAD.

Several simulation tests were done with IEEE14. A few of these results run by using this test system are presented in this section. Below is a test run on the IEEE14 bus test system when there was a single-fault blackout.

7.6.1.1 Comparison of the Minimisation of the Out-of-Service Area (F1)

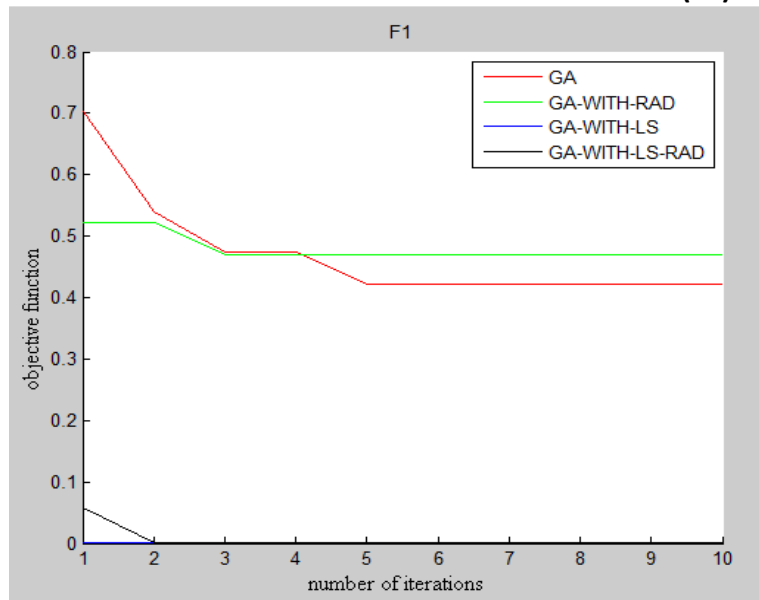


Figure 7.2: Graph of F1 for the IEEE14 bus system.

The figure above shows only ten iterations. After the tenth iteration, all four types of GAs had converged. The results below show the average number of ten executions of the algorithm. In

the above graph it is not possible to see the lines of the GA + LS and the GA + LS + RAD as they are almost close to zero. Their closeness to zero thus shows both algorithms were able to minimise the first objective functions. The GA + LS was the best algorithm based on its lowest F1 value. As F1 is the most important parameter among the three objective functions, GA + LS is superior in this category; hence it is to be monitored for the rest of the other objective functions as well as the parameters.

	GA	GA + RAD	GA + LS	GA + LS + RAD
F1 (MW)	0.42123	0.46949	0.00025	0.000742

Table 7.5: Values of F1 for the IEEE14 test system (average of ten executions).

7.6.1.2 Comparison of the Switching Operations (F2)

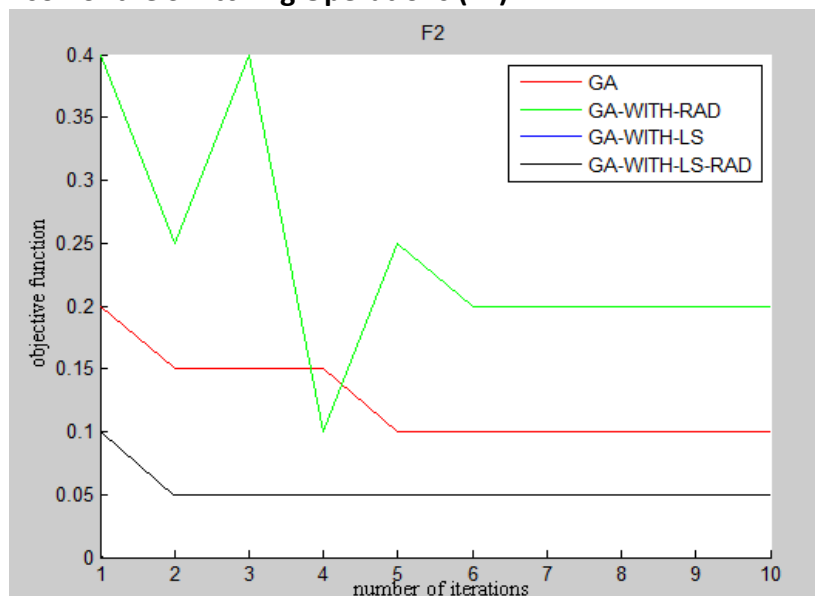


Figure 7.3: Graph of F2 for the IEEE14 bus system.

From the above graph, it can be seen that the GA + RAD has a zigzag shape. Its F2 value was fluctuating. It reduced during the second iteration, then increased again during the next. On the fourth iteration it had a sharp decline but later went up again before converging from the sixth iteration. During the different iterations the GA + RAD picked different reserved switches. The change in the switching operation affected the convergence of the algorithm. This shows that until the GA converges for a consecutive number of iterations. Selecting the GA as the better option would lead to a wrong result. Both the GA + LS and GA + LS + RAD converged at the second iteration till the end of the run. Therefore in this category, there was no superior F2 between the GA + LS and the GA + LS + RAD. The results were the same.

	GA	GA + RAD	GA + LS	GA + LS + RAD
F2 (MW)	0.100	0.200	0.050	0.050

Table 7.6: Values of F2 for the IEEE14 test system (average of ten executions).

7.6.1.3 Comparison of the Losses of the Network (F3)

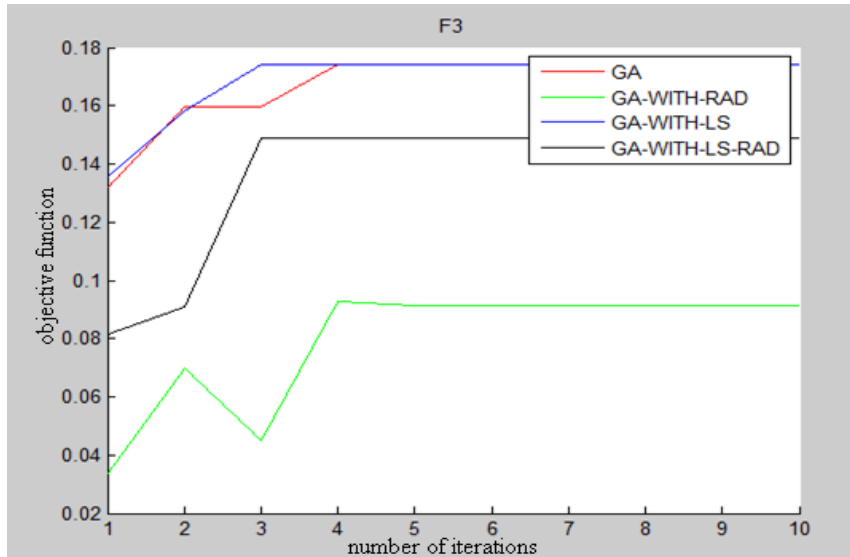


Figure 7.4: Graph of F3 for the IEEE14 bus system.

Losses were considered the least of the three objective functions and hence had the lowest weight. The algorithms all seem to have an increase in losses as the number of iterations increased, because the count of switched ON branches increases step-by-step from the initial string. The initial string has only part of the branches which provide power to the network before a fault. Each active branch provides some losses. When the network is totally turned off there are no losses. However, when we restore power in network, the losses are also restored. The GA + LS and GA performed the worst in this category. Another explanation is that at each iteration the algorithm makes a trade-off between the three objective functions. The outcome is that the first and second objective functions are always chosen or given priority in the selection of the new path for the restoration. This thus leads to a situation where both F1 and F2 are minimised while F3 keeps increasing because of its low importance in the GA process. A slight increase in the losses whereby F1 and F2 are drastically decreased will invariably lead to a lower-weighted sum of the objective function.

	GA	GA + RAD	GA + LS	GA + LS + RAD
F3 (MW)	0.174	0.091	0.174	0.148

Table 7.7: Values of F3 for the IEEE14 test system (average of ten executions).

In another run to test the weight on F3, W3 was given the highest weight among all three objective functions. The graph below shows that in this instance there was a reduction in the losses which is opposite of that which was recorded when the losses were given less weight value. The GA + LS and GA + LS + RAD had the same pattern on the graph and hence only one line appears on the graph. After the point of convergence, the GA, GA + LS, and GA + LS + RAD all had the same losses. By turning off a different set of switches in other executions, a different pattern was shown. This thus tells us that the type of switch turned off plays a big role in how many losses are incurred during the restoration process.

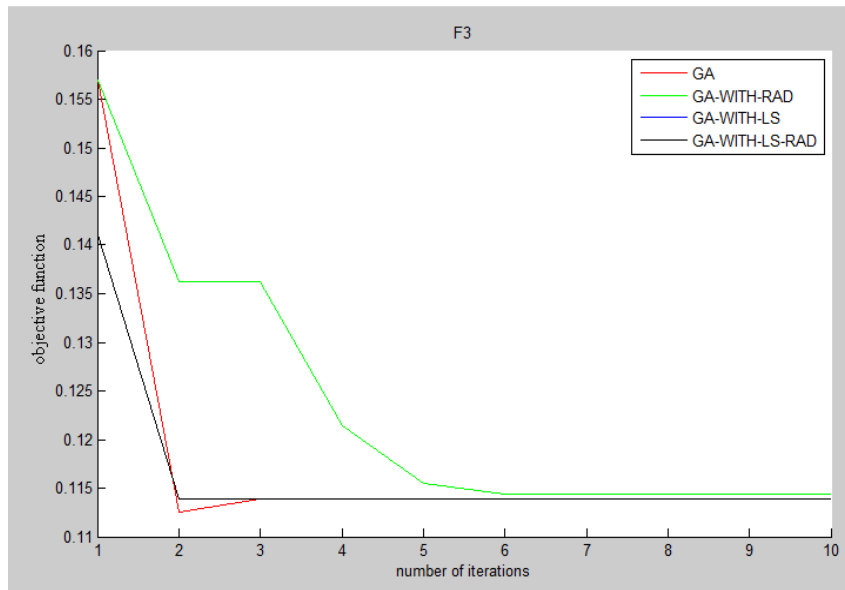


Figure 7.5: Graph of F3 for the IEEE14 bus system when W3 is given priority.

7.6.2 Multi-Fault Blackout

As was stated in the modelling section, when two or more switches are turned off, then we have a multi-fault blackout. Several switches were selected as the reserved branches (switches) of the above network. The result presented here has five switches turned OFF to model the blackout situation. The results of other multi-fault blackouts are presented in Tables 7.5–7.8 of this subsection.

The comparison of the results obtained followed the same format as that of the single-fault blackout. A good restoration algorithm will use less of these reserved switches in order to reduce the overall cost of the restoration as well as restore the maximum load even if all the buses cannot be powered. Reasons for not having a full restoration have been presented in a previous subsection of this thesis.

Multi-Fault for the IEEE30 Bus Test System

In modelling the single fault, IEEE14, which is a small power grid, was chosen. IEEE30 was chosen to model the multi-fault blackout in this section because there are more buses in this grid and therefore it is slightly bigger than the IEEE14 bus test system. This is more than twice that of IEEE14. Also, there is the option to switch off as many buses as possible, which means that there are more variations of turning off these switches at different sections of the grid. Also, it provides a wide range of reserved switches which tests the intelligence of each GA as to whether more switches will be chosen by the GA or vice versa. The results that were achieved shows that the algorithm was able to restore the grid to the IEEE30 test system. Lastly, having to see the result of another test system used gives detailed insight into the working process of the different GAs.

7.6.2.1 Comparison of the Minimisation of the Out-of-Service Area (F1)

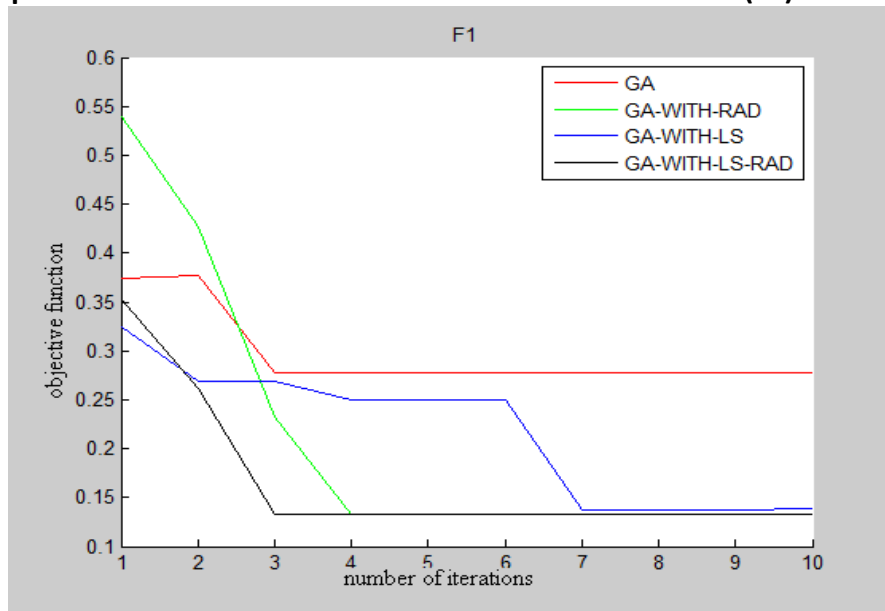


Figure 7.5: Graph of F1 for the IEEE30 bus system.

In the graph, F1 of the hybrid GA has the least value. From the first iteration, it started to decrease until converging on the third iteration. This shows that the hybrid algorithm achieved the aim of minimising F1, which is almost similar to the GA + RAD but the convergence of the hybrid GA was faster as it started at the third iteration. The GA + LS was the third best with the GA performing badly in comparison to the others. Both the GA and the hybrid GA converged. Therefore both algorithms chosen are superior in this test. The exact value of F1 can be seen in Table 7.5 which is the value that is seen in Figure 7.2.

	GA	GA + RAD	GA + LS	GA + LS + RAD
F1 (MW)	0.277	0.133	0.138	0.133

Table 7.8: Values of F1 for the IEEE30 test system (average of ten executions).

7.6.2.2 Comparison of the Switching Operations (F2)

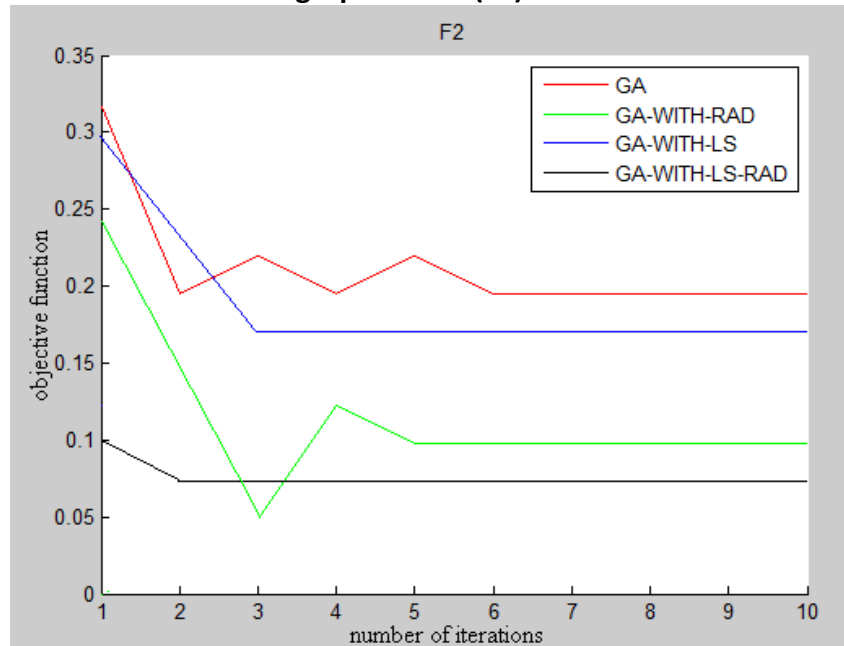


Figure 7.6: Graph of F2 for the IEEE30 bus system.

F2 refers to the number of switching operations that were done during the restoration process. The number of switches used in the restoration is directly proportional to the cost. Therefore the higher number of switches used, the more expensive the restoration process becomes. That is to say, if a restoration process uses five switches during the process, it is more expensive when compared to a restoration process which only uses four or less switches. The GA + LS + RAD did converge quicker than all the other types of GAs. The GA and GA + RAD during the different iterations picked more switches for the restoration. Specifically, on the fourth iteration, the GA + RAD chose a switch which had a lower electrical property than the faulty switch. From the graph, the cost of switching for the hybrid algorithm was lower compared to all other GAs. The exact value of F2 can be seen in Table 7.6 which is the value that is seen in Figure 7.3.

	GA	GA + RAD	GA + LS	GA + LS + RAD
F2 (MW)	0.195	0.097	0.170	0.073

Table 7.9: Values of F2 for the IEEE30 test system (average of ten executions).

7.6.2.3 Comparison of the Losses of the Network (F3)

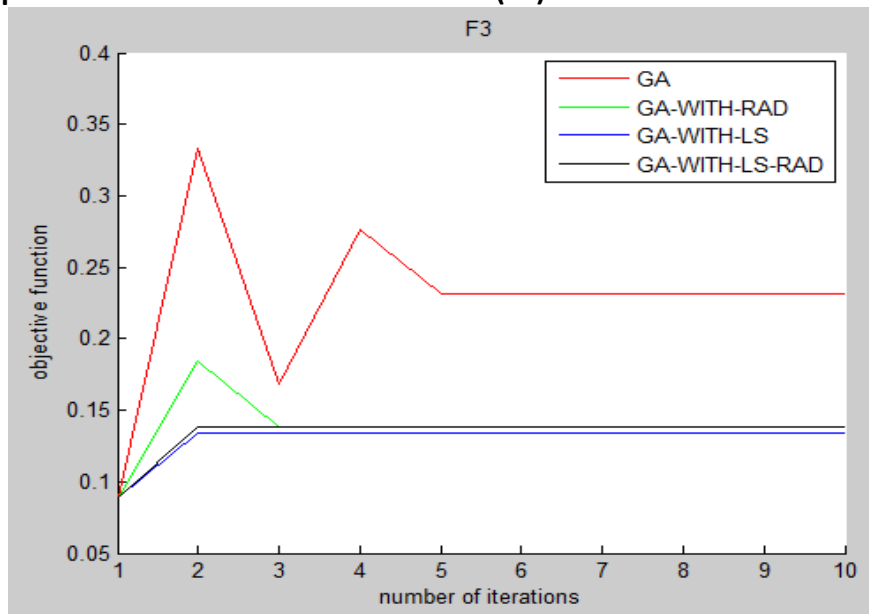


Figure 7.7: Graph of F1 for the IEEE30 bus system.

F3 refers to the losses that were incurred and the aim is to have the minimum amount of losses. But in setting the weight of the objective function, the least important objective was the losses. The losses increased because the count of switched ON branches increases step-by-step from the initial string. The initial string has only part of the branches which provide power to the network before a fault. Each active branch provides some losses. When the network is totally turned off, there are no losses. However, when we restore power in the network, the losses are also restored. In the graph, the GA + RAD had the minimum losses and the GA was the last one but the hybrid algorithm had the highest amount of losses. Even though the hybrid GA performed worse, this is acceptable in the final objective function because losses were placed as the least important objective by setting the weight for F3 to be the lowest. The exact values can be seen in Table 7.10.

	GA	GA + RAD	GA + LS	GA + LS + RAD
F2 (MW)	0.231	0.138	0.133	0.138

Table 7.10: Values of F3 for the IEEE30 test system (average of ten executions).

7.6.2.4 Comparison of the Load Restored after the Restoration

When analysing the results of the project, these should be taken into consideration. First of all, we need to know whether the part of the grid that lost power was first restored. This is followed by checking whether there was a full restoration or partial restoration. In a fully restored network, the areas that lost power due the disconnection of the branches have their full amount of load restored or the final load of the network is almost always equal to the

initial load before the blackout. In a partial restoration, only a portion of the network was able to get reconnected and also only a portion of the load that was lost was restored. This means that some part of the grid will not have the full load required to make it function normally but the power grid will be in a stable state. In the IEEE14 test with one switch turned OFF, for all the other GAs as well as the hybrid GA, the whole network was fully connected. The initial load of the network was 258.999 before the blackout was injected by turning off one switch. When the blackout occurred, the total load on the grid become 233.907 and that was when the restoration was done. After the restoration, by applying the various types of GAs, the results in Table 7.11 were collected for the restored load. It is to be noted that all the various parts of the grid were reconnected but there was a slight difference that was realised in terms of the restored load. For the proposed hybrid GA, besides having to minimise F1 and F2, it was also able to provide almost the full load required by the grid. It fell short of

Type of GA	Load Restored (MW)
GA	250.210
GA + RAD	256.350
GA + LS	258.290
GA + LS + RAD	258.940

Table 7.11: Comparison of the restored load after a single-fault IEEE14 blackout restoration (average of ten executions).

Thus it can be concluded that for a single fault for IEEE14, the proposed hybrid algorithm was superior to all the other GAs.

The result of the IEEE14 test with two switches turned off can be seen in Table 7.12. Once again, all four types of GAs were able to reconnect the areas that were disconnected. The initial load of the network was 258.999 before the blackout was injected by turning off two switches. When the blackout occurred, the total load on the grid became 219.400 before the restoration was done. The GA + LS performed better than all the other GAs, with GA + RAD restoring the least load.

Type of GA	Load Restored (MW)
GA	235.700
GA + RAD	249.800
GA + LS	258.100
GA + LS + RAD	253.400

Table 7.12: Comparison of the restored load after a two-fault IEEE14 blackout restoration (average of ten executions).

Table 7.13 shows the load values received from the IEEE30 network. The initial load of the IEEE30 network was 289.199 before the blackout was injected by turning off two switches. When the blackout occurred, the total load on the grid became 230.600. After the restoration, by applying the various types of GAs, the following results of the restored load were recorded, taking into consideration the fact that the whole grid was fully reconnected (a fully restored grid). The hybrid GA restored more load to the network than the other three types of GAs used in this project.

Type of GA	Load Restored (MW)
GA	256.613
GA + RAD	276.945
GA + LS	273.037
GA + LS + RAD	284.097

Table 7.13: Comparison of the restored load after a two-fault IEEE30 blackout restoration (average of ten executions).

The initial load of the IEEE30 network was 289.199 before the blackout was injected by turning off five switches. When the blackout occurred, the total load on the grid became 102.300, which means more than 50 percent of the load was lost in this blackout modelling. After the restoration, by applying the various types of GAs, the following results of the restored load were recorded. There was also full restoration in terms of the buses that lost power; they were powered back on. The hybrid GA restored more load to the network than the other three types of GAs used in this project.

Type of GA	Load Restored (MW)
GA	225.700
GA + RAD	230.800
GA + LS	252.900
GA + LS + RAD	239.400

Table 7.14: Comparison of the restored load after a five-fault IEEE30 blackout restoration (average of ten executions).

7.6.3 Computational Effort of the Solution

This subsection is about the computational effort of the algorithm which is measured in terms of time for this blackout power restoration project. It was not measured in CPU time but

rather in restoration time. In any restoration process, the time of restoration as well as the restored load after the restoration process is crucial. It was realised that when the local search was implemented together with the conventional GA, the time of computation was increased due to the additional step that was added to the GA process. The trade-off was that the quality of the solution was improved, which can be seen in the minimised value of F1 and F2 and the improved amount of load that was restored to the network.

Restoration Time for the IEEE14 Test System

Type of Genetic Algorithm	Time of Restoration (seconds)
GA	13.654+/- 3
GA + RAD	12.210+/- 1
GA + LS	27.190+/- 5
GA + LS + RAD	22.766+/- 5

Table 7.13: Time of restoration (average of ten executions).

Restoration Time for the IEEE30 Test System

Type of Genetic Algorithm	Time of Restoration (seconds)
GA	34.088+/-3
GA + RAD	36.372+/-2
GA + LS	84.820+/-1
GA + LS + RAD	81.504+/-7

Table 7.14: Time of restoration (average of ten executions).

As can be seen from the two restoration times for IEEE14 and IEEE30, it can be concluded that the time increases when the network becomes bigger. There is an approximately 1.5 time increase in the restoration time.

Test for Remodelled IEEE118

In this subsection, the results of the experiment carried out on the IEEE118 bus system is presented. A single fault was not modelled for this system because it is a big network, hence injecting a large fault makes much sense to help test how resilient the algorithm is. Single faults were done for the other networks as those are small in nature. With 118 buses, it makes sense to have multiple faults.

Input Parameters for Test 1

The maximum number of reserved branches in the remodelled IEEE118 test is sixty-nine. In this test, this was used. Eighteen branches were turned off to model the multi-fault situation.

When this grid is fully operational, the total load is 4190 MW. When the fault was injected, it began to operate at a lower load and then different types of GAs were applied. In this restoration, a full restoration was achieved. The time taken for the restoration was longer for this test. With a large number of faults present, the hybrid GA was superior and resilient when compared to the other algorithms as it had the lowest F1 and F2 values. However there were more losses which can be seen by looking at its F3 value.

Type of GA	F1	F2	F3	Fault on Load	Load Restored	Time
GA	0.0186	0.279	1.140	1597	Full load	322.573
GA + LS	1.7362	0.150	0.696	1597	Full load	465.148
GA + RAD	0.0251	0.263	0.866	1597	Full load	321.970
GA + LS + RAD	0.0043	0.118	1.001	1597	Full load	466.833

Table 7.15: Results for Test 1 (average of ten executions).

Input Parameters for Test 2

The number of reserved branches in the remodelled IEEE118 used for this test was sixty-six. Sixteen branches were turned off to model the multi-fault situation. When this grid is fully operational, the total load is 4191 MW. The load output when the grid is operational was found to be 4191. When the fault was injected, it began to operate at a lower load and then different types of GAs were applied. In this restoration, only a partial restoration was achieved as the full load could not be restored. The average time taken for the restoration was also a bit shorter compared to Test 1. The hybrid GA + LS was superior in that it had the lowest F1 value but in terms of load restored it had the lowest value. The values of F1 were the same as those for the GA + RAD, hence F2 of the next objective function was taken. The GA + LS + RAD was the second best in this run. There was only a partial restoration as the full load could not be restored to the network.

Type of GA	F1	F2	F3	Fault on Load	Load Restored	Time
GA	0.2973	0.5185	0.4227	1598	2945	154.624
GA + LS	0.0696	0.1975	0.6125	1598	2945	496.218
GA + RAD	0.1527	0.5061	0.6638	1598	3898	145.341
GA + LS + RAD	0.1527	0.1851	0.6963	1598	3898	494.769

Table 7.16: Results for Test 2 (average of ten executions).

Input Parameters for Test 3

The number of reserved branches in the remodelled IEEE118 used for this test was sixty-nine. Seven branches were turned off to model the multi-fault situation. When this grid is fully operational, the total load is 4191 MW. The load output when the grid is operational was found to be 4191. When the fault was injected, it began to operate at a lower load and then different types of GAs were applied. In this restoration, only a partial restoration was achieved. In this run, the GA + LS was superior as it had the lowest F1 value and restored the maximum amount of load. From the simulation results, it was realised that the hybrid GA + LS + RAD performed less efficiently as it had the second best F1 value when there were a fewer number of faults for the big network but it outperformed the other GAs when the fault was high. Thus it can be concluded that the hybrid GA is more resilient compared to the other GAs.

Type of GA	F1	F2	F3	Fault on Load	Load Restored	Time
GA	0.0361	0.040	1.128	3224	4074	254.624
GA + LS	0.0143	0.086	0.825	3224	4130	432.218
GA + RAD	0.0433	0.030	1.099	3224	4027	135.341
GA + LS + RAD	0.0321	0.052	1.116	3224	4021	412.769

Table 7.17: Results for Test 1 (average of ten executions).

The next subsection presents the discussion, rules or guidelines for selection which type of Genetic algorithm.

7.7 Discussion, Guidelines, and Rules for Using the Algorithms

Even though the performance of the algorithm was chosen as superior based on the objective functions, the restored load, and the time taken for restoration, several things should be taken into consideration. In this project it was assumed that there was only one type of switch which is the automated switch. In power grids there are mainly two types of switches, namely, automatic switches and manual switches, and the restoration time associated with these two are always different. Therefore minimisation of the number of switching operations should take into consideration the different timings of the different types of switches where applicable.

Furthermore, as was stated, the GA which is superior is the one that has minimised the out-of-service area the most. This means the GA that has the lowest F1 value, as it is the one on the first front with the highest weight or the highest importance. But it is possible to have a situation where the GA has the lowest F1 but a very high F2, value which means that the solution used a large number of switches during the restoration process. Therefore one way to keep from having a very high F2 value is to set a constraint on the maximum number of switches that can be used during the restoration and also penalizing the GA that tries to violate this. Any solution that violates this will be taken to the lower front of the solution.

Additionally, based on the results of the project, when the goal of any restoration is to have a quick time of restoration, then the best GA to choose would be the conventional GA. This is because the GA worked faster in the restoration, but it did not provide the quality required when assessed based on the objective function, thus leading to a poor solution.

To summarise, the advantages of the GA make it flexible to use in several complex problems. In this project it was used to solve an NP-hard problem which is power restoration. When working with GAs, one of the limitations is finding the optimal parameters (e.g., population size, crossover, and mutation probability) as well as the number of iterations. Any change in the parameters affects the results. In this project, the population size was first chosen to be very high and then it was reduced consistently. At each run, the result of the GA was examined to see if the required goal had been achieved. In some cases it got stuck at the global optima and had to be changed. A half-uniform crossover and a three-way crossover were tested but there wasn't a big difference in the final outcome of the GA. The half-uniform crossover was slightly better than the three-way crossover. So it was decided to use only the half uniform for all the executions. The mutation probability also had to be adjusted to find the optimal value. It is hard to know how different combinations can affect the results

since any change in the parameters also has some effect on the final results of the model. There is still on-going research with the aim of finding a good range of parameters when using GAs for power restoration.

Chapter 8

This chapter presents the conclusions from the MSc thesis that proposes a new hybrid genetic algorithm on solving the power restoration problem.

Conclusion and Future Work

The aim of this project was to restore power to a grid using a hybrid GA. By answering the sub-research questions and the main research questions we show how this was done:

1. *What is the current state of existing power grids?*

The current state of the grid is ageing, in addition to being overloaded. The electricity required by households and industry comes from main power suppliers. Since the power system is ageing and degrading, some parts must undergo regular maintenance so that the entire grid remains in top working conditions in order to provide the electricity that the consumers need. Besides the regular maintenance, they are also more likely to be operating close to the security boundaries because the number of people dependent on electric power is increasing while the grid structure is still the same. As such when an overload of the lines occur, the line is tripped in order to protect it from further damage by cutting electricity supply to that part of the grid which can lead to blackout (by cascading failures).

2. *What is a power blackout and how are these caused?*

When a major portion of, or even an entire, electrical power grid fails, it is called a blackout. This can be due to the malfunctioning of closed breakers which link the grid system or an external event such as a natural disaster like tornados breaking part of the grid. Furthermore, human errors, natural disasters like earthquakes, heavy rain falls and strong winds, a lack of evolving regulatory rules, power system deregulation and reduced investment in the power system industry also cause blackouts to occur in the power grid. Natural disasters like earthquakes and heavy rains can only be monitored but cannot be controlled. The monitoring of natural disasters is out of scope of this project, hence they were not discussed in the thesis. In addition, major failures of the grid can be caused by a series of cascading failures. Therefore, developing a reliable algorithm which restores power to the system after a power blackout is crucial in order to guarantee the reliability of the power supply.

3. *What are the steps that need to be taken in order to restore power to customers when there is a blackout on the power grid?*

There are two main types of power restoration methods namely “bottom up” and “top down” restoration method. Bottom up begins with restoration of the main network. Once that is

completed and transmission and substations are energised, the generators are resynchronised and the loads are picked up. Under this strategy, there is a parallel restoration of several areas. Another major strategy is the “top-down” approach. Here, the power system is sectionalized into a number of islands. In each island, the existence of at least one black-start generator is necessary. Once each section is energized, they are reconnected to form the whole grid and restore it to its original working condition.

4. *“How is hybrid genetic algorithm approach used to restore power in a smart grid”*

The hybrid GA proposed in this thesis uses local search to improve the performance of conventional GAs in power distribution. By using local search, the algorithm is able to find the best offspring among the parent population. By also using the local knowledge provided by the local search, the solution is improved and also individuals with better fitness functions are chosen to propagate to the next stage of the generation. The hybrid GA proposed in this thesis can help power operators restore power in real scenario of power blackout restoration. Furthermore it is up-to-date with the current power system models that can be found in the literature.

After answering the four main research questions, we can now answer the main research question of the thesis, which is stated below.

“How to efficiently restore power in a smart grid after such a power blackout by minimizing the number of switching operations, minimizing the out-of-service area, minimizing the losses, restoring a high percentage of the load thereby leading to a reduction of cost.”

Conventional GAs are able to solve the power restoration problem but they fail to provide quality solution. This means that in restoring power, GAs use as many reserved switches as possible and it takes a long time to converge to a solution. The proposed GA uses a radiality check and also local search to improve the quality of the solution. With the radiality check enabled in hybrid GAs, a smaller number of switches is chosen when restoring power to the grid and also by enhancing the search process of the genetic algorithm. Local search also helps to improve the fitness function because only the fittest are chosen and candidates with the lowest fitness function propagate to the next stage. By so doing, the first objective function is minimized. When tested on bigger networks, it was found to be very resilient because when a lot of fault was injected into the grid by turning off a lot switches, the hybrid GA was found to be superior in restoring the grid due to its resilience when compared to the other GAs. The hybrid algorithm was able to minimize the out of service area and the number of switches used in the restoration.

One of the problems that was encountered was that the algorithm wasn't able to minimize the losses. Even though the proposed hybrid GA finds a better quality solution, there was a computation time problem as the selection process of the best candidates has to be done twice during the search process. From the results of this thesis, using local search in addition to the proposed radiality check, the out of service area was minimized as well as the number of switching operations during power restoration which leads to a reduction in cost when power restoration is performed.

The following studies are suggested for future research in this field

- 1) Several different distributed power generators like solar power or wind power which are both renewal energies can be added to the system. With a lot of focus being put on the use of green energy, the effect of large-scale DG penetration can be an area of study.
- 2) In this project only automatic switches were used. Some power grid systems have manual switches which are operated manually by power operators. The effect of using manually operated switches can be studied.
- 3) Further work could be conducted to explore the effect of the several different types of crossover operators. In practice there are a number of crossover operators which all provide different ways of improving the genetic algorithm.
- 4) More research could be done to find ways to reduce the computational effort in terms of the time when using the hybrid GA.

Bibliography

- [1] U.S Department of energy, “U.S.- Canada Power System Outage Task Force,” Office of Electricity Delivery & Energy Reliability, Washington, DC 20585, 2006.
- [2] Energy.gov, “Blackout in the United States and Canada: Causes and recommendations,” Canada Power System Outage Task Force, Vancouver, 2004.
- [3] G. Andersson, P. Donalek, R. Farmer, N. Hatziargyriou, I. Kamwa, P. Kundur, N. Martins, J. Paserba, P. Pourbeik, J. Sanchez-Gasca, R. Schulz, A. Stankovic, C. Taylor and V. Vittal, “Causes of the 2003 Major Grid Blackouts in North America and Europe, and Recommended Means to Improve System Dynamic Performance,” *IEEE TRANSACTIONS ON POWER SYSTEMS*, vol. 20, no. 4, pp. 1922-1923, 2005.
- [4] Y. Koç, M. Warnier, P. Van Mieghem, R. E. Kooij and F. Brazier, “The Impact of the Topology on Cascading Failures in Electric Power Grids,” *Physics and Society*, 29 Nov 2013.
- [5] M. Bruch, V. Münch, M. Aichinger, M. Kuhn, M. Weymann and G. Schmid, “Power Blackout Risks,” CRO Forum, Munich, 2011.
- [6] M. Adibi, L. Fink, J. Giri, D. Kirschen, S. Shahidehpour and J. Zaborszky, “New approaches in power system restoration,” in *IEEE Trans. on Power Systems*, Vol. 7, 1992.
- [7] Y. Koç, T. Verma, N. A. Araujo and M. Warnier, “MATCASC: A tool to analyse cascading line overloads in power grids,” in *Proceedings of the IEEE International Workshop on Intelligent Energy Systems*, Piscataway, 2013.
- [8] R. Zimmerman and E. Murillo-Sanchez, Matpower 5.0b1 User’s Manual July 1, 2014 Power Systems Engineering Research Center (Pserc), Power Systems Engineering Research Center (Pserc), 2014.
- [9] University of Washington, “Power Systems Test Case Archive,” [Online]. Available: http://www.ee.washington.edu/research/pstca/pf14/pg_tca14bus.htm. [Accessed 2014 June 2].
- [10] L. Lindgren, “Automatic Power System Restoration Application of a Search Algorithm,” Department of MT and IEE 2009, Lund, 2009.

- [11] Y. Koc, M. Warnier, R. E. Kooij and F. T. Brazier, “An entropy-based metric to quantify the robustness of power grids against cascading failures,” *Elsevier*, vol. 59, p. 126–134, 2014.
- [12] Virtual lecture course, “Problem Solving by Search,” [Online]. Available: <http://www-g.eng.cam.ac.uk/mmg/teaching/artificialintelligence/nonflash/problemframenf.htm>. [Accessed 2014 November 3].
- [13] M. Zima, *Operation, Monitoring and Control Technology of Power Systems*, Swiss Federal Institute of Technology, 2007.
- [14] P. Schulz and W. Price, “Classification and Identification of Power System Emergencies,” *IEEE Trans. Power Apparatus and Systems*, Vols. vol. PAS-103, pp. 3471-3479, 1984.
- [15] Public Emergency and Safety, “Incident Analysis - Impacts on Critical,” Ontario, 2006.
- [16] Y. Koc, M. Warnier, R. E. Kooij and F. Brazier, “Structural vulnerability assessment of electric power grids,” in *Networking, Sensing and Control (ICNSC)*, Miami, 2014.
- [17] M. Adibi, P. Clelland, L. Fink, H. Happ, R. Kafka, J. Raine, D. Scheurer and J. Trefny, “Power System Restoration - A Task Force Report,” in *IEEE Transactions on Power Systems*, 1987.
- [18] Office of Electricity Delivery and Energy Reliability, “Large Power Transformers and the U.S. Electric Grid,” DOE / OE / ISER, Ontario, 2012.
- [19] M. Adibi and H. Fink, “Power System Restoration Planning,” in *IEEE Transactions on Power Systems*, MD, 1994.
- [20] M. Adibi, “Special Consideration in Power System Restoration. The second working group report,” in *IEEE Transactions on Power Systems, Vol 9*, 1994.
- [21] S. Borlase, “Smart Grid Technologies,” in *Smart Grids: Infrastructure, Technology, and Solutions*, NW, CRC Press, 2012, pp. 202-203.
- [22] UCTE Report, “Final Report of the Investigation Committee on the 28 September 2003 Blackout in Italy,” UCTE, Italy, 2004.
- [23] CERC Order in Petition No. 167, “Report on the grid disturbance on 30-31st July 2012,” <http://www.cercind.gov.in/>, New Delhi, 2012.
- [24] A. Kannampilly, “Blackout hits 300 million in northern India,” *phys.org*, Delhi, 2012.
- [25] Detroit Free Press, 2003.

- [26] C. Hyu and B. Yonio, "Blackout on Small Businesses," *Advanced Research in Economic and Management Sciences*, vol. 18, pp. 108-110, 2014.
- [27] M. Adibi, W. Alexander and P. Milanicz, "Energizing high and extra-high voltage lines during restoration," in *IEEE Transactions on Power Systems*, 1999.
- [28] M. Adibi and R. Kafka, "Power system restoration issues," in *IEEE Comput Appl Power*, 1991.
- [29] M. N. Omara, "Power System Blackout Restoration," in *INTERNATIONAL CONFERENCE ON COMMUNICATION, COMPUTER AND POWER*, Muscat, 2009.
- [30] F. Chu, J. Dobrowolski and K. Nodehi, "The uses of an operator training simulator for system restoration," in *IEEE Proceedings of the PICA Conference*, 1991.
- [31] E. Agneholm, "The restoration process following a major breakdown in a power system. Technical report," Department of Electric Power Engineering, Goteborg, 1996.
- [32] X. Li, D. Xu and Y. Zhang, "Distribution service restoration with DGs based on multiagent immune algorithm," in *2nd International Conference on Power Electronics and Intelligent Transportation System*, , 2009.
- [33] D. Lindenmeyer, W. Dommel and M. Adibi, "Power system restoration – a bibliographical survey," in *Electr Power Energy System*, 2001.
- [34] K. Salek, U. Spanel and G. Krost, "Flexible support for operators in restoring bulk power systems," in *Quality and Security of Electric Power Delivery Systems*, 2003.
- [35] Y. Y. Hsu, H. M. Hung, C. H. Kuo, S. K. Peng and H. S. Yu, "Distribution System Service Restoration Using a Heuristic Search Approach," in *IEEE Trans. Power Del.*, 1992.
- [36] A. Chipperfield, P. Fleming and C. Fonseca, *Genetic algorithm tools for control systems engineering. Adaptive computing in engineering design and control.*, Plymouth, 1994.
- [37] G. Anastasios , N. Pandel and Christo B, "Optimal Power Flow by Enhanced Genetic Algorithm," *IEEE transactions on Power Systems*, pp. 229-236, 2002.
- [38] M. Dorigo and M. Gambardella, "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, pp. 53-66, 1997.
- [39] L. Cechin, V. Canto dos Santos, A. Mendel and T. Gomez, "Genetic algorithms to

- solve the power system restoration planning problem,” in *Springer-Verlag London Limited*, London, 2009.
- [40] N. Fountas and N. Hatziaargyriou, “Artificial intelligence approaches to the problem of power system restoration,” National Technical University of Athens, 2009.
- [41] H. Mori and A. Furuta, “A New Approach of Hierarchical Optimization to Distribution System Service Restoration,” in *IEEE International Symposium on Circuits and Systems* , 2005.
- [42] W. Luan, M. Irving and J. Daniel, “Genetic algorithm for supply restoration and optimal load shedding in power system distribution networks,” in *Generation, Transmission and Distribution, IEE Proceedings*, 2002.
- [43] H. Mori and Y. Ogita, “A Parallel Tabu Search Based Approach to Optimal Network Reconfigurations for Service Restoration in Distribution Systems,” in *Proceedings of the 2002 IEEE International Conference on Control Applications*, Glasgow, 2002.
- [44] N. Razali and J. Geraghty, “Genetic Algorithm Performance with Different Selection Strategies in Solving TSP,” *Proceedings of the World Congress on Engineering* , vol. II, pp. 4-5, 2011.
- [45] J. Csatar, A. Dan and I. Vokony, “Evolutionary algorithm application to medium voltage power system restoration,” in *International Conference on Renewable Energies and Power Quality*, Santiago de Compostela, 2012.
- [46] E. Goldberg , *Genetic Algorithms in Search, Optimization and Machine Learning*, Boston: Addison-Wesley Longman Publishing Co, 1989 .
- [47] Narmatha R and D. Devaraj, “Enhanced Genetic Algorithm Approach for Security Constrained Optimal Power Flow Including FACTS Devices,” *International Journal of Electrical and Electronics Engineering*, pp. 552-557, 2009.
- [48] J. Figueiredo, J. Sauve, P. Nicolletti, E. Rocha, S. Araujo, F. Amorim, A. Feitosa, R. Agra and W. Ribeiro, “Smart action: A tool to help power system restoration,” in *41st Annual Hawaii International Conference on System Sciences*, Hawaii, 2008.
- [49] H. Fink, K. Lee Liou and C. Ching Liu, “From generic restoration actions to specific restoration strategies,” in *IEEE Trans. on Power Systems*, 1995.
- [50] R. Irving, P. Luan and S. Daniel, “Supply restoration in distribution networks using a genetic algorithm,” *Elsevier Science Ltd*, pp. 447-457, 2001.

- [51] Y. Kumar, B. Das and J. Sharma, "Genetic algorithm for supply restoration in distribution system with priority customers," in *International Conference on Probabilistic Methods Applied to Power Systems*, Sweden, 2006.
- [52] S. Guo, *The Application of Genetic Algorithms to Parameter Estimation in Lead-Acid Battery Equivalent Circuit Models*, University of Birmingham, 2010.
- [53] C. Salomon¹, P. M. Coutinho, C. Valério de Moraes¹, L. Borges da Silva, G. Lambert-Torres¹ and R. Aoki, "Applications of Genetic Algorithm," IUNIFEI - Itajuba Federal University, 2005.
- [54] N. M. Razali and J. Geraghty, "Genetic Algorithm Performance with Different Selection Strategies in Solving TSP," in *Proceedings of the World Congress on Engineering*, London, 2011.
- [55] D. Goldberg and S. Voessner, *Optimizing Global-Local Search Hybrids*, 1999.
- [56] C. M. Wankhade and A. P. Vaidya, "Optimal Power Flow Using Genetic Algorithm: Parametric Studies for Selection of Control and State Variables," *British Journal of Applied Science & Technology*, vol. 4, pp. 279-301, 2014.
- [57] M. H. El-werfelli, *Genetic Algorithms Applications to Power System Security Schemes*, Bath: University of Bath, 2009.
- [58] R. Kumar and Jyotishree, "Blending roulette wheel selection & rank selection in genetic algorithms," in *In Proceedings of 3rd International conference on machine learning and computing*, 2011.
- [59] O. Al Jadaan, L. Rajamani and C. Rao, "Improved Selection Operator for GA," *Journal of Theoretical and Applied Information Technology*, pp. 269-271, 2005-2008.
- [60] D. Beasley, R. D. Bull and R. Ralph, "Part 1: Fundamentals," in *An Overview of genetic algorithms*, University Computing, 1993.
- [61] L. Davis, *Handbook of GA*, New York: Van Nostrand Reinhold, 1991.
- [62] R. Malhotra, N. Singh and Y. Singh, "Genetic Algorithms: Concepts, Design for Optimization of Process Controllers," *Computer and Information Science*, pp. 41-43, March 2011.
- [63] A. Diaz-Gomez and F. Hougen, "Initial Population for Genetic Algorithms," School of Computer Science, Oklahoma.
- [64] K. De Jong, *Learning with Genetic Algorithm: An overview*, Machine Learning,

Kluwar Academic, 1998.

- [65] J. R. Koza, "Genetic Programming: A Paradigm for Genetically Breeding Populations of Computer Programs to Solve Problems," Technical Report No. STAN-CS-90-314, Computer Sciences Department, Stanford University, 1990.
- [66] K. E. Kinnear, *Advances in Genetic Programming*, Cambridge: MIT Press, 1994.
- [67] K. De Jong, *An Analysis of the Behaviour of a Class of Genetic Adaptive*, Michigan: University of Michigan, 1975.
- [68] J. Grefenstette, "Optimization of Control Parameters for Genetic," in *IEEE Transaction on Systems, Man, and Cybernetics*, 1986.
- [69] N. Deepanshu, *OPTIMAL POWER FLOW USING HYBRID GENETIC ALGORITHM*, 2012.
- [70] J. R. Koza and M. A. Keane, "Genetic Breeding of Non-Linear optimal Control Strategies for Broom Balancing," in *In Proceedings of the Ninth International Conference on Analysis and Optimization of Systems.*, Antibes, 1990b.
- [71] P. J. Angelene and J. P. Pollack, "Competitive Environments Evolve Better Solutions for Complex Tasks," in *In Proceedings of the Fifth International Conference on Genetic Algorithms*, San Mateo, 1993.
- [72] K. Li, L. Kang , W. Zhang and B. Li, "Comparative Analysis of Genetic Algorithm and Ant Colony Algorithm on Solving Traveling Salesman Problem," *Semantic Computing and Systems*, 2008.
- [73] J. D. McCalley, *The Power Flow Problem*, Iowa: Iowa State University.
- [74] E. Acha, *FACTS : modelling and simulation in power networks.* , Chichester: Wiley, 2004.
- [75] H. Saadat, *Power System Analysis*, New York: McGraw-Hill, 1999.
- [76] A. Bergen and V. Vittal, *Power Systems Analysis*, New Jersey: Prentice Hall, 2000.
- [77] A. Bergen and V. Vittal, *Power Systems Analysis*, New Jersey: Prentice Hall, 2000.
- [78] B. Weedy, *Electric Power Systems*, London: John Wiley & Sons, 1979.
- [79] O. Elgerd, *Electric Energy Systems Theory*, New York: McGraw-Hill, 1982.
- [80] S. Ray, *Electrical Power Systems: Concept, Theory and Practice*, Prentice-Hall of India Pvt.Ltd, 2014.

- [81] A. Dimitrovski and K. Tomsovic, "Slack Bus Treatment in Load Flow Solutions with Uncertain Nodal Powers," in *International Conference on Probabilistic Methods Applied to Power Systems*, Iowa, 2004.
- [82] A. Ghosh, *Power System Analysis: Chapter 4*, Kanpur: IIT-KANPUR.
- [83] P. D. Kothari and S. J. Dhillon, *Power System Optimization*, Prentice-Hall of India, 2010.
- [84] G. Duncan, M. Sarma and H. Overby, "Power Flows," in *Power Systems Analysis and Design*, London, Thompson, 2008, pp. 60-62.
- [85] G. Andersson, *Modelling and Analysis of Electric Power Systems*, Zurich: ETH Zurich, 2004.
- [86] F. Milano, "Power system modelling and scripting," Springer, New York, 2010.
- [87] J. Zhu, *Optimization of Power System Operation*, Wiley-IEEE Press, 2009.
- [88] S. S. Shrawane and M. Diagavane, "Application of Genetic Algorithm for Power Flow Analysis," *International Journal of Engineering Research & Technology*, vol. 2, no. 9, p. 454, 2013.
- [89] F. Milano, *Power system modelling and scripting*, New York: Springer, 2010.
- [90] H. E. Brown, *Solution of large networks by matrix methods*, New York: Chichester: Wiley, 1985.
- [91] J. Arrillaga and P. C. Arnold, *Computer analysis of power systems*, Chichester: New York: Wiley, 1990.
- [92] R. Zimmerman, "MATPOWER's Extensible Optimal Power Flow Architecture," *IEEE journal* 2009, 2009.
- [93] . K. Tarjei, "Utilizing matpower in optimal power flow," Harvard University, Boston.
- [94] Y. Koc, M. Warnier, R. E. Kooij and F. M. Brazier, "A robustness metric for cascading failures by targeted attacks in power networks," in *Networking, Sensing and Control (ICNSC)*, Evry, 2013.
- [95] Elirank, "Tournament selection vs roulette wheel," [Online]. Available: <http://www.cs.bgu.ac.il/~elirank/evo%20web%20crap/q3.htm>. [Accessed 2014 December 21].
- [96] The Computer action team, "An Introduction to MATLAB," [Online]. Available: <http://web.cecs.pdx.edu/~gerry/MATLAB/intro/intro.html>. [Accessed 12 November

- 2014].
- [97] W. Kahan, *Matlab's Loss is Nobody's Gain*, California: University of California, 2004.
- [98] L. Di Gaspero, *Local Search Techniques for Scheduling Problems: Algorithms and Software Tools*, Udine: Universita degli Studi di Udine, 2003.
- [99] Modern Ghana, *How to conquer mountain Afadjato*, Accra: <http://www.modernghana.com/news/294133/1/how-to-conquer-mountain-afadjato.html>, 2010.
- [100] R. Kumar, R. Kumar and G. Gopal, "Hybridization in Genetic Algorithms," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 4, pp. 403-406, 2013.
- [101] R. Garg and S. Mittal, "Effect of Local Search on the Performance of Genetic Algorithm," *International Journal of Emerging Research in Management & Technology*, vol. 3, no. 6, pp. 404-407, 2014.
- [102] T. M. Ferreira dos Santos, "Mesh grid structure vs. radial structure," Instituto Superior tecnico, Lisboa, 2013.
- [103] ALSTOM, "What are electrical grids?," ALSTOM, 2 May 2013. [Online]. Available: <http://www.alstom.com/grid/about-us/understanding-electrical-grids/What-are-electrical-grids/>. [Accessed 15 January 2015].
- [104] "Structural Vulnerability Analysis of Electric Power Distribution Grids," in *Physica and Society*, New York, 2015.
- [105] S. Tongsopit, "Topological standards and the problem of lock-in," in *The Political Economy of Grid-connected Distributed Power Generation Systems*, California, ProQuest, 2011, pp. 32-35.
- [106] J. Keiren, *System Validation Course*, Delft: Delft University of Technology, 2015.
- [107] Y. Kumar, B. Das and J. Sharma, "Service restoration in distribution system using," *Electric Power Systems Research*, p. 768-777, 2005.
- [108] O. Roeva, S. Fidanova and M. Paprzycki, "Influence of the Population Size on the Genetic Algorithm Performance in Case of Cultivation Process Modelling," *Proceedings of the 2013 Federated Conference on Computer Science and Information Systems*, p. 371-376, 2013.
- [109] R. Barr, L. Golden, P. Kelly, M. Resende and W. Stewart, "Designing and Reporting

- on Computational Experiments with Heuristic Methods.,” *Journal of Heuristics*, vol. 1, p. 9–32, 1995.
- [110] A. Silva, Z. Vale and C. Ramos, “Cooperative Training of Power Systems 'Restoration Techniques,” in *Intelligent Systems Application to Power Systems 2005*, Arlington, 2005.
- [111] C. Wang , K. Sun and V. Vittal, “OBDD-Based Sectionalizing Strategies for Parallel Power System Restoration,” in *Power Systems, IEEE Transactions*, Tempe, 2010.
- [112] North American Electric Reliability Council, “Electric System Restoration Reference Document,” 1993.
- [113] J. Gutierrez, M. Starpolsky and M. Garc, “Policies for Restoration of a PowerSystem,” in *IEEE Transactions on Power Systems vol. PWRS-2*, 1987.
- [114] P. Kundur, *Power System Stability and Control*, McGraw-Hill, 1994.
- [115] H. J. Lee and Y. M. Park, “A Restoration Aid Expert System for Distribution Substations,” in *IEEE Trans. Power Del., vol. 11*, 1996.
- [116] T. Brunner, W. Nejd, H. Schwarzjirg and M. Sturm, “Online Expert System for Power System Diagnosis and Restoration,” in *Intelligent Systems Engineering* , 1993.
- [117] Y. Kojima, S. Warashina, S. Nakamura and K. Matsumoto, “Development of a guidance method for power system restoration,” in *Power Systems, IEEE Transactions on (Volume:4 , Issue: 3)*, 1989.
- [118] Y. Kojima, S. Warashina, M. Kato and H. Watanabe, “The development of power system restoration method for a bulk power system by applying knowledge engineering techniques,” in *Power Systems, IEEE Transactions*, 2002.
- [119] K. Matsumoto, T. Sakaguchi, R. Kafka and M. Adibi, “Knowledge-based systems as operational aids in power system restoration,” in *Proceedings of the IEEE (Volume:80 , Issue: 5)*, 2002.
- [120] K. Miura, N. Takamo, Y. Ueki and Y. Fukuyama, “An expert system for procedure generation of power system restoration in regional control center,” in *Advances in Power System Control, Operation and Management*, 1991.
- [121] T. Nagata, H. Sasaki and R. Yokoyama, “Power system restoration by joint usage of expert system and mathematical programming approach,” in *Power Systems, IEEE Transactions on (Volume:10 , Issue: 3)*, 1995.

- [122] Y. Park and K. Lee, "Application of expert system to power system restoration in sub-control center," in *Power Systems, IEEE Transactions on (Volume:12 , Issue: 2)*, 1997.
- [123] T. Nagata, H. Watanabe, M. Ohno and H. Sasaki, "A multi-agent approach to power system restoration," in *Power System Technology, 2000. Proceedings. PowerCon 2000. International Conference on (Volume:3)*, 2000.
- [124] H. Yang, J. T. Liao and H. X. Su, "A fuzzy-rule based power restoration approach for a distribution system with renewable energies," in *Fuzzy Systems (FUZZ), 2011 IEEE International Conference on*, 2011.
- [125] M. Huang, H. Kuo, S. Peng, C. Chang, K. Chang, H. Yu, C. Chow and R. Kuo, "Distribution system service restoration using a heuristic search approach," in *Transmission and Distribution Conference*, 1991.
- [126] D. Kirschen and T. Volkmann, "Guiding a power system restoration with an expert system," in *Power Systems, IEEE Transactions on (Volume:6 , Issue: 2)*, 1991.
- [127] C.-C. Liu, R. Chu and A. Holen, "Generation Capability Dispatch for Bulk Power System Restoration: A Knowledge-Based Approach," in *IEEE Transactions on Power Systems, Vol 8*, 1993.
- [128] R. Kumar, G. Gopal and R. Kumar, "R Kumar, G Gopal & R Kumar," *International Journal of Advanced Research in Computer Science and Software Engineering*, 2001.
- [129] J. Holland, *Adaptation in Natural and artificial Systems*, Michigan: the university of Michigan, 1975.
- [130] O. Al Jadaan, L. Rajamani and R. Rao, "IMPROVED SELECTION OPERATOR FOR GA," *Journal of Theoretical and Applied Information Technology*, pp. 226-229, 2008.
- [131] DP Chassin, "Grid LAB-D Technical Support Document," vol. Network Module Version 1.0, 1 May 2008.
- [132] L. Dai and L. Chih, "Low-Complexity Soft-Output Signal Detection Based on Gauss-Seidel Method for Uplink Multi-User," *Large-Scale MIMO Systems IEEE Conference*, 2013.

Appendices

Appendix 1

Pseudocode of the GA with tournament selection.

```
N = population size
P = create parent population by randomly
creating N individuals
while not done
    C = create empty child population
    while not enough individuals in C
        parent1 = select parent using
tournament selection
        parent2 = select parent using
tournament selection
        child1, child2 = crossover (parent1,
parent2)
        mutate child1, child2
        evaluate child1, child2 for fitness
        insert child1, child2 into C
    end while
    P = combine P and C somehow to get N
new individuals
end while
```

Figure 5.7: Pseudocode of the GA [50].

The expanded form of the Genetic algorithm is given below.

Algorithm

1. Run the powerflow on prefaulted network.
2. Calculate total load (TL) as the sum of the loads on each bus.
3. Remove the faulted branch from the branch matrix(Radiality check- ON or OFF).
4. Create initial string from the status of each remaining branch (1 closed, 0 open).
5. Create initial population.
 - a. numInitSeeds are created by applying mutation to the initial solution.
 - b. The rest is generated at random.
6. While cost < errorTreshold or the maximum number of iteration is reached, do the following.
7. Evaluate the population.
 - a. For each string, calculate the cost function (see the sub-procedure below).
 - b. Sort the genomes into ascending order.
8. Generate the new population.

- a. Take first numParents elements from the sorted genomes to form parentPool.
 - b. For every new genome do the following:
 - i. Select parents from the parentPool by using tournament selection.
 - ii. Perform crossover to create a new genome (half uniform and three-way available).
 - c. Apply mutation to each new genome according to mutationProb.
 - d. $\text{mutationProb} = \text{mutationProb} * \text{mutationDecay}$
9. Perform the proposed local search which does a re-evaluation of fitness in order to have an improved solution.
 10. Go to Step 6.
 11. The solution is a best-fit genome from the current population.

CalcCost Sub-Procedure

1. Change the status of each branch according to the genome.
2. Run powerflow.
3. Calculate the total load on the current network as a sum of the loads on each bus.
 - a. Load on a bus is equal to the difference of the outgoing load and the incoming load.
 - b. The incoming load is read from the branches going into the bus.
 - c. The outgoing load is read from the branches going from the bus.
4. $f1 = TL - \text{currentTotalLoad}$
5. $f2 = \text{hamming distance between current and initial switch state}$
6. $f3 = \text{total losses}$
 - a. Total losses are the sum of losses on each active branch.
 - b. Losses on a branch are the difference of the outPower and inPower of the branch.
7. $f4 = 1$, if some voltage constraint is violated or 0 otherwise
 - a. Each bus has V (current voltage) and V_{min} and V_{max} .
 - b. If V is not between V_{min} and V_{max} , the voltage constraint is violated.
8. $f5 = 1$, if some priority customer is not powered or 0 otherwise
 - a. Each bus has a flag whether it is a priority or not.
9. $f = \text{sum}(w_i * f_i)$, w_i are weights of the cost function

Appendix 2

Check Radiality Algorithm Description

1. Main principles: The count of connection between buses is equal to the count of buses decreased by one.

Base restriction: The tested network must be without blackouts.

2. Implementation:

Count of connection is estimated by calculating the count of 1 in the adjacency matrix. In a common adjacency matrix, the count of 1 will be double by the count of connection. For review we will use half of the adjacency matrix – divided by the main diagonal (the top-right field).

Connection of blackout buses must be excluded from the adjacency matrix.

№	Algorithm Step	Example
1	<p>Init the adjacency matrix.</p> <p>Lower-left field under the main diagonal must be set to zero.</p> <p>Init the bus state vector string, $M = \{m_1, m_2, \dots, m_n\}$.</p> <p>Where $m_i =$ state of i bus, it is 1 if connected to the main generator of the network and 0 if it is unconnected.</p>	<p>Example network with five buses, and the adjacency matrix is:</p> <p>$M = \{1, 0, 0, 0, 0\}$</p> <p>In first step, the network contains only bus with number 1.</p>
2	<p>For each column in the adjacency matrix, A_j must be calculated:</p> <p>$C = M \times A_j$</p> <p>If C is a greater than zero, then perform next operations:</p> <p>$M(j) = 1$; and</p> <p>$M = M \text{ logical or } A_j$.</p>	<p>Column 1: Skip because it is a dummy operation since the same value will be obtained.</p> <p>Column 2: $M * A_2 = 1 * 1 + 0 * 0 + 0 * 0 + 0 * 0 + 0 * 0 = 1$.</p> <p>If result 1 is greater than 0, we perform:</p> <p>$M(2) = 1$; $M = \{1, 1, 0, 0, 0\}$</p> <p>$M = M \text{ logical or } A_2$ (here M is still the same)</p>

		<p>Column 3: $M \cdot A3 = 1 \cdot 1 + 1 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 = 1$.</p> <p>If result 1 is greater than 0, we perform:</p> <p>$M(3)=1$; $M = \{1, 1, 1, 0, 0\}$</p> <p>$M = M$ logical or $A3$ (here M is still the same)</p> <p>Column 4: $M \cdot A4 = 1 \cdot 0 + 1 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 + 0 \cdot 0 = 0$.</p> <p>If result 0 is equal to 0, go to next iteration,</p> <p>$A4 = \{0, 0, 0, 0, 1\}$.</p> <p>Column 5: $M \cdot A5 = 1 \cdot 0 + 1 \cdot 0 + 1 \cdot 0 + 0 \cdot 1 + 0 \cdot 0 = 0$</p> <p>If result 0 is equal 0, go to next iteration.</p>
3	Perform Step 2 when result M is different from start M on each iteration of Step .2	<p>Perform Step 2. Result $M_k = \{1, 1, 1, 0, 0\}$ will be the same as</p> <p style="text-align: center;">$M_{k-1} = \{1, 1, 1, 0, 0\}$</p>
4	<p>Exclude the connection of an area with a blackout in the adjacency matrix by multiplying (as an array) vector M and vector-column A_j:</p> <p>$A_j = A_j$ multiply M</p>	<p>A1: $A1 = \{0, 0, 0, 0, 0\}$</p> <p style="text-align: center;"><u>$M = \{1, 1, 1, 0, 0\}$</u></p> <p>Result $A1 = \{0, 0, 0, 0, 0\}$</p> <p>A2: $A2 = \{1, 0, 0, 0, 0\}$</p> <p style="text-align: center;"><u>$M = \{1, 1, 1, 0, 0\}$</u></p> <p>Result $A2 = \{1, 0, 0, 0, 0\}$</p> <p>A3: $A3 = \{1, 0, 0, 0, 0\}$</p> <p style="text-align: center;"><u>$M = \{1, 1, 1, 0, 0\}$</u></p> <p>Result $A3 = \{1, 0, 0, 0, 0\}$</p> <p>A4: $A4 = \{0, 0, 0, 0, 0\}$</p>

		<p>$M = \{1, 1, 1, 0, 0\}$</p> <p>Result $A_4 = \{0, 0, 0, 0, 0\}$</p> <p>A5: $A_5 = \{0, 0, 0, 1, 0\}$</p> <p>$M = \{1, 1, 1, 0, 0\}$</p> <p>Result $A_5 = \{0, 0, 0, 0, 0\}$ The connection between Buses 4 and 5 is excluded.</p> <p>The new adjacency matrix is</p> $A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$
	<p>Calculate the count of 1 in the upper-right field of the adjacency matrix.</p> <p>Calculate the count of 1 in the M vector.</p>	<p>The count of 1 in the upper-right field of the adjacency matrix is a 2.</p> <p>The count of 1 in the M vector is a 3.</p> <p>Checking the main criteria: $2 = (3-1)$, so the network is radial.</p>
	<p>Optional: Calculate the power flow.</p> <p>Obtain the existing vector of bus loads P.</p> <p>Calculate the power flow</p> $P_{\Sigma} = M \times P$	

Appendix 3

How the reserved switches were found in the network

1. Load the network by executing. Example:

```
net = eval("case14")
```

2. The field "branch" in the structure gives us information about the branches. The number of rows tells us the number of branches in the network.
3. The first two columns of the matrix branch is of importance for us. This two columns tells us the two buses that the particular branch is connecting. It can be viewed by executing the below command. For explanatory purposes, let us call Column 1 the source and Column 2 the destination.

```
net.branch(:,1:2)
```

4. Now the task is to find reserved branches, i.e., to make the network radial or to remove redundant branches. The resulting network must be a spanning tree with no loops. For this, it can be observed that any branch must terminate into any destination bus only once. In other words, the second column of the above matrix must not have any repetitions. So to find the duplicates we can execute the following.

```
[b m n] = unique(net.branch(:,2),'first');
```

5. From the output variable, "[b m n]". b gives the bus numbers, which is not important for us. m gives the indices of the branches that are unique, which means the branches present in the variable m must be ON while the rest of them must be OFF.
6. The next step would be to simply list all the remaining branches that are not present in the variable m. This can be done by executing the following line.

```
setdiff(1:size(net.branch,1),m)
```

For any network, replace its name in the first command.