TABU SEARCH-BASED OPTIMAL CHOICE OF LINES FOR VOLTAGE DERATING IN LIGHTLY LOADED, HIGH VOLTAGE CONSTRAINED NETWORKS

Dissertation Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Philosophy (MPhil) Electrical and Electronic Engineering

Faculty of Engineering and the Built Environment

UNIVERSITY OF JOHANNESBURG

Popi Elizabeth Melato

Date: 30/08/2016

Supervisor: Dr. Nhlanhla Mbuli

Co-Supervisor: Professor Jan-Harm Pretorius

Declaration

I........Popi Elizabeth Melato.. declare that

(i) The research reported in this dissertation/thesis, except where otherwise indicated, is my original work.

(ii) This dissertation/thesis has not been submitted for any degree or examination at any other university.

(iii) This dissertation/thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

(iv) This dissertation/thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

a) their words have been re-written but the general information attributed to them has been referenced;

b) where their exact words have been used, their writing has been placed inside quotation marks, and referenced.

(v) Where I have reproduced a publication of which I am an author, co-author or editor, I have indicated in detail which part of the publication was actually written by myself alone and have fully referenced such publications.

(vi) This dissertation/thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the dissertation/thesis and in the References sections.

Signed: ____________________________

Table of Contents

List of Figures

List of Tables

Acknowledgements

I wish to thank the University of Johannesburg and Eskom Holding SoC Limited, both in South Africa, for funding this study. My deepest gratitude to my co-supervisor, Prof. Jan-Harm Pretorius for been a great supervisor and for the meaningful discussions we've had since the beginning of this study. I am truly grateful to him for his willingness to listen and provide guidance to ensure that I don't lose focus. I am sincerely grateful to my supervisor, Dr. Nhlanhla Mbuli for the meaningful discussions we've had since the beginning of this study. I am also thankful to him for teaching me how to conduct a scientific research and to work systematically. He has taught me how to probe meaningful discussions and ideas. His patience and support helped me to stay focused and disciplined. I'll forever be grateful to him for his significant contribution to my career development.

I owe my gratitude to my parents, my dad, Mr. Motseokae Henry Melato and my mom, Mrs. Semakaleng Lucia Melato, for their love, encouragement and support from the day I was born. They were both constantly in my mind and from them my strength was always restored when I wavered. My heartfelt thank you goes to my dad, for always instilling the importance of knowledge in us from the developmental stage of our childhood.

I am thankful to my partner Dr. Delani Njapha for his unwavering support and for his willingness to always listen and give exceptional advice. I am also thankful to him for his continuous encouragement and mostly, for tirelessly ensuring that I stayed afloat.

I wish to express my gratitude to my siblings, Lehlohonolo Melato, Prince Melato and Kedibone Tshegare, for never giving up on me; as well as the Tshegare's family for their support. I am also grateful to my friends for their overwhelming support.

Most of all, I thank God for His merciful hand which is always upon me.

Without all of you this dissertation would have not been possible.

Abstract

In power system planning, the general concern is the ability of a power system to supply increasing loads without undervoltages. Nonetheless, decrease in loads can also occur in power systems, e.g., when mines are decommissioned leading to shrinkage in load. Depending on the scale of the load reduction, problem of steady overvoltages can arise.

This study proposes an investigation of using of step-down transformers to reduce steady state overvoltages in a lightly loaded network comprising of long corridors. The investigation evaluates the proposal of installing transformers at both ends of the length network, forming a ring of transformers to derate the operating voltage of the derated network. The load flow simulation studies are initially carried out manually by scaling the load down using the Power System Software for Engineers (PSS/E). The load flow studies for the lowest load served without violations are carried out for the case without transformers and the derated case with transformers.

The study further proposed the use of Python program to automate the studies through scaling the load down until the voltage limits are reached by sending commands to PSS/E using Python syntax. In addition to this, the use of Tabu Search (TS) algorithm is proposed for optimally selecting a combination of lines to accommodate the least load within the voltage limits. The load is scaled down automatically in PSS/E using Python program with the least load served as an objective function and the constraints being the voltage limits.

The study found that the use of step-down transformers can successfully reduce the steady state overvoltages in a lightly loaded network. The introduction of Python program to automate the studies was successful. Lastly, intelligence of Tabu Search algorithm can successfully select the optimal combination of lines to serve the least load without voltage violations.

Chapter 1: Introduction

This chapter presents an introduction of the study. Section 1.1 covers basic requirements of a reliable power system for delivery of power to customers. Section 1.2 presents the power system limits requirements for long term and real-time operation of the grid. In Section 1.3 types of overvoltages and their causes are briefly discussed. In Section 1.4, presents the problem statement for the study. Section 1.5 presents the study hypothesis. Section 1.6 presents the structure of the dissertation. Finally, Section 1.7 covers the list of publications for the conference proceedings and publications made during the course of this study.

1.1. Requirements for Power Transmission and Delivery to Customers

Power systems are designed to evacuate electrical power from the sources, i.e., generation power stations through transmission and distribution systems to load centres for consumption by customers. A power system is deemed stable when the total loads and the power losses are equal. In order to ensure a stable power system [1], it is of outmost importance to maintain voltages constant in order to limit varying of reactive power.

Acceptable level of service to the customers must be ensured by virtue of power meeting acceptable levels [2] of reliability and quality of supply. The former is characterised by adequacy and availability of supply and the latter by constancy of frequency and voltage.

1.2. Planning and Operation of a Reliable Power System

The planning and operation of a reliable power system can become exceedingly complex due to network disturbances which are likely to violate the steady state network conditions. The power system is deemed to be successfully designed and operated when the following fundamental requirements are met [2]:

(i) When the system meets the fluctuation of the load demand for both active and reactive power,

(ii) When the energy is supplied as a minimum cost and less impact on the environment, as well as ensuring

(iii) Adherence to quality of supply requirements through a controlled stable frequency, voltage and meeting a level of reliability.

This means, a reliant electrical grid on the ability to deliver power to the customers in a mode which is reliable, secure and stable. Planning and operation of a reliable supply for most electrical utilities are governed by regulators where energy policies and standards are to be adhered to in order to ensure that electricity supply licence is retained and uniformity to governing policies in the energy sectors, Eskom by the National Electricity Regulator of South Africa (NERSA).

In the aspect of voltage, the attempt is to ensure that the network is planned and operated within acceptable voltage limits. For long term planning purpose the voltages should be kept between 0.95 per unit and 1.05 per unit under system healthy and after a contingency [3]. The operational criteria [4] recommends a minimum voltage level of 0.95 per unit with a healthy system or per unit with design contingency and the voltage maximum limit of 1.09 per unit under a loss of two circuits, i.e., N-2.

1.3. Types of Overvoltages in Power Systems

Overvoltages can be categorised as transient overvoltages, sustained overvoltages, harmonic resonance overvoltages, and may also result from ferroresonance phenomenon [5]. The common types of overvoltages in power systems are steady state and transient overvoltages.

1.3.1 Transient Overvoltages

Transient network instabilities are often characterized by voltage fluctuations beyond acceptable limit. Uncontrolled overvoltages (harmonics) will compromise the security of network supply; lead to loss of major circuits in the grid and result in cascading network blackouts.

Transient overvoltages can also emanate from switching of long transmission lines as well as switching of capacitive devices. It has been found that during system restoration the sending end voltage will rise above the maximum limits [6] and result into temporary overvoltages. This was attributed to the network being lightly loaded at the time of network restoration.

The following solutions were therefore proposed to prevent the harmonic resonance overvoltage:

(i) Connecting a highly resistive load to the relevant bus to reduced amplification of injected currents and
(ii) Bringing more generators on line to reduce the inductance and higher frequency resonance.

1.3.2 Steady Overvoltages

Steady state overvoltages are a common problem in lightly loaded networks, and this can be more complex if lines supplying a load center have relatively long corridors [7]. If sustained they result in excessive charging currents due to lightly loaded transmission lines [8].

These high currents may force the generator under-excitation operation mode or even self-excitation and instability. The transformers with sustained overvoltages also overexcite and can generate harmonics.

1.4. Problem Statement

Overvoltages, both transient and steady state, have negative effects in the operation of a reliable power system and therefore, are not acceptable for its performance. Steady state overvoltages are frequently associated with lightly loaded system which can be precipitated by, for example, off-peak loads in the network or reduction in loads supplied by the system. Common practises in solving this problem entail the use of shunt reactors, series reactors, also by varying the transformers tap changer, and ultimately switching off of lines. However, as loads are reduced far too low, switching off lines also does not provide an adequate solution.

In this study, a lightly loaded network suffering from steady state overvoltages spreading within a wide load centre comprising of lengthy 132kV distribution networks and a number of substations, is used. This study attempts to provide a solution to this inadequacy by investigating an option of using step-down transformer and locating them along corridors feeding a load centre, so that a ring of such transformers is formed, and the voltages of all the lines encircled, and on the secondary side of the transformers, become lower than the former.

The use of step-down transformers by forming a ring in a load centre with relatively long lines and lightly loaded is expected to have a positive effect in derating line voltages and ameliorating steady state overvoltages.

1.5. Hypothesis

The use of step-down transformers to derate lines can reduce steady state overvoltages due to additional reactance added in the system. The use of step-down transformers can also reduce operating voltages on the secondary side of the transformer, thus the reactive power flow on the derated lines can also reduce system voltages.

1.6. Structure of the Dissertation

The following section covers the layout of the dissertation.

Chapter 2 presents literature review of overvoltages in power systems. The normal operation of the power system and the delivery requirements while adhering to set limits is discussed. The types of overvoltages in the power system and the associated problems as well as the solutions of steady overvoltages are also covered. Lastly, this chapter discusses the causes of voltages caused by the penetration of distributed generation (DG).

Chapter 3 reviews the theory applicable in overvoltages. The impact of using step-down transformers to reduce steady overvoltages is also presented. The mathematical concepts for the relationship between load and system voltages are also discussed. The benefit of installing step-down transformers to increase system reactance thus reducing both the reactive power flow and line charging currents is also discussed.

Chapter 4 presents the methodology of the study. Chapter 4 also covers the software models used in the study, the type of load flow simulations carried out as well as the technical criterion used for the technical assessment.

Chapter 5 presents the case study used as an illustrative example. The IEEE 14 bus system is used as an illustrative example where a demonstration of the proposed solution for the study, is shown. This chapter also shows the lines, transformers, generation and load modelling data. The automation of the load flow studies using Python program is also covered.

Chapter 6 presents both the results for scaling the load down manually and the automated scaling down of load using a Python program. This section presents the results comparison between Case A and Case B.

Chapter 7 presents the introduction and literature review on Tabu Search applications in power systems. The Tabu Search literature survey also covers its application on combinatory problems in power systems. Tabu Search algorithm use in optimisation problems in power systems and how it compares to other algorithms is presented.

Chapter 8 presents the methodology of the study for Tabu Search algorithm for solving optimisation problem in this study. This section also discusses how Tabu Search works. This chapter also discusses the problem formulation for the study. An illustrative example, case study preparation and lastly the software models used are presented.

Chapter 9 presents the numerical results of generated lines combinations by Tabu Search algorithm. A comparison between the least loads served from different number of combinations studied is also shown.

Chapter 10 presents the conclusions on the investigated use of step-down transformers in a lightly loaded network suffering from overvoltages and comprising of lengthy corridors. The observations made during the analysis of the results where the loads were scaled down manually to connect the least load were covered. The findings of the use of Python program to automate the load flow studies were also presented. The conclusions are also made based on the use of the Tabu Search algorithm to select the best combination of lines to connect the least load without voltage violations are presented. Lastly, the proposal of further work on optimal placement of distributed generation (DG) using Tabu Search algorithm is made.

1.7. Publications

The following conference papers were part of the conference proceedings and were published internationally at peer reviewed conferences, i.e., IEEE, as part of conducting this study.

2015 **Serving Low Load Levels by Derating Line Voltage Using Step Down Transformers**, Universities Power Engineering Conference (UPEC 2015), England, UK, 1-4 September 2015.

2014 **Derating Line Voltage as an Option for Reliability Enhancement of Lightly Loaded Networks,** IEEE Universities Power Engineering Conference (UPEC 2014), Cluj-Napoca, Romania, 2-5 September 2014.

2014 **Derating Line Voltages to Limit Steady Overvoltages and Enhance Network Operational Efficiency,** 6th IEEE Power India International Conference (PIICON-2014), Delhi, India, 5-7 December 2014.

Chapter 2: Literature Review on Overvoltages

2.1. Introduction

This chapter discusses the literature scan and an overview of problems associated with steady overvoltages in power systems. Section 2.2 discusses power system power delivery requirements and the damages associated with sustained overvoltages. In Section 2.3 covers the steady overvoltages in lightly loaded networks. Section 2.4 discusses the causes of overvoltages. In Section 2.5 problems that associated with overvoltages are discussed. A briefly explanation of overvoltage solutions is discussed in Section 2.6. A solution for overvoltages in a Russian power system grid is reported. In Section 2.8 causes of overvoltages in the network with the presence of distributed generation (DG) is explained. Lastly the summary for the literature scan is included.

2.2. Literature Review

Power systems are designed to deliver electrical power from generation stations, via transmission and distribution systems to load centres for consumption by customers. The power generation stations are normally located near the natural energy resources, i.e., coal, water as raw materials and transmission infrastructure is established and used to transmit higher voltages to large power users and the distribution network once the voltage is stepped down via transformers is distributed to relatively small customers.

To ensure acceptable level of service to the customers, the transmitted power must meet acceptable levels [2] of reliability and quality of supply. The acceptable level of service to the customers is characterised by adequacy and availability of supply and acceptable power limits largely dependent on frequency and voltage. Voltage is one of the main component of power systems, ensuring that voltages are maintained within limits can prove to be a challenging task [2]. This can be as a result of the system reactive power varying as the load varies.

There are two types of overvoltages, i.e., steady state and transient. The transient overvoltages are often classified as momentary and often emanate from switching operations. Steady overvoltages are often triggered by line charging currents and occur at the receiving end of lowly loaded transmission lines [9,10]. This study focuses on the steady overvoltages.

Overvoltages lasting for a time duration of a minute or longer are classified as steady overvoltages [11]. Steady state overvoltages are not desirable as they may result in problems in the power system. Steady overvoltages have adverse effects on power systems sustainability, including, inter alia, (I) violation of prescribed planning [3] and operational limits [4].

These overvoltages, if prolonged, may lead to:

(i) Accelerated electrical aging of plant,
(ii) Damage to equipment if voltages exceed withstand capability of equipment [12], and
(iii) Increased current flow and active power losses in the system [13].

They can also lead to violation of quality of supply stipulations [14], and this means violation of the supply license conditions which is a serious matter from the regulatory perspective.

Increased flow of capacitive reactive power flow in the system [13], can emanate as a result of overvoltages, leading to increased loading of equipment and increased active power system losses. Power system losses are undesirable and their costs ultimately have to be endured by the customers. In order to enhance the system efficiency these losses must be reduced to least levels possible [15].

Some the common practices to control overvoltages and reducing steady overvoltages comprises (I) controlling reactive power in the system by regulating the generator set point voltage [16, 2], disconnection of lines during low load conditions (II) use of autotransformer tap changing,(III), (IV) switching in of shunt reactors and disconnecting shunt capacitors, and (V) operating synchronous machines in underexcited mode [16]. The following section discusses the causes of steady state overvoltages.

2.3. Steady State Overvoltage

Steady state overvoltages are undesirable due to the problems they lead to within the power system, particularly with lightly loaded networks and loads that are expected to diminish. Steady state overvoltages can result in violation of quality of supply requirements [17], and utilities can be ripped off their supply license by the regulatory if these conditions are violated. Overvoltages can result in an increased capacitive reactive power flow in the power grid [13], which can lead to excessive loading of equipment and increased active power system losses. The reduction of losses needs to be reduced to minimum levels possible [15].

2.4. Causes of Steady State Overvoltage in Power Systems

The IEEE 1159 [18] standard, describes steady overvoltages as long-duration overvoltages with durations of 1 minute or longer. They can intensify if lines feeding a load centre are relatively long and the power sources are isolated.

Steady state overvoltages in the power system can be caused by a substantial reduction in loads due to , for example, mining and ,or big industrial loads closing down their operations , decommissioning and moving to a remote location, therefore less demand for power is sought in comparison of the previous amounts supplied. Another cause could be load fluctuations [19], resulting in lesser load demand for off-peak network conditions.

The need to have high sending end voltage can prompt a desire to operate the system with high voltages to meet the delivery of a suitable voltage at the customer point and this can be in a case of poor power factor at the customer connection. The described load issues above can be more complex if a system is well-meshed and has long lines generating excessive reactive power [20, 24].

2.5. Problems Associated with Overvoltages

Elongated presence of overvoltages above the equipment withstand capability; can induce damage to equipments such as, shunt capacitors, reactors and current transformers [8]. These equipments have protection settings which will be affected under these conditions to remove them from the system during abnormal system conditions, by virtue of protection being initiated. This operation can damage the insulation and flashovers will result, leading to damage and failure of equipment.

The equipment may need to be replaced or repaired thus leading to potential compromise on reliability of supply. Similarly, the equipment with overvoltage protection if exposed to voltages above protection settings may lead to an outage of those equipment as a trip would have resulted during the voltage reading above protection levels [21]. This can also lead to loss of supply and load to be shedding may be required and reliability may be compromised.

Voltage control if maintained to values within the acceptable voltage limits, can improve the effectiveness and reliability of a power system. Prolonged system operation with voltages exceeding acceptable limits can lead to damaged equipment due to operation of equipment above allowable range [2]. In order to accomplish voltage control levels, there needs to be a control in production and absorption of reactive power in an electrical system [2]. In a question of the relationship between load and reactive power in the system, as load varies, the reactive power also varies.

Therefore, since reactive power cannot be supplied over lengthy lines, voltage control requires special local devices dispersed throughout the system and areas with fluctuating loads, between peak and off peak network conditions. These special devises are discussed in the next section. It should be noted that the balance between selecting and coordinating equipments to address voltage and reactive power control is amongst challenging endeavours in power systems.

2.6. Overvoltages Solutions

The common practices to control steady state overvoltages in power systems [16] are; disconnection of supply power lines into the lightly loaded supply area, installation of shunt reactors as well as reactive compensation devices, i.e., Static Var Compensator (SVC), online tap changers, and controlling of the generator voltage set point.

One of the common solutions to overvoltage problems emanating from long transmission lines [2,5], is to install shunt reactors at the end of the lines. It has been proven that, as loads vary, the reactive power requirements of the power system vary. Hence the known Ferranti Effect phenomenon in power systems tend to emanate on long lightly loaded transmission lines.

Therefore, since reactive power cannot be transmitted over long distances, voltage control is usually controlled by using special device dispersed throughout the system. In the case of lightly loaded networks, line reactors are therefore installed to control voltage rise as well as [1] disconnection of supply power lines into the lightly loaded supply area in order to ensure that the power system adhere to voltage limits.

2.6.1 Voltage Regulation

The regulation of voltage levels can be achieved by controlling the production, absorption, and flow of reactive power at all levels in the system [2].

This voltage control can be achieved by controlling voltage at the generating units through the automated voltage regulators which ensures the control of field excitation thus maintaining a required magnitude of scheduled voltage. The devices used for voltage regulation are sources or sinks of reactive power such as shunt capacitors, SVCs and series capacitors on lines, transformer tap changers.

2.6.2 Shunt Reactors

Shunt reactors provide passive compensation, and the same is true for shunt capacitor and series capacitors [2]. The effects of line capacitance in the system are compensated by use of shunt capacitors and this is particularly under light loading condition or open circuit to limit rise in voltage. Shunt reactors are often required for overhead high voltage lines longer than 200km. They can be switched in and out depending on the load condition, for example under light load condition they can be in service and the opposite is true, under peak load conditions.

2.6.3 Generator Set Point Voltage Control

The limits of real and reactive power which can be supplied into the network are pre-determined by the generator [22] by the nature of its design. In addition to this, real power produced by the generator is derived from the degree of excitation applied to the generator.

The generator terminal voltage is regulated by the excitation. In a question of voltage control, the generator can effectively control the voltage at the bus directly connected to, provided it is the only generator connected to it. In the case of multiple generation sources, connected to one bus, each will contribute reactive power to keep voltage at a certain level. However the extent of voltage control is limited in such cases due to voltage inconsistent change as the function of load.

2.6.4 Transformer Taps

Voltage control can also be achieved by installing step down transformers. The transformer makes use of a tap changer to regulate voltage up or down. The transformer tap changer also provides a means of regulating reactive power flow [2] in the system, and the tap ratio adjusts to voltage fluctuations in the system.

The use of transformers in power systems leads to increased system reactance by the contribution they add to the overall system reactance and results in further reduction of overvoltages for a network suffering from overvoltages.

2.6.5 Synchronous Machines

Another solution to reduce overvoltages is through synchronous machines. A synchronous machine if operated in underexcited mode [6], it absorbs excessive reactive power. However, if operated regularly for this purpose, equipment may accelerate electrical aging of the equipment due to mechanical stress, thus resulting in increased maintenance costs.

2.7. Reducing Steady State Overvoltages in Russian Power System Grid

Previous study [5] outlined the conventional solutions of reducing the steady state overvoltages in the Russian 500 to 750 kV power system grid. An algorithm was developed with an aim of addressing the steady stage overvoltages and also determining the sequence of transmission lines and shunt reactors switching in power systems.

The following possible solutions were found; generators set voltage can be controlled accordingly based on the voltage requirements of the network and reactive power control, transformers tap changing, EHV transmission lines disconnection, and shunt reactors.

2.8. Impact of Distributed Generation on Steady State Overvoltage

Another factor that affects the power system network behaviour is the introduction of Distributed Generation (DG). In a study conducted by [23] found that the operation of DG on lightly loaded grids results in steady state overvoltages in the localised network where DG is present. The voltage rise was linked to a widespread application of DG in power system. The first method this study looks at is based on the voltage level at which the DG is connected and the location of the machine, while the second method considers the three-phase fault level at the DG's point of connection.

The cited methods both tend to be overly conservative. Therefore, this study aimed to develop an accurate, generalised study method that could be used for this purpose. In developing a method to analyses the voltage rise a simple 4-node network was used and a simple algebraic equation, derived from a generalized model of a radial distribution line was used to gauge the effect of different network variables on the extent of network voltage rise in the presence of DG.

Similar studies conducted in [24] found that introduction of DG induces the local voltage constraints. The studies further demonstrated that DG influences voltage control when connected in the grid as it creates overvoltage during off peak load conditions, results in under voltages during peak load conditions when generation is not present, and un-optimize voltage settings at the High Voltage (HV)/Medium Voltage (MV) substation.

Therefore, an auto-adaptive regulator was tested and assessed. The results obtained shows that an auto-adaptive voltage regulator is able to maintain voltage plan on distribution network on normal and emergency conditions. Further study studies supported that one of the technical challenges of active networks is to maintain an acceptable voltage level [25]. This study presents a literature review on the various voltage control methods that have been implemented in distribution networks. The review used intelligent systems such as, genetic algorithm, simulated annealing, Tabu search and multi agent system for the decentralized voltage control in the system.

2.9. Summary

Under any standard, the proper approach to overvoltage is to control it or avoid it. Voltages can be grouped by being classified into number of sub-phenomena based on the duration in which an overvoltage event lasts [26]. The IEEE 1159 standard demonstrates how overvoltages [19,18], can be classified. The duration of the root mean square (rms) value is used for overvoltage classification.

Overvoltages are classified [12] into (I) instantaneous overvoltages if the duration is between 0.5 and 30 cycles, (II) transient overvoltages if the duration is between 30 cycles and 3s, (III) temporary overvoltages for durations lasting between 3s and 1 minute, and, lastly, (IV) steady state or long-duration overvoltages if the duration of the overvoltage is 1 minute or longer. This study focuses on the latter. Overvoltages can result in an increased flow of capacitive reactive power flow in the system [13], resulting in increased loading of equipment and increased active power system losses. Losses are undesirable in power systems, as they are regarded as wastage of power, and their costs ultimately have to be endured by the customers. Therefore, the reduction in losses presents a prospect for an efficient system. They need to be reduced to the lowest levels possible [15].

Sustained presence of overvoltages [13] above a specified safe limit of equipment can damage sensitive equipment such as shunt capacitors, reactors, and current transformers. Protection operation removes these equipments from the power system during limit exceeded conditions. This act may result in damaged insulation and flashovers, consequently resulting in damage and failure of equipment.

Also, some equipment are fitted with an overvoltage protection [27], and if voltages exceed protection levels, these may trip, thus leading to an outage of equipment. This can also lead to loss of supply in certain instances and requiring the load to be shed, thus reliability will be compromised.

In some instances, to reduce overvoltages, it is necessary to switch off equipment, such as capacitors through operating of the breakers. However, this can cause mechanical stresses on equipment, leading to accelerated mechanical aging. Recurring operation of breakers will result in additional maintenance and likewise, when overvoltages occur repeatedly, even if they are not destructive, they can cause accelerating electrical aging of equipment [28].

Synchronous machines can be used to reduce overvoltages if they operate in an under-excited state [2] in order to absorb excessive reactive power. However, this can affect stable operation if levels of excitation are too low, unless under-excitation limiters are used.

The most common solution to overvoltage problems is to install shunt reactors at the substations or at the ends of long transmission lines [2, 16]. These reactors can be fixed, breaker-switched, variable or controllable. Transformers [2, 19,29] also provide a means for controlling voltages via tap changers and can be used to buck overvoltages through the insertion of fewer turns of the primary winding.

Generators operated in the under-excited state can be used to absorb capacitive reactive power [12], thereby ameliorating overvoltages imposed on the system. The practice of switching out of lines to reduce overvoltages [19] is also used as an operational solution when no other facilities are available to control overvoltages. However, when load levels are significantly low, switching out of lines may no longer be a feasible solution, as overvoltages can no longer be reduced in this manner.

Chapter 3: Theory Review

3.1. Introduction

In this chapter the impact of using step-down transformers to reduce steady overvoltages is discussed. In Section 3.2 discusses the phenomenon of voltage rise under light loading network conditions and a simple 2 bus system diagram is used to explain the mathematically concepts such as the correlation between load and voltage; the impedance and voltage drop across the line as well as the induced line currents with the rise in voltage. In addition to this, Section 3.2 discuss the impact of using step-down transformers to reduce overvoltages, and mathematical equations to explain how the overall system reactance is increased by adding the transformers and the reduction in active power loses. Finally, the summary is given in Section 3.3.

3.2. Influence of Step-Down Transformers on Overvoltages

Under light loading conditions, voltage rises at the receiving end of the line [2]. This voltage rise results from the line charging currents from the line reactance. This usually induced by long lowly loaded overhead lines supplying lightly loaded networks. For radial lines with fixed sending end voltage, the load at the receiving end is expressed as:

$$P_R + jQ_R \tag{3.1}$$

where,

P_R is real power at the load bus in MW
Q_R is reactive power at the load bus in MVAr

Fig.3.1, demonstrates a simple two-bus system, comprising a transmission of reactance X, supplying a load, $P_R + jQ_R$, positioned at the end of the line and with an assumption that resistance R is negligible.

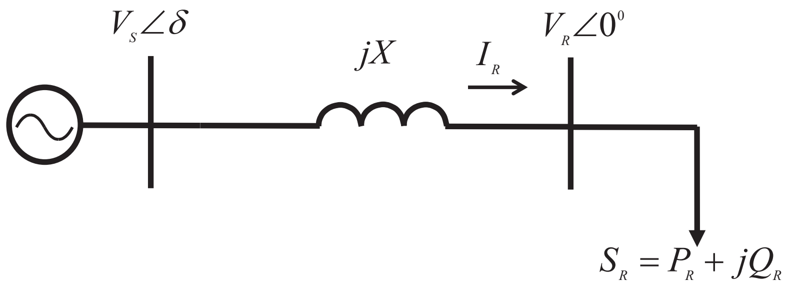

Fig.3.1. Simple 2-bus system showing a transmission line supplying a load.

The voltage drop along the line between the sending end voltage and the receiving end voltage at the load side can be expressed as follows:

$$V_D = ZI_R \tag{3.2}$$

The relationship between sending end voltage, receiving end voltage, and the complex load can be expressed as follows [30]:

$$V_S\angle\delta = V_R\angle 0^0 + \frac{XQ_R}{V_R} + j\frac{XP_R}{V_R} \tag{3.3}$$

where,

V_S is the sending end voltage in kV

V_R is the receiving end voltage in kV

S_R is the apparent power in MVA

P_R is the active power in MW

Q_R is the reactive power in MVar

Equation (3.3), demonstrates that the load at the receiving end of the line in equation (3.1), can influence the voltage at the receiving end of the line V_R. For a constant sending end voltage V_S, V_R will increase as the load decreases.

Consider the relationship in, between the load in equation (3.1) and equation (3.2) with respect to Equation (3.3); where the change in voltage drop $V_D = ZI_R$, is proportional to the load $P_R + jQ_R$, and the receiving end voltage, V_R is inversely proportional to the load.

The reduction in load leads to the reduction in reactive power, therefore freeing some capacitive reactive power in the system. This will cause a rise in the receiving end voltage, V_R, and if there is a significant drop in load, acceptable voltage limits can be exceeded.

A rise in the receiving end voltage on an open circuit results due to line charging [2]. Further to this, the transmission line has capacitive properties due to the capacitance formed between phases of the transmission line and formed between transmission lines and ground. This transmission line generates capacitive current through the capacitance, which can be written [2] as,

$$I_C = YV \tag{3.4}$$

where,

I_C is the capacitance charging line current in kA
Y is the shunt admittance per unit length in SI
V is the operation Voltage of the line

It can be seen from the charging current that the line is a source of capacitive reactive power and this reactive power is proportional to the square of the line operating voltage and may be written as

$$Q_C = BV^2 \tag{3.5}$$

Where,

Q_C is the charging reactive power in MVar

B is the line suceptance of the line in Siemens (SI)

V is the operation Voltage of the line

The active power losses when considered, can be expressed as,

$$P_{loss} = I^2 R \tag{3.6}$$

Where,

P_{loss} is the active power loss in MW

I is the current along the line in kA

R is the resistance of the line in Ω

The light loading condition on the network results the reactive power will be consumed to offset the generation of capacitive reactive power by the line. Subsequently, the network will be exposed to increased total current flow which will lead to a rise in active power losses.

3.3. Summary

It can be seen from theory that there is a significant correlation between voltage and load. This is due to the impact the load had on the receiving end voltage when the line is lightly loaded, which results in increased receiving end voltage.

This voltage when the load is significantly low, it will exceed the maximum voltage limit. When the voltages on the lines are derated the line charging parameter will reduce the generated capacitive reactive power, considerably; therefore reducing the system voltages.

To start with, as shown in equation (3.1), the additional reactance the line reactance, and thus the voltage drop along the corridor will increase. Therefore with the introduction of transformers the bus voltages will decrease as a result of the increase in voltage drop flowing along the corridor.

Secondly, the introduction of reactance by the installation of transformers will induce means of reactive power being generated by the line being consumed. The generated reactive power is described in equation (3.5). High voltage problems will be reduced with the offsetting of excessive generated reactive power. In addition to this, a reduction in current flow on the lines will be effected by the decreased reactive power generation. Active power losses will reduce with the reduction in currents flowing in the system.

Lastly, the transformer installation comes with the on load tap changers. The on load tap changers are able to regulate voltage directly by means of a tap changer action so as to retain voltages within acceptable limits. Therefore the transformer functionality provides an additional voltage control that leads to further reduction in voltage.

The reduction in system voltages will enable lesser load to be accommodated without excessive steady state overvoltages in the power system, thus improvement can be observed in steady overvoltages.

The expectation of this study is that with the installation of step down transformers along the network corridor with long distribution lines will minimize both high voltages and reactive power flow on derated lines. The tap changing capacity of the transformer will enable additional voltage regulation of voltage; therefore further reduction in high voltages is expected.

Chapter 4: Methodology of the Study

4.1. Introduction

This section discusses the methodology of the study. Section 4.2 covers the software models used for the study. The load flow simulations are covered in Section 4.3. Section 4.4 covers the technical criterion used for the technical assessment of the case study. The summary is drawn in Section 4.5.

4.2. Software Model(s) used for Technical Evaluation

The software used for the technical evaluation of the study for conducting the power system simulations is the PSS/E [31]. To assess the effectiveness of installing the step-down transformers in order to reduce overvoltages, a 14-bus IEEE test system in Fig. 5.1, is built and modelled in PSS/E and the results are discussed in Chapter 0.

Part of additional work within this study considered the automation of the load flow calculations where the load flows were automated through programming. The author chose Python Programming language [32] to implement the program.

4.3. Load Flows

The steady state analysis was carried out for system healthy and contingency conditions for various load levels under off-peak conditions. The load flow analysis was conducted for light load conditions based on the original case without step-down transformers (Case A) and one with step-down transformers (Case B).

4.4. Criteria used for Technical Assessment

The planning criterion [3] was used for the technical evaluation of the load flows carried out for monitoring of the distribution busbar voltages in both Case A and Case B. The busbar voltages we kept within the planning minimum and maximum limits of 0.95 and 1.05 per unit [3], respectively. For instances where the bus bar voltages exceed the maximum limit of 1.05 per unit after a line contingency, a second line was disconnected to further reduce the voltages below 1.05 per unit.

4.5. Summary

The evaluation of the impact of using step-down transformers is carried out through conducting load flows for system healthy conditions for Case A and Case b, and voltages are captured. Line contingencies are carried out for the recorded voltages above maximum limit of 1.05 per unit, and busbar voltages are monitored. In some occurrences in order to restore voltages within the safe operating voltage limit of 1.05 per unit [4], more than one line must be switched out.

A second line is only switched out for busbar voltages that are still above 1.05 per unit following switching out of the first line. This is to restore voltages within maximum voltage limits. Voltages are once more monitored to observe the effect of a derating voltage, and the results are captured and discussed in the results section.

Chapter 5: Case Study for Regulation of High Voltages Through the use of Step-Down Transformers

5.1. Introduction

Chapter 5 present the case study for both manual and automated load flow simulations. Section 5.2 discusses the case study using the IEEE 14 bus system for the assessment of the proposed solution for the study. Section 5.3 explains the data used for the study and how the software models are configured to represent a power system. Section 5.4 discusses how the introduction of transformers reduces active power losses to improve network efficiency. The automation of the load flow studies using Python program, the illustrative example and data used is explained in Section 5.5. The summary is covered in Section 5.6.

5.2. Case Study

A study was conducted on the evaluation of the impact of line derating using the step down transformers to attempt to enhance the system reliability. A case study was conducted using the IEEE 14 bus system, and its single line diagram is shown in Fig. 5.1. The parameters of these lines are shown in Case Preparation Section 5.2.the section below.

The network used has relatively long lightly loaded transmission lines which supplies lowly loaded distribution network which also comprises of long corridors. The length of these corridors reduces the effectiveness of voltage regulation using transformers. The section below presents the step followed in reducing the steady overvoltages to improve the reliability of this network.

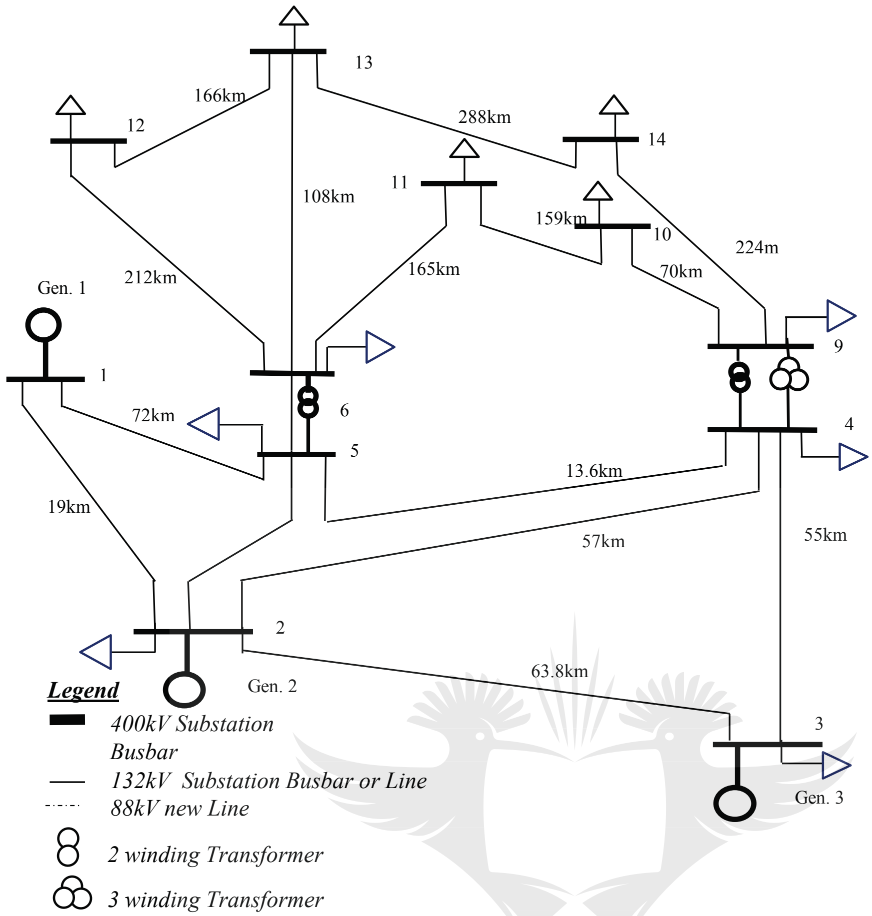

Fig. 5.1. IEEE 14 bus system diagram

5.3. Case Study Preparation

To prepare the case, generation, transmission and distribution data is prepared to represent the power system using the IEEE 14 bus system. For the business case i.e., case without de-rating, Case A and the case with de-rated lines, Case B, the transmission voltages of 400kV, is used and remains unchanged.

The model utilises 400/132kV transformers for Case A. Both voltage levels are calculated using an MVA base of 100MVA [33]. The system is then modified by replacing 400/132kV transformers with 400/66kV transformers (to create Case B).

TABLE 5.1: Transmission and distribution line parameters – Case A

Lines	Voltage (kV)	Resistance (R)	Impedance (X)	Charging (B)	Line Length (km)
L1-2	400	0.00029	0.00381	0.11229	19
L1-5	400	0.00108	0.01437	0.42326	72
L2-3	400	0.00096	0.01276	0.37569	64
L2-4	400	0.00085	0.01136	0.33460	57
L 2-5	400	0.00084	0.01120	0.32997	56
L3-4	400	0.00083	0.01102	0.32456	55
L 4-5	400	0.00020	0.00271	0.07991	14
L6-11	132	0.18125	0.40088	0.07744	165
L6-12	132	0.23310	0.51559	0.09960	212
L6-13	132	0.11871	0.26256	0.05072	108
L 9-10	132	0.07700	0.17031	0.03290	70
L9-14	132	0.24638	0.54495	0.10527	224
L10-11	132	0.17502	0.38712	0.07478	159
L12-13	132	0.18214	0.40286	0.07782	166
L13-14	132	0.31713	0.70144	0.13550	288

The type of conductor used for transmission lines is the Tern conductor and for distribution is the Wolf conductor. The line parameters for Case A are shown in Table 5.1. To prepare Case B, the 132kV distribution lines were de-rated to 66kV and the correct line the parameters, i.e., resistance (R), impedance and charging (B) were updated accordingly.

Table 5.2, shows how the line parameters change when the distribution network is derated from 132kV to 66kV. It should be noted that the line lengths are not changed and only the R, X and B values were calculated.

TABLE 5.2: Transmission and distribution line parameters – Case B

Lines	Voltage (kV)	Resistance (R)	Impedance (X)	Charging (B)	Line Length (km)
L1-2	400	0.00029	0.00381	0.11229	19
L1-5	400	0.00108	0.01437	0.42326	72
L2-3	400	0.00096	0.01276	0.37569	64
L2-4	400	0.00085	0.01136	0.33460	57
L 2-5	400	0.00084	0.01120	0.32997	56
L3-4	400	0.00083	0.01102	0.32456	55
L 4-5	400	0.00020	0.00271	0.07991	14
L6-11	66	0.72498	1.60353	0.01936	165
L6-12	66	0.93242	2.06234	0.02490	212
L6-13	66	0.47483	1.05024	0.01268	108
L 9-10	66	0.30800	0.68124	0.00823	70
L9-14	66	0.98553	2.17981	0.02632	224
L10-11	66	0.70009	1.54847	0.01870	159
L12-13	66	0.72856	1.61143	0.01946	166
L13-14	66	1.26852	2.80574	0.03388	288

5.3.1. Transformer Modeling

The power transformers modelled in Case A and Case B are rated at 50MVA, and placed as shown in Fig. 5.2. The impedance value to transformers on buses 5 and 6 is 0.00275 per unit.

The impedance value for transformers on buses 4 and 9 is 0.00608 per unit and parallel with a three-phase transformer with the following parameters, primary to secondary reactance of = 0.32527 Ω per unit, secondary to tertiary reactance = 0.28616 Ω per unit, and primary to tertiary reactance =0.31913 Ω per unit.

5.3.2. Generator Modeling

The data in Table 5.3 illustrates the data used to model three generators within the used model in PSS/E for the network supplying the transmission and distribution load centre. The busses for the three generators are as shown in Fig. 5.2.

TABLE 5.3: Generator bus data

Generator bus no	1	2	3
P (MW)	615	60	60
Pgen (MW)	615	0.4	60
Qgen(MVAr)	0	-0.424	-10
Pmax(MW)	700	70	70
Pmin(MW)	0	0	0
Qgen. min(MVAr)	10	10	10
Qgen. max(MVAr)	-10	-10	-10

5.3.3. Load Modeling

The distribution load centre loads shown in Table 5.4 are the loads used in the study for light load network condition. The loads remain unchanged for both scenarios, i.e., Cases A and B. The number of load levels starts from the highest load level assuming a peak load value as the highest load level and the least load level as the last load level for lightly loaded network conditions. The loads remain unchanged for both scenarios, i.e., Cases A and B.

The first section of the study uses the load levels shown in Table 5.4, where the simulation model for both Case A and Case B studies simulates the off-peak load conditions and starts from the highest level (Level 1) to the lowest load level (Level 10).

TABLE 5.4: Transmission and distribution load bus data

Bus number	P (MW)	Q (MVAr)
2	21	0.127
3	94	0.19
4	48	15.7
5	7	0.016
6	1.1	0.36
9	2.9	0.99
10	0.9	0.3
11	1.35	0.44
12	0.6	0.2
13	1.3	0.43
14	1.5	0.5

For this section of the study the technical assessment is only carried out for system healthy network conditions for all load levels and the following are recorded, i.e., busbar voltages, reactive power flow on the lines and the active power losses. The results for Case A and Case B are evaluated by comparing busbar voltages and active power losses in every load level and the results are plotted and discussed in Chapter 6.1.

5.4. Derating Line Voltage to Limit System Overvoltages and Enhancement of Network Operational Efficiency

The transformer are located at the ends of corridors comprising a number of substations and in such a manner that at one end voltage is stepped up and at the other it is stepped down; this is to ensure that the operating voltage of the corridor between these two transformers is derated.

The improvement of reducing steady overvoltages and decreasing active power losses in the area of study through the introduction the transformers can be explained mathematically.

The mathematical relationship between the generation of capacitive reactive power produced by a transmission line is proportional to the square of voltage. Thus, it is expected that the use of transformer reduces both reactive power production, steady overvoltages and reduced system active power losses.

Table 5.5, shows the load levels used to carry out the load flows to calculate and record the reactive power flow on the lines, the voltages at various busbars and the active power losses on the lines, for both Case A and Case B. The results of this assessment are captured in Section 6.2.

TABLE 5.5: Transmission and distribution load bus data

Bus number	LL1	LL2	LL3	LL4	LL5
	P(MW)	P(MW)	P(MW)	P(MW)	P(MW)
2	8.4	4.2	2.1	1.05	0.84
3	38	18.8	9.4	4.7	3.76
4	19	9.6	4.8	2.4	1.92
5	2.8	1.4	0.7	0.35	0.28
6	4.4	2.2	1.1	0.55	0.44
9	12	5.8	2.9	1.45	1.16
10	3.6	1.8	0.9	0.45	0.36
11	5.4	2.7	1.35	0.675	0.54
12	2.4	1.2	0.6	0.3	0.24
13	5.2	2.6	1.3	0.65	0.52
14	6.0	3	1.5	0.75	0.6

5.5. Algorithm used for Performing Load Flows at Various Load Levels

This section presents the advancements of the study by introducing an automation of the load flows. The approach of performing power system analysis at various load levels through gathering data for the assessment of study can cumbersome, time consuming and laborious. Hence, an effective approach of introducing automation of the load flow studies by use of Python programming language was considered.

To determine the lowest load that can be accommodated by the network without violating voltage limits and thermal capacities of lines, the assessment is carried out through the steps shown in the pseudo code in Fig. 5.2.

Python is used to interface with PSS/E using the Application Program Interface (API) [34]. The software code is programmed in Python and the interfacing between Python and PSS/E is made possible by using the API to produce a syntax. The program is able to call files in PSS/E, reduce load and extract results. The pseudo code of the program is shown in Fig. 5.2.

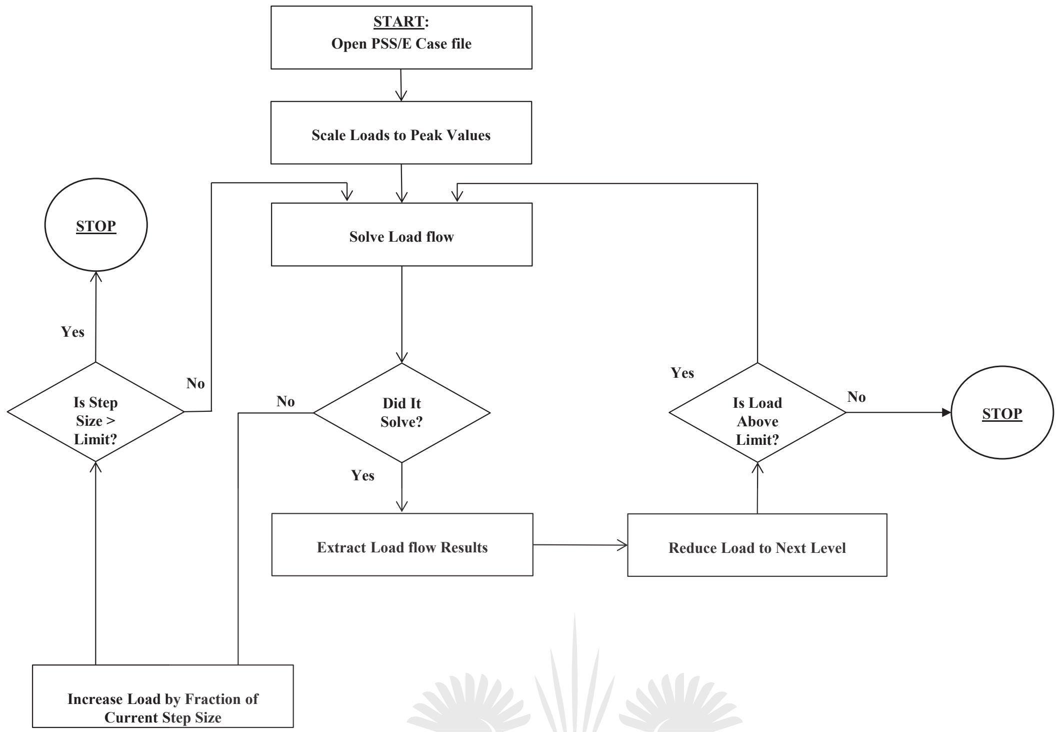

Fig. 5.2. Flow chart showing steps for load flow studies at various load levels.

The first step from the pseudo code is:

- Issuing of an instruction by Python to PSS/E to open a required file which contains data required for the load flow analysis.
- Once the file is open, the next instruction is to scale the load in the network to its peak value and then solves the load flow.
- If the case converges successfully, the results are extracted and the results of this tested load level are extracted and stored in a solution array. The recorded data includes voltages for selected busses, reactive power flowing in selected lines, reactive power in the network and active power losses in the system.
- If at any given point the solution of the load flow fails to converge, the program attempts to obtain a solution by reducing the step size and checks if the step size is not less than some lowest size desired, adding the new step size and attempting the loadflow solution.
- If the solution or the case file converges, the iteration is repeated for the next load level, at a percentage drop of 10%, and the scaled load is now reduced from the previous step.

If the case converges and the solution is met with no network violations, the results are stored into the solution array for this load level.

- The steps are repeated until the least load level for both Case A and Case B, are reached.

The results for voltages, system reactive power flow and reactive power flow on the lines, as well as system losses are recorded and extracted for all load levels, and the results are included in Section 6.4.

5.5.1. Illustrative Example for Automation using Python

An IEEE 14 bus system single line diagram is shown in Fig. 5.1 was used for the case study to evaluate the impact of line derating using the step down transformers. The network parameter used was conducted using the IEEE 14 bus system are as shown in Table 5.1 and Table 5.2, in former sections and also available in [35].

5.5.2. Automated Python Program Data

The written Python program described in the previous sections was used in conducting the simulations, achieving adjusted loading of the changes to the network and extracting data from the PSS/E network case file.

The initial peak load of 120MW was set as the highest load level and it was reduced in steps of 1MW down to the lowest load of 1MW. The load flows were carried out for system healthy network conditions and carried out at each of the load levels.

The voltages at busses 6, 9, 10, 11, 12, 13 and 14, reactive power flow in the system and flowing through lines 6-11, 6-12, 6-13, 9-10, 9-14, 10-11, 12-13 and 13-14, and active power system losses recorded at each load level.

5.6. Summary

In this chapter the simulation studies though the steady state analysis for a lightly loaded network comprising of long corridors was carried out. Due to the length of these corridors the effectiveness of voltage regulation using transformers was reduced.

The load flows were carried out under off-peak network condition for system healthy and contingency conditions at various load levels. The load flow analysis was conducted for both the original case without step-down transformers (Case A) and one with step-down transformers (Case B).

A case study was conducted using the IEEE 14 bus system, and its single line diagram is shown in Fig. 5.1. This bus system was used for the evaluation of the impact of line derating using the step down transformers with the objective of reducing steady overvoltages and enhancing the system reliability. The parameters for the lines, transformers, generators, and loads are also covered in this section.

The planning criterion [3] was used for the technical evaluation of the load flows. The monitored distribution busbar voltages had to be within the planning minimum and maximum limits of 0.95 and 1.05 per unit [3], respectively. The load flows were carried out and thee busbar voltages are captured for system healthy conditions for both cases. The line contingencies were carried out and in some instances recorded voltages exceeded the maximum limit of 1.05 per unit. In order to restore voltages within the safe operating voltage limit of 1.05 per unit [4], more than one line had to be switched out.

In instances where recorded voltages were still above 1.05 per unit following switching out of the first line, a second line is only switched out to restore voltages within maximum voltage limits. Voltages are once more monitored to observe the effect of derating voltage, and the results are captured and discussed in the results section. Part of the technical assessment evaluates the active power losses and reactive power flow on the lines for both cases with and without transformers, recordings are made and a comparison is made and discussed fully in Chapter 6.1.

The latter part of this chapter presents the advancements of the study by introducing an automation of the load flows. This is so as to reduce the cumbersome, time consuming and laborious way of running load flows, manually. Automation of loadflows was achieved through an introduction of Python programming language.

Chapter 6: Results and Discussions

6.1. Introduction

This section presents the results of the illustrated case study in Section 5.2. The load flow results for maul loaf flows is briefly explained in Section 6.2 where voltages and reactive power flow for Case A and Case B comparisons. Section 6.3 discusses the results of the enhancement of the network efficiency. Section 6.4 explains the automation results obtained through the use of Python program. Section 6.5 briefly summaries this chapter.

6.2. Loadflow Simulation Results

The results for the technical evaluation of the recorded voltages during the load flow analysis for system healthy network conditions is demonstrated and summarised in Fig. 6.1. This is where a comparison of two studied scenarios of the recorded voltages at all distribution busbars under system healthy is performed.

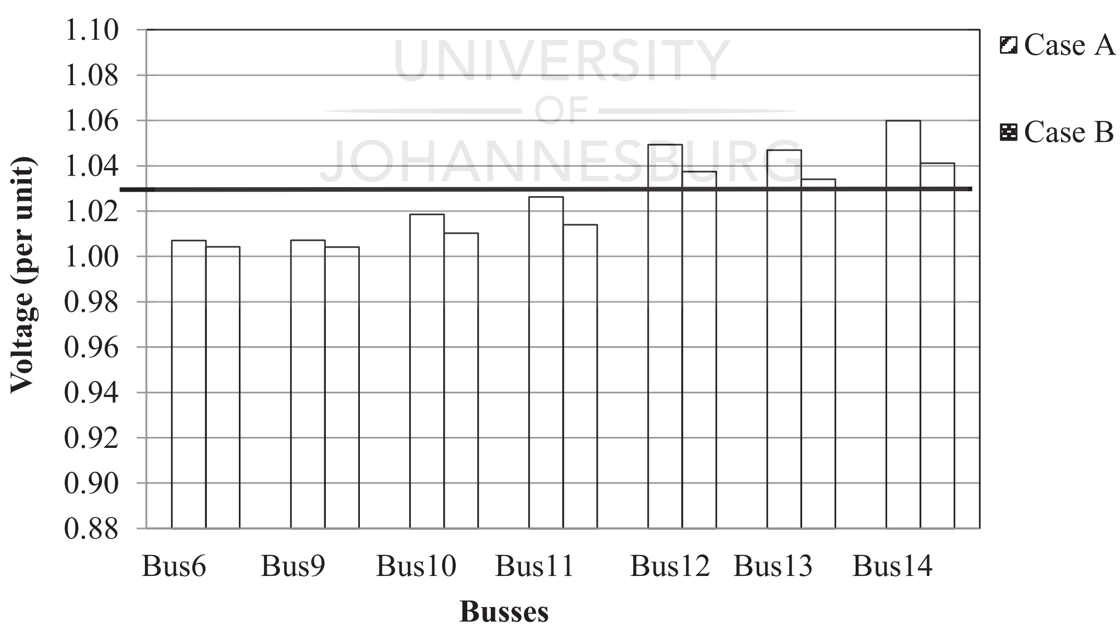

Fig. 6.1. Recorded voltages for system healthy assessment at all distribution busbars for both Cases A and B.

For the network with 132kV lines, i.e., base case, calculated voltages resulted in relatively high values under very light loading conditions. In Fig. 6.1 for Case A voltages on bus 14 are recorded at 1.06 per unit, therefore exceeds the maximum 1.05 per unit voltage limit.

The restoration of the voltage within the maximum voltage limit is achieved by switching out line 9-14. This mitigation of voltages is above the maximum limit and rise beyond the N-2 operating limit of 1.09 per unit to 1.1 per unit on bus 14.

In order to reduce and restore the voltages to acceptable voltage levels, the only option is to switch out a second line. In this case line 13-14, is switched out. With line 13-14 switched out, the voltages are restored to 1.05 per unit; however this result in the load on bus 14, islanded.

As a result of this, the customer at bus 14 cannot be served and this event can also be understood as an unreliability event. For the modified case i.e., Case B, where derating to 66kV has been effected, there is noticeable drop in the recorded voltages and this is demonstrated in Fig.6.1. The voltages in Case B are all recorded within the 0.95 and 1.05 per unit planning limit; thus no contingency study is required. Therefore, in this scenario, there is no risk of an unreliability even resulting from high voltages.

The load flow results of the reactive power flow on the lines for both Case A and B are captured in Fig. 6.2. Case A, results demonstrate a high recording of the reactive power flows with values reaching 18MVAr on line 6-13. However, for the same line, in Case B, the recorded reactive power flow is 4.5MVAr.

The mathematical relationship between reactive power flow and voltage is clearly solidified by the results of this study. The results prove that the steady overvoltages can be reduced by installing step-down transformers along the corridors of long and lightly loaded lines.

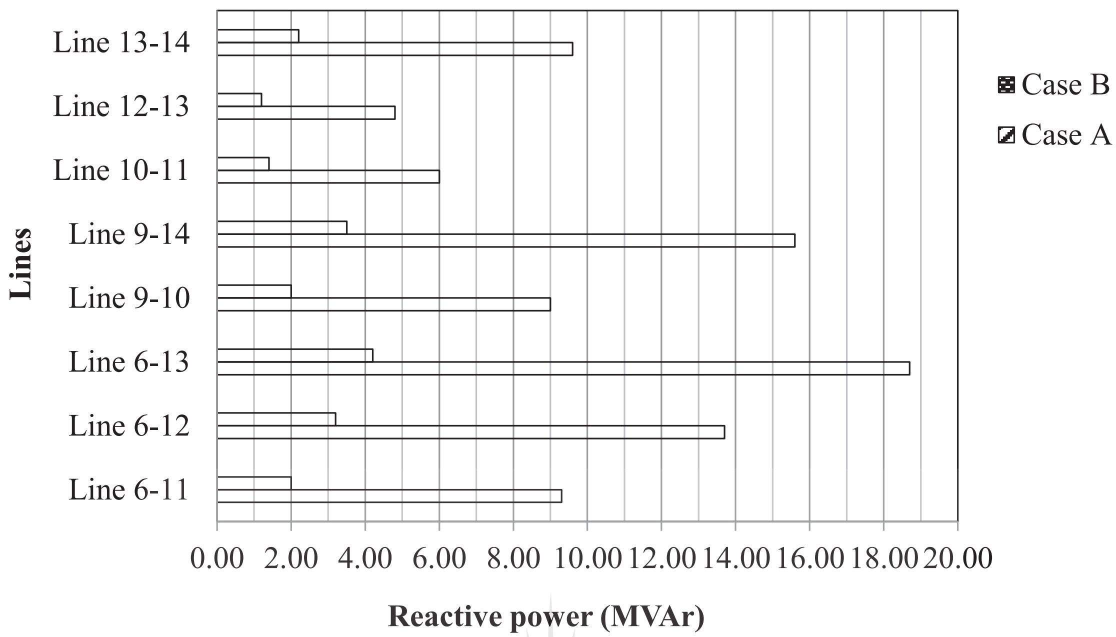

Fig. 6.2. Reactive power flow on the distribution lines for Cases A and B.

The study also shows that with the derated case, system reliability of lightly loaded networks can be improved through installing transformers, without customers supply compromised.

6.3. Loadflow Simulations Results for the Enhancement of the Network Operation Efficiency

This section presents the results of the illustrative case study in Section 5.2, where load flows are carried out to assess the power system with embedded transformers. The load levels used to assess and record the reactive power flow, active power flow on the lines and voltages, can be obtained in Table 5.5. The load flows are carried out for all five load levels.

6.3.1. Reactive Power Flow on Distribution Lines Results

Fig. 6.3 presents the calculations of reactive power in the distribution lines. This calculation is carried out at various load levels. The load levels assessment starts form from load level 1 to 5.

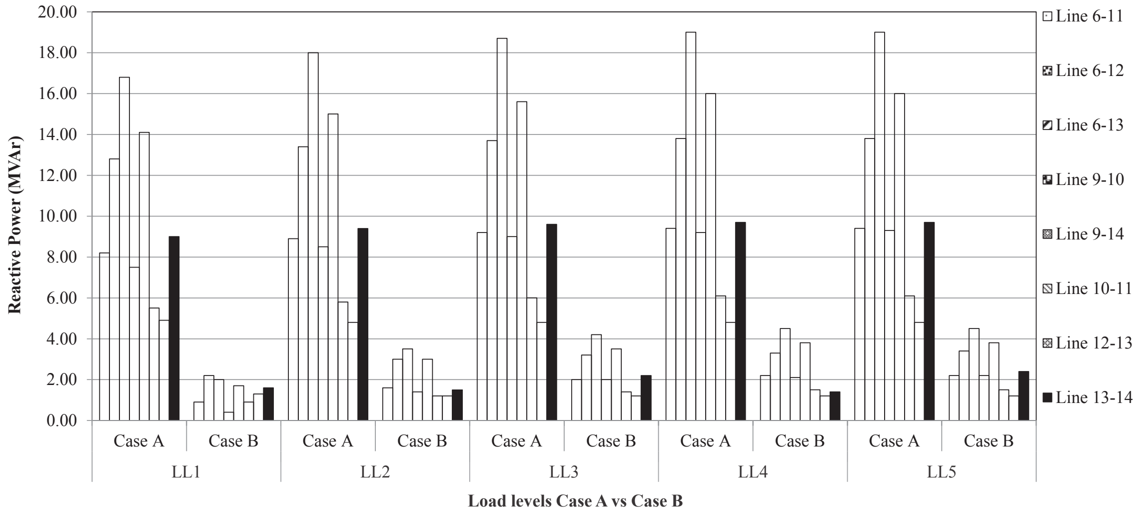

Fig. 6.3. Reactive Power Flow on the lines.

It can be seen in Fig.6.3 that as the load decreases, from load level 1 to level 5, there is an overall increase in the reactive power flowing in the system for both cases, i.e., without (Case A) and with (Case B) step down transformers. Additionally, it is demonstrated from Fig. 6.3 that the impact of introducing step down transformers is to commonly reduce the reactive power flowing in the lines in the network.

6.3.2. Case A and B System Healthy Voltage Results

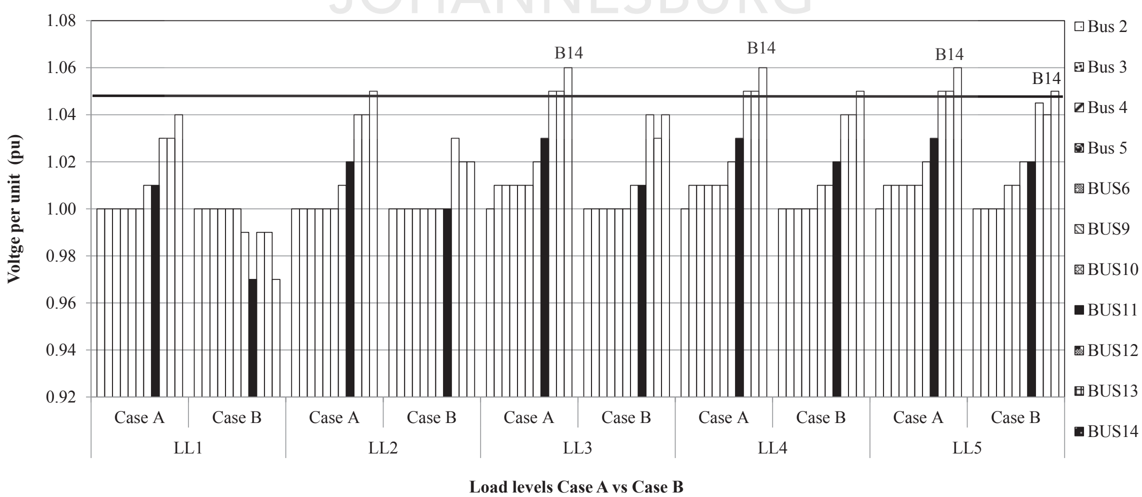

Fig. 6.4. Recorded voltages for system healthy network for cases A and B.

The voltages in the system are summarized in Fig. 6.4. It can be observed that without the step down transformers the voltages in the network are generally greater than those recorded from the case with step down transformers. In Fig. 6.4 voltage decrease at some of the load levels is noticeable however the voltage at bus 14 exceeds the planning limits. In the case of a derated case i.e., Case B, all voltages at all load levels are within the acceptable planning voltage limits.

6.3.3. Change in System Losses Results

Lastly, Fig. 6.5 presents the active power losses at various load levels for both Case A and Case B. Fig. 6.5 also shows the variation of active power losses with the introduction of step down transformers.

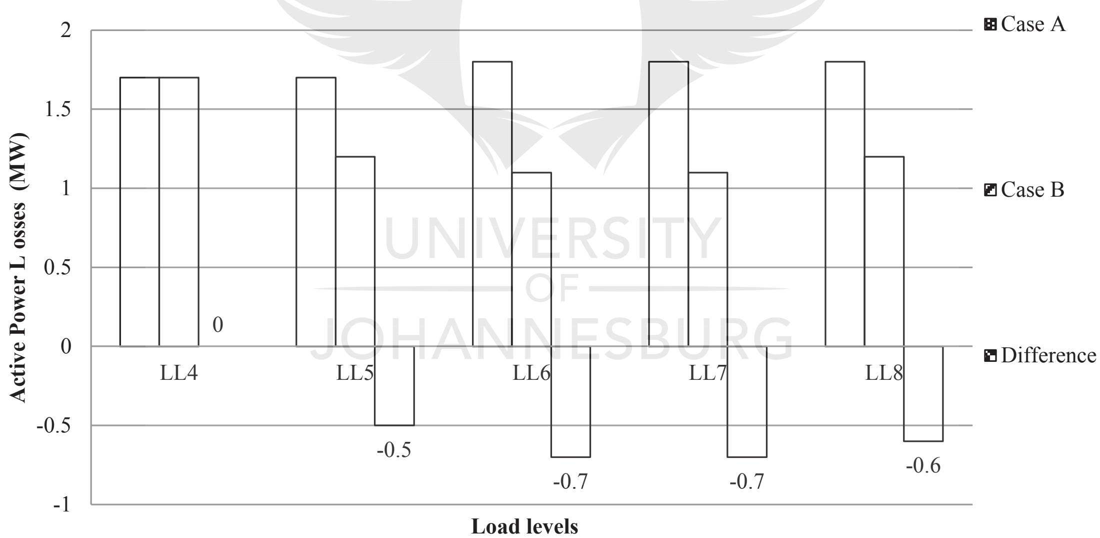

Fig. 6.5. Reactive power flow on the distribution lines for Cases A and B.

It can be seen that with Case B, the active power losses are generally low in the network. It can also be seen that with the decrease in load towards the lower load levels, there is also a decrease in active power flow. The results on the investigation to use step down transformers as a solution for reducing high steady state voltages is proven by the generated positive results.

The assessment further shows how the derating of lines impacts the network reactive power flow and active power losses, under light loading conditions. The findings and results demonstrated that the installation of step down transformers was extremely effective in reducing reactive power flow in the system. Further to the reduction of reactive power flow in the system, an enhanced the ability of the network to control voltages effectively resulted in the set upper voltage limits not exceeded.

Lastly, the overall reduction of the active power losses in the system was achieved. The study proved that the proposal of using of transformers can be an effective solution for reducing high system steady voltages and therefore should be one of the considerations of the solutions for steady overvoltages reduction.

6.4. Automated Loadflow Simulation Results for the Serving low load levels Using Stepdown Transformers

6.4.1. Active Power Losses Results

The results for the active power load in the system against the steady state overvoltage variation at bus 10, are plotted in Fig. 6.6. It can be seen that generally the reduction in load leads to voltage increasing above the acceptable limit without line derating.

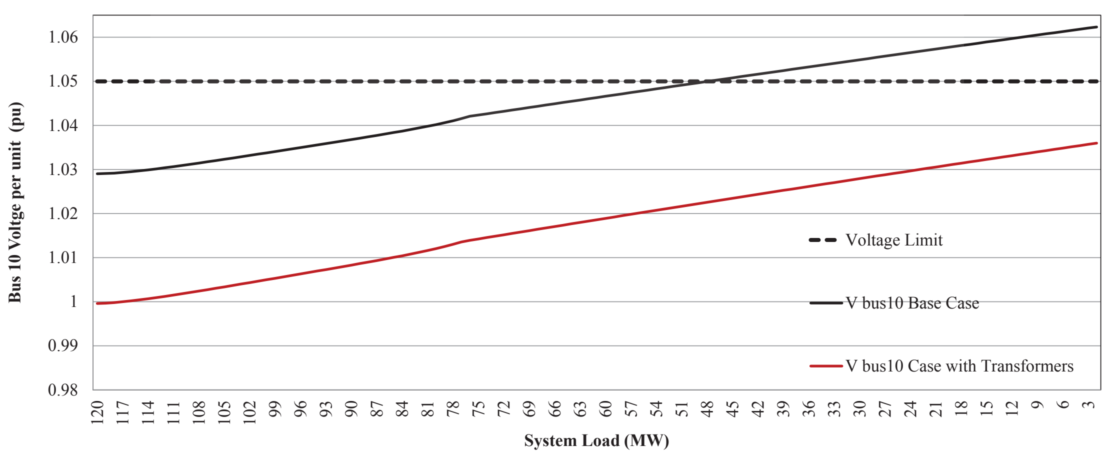

Fig. 6.6. Variation of voltage at bus 10 with active power load in the system for (I) case without step down transformer and (II) case with transformers.

For the base case, at initial system load level of 120MW the voltage at bus 10 is 1.03 per unit and reaches the maximum limit of 1.05 per unit and beyond at the 48MW load level. In the case of derated lines, the bus 10 voltage starts at about 1.0 per unit and reaches the highest voltage of 1.035 per unit at the lowest load level, therefore is regulated below the upper 1.05 per unit steady state limit.

6.4.2. Reactive Power Flow Results

In Fig. 6.7, the analysis is on line 6-13, where the variation of reactive power flowing on the line is plotted against the reduction in load.

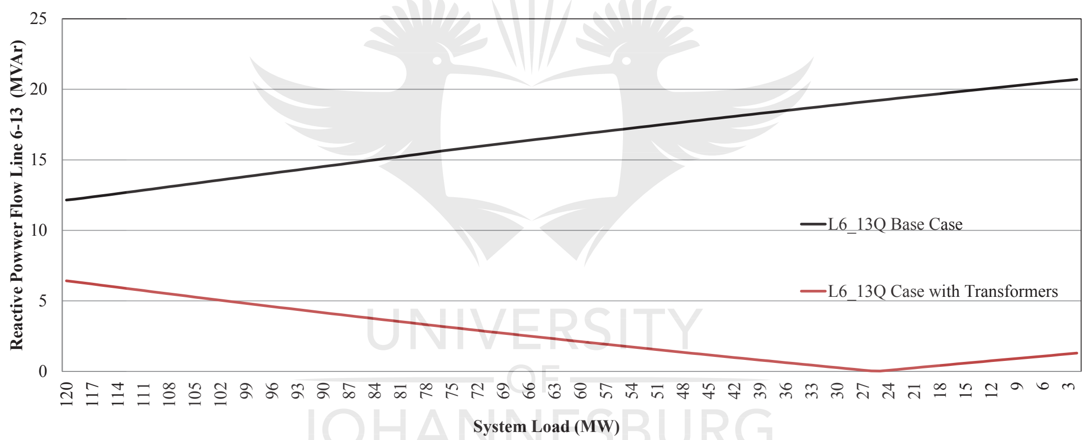

Fig. 6.7. Variation of reactive power in the network with the active power load in the system for (I) case without step down transformer and (II) case with transformers.

The reactive power flow in the system in Fig. 6.7 increases from 12.5MVar to 21MVar as the load is reduced from the peak value to the lowest size, for the base case. The introduction of step down transformers restrains the reactive power within 0 to 5MVar.

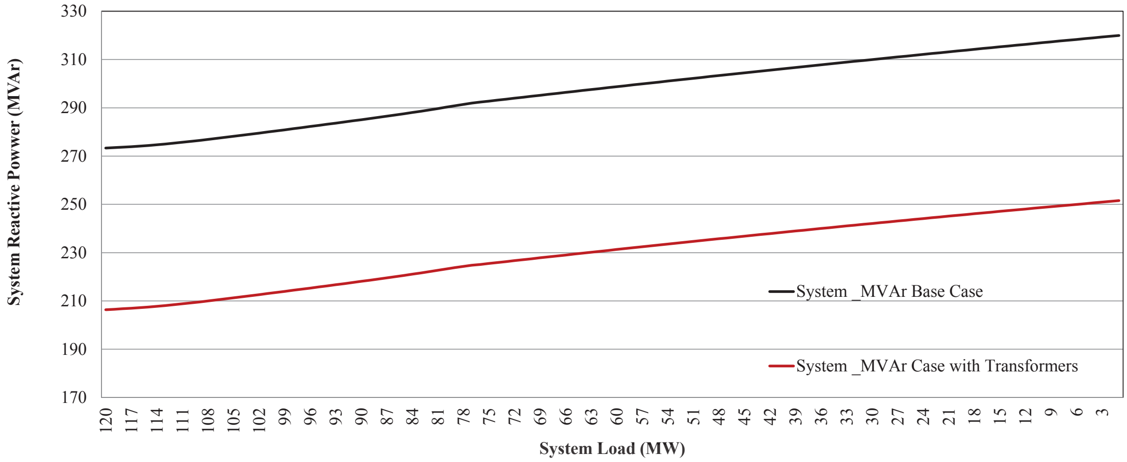

Fig. 6.8. Variation of system reactive power with active power load in the system for (I) case without step down transformer and (II) case with transformers.

An equivalent observation in Fig. 6.7, is also made in Fig. 6.8, where in the case without step down transformers, the system reactive power varies between 271 and 320 MVar.

The incorporation of derating transformers retains the system reactive power between only 208 and 252MVar. This is a significant improvement for the overall system reactive power flow wen a comparison is made between the two cases.

6.4.3. Active Power Losses Results

Fig. 6.9, presents the results of the calculated total system active power losses at various load levels. The active power losses at 120MW, 50MW and lowest load level, ranges at 2.5MW, 1.75MW and 2.0MW, respectively.

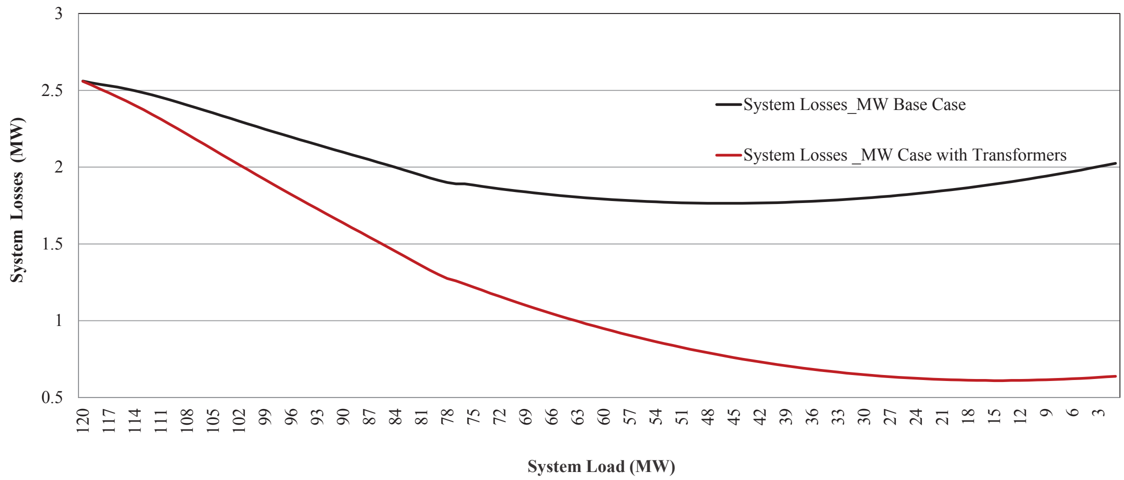

Fig. 6.9. Variation of system active power losses with active power load in the system for (I) case without step down transformer and (II) case with transformers.

For the derated transformer case, the losses vary from 2.5MW at peak load and at the lowest load level to 0.6MW. It can also be seen from the plotted results in Fig. 6.9, that the variance in losses is substantial for the base case and the derated case, with lower losses in the derated case.

6.5. Summary

In this section reports on an investigation into the use of step-down transformers located along the corridors feeding a long centre to control overvoltages. It has been proven in the results that in regulating these overvoltages lines are required to be disconnected to restore voltage to values within acceptable limits.

It has also been demonstrated that as the load drops further, additional lines have to be disconnected. This disconnection of additional lines may lead to some loads being islanded, and therefore, not being served. The reliability of supply is adversely affected by this. Additional possible impacts of such a solution under light loading conditions on reactive power flow and active power losses were assessed.

The outcome of this assessment found that the use of step down transformers to derate the network was very effective in reducing total system reactive power flow. By achieving this, the ability of the network to control voltages not to exceed set upper limits, was improved. Further, a significant reduction in the active power losses in the system, were achieved.

The latter part of the study investigated the use of line derating to accommodate lowest load in the network through also performing load flows. The loads flows were carried out at various load levels and network performance was assessed at each load level. A software program, written in Python and using PSS/E to perform calculations, was used to automate the calculations.

The study revealed that numerous beneficial effects on system performance as the system load was reduced can be achieved through the introduction of line derating. Reduced voltages and lesser reactive power flow and active power losses were recognized in the case of derated lines, demonstrating an enhanced capability of the system to serve low loads.

Chapter 7: Literature Review on Tabu Search

7.1. Introduction

This section discusses the tabu search (TS) literature and its optimisation applications in power systems. Section 7.2 discusses the literature survey of TS and its ability to solve combinatory problems in power systems. In section 7.3, summarises the techniques TS applications in power systems. Section 7.4 explains other type of existing algorithms with reference to TS as well as its application to solve optimisation problems in power systems. Section 7.5 summaries the TS literatures scan.

7.2. Literature Survey

TS is a meta-heuristic that controls a local heuristic search procedure to search the solution space beyond local optimality [36]. Tabu Search uses adaptive memory to generate flexible search behaviour.

It is an iteration algorithm which uses a neighbourhood search technique and it is restricted by it [37]. Its fundamental use of flexible memory of search history guides the search process to overcome local optimal solutions.

The fundamental elements of the of the TS are the moves, tabu list and objective level [36, 37]. The TS method is a rapidly chosen technique of choice for designing solution procedures for complex combinatorial optimization problems [36,38].

TS methods have also been applied to create hybrid techniques procedures with other heuristic and algorithmic approaches, to deliver improved solutions to problems in scheduling, sequencing, resource distribution, investment planning, telecommunications and many other areas [36].

The foundation of TS is based on problem solving [36], and its ability to incorporate adaptive memory and responsive exploration, qualifies its intelligence. The adaptive memory feature enables the application of procedures that are capable of searching the solution space carefully and efficiently.

TS algorithm is applied in order to limit the solution effort, therefore making it a modern optimisation technique that can achieve an optimal or sub-optimal solution within a realistic short time [39].

7.3. Tabu Search Applications in Power Systems

Tabu Search has been applied to a number of optimization problems, successfully [40]. Its efficiency and flexibility contribute vastly to its applications being successful. Hence, numerous experiments have demonstrated that when TS is suitably implemented, the results obtained from tabu search often surpass those of the previous known methods [40].

TS is an approach used to solve combinatorial optimization problems [41]. TS is capable of finding solutions that are of high quality and it's applied in numerous fields with moderate computing time. It uses a search process mechanism and this mechanism biases the search toward points with least objective function values, while including special features for avoid being entrapped in the local minimum.

Tabu search (TS) metaheuristic can be imposed on other processes to prevent them from becoming stuck in a local minimum [42]. Tabu search intelligence incorporates adaptive memory and responsive exploration for the foundation of problem solving [43]. It has an adaptive memory feature which triggers the implementation of procedures capable of searching the solution space [43] economically and effectively.

TS neighbourhood search technique is restricted and [44] fundamentally uses flexible memory of search history and consequently guides the search progression to overcome local optimal solutions. It is capable of optimising, not only the integer variable, but also the continuous variable, by discretising technique.

7.4. Applications of Tabu Search for Power System Optimization

Tabu search is presented as an optimisation algorithm by [44] for optimum planning of series compensation devices in order to achieve an improvement in power quality within distribution systems. In this paper, the proposed algorithm models the optimal placement Series Voltage Restorer (SVR) and the Fault Current Limiter (FCL).

The design procedure optimises the performance by minimising three objectives, i.e., capital cost expenditure, voltage sag improvement on sensitive loads and other nodes. The optimisation problem is based on TS and fuzzy performance indices. In this case TS is used for its high efficiency search technique, and fuzzy performance is used to provide a trade-off among the three objectives [44]. Satisfactory results based on the five case studies were presented and achieved.

Studies by [45], proposed a two staggered TS within distribution systems to optimize the placement of distributed FACTS with wind generation. An optimization technique, i.e., two staggered tabu search is proposed for determining the optimum location and the output of the SVCs in distribution systems with wind power generation in order to suppress voltage variation.

The first Stage determined the optimal distribution while Stage 2 evaluated the optimum output. The findings of the recent studies proved that TS was superior to simulated annealing (SA) and genetic algorithm (GA) hence the use of two-staged TS was used for the reduction of the voltage deviation from the reference voltage with SVCs.

The proposed method was successful and the simulation results proved that the proposed method was effective for defining the allocation and the output of SVC devices in the distribution system with unreliable distributed generation. In distribution systems [41] in order to improve the reliability of the system, reduce energy losses and improve voltage profile, capacitors are installed.

The installation of capacitors has economic benefits, depending on the number of capacitors installed, as well as timely control at various load levels. In [41], TS based solution algorithm is adopted for optimum allocation of the capacitor in a radial distribution system to solve an optimization problem. The operation constraints are considered when the placement method is done, as well as the operating bounds of the capacitors factoring in the load growth, the capacity of each feeder, lastly the voltage at various load levels for the reduction in investment costs.

The results of the placement of capacitors through using TS revealed that this method offers nearly optimum solutions when compared to Simulated Annealing (SA) method, with less computational duration. Thus it is recommended that tabu search be applied for capacity placement problem as well as other combinatorial optimization problems in power systems.

TS algorithm can be used to estimate the parameters of a generator excitation-system model [40]. Tabu compares the measured response with simulated responses using diverse sets of excitation system parameters. The comparison is iterative until best parameters have been identified for providing the best comparison between the measured response and the simulated response.

TS algorithm was developed in order to deliver speedy in convergence and suitable solution. This algorithm has been applied for a generator and excitation system [40] that has been field-tested.

The findings demonstrated that a set of parameters developed by TS method yielded exceptional comparison of the simulated response when compared to the measured response.

The increase in load growth demand prompts the need for the expansion of the transmission system. The main purpose of the topology modification algorithm is to reduce fault current levels at critical locations.

Subsequent to the completion of long-term planning, the need arise to appraise the network by the system operators (SO) need to check whether the equivalent system satisfies security criteria for the real-time network operation [46]. In real time operation of the power system operation, there are several security issues, fault current related security being one of them.

The transmission expansion causes the reduction in the magnitude of Thevenin impedances at important nodes in the load center. This reduction in thevenin impedance can compromise the reliability of the network due to deteriorating fault level security, as fault current levels increases [46].

To remedy this excessive fault current problems post long term planning, an evaluation for operable solution for power system security is evaluated by system operators, with the focus on fault current. A technique of determining power system topology modification using tabu search to reduce the fault current levels at critical locations, is presented.

Firstly, the problem is defined and formulated, by assuming pre-determining bus splitting method and the decision variables of the problem is described as binary variables representing bus splitting or opening of a line at both corresponding and non-corresponding location.

TS method based topology modification algorithm is therefore proposed in electrical networks to reduce fault current levels at critical locations for satisfying system operating limits [46].

Location of system fault in electrical power distribution can be determined by several methodologies and practices [47], at the present time. The only shortfall is that most of the techniques and methodologies for fault location are developed for transmission systems therefore their application in distribution system can be a great challenge due to the dissimilarity of load currents in the downstream network.

Thus, a compromise can be noted on the accuracy of most fault location techniques by network topology, impedance variations as the network is reconfigured, distributed resources and incorrectness in the source impedance, to list few. A fault location methodology for distribution when derived from transmission system methodologies can yield good results however it is highly dependent on where the Phasor Measurement Units (PMUs) are installed. Therefore, it is necessary to carry out optimized allocation of these devices, trying to minimize fault location inaccuracies.

The PMUs allows control, monitoring and accurate operation of the power systems, when optimally allocated [47]. In this study TS search algorithm technique is proposed on a real life distribution system with 141 busses to optimally allocate the PMUs. The steps sequence is developed on a flow chart for carrying out of the simulations for both pre and post faults scenarios.

When voltage comparison is carried out, the busses with lowest voltages are most likely to have fault location. Therefore, PMU allocation is a combinatory problem, TS algorithm is used to solve this problem and a significant improvement of an urban feeder is achieved.

Under partially shaded conditions MPP are shown by the power-voltage characteristics of the photovoltaic (PV) array [48]. A single MPP is in existence in the P-V characteristic curve when the solar irradiance on the PV array is constant.

However, due to the bypass diodes and the blocking diodes, multiple peaks arising under moderately shaded conditions, different shading patterns would result in different number of peaks.

The conventional power point tracking is incapable of distinguishing between local and global MPP. To resolve this, a new MPPT technique based on tabu search (TSMPPT) for PV systems under partially shaded conditions is introduced in [48].

The conventional maximum power tracking (MPPT), are not able to track multiple peaks under shaded conditions. TS is however able to record and explore areas and filters by avoiding previously visited voltage areas. In this case, the results are refined by use of another algorithm, i.e., hill climbing, by use of small steps. Simulation results confirmed that the proposed method can accurately track the global MPP with low complexity.

7.5. Summary

The applications of TS in power system have been effective in solving combinatorial problems. The literature scan has demonstrated the different achievable methods of optimising the use of TS algorithm.

TS , when compared to other algorithms was found to be more effective due to its ability to escape local minimum and its ability to search and find global solutions without being trapped. The next chapter will discuss the application of TS to solve a combinatorial optimisation problem in this study.

Chapter 8: Methodology of the Study using Tabu Search Algorithm

8.1. Introduction

This section presents a TS based method to solve an optimization problem in this study for an area suffering from overvoltages. This section also discusses an investigation of the study of applying TS algorithm to solve the optimisation problem in a lightly loaded network. Section 8.2 explains the TS technique and how it creates possible moves. Section 8.4 explains TS problem formulation for the study. In Section 8.5 presents an illustrative example of the study, case study preparation and the software models used.

The problem description of the study is covered in Section 8.6. Section 8.7, briefly explains the use of TS in this study on a flow chart with step by step method for selection of best lines combinations. The summary of the chapter is covered in Section 8.8. Lastly, the numerical results of the TS method tested in an IEE 14 bus system and presented in Chapter 9.

8.2. Radom Selection of Lines using TS in an Area suffering form Overvoltages

Combinatorial optimization for a large scale of the distribution system can be a problem due to many possible combinations. It is a challenge to assess the optimal solution due to restriction of a local minimum [49].

This network comprises both a transmission and distribution network. The transmission network has 400kV network and 132kV for distribution network. The aim of the optimization is to randomise the selection of best lines combination to be derated to meet the objective function and to ensure that the constraints are not exceeded. Tabu search is adopted by searching the solution space with the least load as an objective function and ensuring that the voltages limit of 0.95 per unit and 1.05 per unit is not exceeded.

An illustrative example applying the tabu search based topology modification method is demonstrated on a 14 bus test system in Fig. 8.1, below. TS is applied using problem description such as limits and constraints equations.

8.3. Tabu Search Algorithm in the study

Optimizing the selection of the lines to be derated will reduce the voltage and as such the active power losses. The problem of selecting best lines stems in a combinatorial optimization. However, it is a huge challenge to determine an optimal network configuration in far-reaching radial distribution systems due to many possible combinations.

Meta-heuristic methods and techniques are iterative and to reach a global minimum, they use heuristic to reach an estimated solution [49]. Normally, the following meta-heuristics are used, Simulated Annealing (SA), Genetic Algorithm (GA), and TS. In the case of SA and GA, uses probabilistic approach is discovering solutions and the probabilistic reach methods used are flexible in small-scale systems. In large –scale systems, SA and GA probabilistic search is not applicable due to the large solution space due to the difficulty to tune up.

Hence the focus of this study on TS, due to its uniqueness when compared to SA and GA, is based on the following, (i) it's simplicity, (ii) it's adaptive memory, namely tabu list which plays an vital role in storing some attributes, and (iii) the tabu length parameter.

TS is deterministic, literature in [48] it is easy to tune up and relevant in simulations while probabilistic search methods such SA and GA have a limited ability in large-scale problems. TS algorithm is an evolution type algorithm, which searches widely to find better solutions under different constrained conditions, whereas SA and GA relate to the convergence type algorithm.

8.4. Formulation of Lines Selection on Distribution Network

The objective of the combination of lines to be derated is to maximise the least load that can be connected. This is carried out with the following constraints:

a) Power flow equation, the balance between generation and load ,
b) the lower and the upper limit of nodal voltages, $0.95 \leq 1.05$ per unit , and
c) switching conditions for lines to be derated.

The mathematic description for the problem formulation can be written as follows:

$$\min = \sum_{i=1}^{n} P_i \, j_i \tag{8.1}$$

Subject to:

$$g(x) = 0 \tag{8.2}$$

$$V^{\min} \leq V_i \leq V^{\max} \tag{8.3}$$

$$\text{Number combination of lines per iteration } = L - N + 1 \tag{8.4}$$

where,

min is the least load (MW)
P_i is the active power distribution load
L is the number of the transmission line
j_i is the bus number
$g() = 0$ is the power flow equation
x is the nodal voltage vector
N is the number of busses

The active power losses in the network can be expressed as the sum of the $I^2 R$ at every branch. The total active power losses can then be expressed by,

$$P_{loss} = \sum_{i=l}^{n} \sum_{j=0}^{li-1} r_i \, j + 1 (P_{ij}^2 + Q_{ij}^2) / V_{ij}^2 \tag{8.5}$$

where,

$r_i \, j + 1$ is the resistance of the node

From the equation above, it is expected that the introduction of transformers to reduce the operating voltage on the secondary side of the transformers; it is also expected to reduce the reactive power flow on the lines, as well as the active power losses.

8.5. Illustrative Example - Tabu Search Case Study

TS algorithm is used to optimally select the best combination of lines to be derated and connected to the existing network. The study assesses the feasibility of using TS algorithm with the introduction of step-down transformers to primarily reduce steady overvoltages, reactive power flow on the lines and active power losses. A case study was conducted using the IEEE 14 bus system, and its single line diagram is shown in Fig. 8.1.

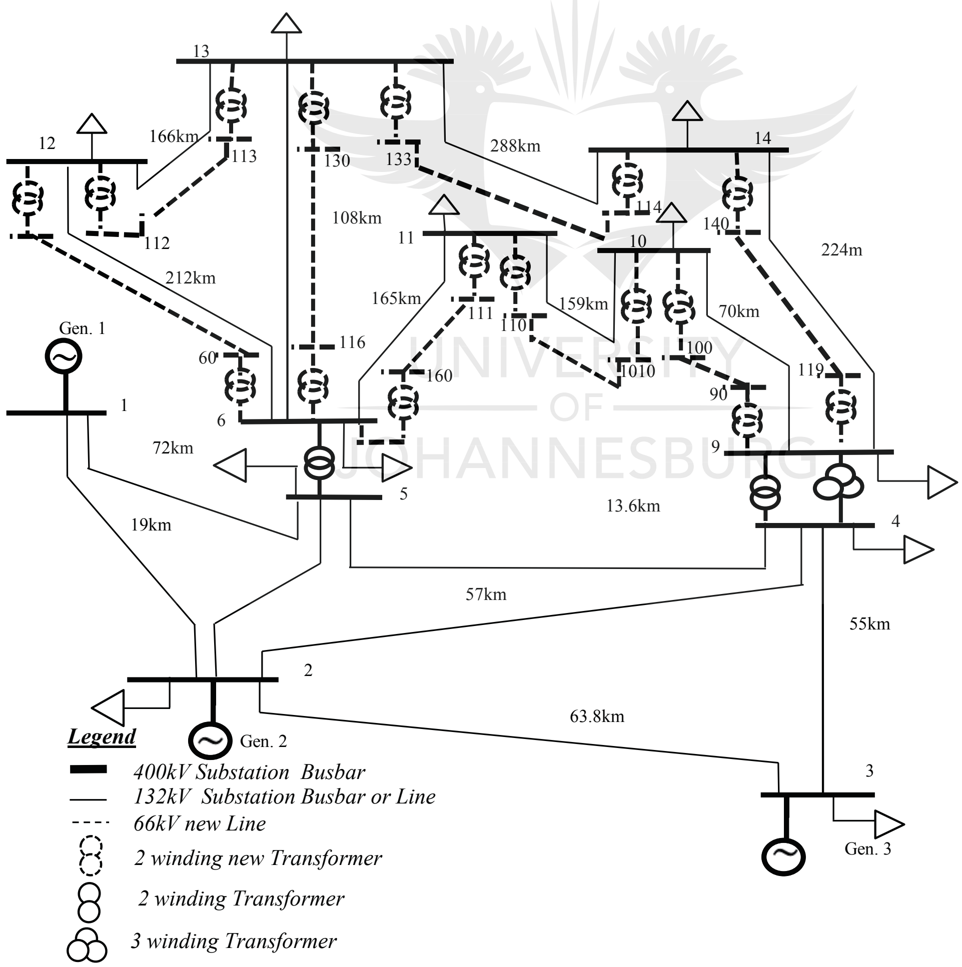

Fig. 8.1. IEEE 14 bus system diagram with parallel derated network

The 1EEE 14 busy system was enhanced by modelling a parallel 88kV network on the sub-transmission side. The parameters of these lines are shown in the Methodology of the Study Section, below.

The network used in this section is similar to the one used in Chapter 2, i.e., Fig. 5.1, which has long lightly loaded transmission corridors, supplying a lightly loaded distribution network with equally long lines. The effectiveness of voltage control is limited by the length of the lines form the power source. Section 8.3.1 presents the step followed in reducing the steady overvoltages to improve the reliability of this network.

8.5.1. Software Model(s) used for Technical Assessment of the Tabu Search Study

The power system simulations software used to carry out the load flows are the PSS/E [31], and [32] API to implement the program. The automation of the study is achieved through writing a Python syntax program through using the API [34]. The Python program is interfaced with PSS/E to issue commands, for the load flows and capturing the results.

The effectiveness of installing the step-down transformers in order to reduce overvoltages is assessed by selecting a combination of two, three, four and lastly five lines from the paralleled derated network shown in Fig. 8.1. The model is built in PSS/E and the results are discussed in Chapter 9.

8.5.2. Case Study Data Preparation

To prepare the case, generation, transmission and distribution data is prepared to represent the power system using the IEEE 14 bus system in Fig. 8.1. The transmission voltage of 400kV is used Fig. 8.1, for both the base case and the derated network.

The transformers for base case utilises 400/132kV transformers for the parallel network which is only switched in through an automated optimisation selection through the use of TS, the 400/66kV transformers are used. The line parameters for Case A are shown in Table 5.1 and for derated case, the 132kV distribution lines were de-rated to 66kV and data can be obtained from Table 5.2. The transformers and generation modelling data is the same as the one used in Chapter 5.

The solid network is the base case 132kV network and to create the derated network the embedded transformers are between buses 6 and 60, 12 and 120, 6 and 160, 12 and 112, 13 and 113, 11 and 111, 11 and 110, 10 and 1010, 10 and 100, 9 and 90, 9 and 119, 13 and 133, 14 and 140, 14 and 140. The data in Table 3 illustrates the data used to model three generators within the used model in PSS/E and the busses for the three generators are as shown in Fig. 8.1.

8.6. Problem Description for Optimization of Line Selection in Distribution Network

The distribution load centre loads for this analysis is based on a peak load of 120MW. The step change reduction used is 1 MW, and the least load tested is governed by the voltage constraints. There are 50 iterations tested and there are 120 load steps.

For this section of the study the technical assessment is only carried out for system healthy network conditions. Fifty iterations for combination of lines are carried out and the following elements are recorded, i.e., lines combinations tried, best lines combinations, remaining load accommodated and the load steps.

8.7. Solution for Tabu Search Method for Selection of lines in a Flow Chart

Figure 8.2 presents a step by step TS algorithm for selection of best lines combinations to connect the least load within minimum and maximum voltage constraints met.

Step 1: Input system data. Input the network confuguration , network data, number of transformers and line parameters as well as constraints , i.e., lower and upper limits of planning voltage, tabu list length , solution arrays, combinations tried, lines id and iterations number.

- Step 2: Innitialise the first iteration by providing the first elements of the lines combinations, i.e, for two combinations (k,l)=(3,2) for three combinations (k,l,a)=(3,2,4) for four combinations (k,l,a,b)=(3,2,4,5) for five combinations (k,l,a,b,c)=(3,2,4,5,1).
- Step3: Generate neigbourhoods i.e., possible moves based on the innitial solution as shown in step 2.
- Step 4: Innitialise the case file to start at 120MW(Pnew) innitially. It should be noted that Pnew=120MW innitially and it reduces with 1MW to simmulate next load level.

1) Run the first iteration with the first combinations in the generated neigborhood.
2) Check if the case file converges.

- Step 5: For each load level execute the distribution power flow to asses the feasibility and if any limits are violated, go to Step 4,and increase load "P" by a fraction or else continue to the next step.
- Step 6: Perfom TS proceedure.

1) Select best moves as next move direction based on the objective function evaluated,
2) Asses the feasibility in Step 2 and Step 4 (scale load down). If not feasible, go to Step 4, or else continue to the next step.

- Step 6: Update best solution state . If the move/combination is in not in tabu, try and run an iteration. If in tabu and aspiration creterion is met, update the best solution state and store in the solutions array.

- Step 7: Check the stopping criterion. If the change of the objective function value is the best this far and the set iterations are not met , go to the next step, or else if the iterations have been met, stop. If not go to the next step.
- Step 8: Record the optimum solution state. Record the best combination solutions , including its position in the solution arrays and the least load connected.

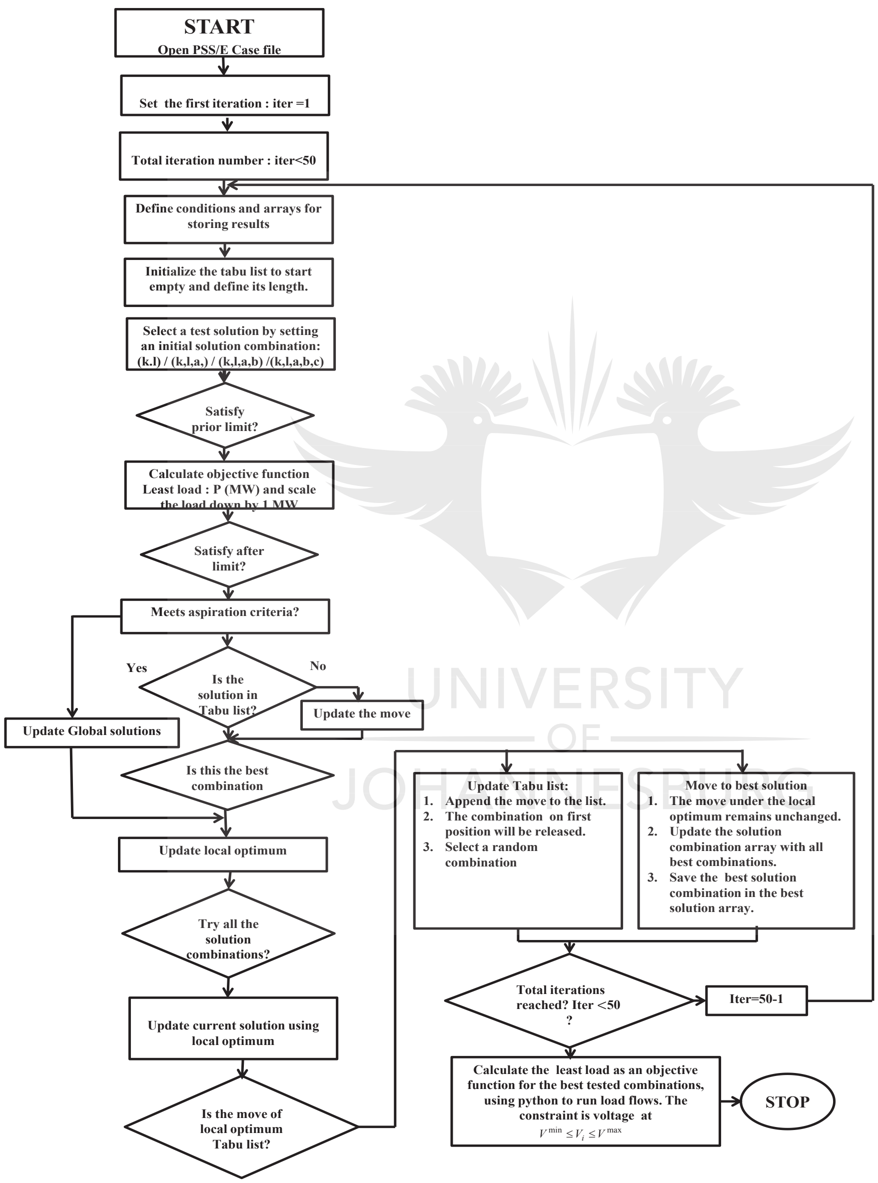

Fig. 8.2. Flow chart showing steps for load flow studies at various load levels using Tabu Search algorithm

8.8. Summary

This section discussed the application of TS algorithm in this study. It also discussed the case study and the TS technique to search the best combination with the least load served as an objective function through the selection of the combination of two, three, four and five lines.

The reults of this tabu search based method is discussed ans analysed in Chapter 9. The selection of best lines combinations and the objective function to connect the least load based on the number of the combinations is also compared in Chapter 9.

Chapter 9: Tabu Search Results

9.1. Introduction

This section presents the results for the best line combination selection using tabu search method. Section 9.2 demonstrates the least load connected per combination. The best lines combinations selection using tabu search method results are compared with the base case results which is the case without line derating.

9.2. System Load Variation with TS Line Combination Selections

The following figures demonstrate the system load variations at various load levels for all TS combinations. The load starts of initially at 120MW for all combinations and the least load that can be accommodated for various combinations explained below.

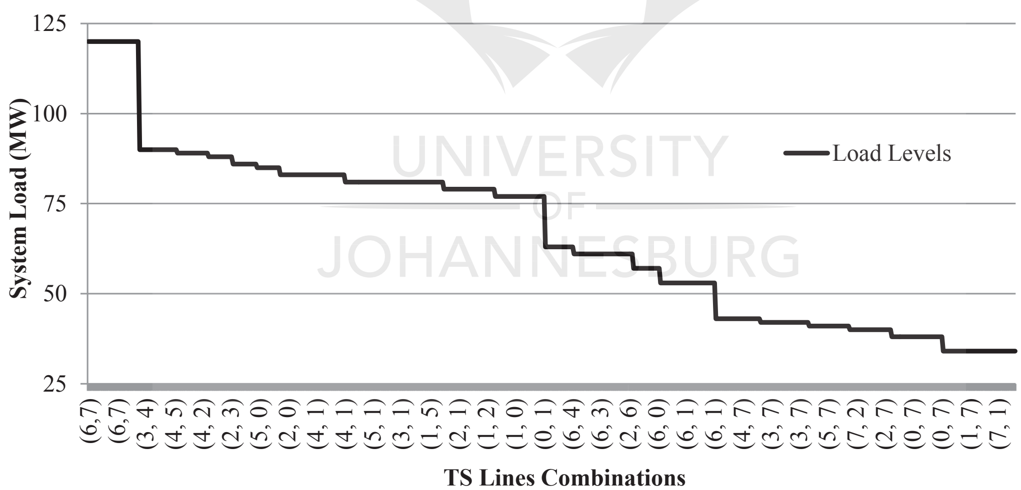

Fig. 9.1. Least load connected on the distribution network for Base case and with Tabu Search two best lines combinations.

In Fig. 9.1 the plot shows the two combination results for all possible TS moves in this scenario. The best combination selected using TS technique is (7, 1). The least load that can be accommodated without violations from (7, 1) combination is 34MW.

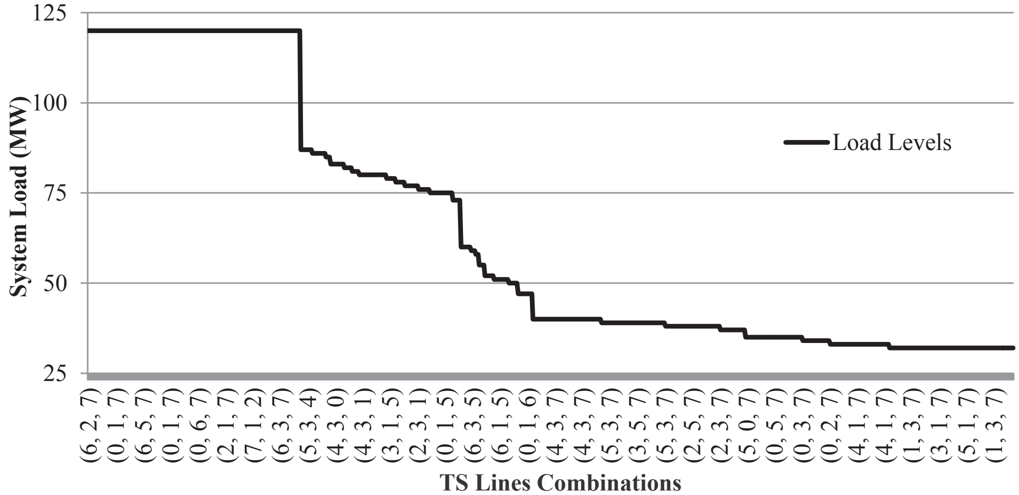

Fig. 9.2. Least load connected on the distribution network for Base case and with Tabu Search three best lines combinations.

In Fig. 9.2 the TS three best lines combination selection is plotted. The best combination selection using TS technique is (1, 3, 7). The least load that can be accommodated by the this best lines combinations is 32MW.

The TS line combinations for four lines are shown in Fig. 9.3. The least load that can be accommodated by selecting four best lines combinations by TS is 29MW. The best line combination in this scenario is (5, 3, 7, 1)

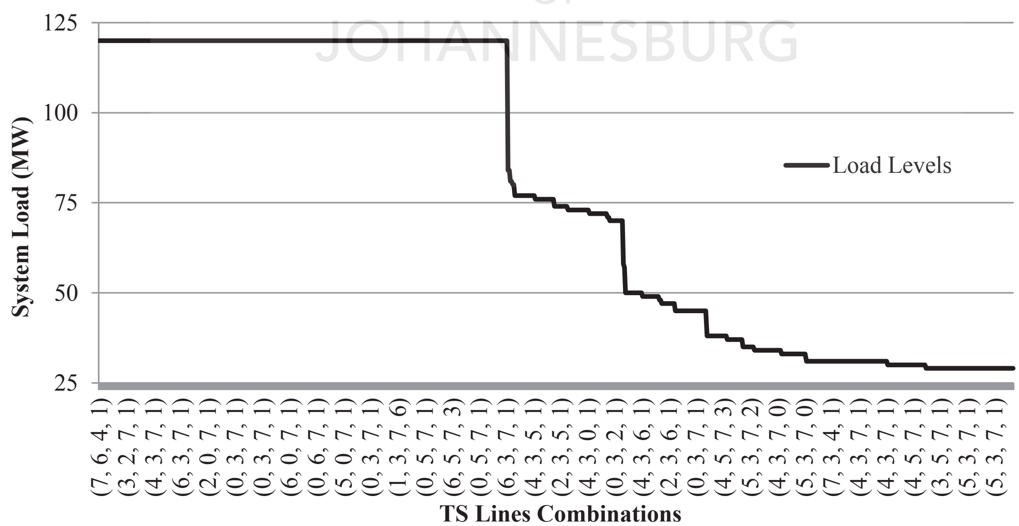

Fig. 9.3. Least load connected on the distribution network for Base case and with Tabu Search four best lines combinations.

Lastly, in Figure 9.4, the least load that can be connected without voltage violations for best five TS line combinations is 28MW. This is 6MW less than the two line combinations, 4MW less than the three line combinations and 1MW less of the four lines combinations. The best TS selection of five lines combinations is (1, 3, 7, 4, 5). Is can be seen in Fig. 9.4 that the more the number of lines selected by use of TS, the lesser load can be connected.

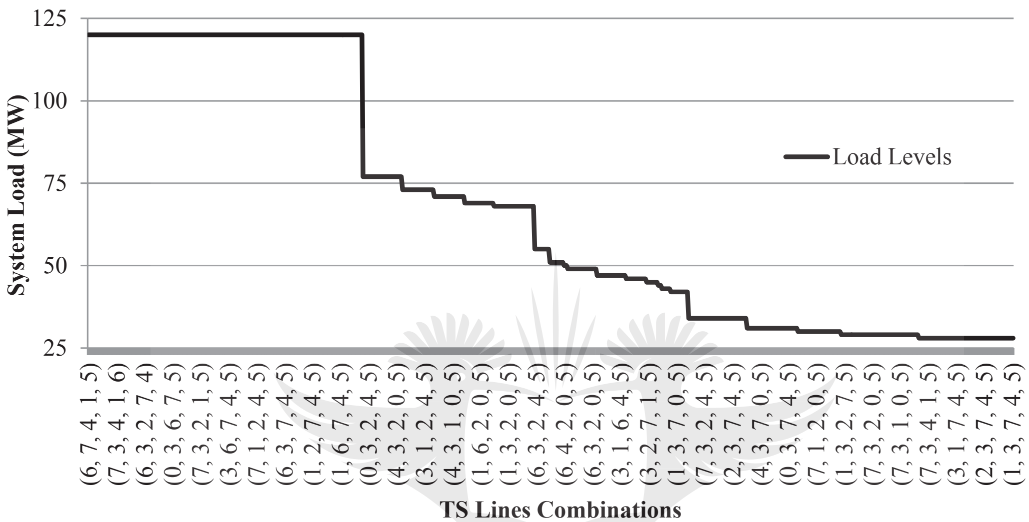

Fig. 9.4. Least load connected on the distribution network for Base case and with Tabu Search five best lines combinations.

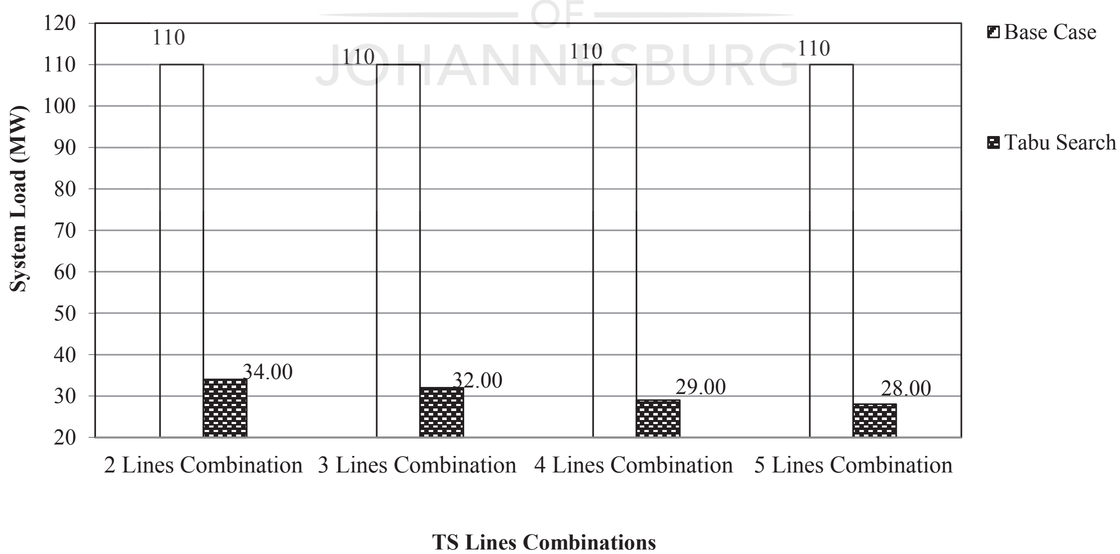

Fig. 9.5. Least load connected on the distribution network for Base case and with Tabu Search best lines combinations.

Fig. 9.5 demonstrates that with the introduction of derating lines, through using tabu search method to select the best line combinations, the least load can be connected without voltage limits being exceeded.

As shown in Fig. 9.5, the base case, the least load that can be accommodated without network constraints is 110MW. In the case of the use of TS best line selection, the least load that can be connected if 34MW for two lines combination , 32MW for three lines combination, 29MW for four lines combination and 28MW for five lines combination. This demonstrates that as the number of best lines combinations increases using TS, further amount of least load can be accommodated without network limits being violated.

The active power losses are shown in Fig. 9.6, where the results comparison between the case without derated lines, i.e., base case and the tabu search best lines combination, is done. The active power losses in the network are lower for the case with tabu search best lines to be derated combinations when compared to base case results.

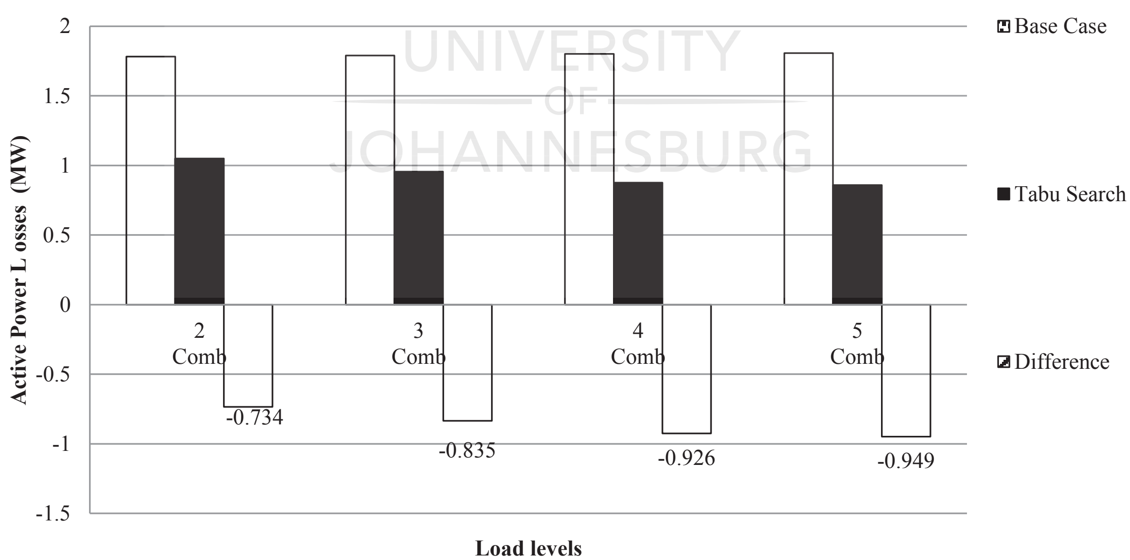

Fig. 9.6. Reactive power flow on the distribution lines for Base case and Tabu Search least load for best combinations.

The active power losses are reduced from the best line selection of two lines, three, four and five lines combinations, therefore demonstrating that towards the lower load levels there is also a decrease in active power flow.

In Fig. 9.7, the assessment between the variation of the system load and system reactive power flow, is conducted. In this assessment it is noted that the network impact on reactive power flow in the system is reduced as the load decreases. A significant increase can be seen as the number of derated lines combinations increases.

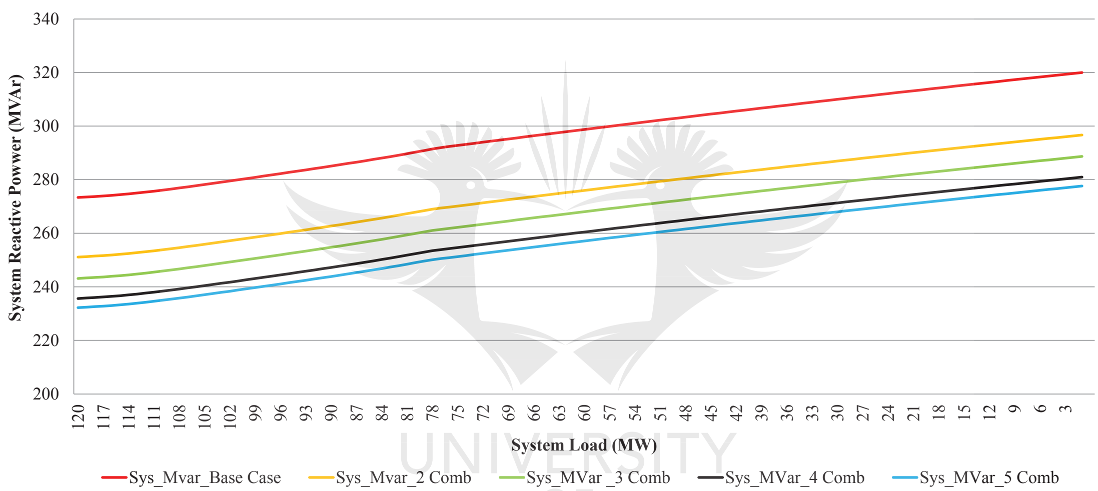

Fig. 9.7. Variation of system load for Base case and with Tabu Search best lines combinations.

Also, in Fig. 9.7, it can be noted that, the issue of overvoltages will significantly improve with the reduction of reactive power flow on the lines. This is due to the direct correlation between reactive power flow generated in the system and voltage.

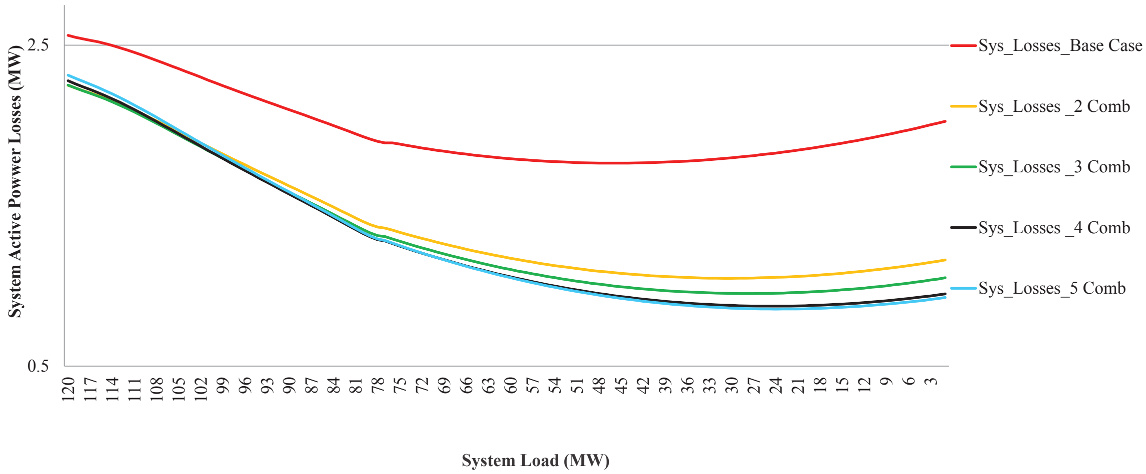

Fig. 9.8. Variation of system active power losses for Base case and Tabu Search best lines combinations.

In Fig. 9.8 the comparison is made between the base case active power losses at a system level and how it changes as the load decreases to towards light loading of the network. It can be seen that the system losses as significantly high for the base case from the peak load of 120MW and that as the load is decreased significantly.

The active power losses also increase exponentially. However, with the best lines selection combinations by TS there is a significant improvement in active power losses that reduce with the load.

9.3. Summary

This section of the study has shown that the use of TS algorithm can efficiently optimise the selection of best lines combination to connect the least load. The best selected line combinations also prove that the active power losses can significantly reduce as the load is reduced to lower levels. This is due to lesser generation of reactive power flow in the system and on the lines.

Chapter 10: Conclusions

The steady state overvoltages are a problem in lightly loaded networks with long corridors. These steady state overvoltages manifests at the receiving end of long lightly loaded transmission lines and can result in excess reactive power as a result of line charging currents. This study investigated derating of line operating voltages by placing step-down transformers at the ends of the long corridors. The secondary side of the step-down transformers resulted in lower design voltages on the lines. Initially the load was scaled down manually and the load flow simulations were carried out until the voltages reached the maximum voltage limit.

The second part of the studies investigated the automation of scaling the loads down using a Python program. The load flow simulations were also carried out using the Python program to connect the lowest load without voltage violations. The study proceeded to investigate the optimal combination of lines than can accommodate the lowest load by using Tabu Search algorithm. The load flow simulations to scale down the load were carried on PSS/E using instructions from a written Python program.

The study demonstrated the use of step-down transformers can successfully reduce the steady state voltages. Derating of lines can also reduce the active power losses in the system. The process of automating these studies using python was successful and the lowest load was connected without exceeding the voltage limit. The use of Tabu Search algorithm to optimise the selection of the best line combinations enable the least load to be connected without voltage violations.

For further work, the use of Tabu Search algorithm can be investigated for optimal placement of Distributed Generators (DG) in a network of interest without violating voltages.

Appendix A: Python Code for Automated Load Scaling and Load Flows

A.1 Introduction

In this appendix, the Python syntax is presented. The program is written in Python using APIs to write the code and instructions to perform the loadflows used PSS/E, and the studies are at various load levels. The program automates the load flows and records, voltages, reactive power flow and active power losses. The results were printed at the end of the program.

Python Code:

```
psspy.report_output(2,r'C:\Users\melatope\Desktop\test1.dat',[0,0])
p_init=120
p_new=120
scale=1
p_list=[]
sys_MW = []
sys_Losses = []
sys_MVar = []
L6_11MVA = []
L6_12MVA = []
L6_13MVA = []
L9_10MVA = []
L9_14MVA = []
L10_11MVA = []
L12_13MVA = []
L13_14MVA = []
L6_11MVAr = []
L6_12MVAr = []
L6_13MVAr = []
L9_10MVAr = []
L9_14MVAr = []
L10_11MVAr = []
L12_13MVAr = []
L13_14MVAr = []
L6_11MW = []
L6_12MW = []
L6_13MW = []
L9_10MW = []
L9_14MW = []
L10_11MW = []
L12_13MW = []
L13_14MW = []
bus6_volts = []
bus9_volts = []
```

```
bus10_volts = []
bus11_volts = []
bus12_volts = []
bus13_volts = []
bus14_volts = []
j=0
load_step=0
while p_new > p_init*0.005:
    p_list.append(p_new)
    psspy.scal_2(0,1,1,[0,0,0,0,0],[0.0,0.0,0.0,0.0,0.0,0.0,0.0])
    psspy.scal_2(0,1,2,[_i,1,0,1,0],[ p_new, 100.0,0.0,-.0,0.0,-17.848, 33.066])
    psspy.fnsl([1,0,0,1,1,0,0,0])
    solution=psspy.solved()
    if solution==0:
       sys_MW.append(psspy.systot('LOAD'))
       sys_Losses.append(psspy.systot('LOSS'))
       sys_MVar.append(psspy.systot('GEN'))
       L6_11MVA.append(psspy.brnflo(6,11,'1'))
       L6_12MVA.append(psspy.brnflo(6,12,'1'))
       L6_13MVA.append(psspy.brnflo(6,13,'1'))
       L9_10MVA.append(psspy.brnflo(9,10,'1'))
       L9_14MVA.append(psspy.brnflo(9,14,'1'))
       L10_11MVA.append(psspy.brnflo(10,11,'1'))
       L12_13MVA.append(psspy.brnflo(12,13,'1'))
       L13_14MVA.append(psspy.brnflo(13,14,'1'))
       L6_11MVAr.append(psspy.brnflo(6,11,'1'))
       L6_12MVAr.append(psspy.brnflo(6,12,'1'))
       L6_13MVAr.append(psspy.brnflo(6,13,'1'))
       L9_10MVAr.append(psspy.brnflo(9,10,'1'))
       L9_14MVAr.append(psspy.brnflo(9,14,'1'))
       L10_11MVAr.append(psspy.brnflo(10,11,'1'))
       L12_13MVAr.append(psspy.brnflo(12,13,'1'))
       L13_14MVAr.append(psspy.brnflo(13,14,'1'))
       L6_11MW.append(psspy.brnflo(6,11,'1'))
       L6_12MW.append(psspy.brnflo(6,12,'1'))
       L6_13MW.append(psspy.brnflo(6,13,'1'))
       L9_10MW.append(psspy.brnflo(9,10,'1'))
       L9_14MW.append(psspy.brnflo(9,14,'1'))
       L10_11MW.append(psspy.brnflo(10,11,'1'))
       L12_13MW.append(psspy.brnflo(12,13,'1'))
       L13_14MW.append(psspy.brnflo(13,14,'1'))
       bus6_volts.append(psspy.busdat(6,'PU'))
       bus9_volts.append(psspy.busdat(9,'PU'))
       bus10_volts.append(psspy.busdat(10,'PU'))
       bus11_volts.append(psspy.busdat(11,'PU'))
       bus12_volts.append(psspy.busdat(12,'PU'))
       bus13_volts.append(psspy.busdat(13,'PU'))
       bus14_volts.append(psspy.busdat(14,'PU'))
       p_new=p_new-scale
       load_step=load_step+1
print'%10s'%'sys_MW','%10s'%'sys_Losses'                                              ,
        '%10s'%'sys_MVar','%10s'%'L6_11P','%10s'%'L6_11Q','%10s'%'S6_11S','%10s'%'L6_12P','%10s'%'
        L6_12Q','%10s'%'S6_12S','%10s'%'L6_13P','%10s'%'L6_13Q','%10s'%'S6_13S','%10s'%'L9_10P','%
        10s'%'L9_10Q','%10s'%'S9_10S','%10s'%'L9_14P','%10s'%'L9_14Q','%10s'%'S9_14S','%10s'%'L10_
```

```
11P','%10s'%'L10_11Q','%10s'%'S10_11S','%10s'%'L12_13P','%10s'%'L12_13Q','%10s'%'S12_13S','
%10s'%'L13_14P','%10s'%'L13_14Q','%10s'%'S13_14S','%10s'%'bus6_volts','%10s'%'bus9_volts','%
10s'%'bus10_volts','%10s'%'bus11_volts','%10s'%'bus12_volts','%10s'%'bus13volts','%10s'%'bus14
_volts'
while j < len(sys_MW):
 while j < len(sys_Losses):
  while j < len(sys_MVar):
    while j < len(L6_11MVA):
      while j < len(L6_12MVA):
        while j < len(L6_13MVA):
          while j < len(L9_10MVA):
            while j < len(L9_14MVA):
              while j < len(L10_11MVA):
                while j < len(L12_13MVA):
                  while j < len(L13_14MVA):
                    while j < len(bus6_volts):
                      while j < len(bus9_volts):
                        while j < len(bus10_volts):
                          while j < len(bus11_volts):
                            while j < len(bus12_volts):
                              while j < len(bus13_volts):
                                while j < len(bus14_volts):
                                  sys_PandQ=sys_MW[j]
                                  MVA=sys_PandQ[1]
                                  Psyst=MVA.real

                                  sysL_PandQ=sys_Losses[j]
                                  MVAL=sysL_PandQ[1]
                                  PLosses=MVAL.real

                                  sys_P1andQ1=sys_MVar[j]
                                  MVA1=sys_P1andQ1[1]
                                  Qsyst=abs(MVA1.imag)

                                  L6_11PandQ=L6_11MVA[j]
                                  MVAL6_11=L6_11PandQ[1]
                                  PL6_11=MVAL6_11.real

                                  L6_11PandQ=L6_11MVA[j]
                                  MVAL6_11=L6_11PandQ[1]
                                  QL6_11=abs(MVAL6_11.imag)

                                  SL6_11=(PL6_11**2+QL6_11**2)**0.5

                                  L6_12PandQ=L6_12MVA[j]
                                  MVAL6_12=L6_12PandQ[1]
                                  PL6_12=MVAL6_12.real

                                  L6_12PandQ=L6_12MVA[j]
                                  MVAL6_12=L6_12PandQ[1]
                                  QL6_12=abs(MVAL6_12.imag)

                                  SL6_12=(PL6_12**2+QL6_12**2)**0.5
```

```
L6_13PandQ=L6_13MVA[j]
MVAL6_13=L6_13PandQ[1]
PL6_13=MVAL6_13.real

L6_13PandQ=L6_13MVA[j]
MVAL6_13=L6_13PandQ[1]
QL6_13=abs(MVAL6_13.imag)

SL6_13=(PL6_13**2+QL6_13**2)**0.5

L9_10PandQ=L9_10MVA[j]
MVAL9_10=L9_10PandQ[1]
PL9_10=MVAL9_10.real

L9_10PandQ=L9_10MVA[j]
MVAL9_10=L9_10PandQ[1]
QL9_10=abs(MVAL9_10.imag)

SL9_10=(PL9_10**2+QL9_10**2)**0.5

L9_14PandQ=L9_14MVA[j]
MVAL9_14=L9_14PandQ[1]
PL9_14=MVAL9_14.real

L9_14PandQ=L9_14MVA[j]
MVAL9_14=L9_14PandQ[1]
QL9_14=abs(MVAL9_14.imag)

SL9_14=(PL9_14**2+QL9_14**2)**0.5

L10_11PandQ=L10_11MVA[j]
MVAL10_11=L10_11PandQ[1]
PL10_11=MVAL10_11.real

L10_11PandQ=L10_11MVA[j]
MVAL10_11=L10_11PandQ[1]
QL10_11=abs(MVAL10_11.imag)

SL10_11=(PL10_11**2+QL10_11**2)**0.5

L12_13PandQ=L12_13MVA[j]
MVAL12_13=L12_13PandQ[1]
PL12_13=MVAL12_13.real

L12_13PandQ=L12_13MVA[j]
MVAL12_13=L12_13PandQ[1]
QL12_13=abs(MVAL12_13.imag)

SL12_13=(PL12_13**2+QL12_13**2)**0.5

L13_14PandQ=L13_14MVA[j]
MVAL13_14=L13_14PandQ[1]
PL13_14=MVAL13_14.real
```

```
L13_14PandQ=L13_14MVA[j]
MVAL13_14=L13_14PandQ[1]
QL13_14=abs(MVAL13_14.imag)

SL13_14=(PL13_14**2+QL13_14**2)**0.5

V6=bus6_volts[j]
bus6Volts=V6[1]

V9=bus9_volts[j]
bus9Volts=V9[1]

V10=bus10_volts[j]
bus10Volts=V10[1]

V11=bus11_volts[j]
bus11Volts=V11[1]

V12=bus12_volts[j]
bus12Volts=V12[1]

V13=bus13_volts[j]
bus13Volts=V13[1]

V14=bus14_volts[j]
bus14Volts=V14[1]

print10.3f'%Psyst,'%10.3f'%PLosses,'%10.3f'%Qsyst,'%10.3f'%PL6_11,'%10.3f'%QL6_11,'%10.3f'%SL6
_11,'%10.3f'%PL6_12,'%10.3f'%QL6_12,'%10.3f'%SL6_12,'%10.3f'%PL6_13,'%10.3f'%QL6_13,'%10.3f
'%SL6_13,'%10.3f'%PL9_10,'%10.3f'%QL9_10,'%10.3f'%SL9_10,'%10.3f'%PL9_14,'%10.3f'%QL9_14,'
%10.3f'%SL9_14,'%10.3f'%PL10_11,'%10.3f'%QL10_11,'%10.3f'%SL10_11,'%10.3f'%PL12_13,'%10.3f'
%QL12_13,'%10.3f'%SL12_13,'%10.3f'%PL13_14,'%10.3f'%QL13_14,'%10.3f'%SL13_14,'%10.5f'%bus
6Volts,'%10.5f'%bus9Volts,'%10.5f'%bus10Volts,'%10.5f'%bus11Volts,'%10.5f'%bus12Volts,'%10.5f'%b
us13Volts,'%10.5f'%bus14Volts
 j=j+1
print load_step
```

Appendix B: Tabu Search Code for Optimal Selection of Two Lines Combination

B.1 Introduction

In this appendix, presents the Python syntax for a TS optimal selection of two lines combination. The program is written such that the commands will select the neighbourhoods based on the first iteration of (k, l)=(3, 2). TS will use its own intelligence to replace the initial k and l, with the found possible moves. The results are recorded for all possible moves and each will have dedicated set of results and printed as shown in the program. Lastly, TS will generate the best two lines combination for and the least load connected without violating the voltage limits.

Tabu Search Code:

```
import random
from random import randint
iter=1
all_sol_comb=[]
all_obj_fun=[]
sys_load_rem=[]
(k,l)=(3,2)
tabu=[0,0,0,0,0]
while iter<50:
    lines_id=[0,1,2,3,4,5,6,7]
    sol_comb=[]
    obj_fun=[]
    print 'The current value of (k,l) is:',(k,l)
    if (k,l) not in sol_comb:
        i=0
        j=0
        neighbourhood=[]
        while i<len(lines_id):
            ne_soln=(k,i)
            if k!=i:
                neighbourhood.append(ne_soln)
            i=i+1
        while j<len(lines_id):
            ne_soln=(j,l)
            if j!=l:
                neighbourhood.append(ne_soln)
            j=j+1
        print 'This neighbourhood is based on solution:', (k,l)
        comb=(k,l)
```

```
for comb in neighbourhood:
    if comb not in tabu:
        print comb
        print sol_comb
        print tabu
        tabu.pop(0)
        tabu.append(comb)
        m=comb[0]
        n=comb[1]
        psspy.case(r"""C:\Users\melatope\Desktop\Exp Training\UJ\IEEE\IEEE 2015\Tabu Search prog and casefiles\Case files_TS\132kV_Off Peak_Line 60_120_ additional network.sav""")
        lines=[(6, 12, 1), (12, 13, 1), (6, 13, 1), (6, 11, 1),  (9, 10, 1), (10, 11, 1), (9, 14, 1), (13, 14, 1)]
    corridors=[(6,60,20,12),(12,112,113,13),(6,116,130,13),(6,160,111,11),(9,90,100,10),
    (10,1010,110,11),(9,119,140,14),(13,133,114,14)]
        off_line=lines[m]
        on_corridor=corridors[m]
        bus_s=on_corridor[0]
        bus_r=on_corridor[3]
        bus_ss=on_corridor[1]
        bus_rr=on_corridor[2]
        psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
        psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
    psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
    ,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
    psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
    _f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
    psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
        off_line=lines[n]
        on_corridor=corridors[n]
        bus_s=on_corridor[0]
        bus_r=on_corridor[3]
        bus_ss=on_corridor[1]
        bus_rr=on_corridor[2]
        psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
        psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
        p_init=120
        p_new=120
        scale=1
        p_list=[]
        sys_MW = []
        sys_Losses = []
        sys_MVar = []
        L6_11MVA = []
        L6_12MVA = []
        L6_13MVA = []
        L9_10MVA = []
        L9_14MVA = []
```

```
L10_11MVA = []
L12_13MVA = []
L13_14MVA = []
L6_11MVAr = []
L6_12MVAr = []
L6_13MVAr = []
L9_10MVAr = []
L9_14MVAr = []
L10_11MVAr = []
L12_13MVAr = []
L13_14MVAr = []
L6_11MW = []
L6_12MW = []
L6_13MW = []
L9_10MW = []
L9_14MW = []
L10_11MW = []
L12_13MW = []
L13_14MW = []
bus6_volts = []
bus9_volts = []
bus10_volts = []
bus11_volts = []
bus12_volts = []
bus13_volts = []
bus14_volts = []
j=0
bus6kV=1
bus9kV=1
bus10kV=1
bus11kV=1
bus12kV=1
bus13kV=1
bus14kV=1
LL6_11MVA=10
LL6_12MVA=10
LL6_13MVA=10
LL9_10MVA=10
LL9_14MVA=10
LL10_11MVA=10
LL12_13MVA=10
LL13_14MVA=10
sol_load_step=[]
load_step=0
while p_new > p_init*0.005 and bus6kV > 0.95 and bus6kV < 1.05 and bus9kV > 0.95 and bus9kV < 1.05 and bus10kV > 0.95 and bus10kV <1.05 and bus11kV > 0.95 and bus11kV <1.05 and bus12kV > 0.95 and bus12kV <1.05 and bus13kV > 0.95 and bus13kV <1.05 and bus14kV > 0.95 and bus14kV <1.05 and LL6_11MVA < 84.5 and LL6_12MVA < 84.5 and LL6_13MVA < 84.5 and LL9_10MVA < 84.5 and LL9_14MVA < 84.5 and LL10_11MVA < 84.5 and LL12_13MVA < 84.5 and LL13_14MVA < 84.5 :
    p_list.append(p_new)
    psspy.scal(0,1,1,[0,0,0,0],[0.0,0.0,0.0,0.0,0.0,0.0,0.0])
    psspy.scal(0,1,2,[1,0,1,0],[ p_new, 120.0,0.0,-.0,0.0,-17.848, 33.066])
    psspy.fnsl([1,0,0,1,1,0,0,0])
```

```
solution=psspy.solved()
bus14=psspy.busdat(14,"PU")
bus14kV=bus14[1]
LL6_11_0_PandQ=psspy.brnflo(6,11,'1')
LL6_11_PandQ=LL6_11_0_PandQ[1]
LL6_11_P=LL6_11_PandQ.real
LL6_11_0_PandQ=psspy.brnflo(6,11,'1')
LL6_11_PandQ=LL6_11_0_PandQ[1]
LL6_11_Q=LL6_11_PandQ.imag
LL6_11MVA=(LL6_11_P**2+LL6_11_Q**2)**0.5
LL6_12_0_PandQ=psspy.brnflo(6,12,'1')
LL6_12_PandQ=LL6_12_0_PandQ[1]
LL6_12_P=LL6_12_PandQ.real
LL6_12_0_PandQ=psspy.brnflo(6,12,'1')
LL6_12_PandQ=LL6_12_0_PandQ[1]
LL6_12_Q=LL6_12_PandQ.imag
LL6_12MVA=(LL6_12_P**2+LL6_12_Q**2)**0.5
LL6_13_0_PandQ=psspy.brnflo(6,13,'1')
LL6_13_PandQ=LL6_13_0_PandQ[1]
LL6_13_P=LL6_13_PandQ.real
LL6_13_0_PandQ=psspy.brnflo(6,13,'1')
LL6_13_PandQ=LL6_13_0_PandQ[1]
LL6_13_Q=LL6_13_PandQ.imag
LL6_13MVA=(LL6_13_P**2+LL6_13_Q**2)**0.5
LL9_10_0_PandQ=psspy.brnflo(9,10,'1')
LL9_10_PandQ=LL9_10_0_PandQ[1]
LL9_10_P=LL9_10_PandQ.real
LL9_10_0_PandQ=psspy.brnflo(9,10,'1')
LL9_10_PandQ=LL9_10_0_PandQ[1]
LL9_10_Q=LL9_10_PandQ.imag
LL9_10MVA=(LL9_10_P**2+LL9_10_Q**2)**0.5
LL9_14_0_PandQ=psspy.brnflo(9,14,'1')
LL9_14_PandQ=LL9_14_0_PandQ[1]
LL9_14_P=LL9_14_PandQ.real
LL9_14_0_PandQ=psspy.brnflo(9,14,'1')
LL9_14_PandQ=LL9_14_0_PandQ[1]
LL9_14_Q=LL9_14_PandQ.imag
LL9_14MVA=(LL9_14_P**2+LL9_14_Q**2)**0.5
LL10_11_0_PandQ=psspy.brnflo(10,11,'1')
LL10_11_PandQ=LL10_11_0_PandQ[1]
LL10_11_P=LL10_11_PandQ.real
LL10_11_0_PandQ=psspy.brnflo(10,11,'1')
LL10_11_PandQ=LL10_11_0_PandQ[1]
LL10_11_Q=LL10_11_PandQ.imag
LL10_11MVA=(LL10_11_P**2+LL10_11_Q**2)**0.5
LL12_13_0_PandQ=psspy.brnflo(12,13,'1')
LL12_13_PandQ=LL12_13_0_PandQ[1]
LL12_13_P=LL12_13_PandQ.real
LL12_13_0_PandQ=psspy.brnflo(12,13,'1')
LL12_13_PandQ=LL12_13_0_PandQ[1]
LL12_13_Q=LL12_13_PandQ.imag
LL12_13MVA=(LL12_13_P**2+LL12_13_Q**2)**0.5
LL13_14_0_PandQ=psspy.brnflo(13,14,'1')
LL13_14_PandQ=LL13_14_0_PandQ[1]
```

```
                LL13_14_P=LL13_14_PandQ.real
                LL13_14_0_PandQ=psspy.brnflo(13,14,'1')
                LL13_14_PandQ=LL13_14_0_PandQ[1]
                LL13_14_Q=LL13_14_PandQ.imag
                LL13_14MVA=(LL13_14_P**2+LL13_14_Q**2)**0.5
                if solution==0:
                    sys_MW.append(psspy.systot('LOAD'))
                    sys_Losses.append(psspy.systot('LOSS'))
                    sys_MVar.append(psspy.systot('GEN'))
                    L6_11MVA.append(psspy.brnflo(6,11,'1'))
                    L6_12MVA.append(psspy.brnflo(6,12,'1'))
                    L6_13MVA.append(psspy.brnflo(6,13,'1'))
                    L9_10MVA.append(psspy.brnflo(9,10,'1'))
                    L9_14MVA.append(psspy.brnflo(9,14,'1'))
                    L10_11MVA.append(psspy.brnflo(10,11,'1'))
                    L12_13MVA.append(psspy.brnflo(12,13,'1'))
                    L13_14MVA.append(psspy.brnflo(13,14,'1'))
                    L6_11MVAr.append(psspy.brnflo(6,11,'1'))
                    L6_12MVAr.append(psspy.brnflo(6,12,'1'))
                    L6_13MVAr.append(psspy.brnflo(6,13,'1'))
                    L9_10MVAr.append(psspy.brnflo(9,10,'1'))
                    L9_14MVAr.append(psspy.brnflo(9,14,'1'))
                    L10_11MVAr.append(psspy.brnflo(10,11,'1'))
                    L12_13MVAr.append(psspy.brnflo(12,13,'1'))
                    L13_14MVAr.append(psspy.brnflo(13,14,'1'))
                    L6_11MW.append(psspy.brnflo(6,11,'1'))
                    L6_12MW.append(psspy.brnflo(6,12,'1'))
                    L6_13MW.append(psspy.brnflo(6,13,'1'))
                    L9_10MW.append(psspy.brnflo(9,10,'1'))
                    L9_14MW.append(psspy.brnflo(9,14,'1'))
                    L10_11MW.append(psspy.brnflo(10,11,'1'))
                    L12_13MW.append(psspy.brnflo(12,13,'1'))
                    L13_14MW.append(psspy.brnflo(13,14,'1'))
                    bus6_volts.append(psspy.busdat(6,'PU'))
                    bus9_volts.append(psspy.busdat(9,'PU'))
                    bus10_volts.append(psspy.busdat(10,'PU'))
                    bus11_volts.append(psspy.busdat(11,'PU'))
                    bus12_volts.append(psspy.busdat(12,'PU'))
                    bus13_volts.append(psspy.busdat(13,'PU'))
                    bus14_volts.append(psspy.busdat(14,'PU'))
                else:
                    break
                p_new=p_new-scale
                load_step=load_step+1
                load_rem=p_init-(load_step-1)*scale
        sol_comb.append(comb)
        obj_fun.append(load_step)
        all_sol_comb.append(comb)
        all_obj_fun.append(load_step)
        sys_load_rem.append(load_rem)
        print 'All solutions that were investigated are', sol_comb
        print 'Load_steps array is ',obj_fun
max_value = max(obj_fun)
print max_value
```

```
    max_index = obj_fun.index(max_value)
    print max_index
    (k,l)=sol_comb[max_index]
    print (k,l)
    print (k,l)
    if (k,l) in all_sol_comb:
        k=randint(0,len(lines_id)-1)
    iter=iter+1
print all_sol_comb
print sys_load_rem
print all_obj_fun
max_value = max(all_obj_fun)
print max_value
max_index = all_obj_fun.index(max_value)
print,max_index
(k,l)=all_sol_comb[max_index]
Print (k,l)
opt_load=p_init-(max_value-1)*scale
print opt_load
print'%10s'%'Number'   ,'%10s'%'Combination'   ,'%10s'%'Load'   , '%10s'%'Steps'
while j < len(all_sol_comb):
  while j < len(sys_load_rem):
    while j < len(all_obj_fun):
        t=j+1
        comb=all_sol_comb[j]
        load=sys_load_rem[j]
        objfunc=all_obj_fun[j]
        print '%10.0f'%t,comb,'%10.3f'%load,objfunc
        j=j+1
```

Appendix C: Tabu Search Code for Optimal Selection of Three Lines Combination

C.1 Introduction

In this appendix, presents the Python syntax for a TS optimal selection of three lines combination. Similarly as the in Appendix B, the program is given the first iteration, however for three combinations, the 3rd element in the 1st solution were added. The first iteration of (k, l, a) were as follows (3, 2, 4).

TS used its own intelligence to generate neighbourhoods and run loadflows, lastly replace the initial k and l, with the found possible moves. This is an iterative process and at the end TS generated the best three lines combinations for and the least load connected without violating the voltage limits.

Tabu Search Code:

```
import random
iter=1
all_sol_comb=[]
all_obj_fun=[]
all_load_rem=[]
(k,l,a)=(3,2,4)
tabu=[0,0,0,0,0]
while iter<50:
    lines_id=[0,1,2,3,4,5,6,7]
    sol_comb=[]
    obj_fun=[]
    print (k,l,a)
    if (k,l,a) not in sol_comb:
        i=0
        j=0
        s=0
        neighbourhood=[]
        while i<len(lines_id):
            ne_soln=(k,i,a)
            if k!=i and k!=a and i!=a:
                neighbourhood.append(ne_soln)
            i=i+1
        while j<len(lines_id):
            ne_soln=(j,l,a)
            if j!=l and j!=a and l!=a:
                neighbourhood.append(ne_soln)
            j=j+1
        while s<len(lines_id):
```

```
        ne_soln=(k,l,s)
        if k!=l and k!=s and l!=s:
            neighbourhood.append(ne_soln)
        s=s+1
    print (k,l,a)
    comb=(k,l,a)
    for comb in neighbourhood:
        if comb not in tabu:
            print comb
            print sol_comb
            print tabu
            tabu.pop(0)
            tabu.append(comb)
            m=comb[0]
            n=comb[1]
            h=comb[2]
          psspy.case(r"""C:\Users\melatope\Desktop\Exp Training\UJ\IEEE\IEEE 2015\Tabu Search prog and casefiles\Case files_TS\132kV_Off Peak_Line 60_120_ additional network.sav""")
            lines=[(6, 12, 1), (12, 13, 1), (6, 13, 1), (6, 11, 1),  (9, 10, 1), (10, 11, 1), (9, 14, 1), (13, 14, 1)]
            corridors=[(6,60,
120,12),(12,112,113,13),(6,116,130,13),(6,160,111,11),(9,90,100,10),(10,1010,110,11),(9,119,140,14
),(13,133,114,14)]
            off_line=lines[m]
            on_corridor=corridors[m]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[n]
            on_corridor=corridors[n]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[h]
            on_corridor=corridors[h]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
```

```
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            p_init=120
            p_new=120
            scale=1
            p_list=[]
            sys_MW = []
            sys_Losses = []
            sys_MVar = []
            L6_11MVA = []
            L6_12MVA = []
            L6_13MVA = []
            L9_10MVA = []
            L9_14MVA = []
            L10_11MVA = []
            L12_13MVA = []
            L13_14MVA = []
            L6_11MVAr = []
            L6_12MVAr = []
            L6_13MVAr = []
            L9_10MVAr = []
            L9_14MVAr = []
            L10_11MVAr = []
            L12_13MVAr = []
            L13_14MVAr = []
            L6_11MW = []
            L6_12MW = []
            L6_13MW = []
            L9_10MW = []
            L9_14MW = []
            L10_11MW = []
            L12_13MW = []
            L13_14MW = []
            bus6_volts = []
            bus9_volts = []
            bus10_volts = []
            bus11_volts = []
            bus12_volts = []
            bus13_volts = []
            bus14_volts = []
            j=0
            bus6kV=1
            bus9kV=1
            bus10kV=1
            bus11kV=1
            bus12kV=1
```

```
    bus13kV=1
    bus14kV=1
    LL6_11MVA=10
    LL6_12MVA=10
    LL6_13MVA=10
    LL9_10MVA=10
    LL9_14MVA=10
    LL10_11MVA=10
    LL12_13MVA=10
    LL13_14MVA=10
    sol_load_step=[]
    load_step=0
    while p_new > p_init*0.005 and bus6kV > 0.95 and bus6kV < 1.05 and bus9kV > 0.95 and bus9kV < 1.05 and bus10kV > 0.95 and bus10kV <1.05 and bus11kV > 0.95 and bus11kV <1.05 and bus12kV > 0.95 and bus12kV <1.05 and bus13kV > 0.95 and bus13kV <1.05 and bus14kV > 0.95 and bus14kV <1.05 and LL6_11MVA < 84.5 and LL6_12MVA < 84.5 and LL6_13MVA < 84.5 and LL9_10MVA < 84.5 and LL9_14MVA < 84.5 and LL10_11MVA < 84.5 and LL12_13MVA < 84.5 and LL13_14MVA < 84.5 :
        p_list.append(p_new)
        psspy.scal(0,1,1,[0,0,0,0],[0.0,0.0,0.0,0.0,0.0,0.0,0.0])#new line
        psspy.scal(0,1,2,[1,0,1,0],[ p_new, 120.0,0.0,-.0,0.0,-17.848, 33.066])#new line
        psspy.fnsl([1,0,0,1,1,0,0,0])
        solution=psspy.solved()
        bus14=psspy.busdat(14,"PU")
        bus14kV=bus14[1]
        LL6_11_0_PandQ=psspy.brnflo(6,11,'1')
        LL6_11_PandQ=LL6_11_0_PandQ[1]
        LL6_11_P=LL6_11_PandQ.real
        LL6_11_0_PandQ=psspy.brnflo(6,11,'1')
        LL6_11_PandQ=LL6_11_0_PandQ[1]
        LL6_11_Q=LL6_11_PandQ.imag
        LL6_11MVA=(LL6_11_P**2+LL6_11_Q**2)**0.5
        LL6_12_0_PandQ=psspy.brnflo(6,12,'1')
        LL6_12_PandQ=LL6_12_0_PandQ[1]
        LL6_12_P=LL6_12_PandQ.real
        LL6_12_0_PandQ=psspy.brnflo(6,12,'1')
        LL6_12_PandQ=LL6_12_0_PandQ[1]
        LL6_12_Q=LL6_12_PandQ.imag
        LL6_12MVA=(LL6_12_P**2+LL6_12_Q**2)**0.5
        LL6_13_0_PandQ=psspy.brnflo(6,13,'1')
        LL6_13_PandQ=LL6_13_0_PandQ[1]
        LL6_13_P=LL6_13_PandQ.real
        LL6_13_0_PandQ=psspy.brnflo(6,13,'1')
        LL6_13_PandQ=LL6_13_0_PandQ[1]
        LL6_13_Q=LL6_13_PandQ.imag
        LL6_13MVA=(LL6_13_P**2+LL6_13_Q**2)**0.5
        LL9_10_0_PandQ=psspy.brnflo(9,10,'1')
        LL9_10_PandQ=LL9_10_0_PandQ[1]
        LL9_10_P=LL9_10_PandQ.real
        LL9_10_0_PandQ=psspy.brnflo(9,10,'1')
        LL9_10_PandQ=LL9_10_0_PandQ[1]
        LL9_10_Q=LL9_10_PandQ.imag
        LL9_10MVA=(LL9_10_P**2+LL9_10_Q**2)**0.5
        LL9_14_0_PandQ=psspy.brnflo(9,14,'1')
```

```
LL9_14_PandQ=LL9_14_0_PandQ[1]
LL9_14_P=LL9_14_PandQ.real
LL9_14_0_PandQ=psspy.brnflo(9,14,'1')
LL9_14_PandQ=LL9_14_0_PandQ[1]
LL9_14_Q=LL9_14_PandQ.imag
LL9_14MVA=(LL9_14_P**2+LL9_14_Q**2)**0.5
LL10_11_0_PandQ=psspy.brnflo(10,11,'1')
LL10_11_PandQ=LL10_11_0_PandQ[1]
LL10_11_P=LL10_11_PandQ.real
LL10_11_0_PandQ=psspy.brnflo(10,11,'1')
LL10_11_PandQ=LL10_11_0_PandQ[1]
LL10_11_Q=LL10_11_PandQ.imag
LL10_11MVA=(LL10_11_P**2+LL10_11_Q**2)**0.5
LL12_13_0_PandQ=psspy.brnflo(12,13,'1')
LL12_13_PandQ=LL12_13_0_PandQ[1]
LL12_13_P=LL12_13_PandQ.real
LL12_13_0_PandQ=psspy.brnflo(12,13,'1')
LL12_13_PandQ=LL12_13_0_PandQ[1]
LL12_13_Q=LL12_13_PandQ.imag
LL12_13MVA=(LL12_13_P**2+LL12_13_Q**2)**0.5
LL13_14_0_PandQ=psspy.brnflo(13,14,'1')
LL13_14_PandQ=LL13_14_0_PandQ[1]
LL13_14_P=LL13_14_PandQ.real
LL13_14_0_PandQ=psspy.brnflo(13,14,'1')
LL13_14_PandQ=LL13_14_0_PandQ[1]
LL13_14_Q=LL13_14_PandQ.imag
LL13_14MVA=(LL13_14_P**2+LL13_14_Q**2)**0.5
if solution==0:
    sys_MW.append(psspy.systot('LOAD'))
    sys_Losses.append(psspy.systot('LOSS'))
    sys_MVar.append(psspy.systot('GEN'))
    L6_11MVA.append(psspy.brnflo(6,11,'1'))
    L6_12MVA.append(psspy.brnflo(6,12,'1'))
    L6_13MVA.append(psspy.brnflo(6,13,'1'))
    L9_10MVA.append(psspy.brnflo(9,10,'1'))
    L9_14MVA.append(psspy.brnflo(9,14,'1'))
    L10_11MVA.append(psspy.brnflo(10,11,'1'))
    L12_13MVA.append(psspy.brnflo(12,13,'1'))
    L13_14MVA.append(psspy.brnflo(13,14,'1'))
    L6_11MVAr.append(psspy.brnflo(6,11,'1'))
    L6_12MVAr.append(psspy.brnflo(6,12,'1'))
    L6_13MVAr.append(psspy.brnflo(6,13,'1'))
    L9_10MVAr.append(psspy.brnflo(9,10,'1'))
    L9_14MVAr.append(psspy.brnflo(9,14,'1'))
    L10_11MVAr.append(psspy.brnflo(10,11,'1'))
    L12_13MVAr.append(psspy.brnflo(12,13,'1'))
    L13_14MVAr.append(psspy.brnflo(13,14,'1'))
    L6_11MW.append(psspy.brnflo(6,11,'1'))
    L6_12MW.append(psspy.brnflo(6,12,'1'))
    L6_13MW.append(psspy.brnflo(6,13,'1'))
    L9_10MW.append(psspy.brnflo(9,10,'1'))
    L9_14MW.append(psspy.brnflo(9,14,'1'))
    L10_11MW.append(psspy.brnflo(10,11,'1'))
    L12_13MW.append(psspy.brnflo(12,13,'1'))
```

```
                        L13_14MW.append(psspy.brnflo(13,14,'1'))
                        bus6_volts.append(psspy.busdat(6,'PU'))
                        bus9_volts.append(psspy.busdat(9,'PU'))
                        bus10_volts.append(psspy.busdat(10,'PU'))
                        bus11_volts.append(psspy.busdat(11,'PU'))
                        bus12_volts.append(psspy.busdat(12,'PU'))
                        bus13_volts.append(psspy.busdat(13,'PU'))
                        bus14_volts.append(psspy.busdat(14,'PU'))
                    else:
                        break
                    p_new=p_new-scale
                    load_step=load_step+1
                    load_rem=p_init-(load_step-1)*scale
            sol_comb.append(comb)
            obj_fun.append(load_step)
            all_sol_comb.append(comb)
            all_obj_fun.append(load_step)
            all_load_rem.append(load_rem)
            print sol_comb
            print obj_fun
    max_value = max(obj_fun)#for this iteration
    print max_value
    max_index = obj_fun.index(max_value)
    print max_index
    (k,l,a)=sol_comb[max_index]
    print(k,l,a)
    print (k,l,a)
    if (k,l,a) in all_sol_comb:
            k=randint(0,len(lines_id)-1)
    iter=iter+1
print iter-1
print all_sol_comb
print all_load_rem
print all_obj_fun
max_value = max(all_obj_fun)
print max_value
max_index = all_obj_fun.index(max_value)
print max_index
(k,l,a)=all_sol_comb[max_index]
print (k,l,a)
opt_load=p_init-(max_value-1)*scale
print opt_load
print'%10s'%'Number'   ,'%10s'%'Combination'   ,'%10s'%'Load'   , '%10s'%'Steps'
while j < len(all_sol_comb):
  while j < len(all_load_rem):
    while j < len(all_obj_fun):
        t=j+1
        comb=all_sol_comb[j]
        load=all_load_rem[j]
        objfunc=all_obj_fun[j]
        print '%10.0f'%t,comb,'%10.3f'%load,objfunc
        j=j+1
```

Appendix D: Tabu Search Code for Optimal Selection of Four Lines Combination

D.1 Introduction

In this appendix, presents the Python syntax for a TS optimal selection of four lines combination. The first iteration of (k, l, a, b) are set as the first bet of lines. The combinations are recorded for all possible moves and the results are captured at the end of the program.

Tabu Search Code:

```
import random
iter=1
all_sol_comb=[]
all_obj_fun=[]
all_load_rem=[]
(k,l,a,b)=(3,2,4,1)
tabu=[0,0,0,0,0]
while iter<50:
    lines_id=[0,1,2,3,4,5,6,7]
    sol_comb=[]
    obj_fun=[]
    print iter
    print (k,l,a,b)
    if (k,l,a,b) not in sol_comb:
        i=0
        j=0
        s=0
        z=0
        neighbourhood=[]
        while i<len(lines_id):
            ne_soln=(k,i,a,b)
            if k!=i and k!=a and k!=b and i!=a and i!=b and a!=b:
                neighbourhood.append(ne_soln)
            i=i+1
        while j<len(lines_id):
            ne_soln=(j,l,a,b)
            if j!=l and j!=a and j!=b and l!=a and l!=b and a!=b:
                neighbourhood.append(ne_soln)
            j=j+1
        while s<len(lines_id):
            ne_soln=(k,l,s,b)
            if k!=l and k!=s and k!=b and l!=s and l!=b and s!=b:
                neighbourhood.append(ne_soln)
            s=s+1
```

```
    while z<len(lines_id):
        ne_soln=(k,l,a,z)
        if k!=l and k!=a and k!=z and l!=a and l!=z and a!=z:
            neighbourhood.append(ne_soln)
        z=z+1
    print (k,l,a,b)
    print neighbourhood
    comb=(k,l,a,b)
    for comb in neighbourhood:
        if comb not in tabu:
            print comb
            print sol_comb
            print tabu
            tabu.pop(0)
            tabu.append(comb)
            m=comb[0]
            n=comb[1]
            h=comb[2]
            d=comb[3]
            psspy.case(r"""C:\Users\melatope\Desktop\Exp Training\UJ\IEEE\IEEE 2015\Tabu Search
prog and casefiles\Case files_TS\132kV_Off Peak_Line 60_120_ additional network.sav""")
            lines=[(6, 12, 1), (12, 13, 1), (6, 13, 1), (6, 11, 1),  (9, 10, 1), (10, 11, 1), (9, 14, 1), (13, 14, 1)]
corridors=[(6,60,120,12),(12,112,113,13),(6,116,130,13),(6,160,111,11),(9,90,100,10),(10,1010,110,
11),(9,119,140,14),(13,133,114,14)]
            off_line=lines[m]
            on_corridor=corridors[m]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[n]
            on_corridor=corridors[n]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[h]
            on_corridor=corridors[h]
```

```
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)

psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[d]
            on_corridor=corridors[d]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            p_init=120
            p_new=120
            scale=1
            p_list=[]
            sys_MW = []
            sys_Losses = []
            sys_MVar = []
            L6_11MVA = []
            L6_12MVA = []
            L6_13MVA = []
            L9_10MVA = []
            L9_14MVA = []
            L10_11MVA = []
            L12_13MVA = []
            L13_14MVA = []
            L6_11MVAr = []
            L6_12MVAr = []
            L6_13MVAr = []
            L9_10MVAr = []
            L9_14MVAr = []
            L10_11MVAr = []
            L12_13MVAr = []
            L13_14MVAr = []
            L6_11MW = []
            L6_12MW = []
            L6_13MW = []
            L9_10MW = []
```

```
L9_14MW = []
L10_11MW = []
L12_13MW = []
L13_14MW = []
bus6_volts = []
bus9_volts = []
bus10_volts = []
bus11_volts = []
bus12_volts = []
bus13_volts = []
bus14_volts = []
j=0
bus6kV=1
bus9kV=1
bus10kV=1
bus11kV=1
bus12kV=1
bus13kV=1
bus14kV=1
LL6_11MVA=10
LL6_12MVA=10
LL6_13MVA=10
LL9_10MVA=10
LL9_14MVA=10
LL10_11MVA=10
LL12_13MVA=10
LL13_14MVA=10
sol_load_step=[]
load_step=0
while p_new > p_init*0.005 and bus6kV > 0.95 and bus6kV < 1.05 and bus9kV > 0.95 and bus9kV < 1.05 and bus10kV > 0.95 and bus10kV <1.05 and bus11kV > 0.95 and bus11kV <1.05 and bus12kV > 0.95 and bus12kV <1.05 and bus13kV > 0.95 and bus13kV <1.05 and bus14kV > 0.95 and bus14kV <1.05 and LL6_11MVA < 84.5 and LL6_12MVA < 84.5 and LL6_13MVA < 84.5 and LL9_10MVA < 84.5 and LL9_14MVA < 84.5 and LL10_11MVA < 84.5 and LL12_13MVA < 84.5 and LL13_14MVA < 84.5 :
    p_list.append(p_new)
    psspy.scal(0,1,1,[0,0,0,0],[0.0,0.0,0.0,0.0,0.0,0.0,0.0])#new line
    psspy.scal(0,1,2,[1,0,1,0],[ p_new, 120.0,0.0,-.0,0.0,-17.848, 33.066])#new line
    psspy.fnsl([1,0,0,1,1,0,0,0])
    solution=psspy.solved()
    bus14=psspy.busdat(14,"PU")
    bus14kV=bus14[1]
    LL6_11_0_PandQ=psspy.brnflo(6,11,'1')
    LL6_11_PandQ=LL6_11_0_PandQ[1]
    LL6_11_P=LL6_11_PandQ.real
    LL6_11_0_PandQ=psspy.brnflo(6,11,'1')
    LL6_11_PandQ=LL6_11_0_PandQ[1]
    LL6_11_Q=LL6_11_PandQ.imag
    LL6_11MVA=(LL6_11_P**2+LL6_11_Q**2)**0.5
    LL6_12_0_PandQ=psspy.brnflo(6,12,'1')
    LL6_12_PandQ=LL6_12_0_PandQ[1]
    LL6_12_P=LL6_12_PandQ.real
    LL6_12_0_PandQ=psspy.brnflo(6,12,'1')
    LL6_12_PandQ=LL6_12_0_PandQ[1]
```

```
LL6_12_Q=LL6_12_PandQ.imag
LL6_12MVA=(LL6_12_P**2+LL6_12_Q**2)**0.5
LL6_13_0_PandQ=psspy.brnflo(6,13,'1')
LL6_13_PandQ=LL6_13_0_PandQ[1]
LL6_13_P=LL6_13_PandQ.real
LL6_13_0_PandQ=psspy.brnflo(6,13,'1')
LL6_13_PandQ=LL6_13_0_PandQ[1]
LL6_13_Q=LL6_13_PandQ.imag
LL6_13MVA=(LL6_13_P**2+LL6_13_Q**2)**0.5
LL9_10_0_PandQ=psspy.brnflo(9,10,'1')
LL9_10_PandQ=LL9_10_0_PandQ[1]
LL9_10_P=LL9_10_PandQ.real
LL9_10_0_PandQ=psspy.brnflo(9,10,'1')
LL9_10_PandQ=LL9_10_0_PandQ[1]
LL9_10_Q=LL9_10_PandQ.imag
LL9_10MVA=(LL9_10_P**2+LL9_10_Q**2)**0.5
LL9_14_0_PandQ=psspy.brnflo(9,14,'1')
LL9_14_PandQ=LL9_14_0_PandQ[1]
LL9_14_P=LL9_14_PandQ.real
LL9_14_0_PandQ=psspy.brnflo(9,14,'1')
LL9_14_PandQ=LL9_14_0_PandQ[1]
LL9_14_Q=LL9_14_PandQ.imag
LL9_14MVA=(LL9_14_P**2+LL9_14_Q**2)**0.5
LL10_11_0_PandQ=psspy.brnflo(10,11,'1')
LL10_11_PandQ=LL10_11_0_PandQ[1]
LL10_11_P=LL10_11_PandQ.real
LL10_11_0_PandQ=psspy.brnflo(10,11,'1')
LL10_11_PandQ=LL10_11_0_PandQ[1]
LL10_11_Q=LL10_11_PandQ.imag
LL10_11MVA=(LL10_11_P**2+LL10_11_Q**2)**0.5
LL12_13_0_PandQ=psspy.brnflo(12,13,'1')
LL12_13_PandQ=LL12_13_0_PandQ[1]
LL12_13_P=LL12_13_PandQ.real
LL12_13_0_PandQ=psspy.brnflo(12,13,'1')
LL12_13_PandQ=LL12_13_0_PandQ[1]
LL12_13_Q=LL12_13_PandQ.imag
LL12_13MVA=(LL12_13_P**2+LL12_13_Q**2)**0.5
LL13_14_0_PandQ=psspy.brnflo(13,14,'1')
LL13_14_PandQ=LL13_14_0_PandQ[1]
LL13_14_P=LL13_14_PandQ.real
LL13_14_0_PandQ=psspy.brnflo(13,14,'1')
LL13_14_PandQ=LL13_14_0_PandQ[1]
LL13_14_Q=LL13_14_PandQ.imag
LL13_14MVA=(LL13_14_P**2+LL13_14_Q**2)**0.5
if solution==0:
    sys_MW.append(psspy.systot('LOAD'))
    sys_Losses.append(psspy.systot('LOSS'))
    sys_MVar.append(psspy.systot('GEN'))
    L6_11MVA.append(psspy.brnflo(6,11,'1'))
    L6_12MVA.append(psspy.brnflo(6,12,'1'))
    L6_13MVA.append(psspy.brnflo(6,13,'1'))
    L9_10MVA.append(psspy.brnflo(9,10,'1'))
    L9_14MVA.append(psspy.brnflo(9,14,'1'))
    L10_11MVA.append(psspy.brnflo(10,11,'1'))
```

```
                        L12_13MVA.append(psspy.brnflo(12,13,'1'))
                        L13_14MVA.append(psspy.brnflo(13,14,'1'))
                        L6_11MVAr.append(psspy.brnflo(6,11,'1'))
                        L6_12MVAr.append(psspy.brnflo(6,12,'1'))
                        L6_13MVAr.append(psspy.brnflo(6,13,'1'))
                        L9_10MVAr.append(psspy.brnflo(9,10,'1'))
                        L9_14MVAr.append(psspy.brnflo(9,14,'1'))
                        L10_11MVAr.append(psspy.brnflo(10,11,'1'))
                        L12_13MVAr.append(psspy.brnflo(12,13,'1'))
                        L13_14MVAr.append(psspy.brnflo(13,14,'1'))
                        L6_11MW.append(psspy.brnflo(6,11,'1'))
                        L6_12MW.append(psspy.brnflo(6,12,'1'))
                        L6_13MW.append(psspy.brnflo(6,13,'1'))
                        L9_10MW.append(psspy.brnflo(9,10,'1'))
                        L9_14MW.append(psspy.brnflo(9,14,'1'))
                        L10_11MW.append(psspy.brnflo(10,11,'1'))
                        L12_13MW.append(psspy.brnflo(12,13,'1'))
                        L13_14MW.append(psspy.brnflo(13,14,'1'))
                        bus6_volts.append(psspy.busdat(6,'PU'))
                        bus9_volts.append(psspy.busdat(9,'PU'))
                        bus10_volts.append(psspy.busdat(10,'PU'))
                        bus11_volts.append(psspy.busdat(11,'PU'))
                        bus12_volts.append(psspy.busdat(12,'PU'))
                        bus13_volts.append(psspy.busdat(13,'PU'))
                        bus14_volts.append(psspy.busdat(14,'PU'))
                    else:
                        break
                    p_new=p_new-scale
                    load_step=load_step+1
                    load_rem=p_init-(load_step-1)*scale
            sol_comb.append(comb)
            obj_fun.append(load_step)
            all_sol_comb.append(comb)
            all_obj_fun.append(load_step)
            all_load_rem.append(load_rem)
            print sol_comb
            print obj_fun
    max_value = max(obj_fun)
    print max_value
    max_index = obj_fun.index(max_value)
    print max_index
    (k,l,a,b)=sol_comb[max_index]
    print (k,l,a,b)
    print (k,l,a,b)
    if (k,l,a,b) in all_sol_comb:
            k=randint(0,len(lines_id)-1)
    iter=iter+1
print iter-1
print all_sol_comb
print all_load_rem
print all_obj_fun
max_value = max(all_obj_fun)
print max_value
max_index = all_obj_fun.index(max_value)
```

```
print max_index
(k,l,a,b)=all_sol_comb[max_index]
print (k,l,a,b)
opt_load=p_init-(max_value-1)*scale
print 'The optimal load that can be served is:',opt_load
print'%10s'%'Number'   ,'%10s'%'Combination'   ,'%10s'%'Load'   , '%10s'%'Steps'
while j < len(all_sol_comb):
  while j < len(all_load_rem):
    while j < len(all_obj_fun):
        t=j+1
        comb=all_sol_comb[j]
        load=all_load_rem[j]
        objfunc=all_obj_fun[j]
        print '%10.0f'%t,comb,'%10.3f'%load,objfunc
        j=j+1
```

Appendix E: Tabu Search Code for Optimal Selection of Five Lines Combination

E.1 Introduction

In this appendix, TS optimal selection of five lines combination is presented. Similarly to the previous TS programs, the program is given the first iteration to initialise the solution. The first iteration of (k, l, a, b, c) were as follows (3, 2, 4, 1, 5).

TS generate the neighbourhoods based on the possible moves and the loadflows are performed. Once the best selection of best five lines has been found, and maximum iterations have been met, the program termites and results are generated.

Tabu Search Code:

```
import random
iter=1
all_sol_comb=[]
all_obj_fun=[]
all_load_rem=[]
(k,l,a,b,c)=(3,2,4,1,5)
tabu=[0,0,0,0,0]
while iter<10:
    lines_id=[0,1,2,3,4,5,6,7]
    sol_comb=[]
    obj_fun=[]
    print, iter
    print (k,l,a,b,c)
    if (k,l,a,b,c) not in sol_comb:
        i=0
        j=0
        s=0
        z=0
        r=0
        neighbourhood=[]
        while i<len(lines_id):
            ne_soln=(k,i,a,b,c)
            if k!=i and k!=a and k!=b and k!=c and i!=a and i!=b and i!=c and a!=b and a!=c and b!=c:
                neighbourhood.append(ne_soln)
            i=i+1
        while j<len(lines_id):
            ne_soln=(j,l,a,b,c)
            if j!=l and j!=a and j!=b and j!=c and l!=a and l!=b and l!=c and a!=b and a!=c and b!=c:
                neighbourhood.append(ne_soln)
```

```
        j=j+1
    while s<len(lines_id):
        ne_soln=(k,l,s,b,c)
        if k!=l and k!=s and k!=b and k!=c and l!=s and l!=b and l!=c and s!=b and s!=c and b!=c:
            neighbourhood.append(ne_soln)
        s=s+1
    while z<len(lines_id):
        ne_soln=(k,l,a,z,c)
        if k!=l and k!=a and k!=z and k!=c and l!=a and l!=z and l!=c and a!=z and l!=c and z!=c:
            neighbourhood.append(ne_soln)
        z=z+1
    while r<len(lines_id):
        ne_soln=(k,l,a,b,r)
        if k!=l and k!=a and k!=b and k!=r and l!=a and l!=b and l!=r and a!=b and a!=r and b!=r:
            neighbourhood.append(ne_soln)
        r=r+1
    print (k,l,a,b,c)
    print neighbourhood
    comb=(k,l,a,b,c)
    for comb in neighbourhood:
        if comb not in tabu:
            print comb
            print sol_comb
            print tabu
            tabu.pop(0)
            tabu.append(comb)
            m=comb[0]
            n=comb[1]
            h=comb[2]
            d=comb[3]
            e=comb[4]
            psspy.case(r"""C:\Users\melatope\Desktop\Exp Training\UJ\IEEE\IEEE 2015\Tabu Search
prog and casefiles\Case files_TS\132kV_Off Peak_Line 60_120_ additional network.sav""")
            lines=[(6, 12, 1), (12, 13, 1), (6, 13, 1), (6, 11, 1),  (9, 10, 1), (10, 11, 1), (9, 14, 1), (13, 14, 1)]
            corridors=[(6,60,
20,12),(12,112,113,13),(6,116,130,13),(6,160,111,11),(9,90,100,10),(10,1010,110,11),(9,119,140,14),
(13,133,114,14)]
            off_line=lines[m]
            on_corridor=corridors[m]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[n]
            on_corridor=corridors[n]
            bus_s=on_corridor[0]
```

```
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[h]
            on_corridor=corridors[h]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[d]
            on_corridor=corridors[d]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            off_line=lines[e]
            on_corridor=corridors[e]
            bus_s=on_corridor[0]
            bus_r=on_corridor[3]
            bus_ss=on_corridor[1]
            bus_rr=on_corridor[2]
            psspy.bus_chng_3(bus_ss,[1,_i,_i,_i],[_f, 1.0337, 1.504,_f,_f,_f,_f],_s)
            psspy.bus_chng_3(bus_rr,[1,_i,_i,_i],[_f,1.0337,1.504,_f,_f,_f,_f],_s)
psspy.branch_chng(bus_ss,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
psspy.two_winding_chng_4(bus_s,bus_ss,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f
,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.two_winding_chng_4(bus_r,bus_rr,r"""1""",[1,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i,_i],[_f,_f,_f,_f,
_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f],["",""])
psspy.branch_chng(bus_s,bus_r,r"""1""",[0,_i,_i,_i,_i,_i],[_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f,_f])
            p_init=120
```

```
p_new=120
scale=1
p_list=[]
sys_MW = []
sys_Losses = []
sys_MVar = []
L6_11MVA = []
L6_12MVA = []
L6_13MVA = []
L9_10MVA = []
L9_14MVA = []
L10_11MVA = []
L12_13MVA = []
L13_14MVA = []
L6_11MVAr = []
L6_12MVAr = []
L6_13MVAr = []
L9_10MVAr = []
L9_14MVAr = []
L10_11MVAr = []
L12_13MVAr = []
L13_14MVAr = []
L6_11MW = []
L6_12MW = []
L6_13MW = []
L9_10MW = []
L9_14MW = []
L10_11MW = []
L12_13MW = []
L13_14MW = []
bus6_volts = []
bus9_volts = []
bus10_volts = []
bus11_volts = []
bus12_volts = []
bus13_volts = []
bus14_volts = []
j=0
bus6kV=1
bus9kV=1
bus10kV=1
bus11kV=1
bus12kV=1
bus13kV=1
bus14kV=1
LL6_11MVA=10
LL6_12MVA=10
LL6_13MVA=10
LL9_10MVA=10
LL9_14MVA=10
LL10_11MVA=10
LL12_13MVA=10
LL13_14MVA=10
sol_load_step=[]
```

```
        load_step=0
        while p_new > p_init*0.005 and bus6kV > 0.95 and bus6kV < 1.05 and bus9kV > 0.95 and bus9kV < 1.05 and bus10kV > 0.95 and bus10kV <1.05 and bus11kV > 0.95 and bus11kV <1.05 and bus12kV > 0.95 and bus12kV <1.05 and bus13kV > 0.95 and bus13kV <1.05 and bus14kV > 0.95 and bus14kV <1.05 and LL6_11MVA < 84.5 and LL6_12MVA < 84.5 and LL6_13MVA < 84.5 and LL9_10MVA < 84.5 and LL9_14MVA < 84.5 and LL10_11MVA < 84.5 and LL12_13MVA < 84.5 and LL13_14MVA < 84.5 :
            p_list.append(p_new)
            psspy.scal(0,1,1,[0,0,0,0],[0.0,0.0,0.0,0.0,0.0,0.0,0.0])
            psspy.scal(0,1,2,[1,0,1,0],[ p_new, 120.0,0.0,-.0,0.0,-17.848, 33.066])
            psspy.fnsl([1,0,0,1,1,0,0,0])
            solution=psspy.solved()
            bus14=psspy.busdat(14,"PU")
            bus14kV=bus14[1]
            LL6_11_0_PandQ=psspy.brnflo(6,11,'1')
            LL6_11_PandQ=LL6_11_0_PandQ[1]
            LL6_11_P=LL6_11_PandQ.real
            LL6_11_0_PandQ=psspy.brnflo(6,11,'1')
            LL6_11_PandQ=LL6_11_0_PandQ[1]
            LL6_11_Q=LL6_11_PandQ.imag
            LL6_11MVA=(LL6_11_P**2+LL6_11_Q**2)**0.5
            LL6_12_0_PandQ=psspy.brnflo(6,12,'1')
            LL6_12_PandQ=LL6_12_0_PandQ[1]
            LL6_12_P=LL6_12_PandQ.real
            LL6_12_0_PandQ=psspy.brnflo(6,12,'1')
            LL6_12_PandQ=LL6_12_0_PandQ[1]
            LL6_12_Q=LL6_12_PandQ.imag
            LL6_12MVA=(LL6_12_P**2+LL6_12_Q**2)**0.5
            LL6_13_0_PandQ=psspy.brnflo(6,13,'1')
            LL6_13_PandQ=LL6_13_0_PandQ[1]
            LL6_13_P=LL6_13_PandQ.real
            LL6_13_0_PandQ=psspy.brnflo(6,13,'1')
            LL6_13_PandQ=LL6_13_0_PandQ[1]
            LL6_13_Q=LL6_13_PandQ.imag
            LL6_13MVA=(LL6_13_P**2+LL6_13_Q**2)**0.5
            LL9_10_0_PandQ=psspy.brnflo(9,10,'1')
            LL9_10_PandQ=LL9_10_0_PandQ[1]
            LL9_10_P=LL9_10_PandQ.real
            LL9_10_0_PandQ=psspy.brnflo(9,10,'1')
            LL9_10_PandQ=LL9_10_0_PandQ[1]
            LL9_10_Q=LL9_10_PandQ.imag
            LL9_10MVA=(LL9_10_P**2+LL9_10_Q**2)**0.5
            LL9_14_0_PandQ=psspy.brnflo(9,14,'1')
            LL9_14_PandQ=LL9_14_0_PandQ[1]
            LL9_14_P=LL9_14_PandQ.real
            LL9_14_0_PandQ=psspy.brnflo(9,14,'1')
            LL9_14_PandQ=LL9_14_0_PandQ[1]
            LL9_14_Q=LL9_14_PandQ.imag
            LL9_14MVA=(LL9_14_P**2+LL9_14_Q**2)**0.5
            LL10_11_0_PandQ=psspy.brnflo(10,11,'1')
            LL10_11_PandQ=LL10_11_0_PandQ[1]
            LL10_11_P=LL10_11_PandQ.real
            LL10_11_0_PandQ=psspy.brnflo(10,11,'1')
            LL10_11_PandQ=LL10_11_0_PandQ[1]
```

```
LL10_11_Q=LL10_11_PandQ.imag
LL10_11MVA=(LL10_11_P**2+LL10_11_Q**2)**0.5
LL12_13_0_PandQ=psspy.brnflo(12,13,'1')
LL12_13_PandQ=LL12_13_0_PandQ[1]
LL12_13_P=LL12_13_PandQ.real
LL12_13_0_PandQ=psspy.brnflo(12,13,'1')
LL12_13_PandQ=LL12_13_0_PandQ[1]
LL12_13_Q=LL12_13_PandQ.imag
LL12_13MVA=(LL12_13_P**2+LL12_13_Q**2)**0.5
LL13_14_0_PandQ=psspy.brnflo(13,14,'1')
LL13_14_PandQ=LL13_14_0_PandQ[1]
LL13_14_P=LL13_14_PandQ.real
LL13_14_0_PandQ=psspy.brnflo(13,14,'1')
LL13_14_PandQ=LL13_14_0_PandQ[1]
LL13_14_Q=LL13_14_PandQ.imag
LL13_14MVA=(LL13_14_P**2+LL13_14_Q**2)**0.5
if solution==0:
    sys_MW.append(psspy.systot('LOAD'))
    sys_Losses.append(psspy.systot('LOSS'))
    sys_MVar.append(psspy.systot('GEN'))
    L6_11MVA.append(psspy.brnflo(6,11,'1'))
    L6_12MVA.append(psspy.brnflo(6,12,'1'))
    L6_13MVA.append(psspy.brnflo(6,13,'1'))
    L9_10MVA.append(psspy.brnflo(9,10,'1'))
    L9_14MVA.append(psspy.brnflo(9,14,'1'))
    L10_11MVA.append(psspy.brnflo(10,11,'1'))
    L12_13MVA.append(psspy.brnflo(12,13,'1'))
    L13_14MVA.append(psspy.brnflo(13,14,'1'))
    L6_11MVAr.append(psspy.brnflo(6,11,'1'))
    L6_12MVAr.append(psspy.brnflo(6,12,'1'))
    L6_13MVAr.append(psspy.brnflo(6,13,'1'))
    L9_10MVAr.append(psspy.brnflo(9,10,'1'))
    L9_14MVAr.append(psspy.brnflo(9,14,'1'))
    L10_11MVAr.append(psspy.brnflo(10,11,'1'))
    L12_13MVAr.append(psspy.brnflo(12,13,'1'))
    L13_14MVAr.append(psspy.brnflo(13,14,'1'))
    L6_11MW.append(psspy.brnflo(6,11,'1'))
    L6_12MW.append(psspy.brnflo(6,12,'1'))
    L6_13MW.append(psspy.brnflo(6,13,'1'))
    L9_10MW.append(psspy.brnflo(9,10,'1'))
    L9_14MW.append(psspy.brnflo(9,14,'1'))
    L10_11MW.append(psspy.brnflo(10,11,'1'))
    L12_13MW.append(psspy.brnflo(12,13,'1'))
    L13_14MW.append(psspy.brnflo(13,14,'1'))
    bus6_volts.append(psspy.busdat(6,'PU'))
    bus9_volts.append(psspy.busdat(9,'PU'))
    bus10_volts.append(psspy.busdat(10,'PU'))
    bus11_volts.append(psspy.busdat(11,'PU'))
    bus12_volts.append(psspy.busdat(12,'PU'))
    bus13_volts.append(psspy.busdat(13,'PU'))
    bus14_volts.append(psspy.busdat(14,'PU'))
else:
    break
p_new=p_new-scale
```

```
                    load_step=load_step+1
                    load_rem=p_init-(load_step-1)*scale
                sol_comb.append(comb)
                obj_fun.append(load_step)
                all_sol_comb.append(comb)
                all_obj_fun.append(load_step)
                print sol_comb
                print obj_fun
    max_value = max(obj_fun)
    print max_value
    max_index = obj_fun.index(max_value)
    print max_index
    (k,l,a,b,c)=sol_comb[max_index]
    print (k,l,a,b,c)
    if (k,l,a,b,c) in all_sol_comb:
        k=randint(0,len(lines_id)-1)
    iter=iter+1
print iter-1
print all_sol_comb
print all_load_rem
print all_obj_fun
max_value = max(all_obj_fun)
print max_value
max_index = all_obj_fun.index(max_value)
print max_index
(k,l,a,b,c)=all_sol_comb[max_index]
print k,l,a,b,c
opt_load=p_init-(max_value-1)*scale
print opt_load
print'%10s'%'Number'   ,'%10s'%'Combination'   ,'%10s'%'Load'   , '%10s'%'Steps'
while j < len(all_sol_comb):
  while j < len(all_load_rem):
    while j < len(all_obj_fun):
        t=j+1
        comb=all_sol_comb[j]
        load=all_load_rem[j]
        objfunc=all_obj_fun[j]
        print '%10.0f'%t,comb,'%10.3f'%load,objfunc
        j=j+1
```

References

[1] A. Sina, A. Aeenmehr, and H. Mohamadian, “Power system voltage control using lp and artificial neural network,” in *Proc. 2012 World Academy of Science Engineering and Technology Conf.,* pp. 679-683.

[2] P. Kundur, “Power System Stability and Control,” McGraw-Hill, Inc., New York: USA, 1993.

[3] Eskom Holdings SoC Limited, System Operation and Planning Division, Johannesburg, “Transmission System Planning Standard,” March 2009.

[4] Eskom Holdings SoC Limited, System Operator Security Guideline, Unique Identifier: 240-42285014, September 2009.

[5] I. Sadeghkhani, A. Ketabi, and R. Feuillet, “Radial Basis Function Neural Network Application to Measure and Control of Shunt Reactor Overvoltages Based on Analytical Rules,” *Hawayi Publishing Corporation,* in *Proc.,* vol. 2012, pp.1-13, August 2012.

[6] D. Lindenmeyer, H. W. Dommel, A. Moshef, and P. Kunder, “Analysis and Control of Harmonic Over Voltages during Restoration,” in Proc. 1999 IPST Power Systems Transients Conf., pp. 635-640.

[7] P. Melato, N. Mbuli, J. H. Pretorius, “Derating Line Voltage as an Option for Reliability Enhancement of Lightly Loaded Networks,” in Proc. 2014, 49th Int. Conf. Universities Power Engineering, Romania, 2014, pp. 1-6.

[8] J.D. Glover, S. M. Sarma, T. J. Overby, “Power System Analysis and Design,” Cengage Learning., Stamford: USA, 2012.

[9] I. Sadeghkhani, N. S. Monajemi, H. Khosravian,R. Monajemi, and M. Rezaeirad “An Intelligent Estimator for Reactor Switching Studies,” in Proc. 2013, International Journal of Electronics Communications and Electrical Engineering (Volume: 5), May 2013, pp. 18-31.

[10] I. Sadeghkhani, A. Ketabi, and R. Feuillet, “Artificial Neural based Approach to Analyse Transient Overvoltages during Capacitor Bank Switching,” IJC Special Issue on “Artificial Intelligence Techniques-Novel Approach and Practical Applications, AIT, June 2011, pp. 5-11.

[11] A. Moreno-Munoz, “Power Quality Mitigation Technologies in a Distributed Environment,” Power System Series, Springer-Verlag London Limited, 2007.

[12] Energy Efficiency and Power Quality Series, “Power Quality Basics: Fixing High Voltage and Overvoltage Problems,” Utility Systems Technologies, Inc., October, 2009.

[13] Berube, G.R., Berube, G.R., Hajagos, L.M., Beaulieu, R.E.; “Utility Perspective on Under-Excitation Limiters, “ *IEEE Trans. Energy Conversion*, vol. 10, pp. 532-537, , Sep. 1995.

[14] Electricity Supply-Quality of Supply, Part 2: Voltage Characteristics, 2nd Edition, NRS 048-2: 2003.

[15] Suriyamongkol, D, Non-technical Losses in Electrical Power Systems, Master of Science Thesis, Ohio University, November 2002.

[16] V. A. Diachkov, A. A. Gremiakov, and V. A. Stroev, “Alleviation of Steady State Overvoltages in Electrical Power Systems,” in Proc. 2003, IEEE PowerTech Conference (Volume: 3), Bologna: Italy, 23-26 June 2003, pp. 1-5.

[17] Electricity Supply-Quality of Supply, Part 2: Voltage Characteristics, 2nd Edition, NRS 048-2: 2003.

[18] A. Moreno-Munoz, “Power Quality Mitigation Technologies in a Distributed Environment,” Power System Series, Springer-Verlag London Limited, 2007.

[19] I. Sadeghkhani, A. Ketabi, and S. A. Taher, “Mitigation of Shunt Reactor Overvoltages Using Delta-Bar-Delta and Directed Random Search Algorithms.” Przeglad Elektrotechniczny, R. 88 NR 12a/2012, pp.269-274.

[20] M. V. Escudero and M. Redfern, “Effects of Transmission Line Construction on Resonance in Shunt Compensated EHV Lines,” International Conference on Power Systems Transients (IPST’05) in Montreal, Canada on June 19-23, 2005 Paper No. IPST05 – 109, pp. 1-6.

[21] PG&E Transmission Interconnection Handbook, “Section G2: Protection and Control Requirements for Transmission Generation Entities,” May 2013.

[22] N. Hiscock, T. Hazel, and J. A. Hiscock, “Voltage Regulation at Sites with Distributed Generation,” in Proc. 2008, IEEE Transections on Industrial Applications, (Volume: 44), March/April 2008, pp. 445-454.

[23] S. van Zyl and C. T. Gaunt, “A generalised method for evaluating voltage rise in dg equipped networks,” presented at the Cigre Conf., 2005.

[24] T. Tran-Quoc, E. Monnot, G. Rami, A. Almeida, Ch. Kieny, and N. Hadjsaid, “Intelligent voltage control in distribution network with distributed generation,” in *Proc. 2007 International Electricity Distribution Conf.*, pp. 1-4.

[25] T. J. Hashim, A. Mohamed, and H. Shareef, “A review on voltage control methods for active distribution networks,” Prezeglad Elektroteniczny, pp. 304-312, June. 2012.

[26] S. Chattopadhyay, M. Mitra, and S. Sengupta, “Electric Power Quality,” Springer: New York, 2011.

[27] PG&E Transmission Interconnection Handbook, “Section G2: Protection and Control Requirements for Transmission Generation Entities,” May 2013.

[28] Electricity Supply-Quality of Supply, Part 4: Application Guidelines for Utilities, Edition 1.1, NRS 048-4: 1999.

[29] J. D. Glover, M. S. Sarma, and T. J. Overbye, “Power system analysis and design,” 5th ed., Connecticut: CENGAGE Learning, 2010.

[30] C. W. Taylor, “Power System Voltage Stability,” McGraw-Hill, Inc., New York: USA, 1992.

[31] PSS/E, Power System Simulator for Engineering: Online Documentation : Ver. 33 (Siemens Energy, Inc., 2009).

[32] PYTHON, Online Documentation : Ver. 32 (Siemens Energy, Inc., 2009).

[33] S. Kamel Mena Kodsi, and C. A. Canizares, Modelling and Simulation of IEEE 14 Bus System With Facts Controllers.
Available[Online]:https://ece.uwaterloo.ca/~ccanizar/.../IEEEBenchmarkTFreport
Last accessed 05 June 2014

[34] Application Program Interface Power System Simulator for Engineering: Online Documentation : (Siemens Energy, Inc., 2013).

[35] P. Melato, N. Mbuli, J. H. Pretorius, "Derating Line Voltage as an Option for Reliability Enhancement of Lightly Loaded Networks," in Proc. 2014, 49th Int. Conf. Universities Power Engineering, Cluj-Napoca: Romania, 2014, pp. 1-6.

[36] F. Glover, M. Laguna and R. Marti, Tabu Search Principles.
Available[online]:http://citeseerx.ist.psu.edu/viewdoc/download%3Fdoi%D10.1.1.302.8195%26rep%Drep1%26type%3Dp/Principles of Tabu Seearch.pdf.
Last accessed 03 February 2016

[37] D.P. Kothari, "Power Systems," in *Proc. 2012Computational Intelligence and Signal Processing (CISP) 2nd National Conf*., pp. 18-21.

[38] H. Mori and Y. Ogita "Parallel Tabu Search for Capacitor Placement in Radial Distribution Systems,''*IEEE Trans. Power Engineering Society Winter Meeting*, vol. 4, pp. 2334-2339, Jan. 2000.

[39] C. S. Chang, J. S. Low, D. Srinivasan, "Application of Tabu Search in Optimal System Design and Operation of MRT Power Supply Systems, "*IEEE Trans. Energy Conversion*, vol. 146, pp. 75-80, Jan. 1999.

[40] X. Yo, G. G. Karady, R. G. Farmer, and B. L. Agrawal H. Tani, "Estimation of Generator Excitation System Parameters by Tabu Seach,''*IEEE Trans. Power Engineering Society Winter Meeting* , vol. 3, pp. 1185-1190, Jan. 2001.

[41] H. Mori and Y. Ogita "Solving the Capacitor Placement in a Radial Distribution System using Tabu Search Approach,''*IEEE Trans. On Power Systems*, vol. 11, pp. 1868-1873, Nov. 1996.

[42] F. Glover, Tabu Search Tutorial.
Available[online]:http:// www.ida.liu.se/~zebpe83/heuristic/papers/TS_tutorial.pdf .
Last accessed 03 February 2016

[43] F. Glover, Chapter 4 Tabu Search Tutorial.
Available[online]:http:// www.uv.es/rmarti/paper/docs/ts2 .pdf.
Last accessed 04 February 2016

[44] C. S. Chang and S. W. Yang, "Optimal Multiobjective Planning of Dynamic Series Compensation Devices for Power Quality Improvement,''*IEEE Trans. Generation, Transmission and Distribution*, vol. 148, pp. 361-370, Jul. 2001.

[45] H. Mori, and H. Tani, "Two-Staged Tabu Search for Determining Optimal Allocation of D-FACTS in Radial Distribution Systems with Distributed Generation,''*IEEE Trans. Transmission and Distribution Conference and Exhibition*, vol. 1, pp. 56-61, Oct. 2002.

[46] Y.Menuelito, D. Castillo and H. Song, "Tabu-Search Based Topology Modification for Reduction of Fault Current Level in Power Systems,''*IEEE Trans. International Journal of Software Engineering & Its Applications* , vol. 7, pp. 2029-2032, Jul. 2013.

[47] R. A. F. Pereira, L. G. W. da Silva and J. R. S. Mantovani, "PMUs Optimized Allocation Using Tabu- Search Algorithm for Fault Location in Electric Power Distribution System,'' in *Proc. 2004 IEEE/PSE Transmission and Distribution Conference and Exposition.,* pp.143-148.

[48] Z. Zheng, C. Wei and S.Lin, "A Maximum power point Tracking Method on Tabu Search for PV Systems under partially Shaded Conditions,'' in *Proc. 2004 IEEE State Key Laboratory of Electrical Insulation and Power Equipment.,* pp.1-4.

[49] H. Mori and Y. Ogita "A Parallel Tabu Search Based Method for Reconfigurations of Distribution Systems,''*IEEE Trans. Power Engineering Society Summer Meeting*, vol. 1, pp. 73-78, Jul. 2000.