Optimization of Controller for Islanded AC Microgrid System

by

Quazi Nafees Ul Islam

MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONIC ENGINEERING

Department of Electrical and Electronic Engineering
Islamic University of Technology (IUT)
Board Bazar, Gazipur-1704, Bangladesh.
January, 2020

CERTIFICATE OF APPROVAL

The thesis titled,"**Optimization of Controller for Islanded AC Microgrid System**" submitted by Quazi Nafees Ul Islam, St. No. 161021004 of Academic Year 2016-17 has been found as satisfactory and accepted as partial fulfillment of the requirement for the Degree MASTER OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING on 20 January, 2020.

Board of Examiners:

Dr. Ashik Ahmed (Supervisor)
Professor, Chairman
Electrical and Electronic Engineering Department,
Islamic University of Technology (IUT), Gazipur.

Dr. Md. Ruhul Amin (Ex-Officio)
Professor and Head, Member
Electrical and Electronic Engineering Department,
Islamic University of Technology (IUT), Gazipur.

Dr. Md. Ashraful Hoque
Dean, Faculty of Engineering and Technology (FET)
Professor, Electrical and Electronic Engineering Department, Member
Islamic University of Technology (IUT), Gazipur.

Dr. Md. Monirul Kabir
Professor, Member (External)
Electrical and Electronic Engineering Department,
Dhaka University of Engineering and Technology (DUET), Gazipur.

Declaration of Candidate

It is hereby declared that this thesis report or any part of it has not been submitted elsewhere for the award of any Degree or Diploma.

Dr. Ashik Ahmed
Professor,
Electrical and Electronic Engineering department,
Islamic University of Technology (IUT),
Date:20 January, 2020.

Quazi Nafees Ul Islam
Student No.: 161021004,
Academic Year: 2016-17,
Date:20 January, 2020.

Dedicated to my beloved parents and dear wife

Table of Contents

List of Figures

List of Tables

List of Abbreviations

DG	Distributed Generation
MG	Microgrid
DS	Distributed Source
GA	Genetic Algorithm
FA	Firefly Algorithm
SI	Swarm Intelligence
PSO	Particle Swarm Optimization
EP	Evolutionary Programming
ABC	Ant Bee Colony
HEFA	Hybrid Evolutionary Firefly Algorithm
DE	Differential Evolution
WOA	Whale Optimization Algorithm
MGWO	Mean Grey Wolf Optimizer
NSGA	Non-dominated Sorting Genetic Algorithm
NSFA	Non-dominated Sorting Firefly Algorithm
NSWOA	Non-dominated Sorting Whale Optimization Algorithm
IM	Induction Motor
IIDG	Inverter Interfaced Distributed Generation
SPEA	Strength Pareto Evolutionary Algorithm
SPSS	Statistical Package for the Social Sciences

List of Symbols

Symbol	Meaning
δ	Reference angle of Inverter
ω_{com}	Common Reference Frame Frequency
ω_c	Cut off Frequency
ω_{nl}	No Load Frequency
v_{odq}	Output Voltage
s	Laplace Operator
m_p	Active Power Droop Controller
n_q	Reactive Power Droop Controller
K_{pv}	Proportional Gain for Voltage Controller
K_{iv}	Integral Gain for Voltage Controller
K_{pc}	Proportional Gain for Current Controller
K_{ic}	Integral Gain for Current Controller
L_c	Coupling Inductor
R_c	Coupling Resistor
L_f	Filter Inductor
R_f	Filter Resistor
C_f	Filter Capacitor
T_E	Electromagnetic Torque of Motor
T_L	Load Torque of Motor
J	Inertia of Motor
ρ	Number of Poles in a Motor
s_l	Rotor Slip
σ	Real part of Eigen Value
ζ	Damping Ratio
β	Attractiveness of Firefly
α	Mutation Co-efficient of Firefly
γ	Light Absorption Co-efficient of Firefly

Acknowledgment

All praise and gratitude be to Allah, the most beneficent, the most merciful for bestowing His inexhaustible kindness upon me and giving me the ability to explore, think and have patience in utilizing my knowledge for conducting this research work.

I would like to express my wholehearted gratitude, indebtedness and appreciation to my honorable teacher and respected supervisor, Dr. Ashik Ahmed, Associate Professor, Department of Electrical Electronic Engineering, Islamic University of Technology (IUT), Board Bazar, Gazipur, Bangladesh, for the patient guidance, encouragement and support throughout the period of my research work. I have been always well received by supervisor for all kinds of queries about my work and responded to my questions promptly.

I would also like to show deep gratitude to all of the respective teachers of the Department of Electrical Electronic Engineering, IUT, for their motivation, valuable suggestions and cooperation during this research work. Thanks to all my friends and colleagues for their support throughout the research work.

Finally, I am indebted to my beloved parents and dear wife without whom this work would have been almost impossible to carry out successfully. I express my deepest gratitude to them for their moral support and optimistic encouragement throughout the research work.

Abstract

Global optimization algorithms are becoming popular in a fast rate as they are able to solve different real life challenges. Swarm based nature inspired optimization techniques are developing in a fast pace due to its global acceptance and their capability in replicating various real life challenges and solving them efficiently with a better computation rate. Hybridization of these swarm based intelligent techniques with multi-objective based solution techniques is creating a wide door in the field of optimization as many real problems include multiple objectives which needs to be optimized. In this dissertation, two nature inspired hybrid optimizing algorithm is proposed incorporating swarm intelligence based firefly algorithm (FA) with multi objective based non-dominated sorting technique to form Non-dominated Sorting Firefly Algorithm (NSFA) and Non-dominated Sorting Whale Optimization Algorithm (NSWOA) where swarm based intelligence technique of Whale Optimization is hybridized with non-dominated sorting technique. This study also demonstrates their application in optimization of controller parameter of islanded microgrid . Moreover, how the optimized parameter affects the dynamic performance of microgrid during load variation is also demonstrated in this dissertation. The purpose of incorporating FA and WOA with non-dominated sorting technique to form NSFA and NSWOA respectively is to enhance global searching capability of conventional FA and Whale Optimization Algorithm (WOA) for complex optimization problems. Based on statistical tool, SPSS software, statistical analysis was carried out where the performance of the proposed hybrid NSWOA algorithm is compared with the performance of multi objective Non-dominated sorting genetic algorithm (NSGA) and Strength Pareto Evolutionary Algorithm (SPEA) in optimizing the controller parameters for the microgrid model used in this study. Similarly performance of NSFA is also compared with NSGA-II to analyze the ability of NSFA. This dissertation shows that NSWOA and NSFA is able to stabilize the system with faster computation rate with lesser number of iteration. Moreover, the system optimized using NSWOA and NSFA techniques provides better damping performance compared to NSGA-II. From this study it is obtained that NSWOA and NSFA requires 4.033 and 2.3667 iterations respectively to reach to best optimum solution which is significantly less than other existing algorithms. Moreover, the computational time required by NSWOA and NSFA is $2.9201\ sec.$ and $4.5459\ sec.$ respectively which proves that they reach to convergence at a much faster rate which is significantly less than existing NSGA-II and SPEA algorithms.

Chapter 1

Introduction and Background

Electrical energy and use of power grids for production of this electrical energy perhaps opened the door of new era in field of engineering. Although the modernization of these grids for uninterrupted and continuous supply of power to satisfy the electricity demand of the mass became a challenging issue for the engineers. Moreover, due to centralised generation of power there persists transmission loss and sudden blackouts and burnouts cause the total development of the certain area standstill [1]. In the world of increasing energy demand with increased blackouts, burnouts; microgrid came as a competent solution due to it being pollution free, energy saving and smart energy system with diversified energy sources [2].

1.1 Microgrid

Renewable energy sources are gaining more popularity now a day due to production of clean energy with no emission and requiring less dependency on limited fossil fuel resources. Micro-grids (MGs) are a competent solution for power system management, control and integration of renewables as Distributed Sources (DS) within utility grid [3]. Microgrid evolved as a solution to global energy crisis and at the same time it became a talking point in between the researchers for its advancement and development. [4, 5].

Microgrid also came as a solution to centralized power generation system problems like land, establishment and transmission loss constraints and helped in the development of the concept of distributed generation (DG) or localized generation. Microgrid is not an alternative to centralized generation rather it provides power to the load center of the locality with minimum transmission loss, better efficiency and cheaper rate [6,7].

1.2 Importance of Controller Parameter Optimization

MGs have two modes of operation- one is grid connected and the other is islanded. In islanded mode, basically there is one or more reliable energy source or energy storage compensating for generated power fluctuation of renewables [8], [9]. MG performance depends upon the selection of MG sources, controller performance and load scheduling.

Voltage source converters are used to interface a majority of distributed generation (DG) units to network. Optimal selection of controller parameters improves the system performance as well as helps in quality assurance of power developed [10]. Even if there exists any type of disturbance in the system, selection of proper controller parameters and tuning them at optimized value ensures stable operation of the system [11]. Optimizing the system controller parameters of microgrid with a view to enhancing the performance, efficiency and making it inexpensive became the prime concern for the researchers [12–14]. In order to run a microgrid smoothly and efficiently there must be some control over the operation, current, voltage and power supply of the inverters of a microgrid. So, it is inevitable that controller parameters play a vital role in proper operation of microgrid.

1.3 Importance of Multi-objective optimization

In single objective optimization, generally aim is to search for the best design or decision, which is usually the global solution of the optimization problem. But in the case of multiple objectives, there may exist one or more solutions which may be the best (global minimum or maximum) with respect to all objectives [15]. Generally, a set of solutions is obtained for this type of problem which may be better than the rest of solutions in the search space. Multi-objective and single-objective optimization techniques both have different methodology of ranking the generated population. Multi-objective optimization techniques have the advantage of selecting the best optimum global solution from a set of solutions arranged in a pareto-front using hyper volumetric measure which are ranked according to their fitness value and satisfying the constraints as well as finding optimal solution for the multiple objective functions [16]. On the contrary nature inspired single objective optimization techniques ranks the fitness function using bio-type ranking where a single optimal solution is considered as the global solution [17]. Most of our real challenges do not remain limited to only a single objective or single function rather there exist several self contrary objectives with numerous constraints which need to be solved simultaneously. So, it is inevitable

that multi-objective optimization techniques also provides the flexibility in demonstrating the real life challenges and provide the methodology in optimizing them and find the optimal solution of them.

1.4 Literature Review

Global optimization algorithms have gained lot of attention in last few years due to its ability to solve different real life challenges. Due to non-linear and multi-modality nature as well as presence of multi-local optima make it difficult to solve global optimization problems. Researchers are now moving towards stochastic algorithms which mostly include metaheuristics methods like Genetic Algorithm (GA) [18], Particle Swarm Optimization (PSO) [19], Evolutionary Programming (EP) [20], Ant Colony Optimization (ACO) [21], Modified Particle Swarm Optimization [22, 23] and Artificial Bee Colony (ABC) [24] which have shown great potential in solving complex global optimization problems. All these are swarm intelligence (SI) algorithms as they are based on the biological nature of breeding, food searching and other biotic processes of different creatures. Due to it's efficiency in dealing with non-linear multimodal global optimization problem, Firefly Algorithm (FA) is being used in different sectors for optimization.

Firefly Algorithm (FA) was first introduced in 2008 by Xin-She Yang where flashing lights produced by fireflies known as bio-luminescence along with general behavior of the tropical fireflies were taken into consideration for developing the algorithm [25], [26]. Firefly Algorithm (FA) has been used in different engineering applications like in - ref. [27] it was used to solve economic emissions of load dispatch problem, in ref. [28] to design digital controlled re-configurable switched beam concentric ring array antenna, in ref. [29] for minimization in permutation flow shop scheduling problems. In ref. [30] and [31] FA was used to minimize the computational time for digital image compression. It was seen from ref. [32] and ref. [33] that FA is capable of producing better global solution for linear antenna array design compared to Artificial Bee Colony (ABC) and PSO respectively. The pioneering FA algorithm works on the basis of light intensity variation and attraction capability of the fireflies which were further modified to improve the performance of the algorithm. Generation of random directions for determination of the best direction of fireflies with highest brightness and modification of attractiveness was initiated to observe it's effect on objective function [34]. Several Binary FA algorithms were also developed as a potential solution for different problems like in ref. [35] binary FA was developed to decode cipher to plain text and results of this algorithm outperformed Genetic Algo-

rithm (GA) in terms of efficiency. In order to speed up the convergence of solution of the algorithm, Gaussian distribution was introduced so that all fireflies move to the global best after each iteration and eventually this modified algorithm outperformed the classical firefly algorithm [36]. In ref. [37] combination of FA with chaotic maps was proposed where introduction of chaotic sequence helped in escaping local optima efficiently. In ref. [38] a Hybrid Evolutionary Firefly Algorithm (HEFA) was proposed where classical Firefly Algorithm was incorporated with the evolutionary process of the Differential Evolution (DE) algorithm for better searching efficiency.

Whale optimization algorithm (WOA) was first introduced by Mirjalali and Lewis in 2016 where they adopted the hunting behavior of whales [39]. For this they considered a specific breed of whale known as humpback whales as these types of whales use their unique bubble net feeding method for hunting their prey. In reference [40] a hybrid whale PSO algorithm (HWPSO) was developed combining WOA with PSO to address the limitation of PSO algorithm in exploration. Due to better exploration ability of WOA in different uncertain and small space region, many other researchers hybridized this WOA to overcome the time constraints and accuracy in obtaining global optimum solution of the problem. In reference [41] and [42] this ability of WOA was used with PSO and local search strategy respectively where [42] showed the ability of WOA in solving permutation flow shop scheduling problem. In reference [43] mean grey wolf optimizer (MGWO) [44] was combined with WOA to prevent local optima problems and obtain global optima at a faster rate. Similar approach was followed in [45–50] where WOA has been hybridized with chaotic local search in ref. [45], Simulated Annealing (SA) in ref. [46], Moth- Flame optimization in ref. [47], Grey-Wolf Optimizer in ref. [48], Sine-Cosine in ref. [49] and Brain Storm Optimization in ref. [50] to solve various real life problems and find the global optimum solution for those situation with a better accuracy, computation time and efficiency. Ref. [51] used WOA with Colliding Bodies Optimization (CBO) [52] forming WOA- CBO in building layout planning.

The above mentioned algorithms initially have been developed as a single objective based optimization algorithm with a view to optimizing any single objective function and all of these proposed algorithms efficiently handled single objective optimization problems. But in most cases the problems do not remain limited to only a single objective function rather there appears various multiple functions which may be conflicting with the other but all needs to be optimized. For multi objective optimization problems, Non-dominated Sorting Genetic Algorithm (NSGA) which was introduced by N. Srinivas and Kalyanmoy Deb in ref. [53] gained much appreciation due to it's abil-

ity to solve such problems with better search results and efficiency. But NSGA had few limitations i.e. complex computational strategy due to absence of crowding distance calculation; better solutions are lost due to huge number of solutions; and lack of sharing parameters being specified [54]. So to solve the lacking of NSGA a new improved algorithm namely NSGA-II has been introduced in [55] which overcomes the mentioned problems of NSGA. NSGA-II basically introduces crowding distance calculation after non-dominated sorting of population which reduces the number of solutions. It also adds a diversity factor amongst the solutions.

Voltage source converters are used to interface a majority of distributed generation (DG) units to network. Droop control is a well-established approach for autonomous microgrid operation. Due to the low-inertia nature of such converter-dominated systems, the stability of autonomous micro-grids is a critical issue. In ref. [56] and [11] small-signal-based stability analysis has been reported for studying the stability of autonomous droop-controlled microgrid system. Here, static (RL-type) loads are considered to simplify the modeling and analysis tasks addressing microgrid stability issues. In future power networks and recent advances in power converter ratings and topologies, medium-voltage (MV) multi-MW microgrid systems will be created with a wide pattern of both static and dynamic loads. The operation of dynamic induction motor loads in droop-controlled converter based microgrids yields special characteristics due to direct and relatively fast frequency control (e.g., droop control) using load power. Modeling of the small signal dynamic equations is very cumbersome because of complex control strategy for DGs. Ref. [57] characterizes small-signal stability of a hybrid AC/DC microgrid with static and dynamic loads using state-space and dynamic simulation models developed in MATLAB but it faced challenges in maintaining the voltage and frequency within permissible limits during the standalone mode. The impedance mismatch between inverter-interfaced distributed generation (IIDG) units and induction motor (IM) loads, addressed in [58] and [59], showed that the presence of IM load causes frequency instabilities in the system. Ref. [60] investigates the impact of load dynamics and load sharing among IIDG units on the stability and dynamic performance of islanded AC microgrids but lacks on description of controller requirement for damping out unstable, inter-area and low frequency oscillations introduced by dynamics of active and IM loads.

For maintaining required control performance and power quality during severe conditions, excessive trial-and-error-based repeated tuning process is required. The problem of tuning control parameters optimally based on control objectives has been mentioned in many researches. There are traditional methods for tuning PI controllers such

as Ziegler Nichols method, frequency domain methods considering gain and phase margins. Since, change of one PI control parameter tuning is likely to affect other parameter tuning, applying these methods to multiple DG controllers tuning is critical. Ref. [61] first presented control schemes for coordination of multiple microgrid generators, so that they can work on both grid-connected and autonomous modes. They used PSO algorithm for control parameter tuning during islanded mode operation. Ref. [62] proposed an effective control parameter-tuning method for multiple distributed generators in a microgrid using the PSO algorithm and gain-scheduling method which are found to be effective for weak power systems only. In ref. [63] PSO has been used to give optimal settings of the optimized control parameters in each mode. Moreover, they proposed eigenvalue-based objective functions for enhancing the damping characteristics and finally presented non-linear time domain based simulation for minimization of error in the measured power. In ref. [64] (GA) was introduced to optimize the control parameters to improve the dynamic performance of microgrid under load variation. Similarly in ref. [65] artificial fish swarm algorithm (AFSA) was used to optimize the gains of a PI controller to obtain better frequency output during islanded operation of microgrid. All the aforementioned work gained decent performance by optimizing the control parameters and treating the controller parameter tuning as a single objective optimization problem however, in ref. [66] a multi objective based genetic algorithm was used to optimize the control parameters for stable operation of microgrid but left much opportunity for improvement.

1.5 Motivation

The aforementioned literature review gives an acumen about the current progress in optimization of controller design using multi-objective optimization techniques for an islanded microgrid and also addresses the necessity of development of controller design methodologies for islanded microgrids. Most of the aforementioned works were limited to single objective optimization techniques. Single-objective optimization techniques limits the ranking of the fitness function into an individual function with highest fitness value. It considers this individual function with highest fitness value as the optimum global solution. On the contrary in multi-objective optimization techniques a number of objective functions with numeral constraints exist which are needed to be satisfied. Thus, this creates a chance for exploring multi-objective optimization techniques.

Microgrid developed as a blessing in implementation of small scale power grid in localized area but stable operation of this microgrid with stable dynamic performance

is a challenge to modern day engineers. For proper stable operation of a microgrid and maintain the quality of power supply it is essential to analyze the small scale model of a microgrid and observe its dynamic performance [64]. One of the characteristics of microgrid is its low physical intertia which result the system to be more oscillatory compared to traditional systems due to faster dynamic response of microgrid [67, 68]. Development of dynamic model of a microgrid is a difficult issue due to the complexity of the equations. In day to day life usage both static and dynamic type loads are included so, this gives an opportunity to develop a microgrid model consisting of both static and dynamic load and analyze the dynamic performance of the system.

The performance as well as the quality of power of the microgrid along with the proper distribution of power to the load centre depends on the proper operation and selection of the control parameters [66]. In the aforementioned literature review it is seen that several researchers worked in order to obtain optimal controller parameters and obtain steady state performance of microgrid. But most of them were either single objective based optimized parameters or some were trial and error based parameters. This enables the vision of optimizing the controller parameters of a microgrid with both static and dynamic load using the developed multi-objective based optimization techniques.

1.6 Thesis Objectives

The aim of this thesis is addressing the limitations of the aforementioned works and develop a nature-inspired meta-heuristic multi-objective based optimization algorithms for controller design of an islanded AC microgrid. The following objectives need to be satisfied in order to reach the thesis goal:

- To obtain a linearized dynamic model of a microgrid system having both static and nonlinear dynamic loads.
- To identify the controller parameters that play a vital role in the microgrid stability, oscillation and damping under load variation.
- To propose hybrid optimizing algorithms for finding the optimized controller parameters to improve the performance of the system for both static and dynamic load variation.
- To perform comparative study among the proposed and available optimizing algorithms for obtaining the best outcome.

1.7 Thesis Outline

The total thesis has been arranged in different chapters where each chapter describes different portion of the work.

Chapter 1 discusses the background history of different optimization algorithms and their contribution in optimization of the controller parameters of an islanded AC microgrid. A brief description about the importance of controller parameter optimization and use of multi-objective optimization has been also included. In the later part of the chapter motivation of this work and thesis objectives were also mentioned.

Chapter 2 explains the linearized mathematical modelling of the microgrid. In this chapter detailed illustration of the mathematical model of the microgrid consisting of static load and induction motor as dynamic load has been given.

Chapter 3 In this chapter the problem statement along with the objective functions has been mentioned. A root locus analysis was shown as well.

Chapter 4 illustrates the methodology of the proposed nature inspired algorithm i.e. Non-dominated Sorting Whale Optimization Algorithm (NSWOA). Detailed flow chart along with step-wise methodology has been included in this chapter. In this chapter another proposed nature inspired algorithm i.e. Non-dominated Sorting Fire-fly Algorithm (NSFA) has been also described with detailed flow chart and step-wise procedure.

Chapter 5 shows the performance analysis, results and discussion of the proposed NSWOA technique. In this chapter eigen value analysis, time domain analysis and statistical analysis of the proposed algorithms are shown with proper graphical representations. Moreover this chapter also shows comparative analysis of NSWOA with other prominent multi-objective algorithms.

Chapter 6 illustrates the detailed performance analysis, results and discussion of the proposed NSFA technique. A comparative study of NSFA has been carried out with well known multi-objective algorithm.

Chapter 7 lastly discusses the concluding part of the total work as well as gives an insight about the future scope of this work.

Chapter 2

Mathematical Modelling of an AC Microgrid with Dynamic Load

An islanded microgrid consisting of both static and dynamic load has been used for this study. The Microgrid model shown in Figure 2.1 consists of two DG units where static load has been installed on one unit and induction motor as dynamic load on the other unit. In this section the dynamic model of the aforementioned islanded microgrid will be discussed where necessary static and dynamic load equations, line equations and controller equations will be developed for completion of a total state-space model of the studied microgrid. Necessary study was done from [11, 59] for development of the equations for the complete microgrid model.

Modeling of microgrid is basically comprised of three main segments: inverter, loads and network. Inverters were designed in their individual $(d_{ij} - q_{ij})$ reference frame and power sharing controller was used to fix their angular frequency ω_{ij}. In this study, reference frame of one of the inverters was taken as the common reference frame $(D - Q)$ and the load and network state equations were transformed into that $(D-Q)$ frame including the other inverters. $D-Q$ transformation technique was used for this purpose as shown in Eqs. (2.1) and (2.2) [69]. Here D-Q axes was taken as common reference frame where $(d-q)_i$ and $(d-q)_j$ were taken as the the reference frame of i^{th} and j^{th} inverters respectively and angle δ_i is taken as the reference angle for i^{th} inverter as shown in Figure 2.2. $D - Q$ axes rotates at a frequency ω_{com} which is taken as the common reference frame frequency whereas $(d-q)_i$ and $(d-q)_j$ axes rotates with frequency ω_i and ω_j respectively. Now angle δ denotes the difference of angle between individual reference frame and common reference frame which can be calculated following Eq. (2.3). There is exist an angular frequency difference between common $D-Q$ reference frame and individual $d-q$ reference frame as the components in common $D - Q$ reference frame rotates at a synchronous speed where as the speed is not synchronous for the components in their individual $(d - q)$ reference frame.

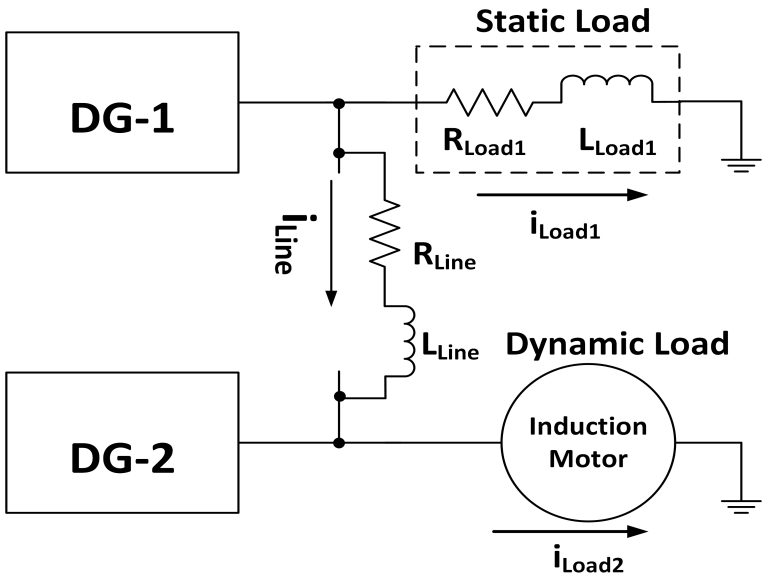


Figure 2.1: DGs with both static and dynamic load

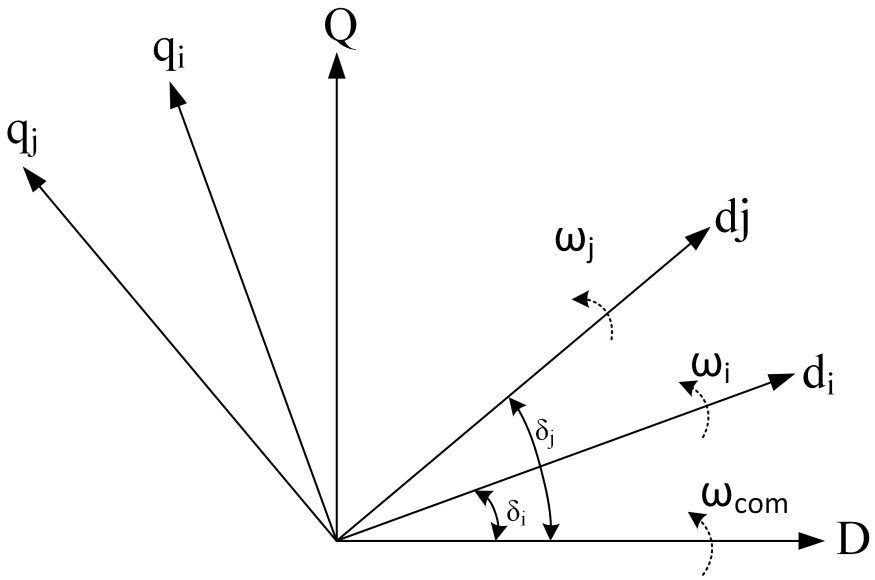


Figure 2.2: Transformation of Reference Frame

$$[f_{DQ}] = [T_i][f_{dq}] \tag{2.1}$$

$$[T_i] = \begin{bmatrix} \cos(\delta_i) & -\sin(\delta_i) \\ \sin(\delta_i) & \cos(\delta_i) \end{bmatrix} \tag{2.2}$$

$$\delta = \int (\omega - \omega_{com})dt \tag{2.3}$$

A DG inverter block diagram has been shown in Figure 2.3 which is connected to the microgrid. There are three controller units of which power controller has been

designed for determining the magnitude and frequency of the inverter output voltage (v_{od}, v_{oq}) following droop characteristics [70]. For obtaining characteristics of the output voltage, voltage control unit has been used and current control unit is used for controlling the LC filter current and damp out the high frequency disturbances [71, 72]. The state space model for each of these controllers is represented in this section.

2.1 Power Controller

A power controller successive block diagram has been depicted in Figure 2.4 which shows power calculation, filtration stage and droop control strategy in a power controller. At first the instantaneous active power, $p(t)$, and instantaneous reactive power, $q(t)$, is calculated from output voltage, (v_{odq}) and output current, (i_{odq}) as shown in Eq. (2.4). The average active power, P, as well as average reactive power, Q, is obtained by passing the instantaneous values of active, $p(t)$ and reactive, $q(t)$ power through a low pass filter as depicted in Eq. (2.5). The droop controller in power controller loop generates the output reference voltage magnitude (v_o) and frequency (ω) of the DG units depending upon their active and reactive power values and the controller also allows them to share their active and reactive power demand.

$$p(t) = 1.5(v_{od}i_{od} + v_{oq}i_{oq}),\ q(t) = 1.5(v_{od}i_{oq} - v_{oq}i_{od}) \tag{2.4}$$

$$P = \frac{\omega_c}{s + \omega_c}p(t),\ Q = \frac{\omega_c}{s + \omega_c}q(t) \tag{2.5}$$

Here, ω_c is the cut off frequency of the low pass filter, s is the Laplace operator and v_{od}, v_{oq} are the output voltage and i_{od}, i_{oq} are the output current in d-q reference frame. Now the power sharing between active and reactive power due to droop controller is depicted in Eqs. (2.6) and (2.7) where m_p and n_q are static active and reactive power droop gains respectively.

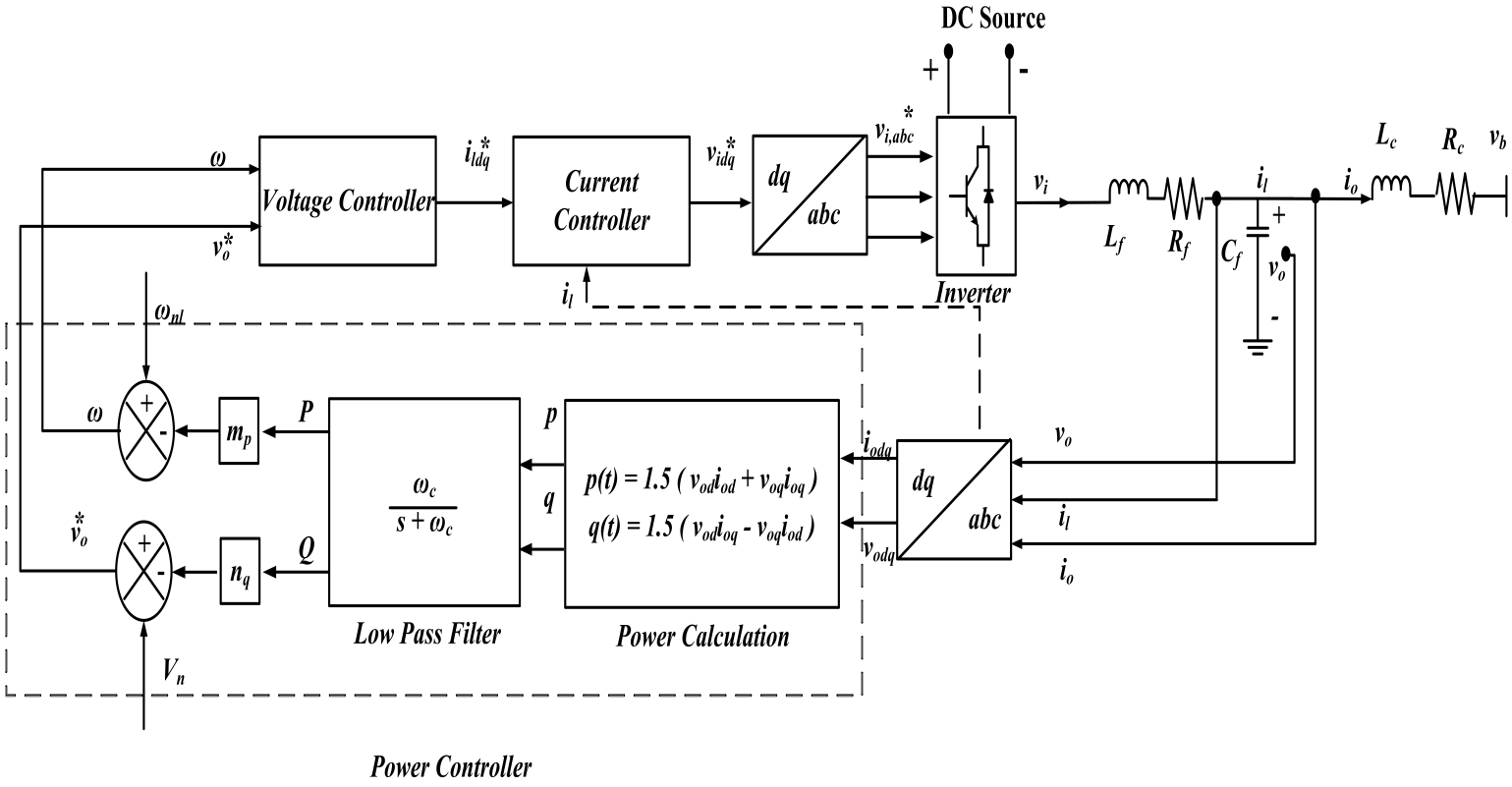


Figure 2.3: Block Diagram of the DG Inverter

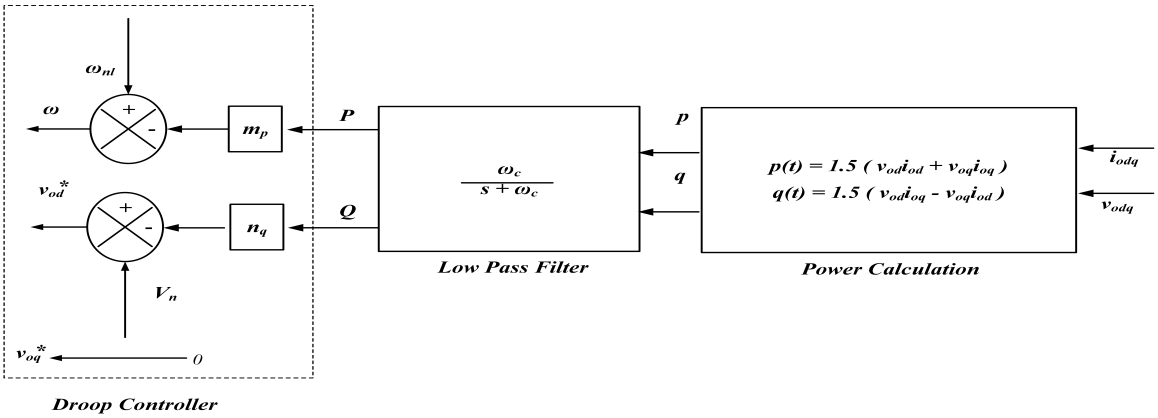


Figure 2.4: Power Controller of the Microgrid

$$\omega = \omega_{nl} - m_p P \tag{2.6}$$

$$v_{od}^* = V_n - n_q Q,\ v_{oq}^* = 0 \tag{2.7}$$

Here, ω represents the output angular frequency reference of the inverter, ω_{nl} is the no load frequency set point, V_n is the output voltage nominal set point and superscript '*' represents a reference value. The output voltage magnitude reference is the direct $d-axis$ component of the inverter reference frame as the quadrature $q-axis$ component is made zero. This strategy is followed to assist and expedite voltage attuned control system. The static droop control gains m_p and n_q are calculated for a certain bandwidth of frequency and magnitude of voltage for real and reactive power respectively using the following equations as shown in Eq.(2.8). The values of ω_{max}, ω_{min}, V_{odmin} and V_{odmax} can be selected based on the system ratings.

$$m_p = \frac{\omega_{max} - \omega_{min}}{P_{max}},\ n_q = \frac{V_{odmax} - V_{odmin}}{Q_{max}} \tag{2.8}$$

2.2 Voltage Controller

A block diagram of output voltage controller consisting of the controller gains and feedback is shown in Figure 2.5. A standard PI regulator has been used here for controlling the output voltage. The main objective of voltage controller is to compare the voltage (v_{od}^*, v_{oq}^*) reference value with the actual value of voltage (v_{od}, v_{oq}) and then using the PI regulator generate inductor output currents (i_{ld}^*, i_{lq}^*). The corresponding voltage controller equations are given below in Eqs.(2.9)-(2.12).

$$\dot{\varphi}_d = v_{od}^* - v_{od} \tag{2.9}$$

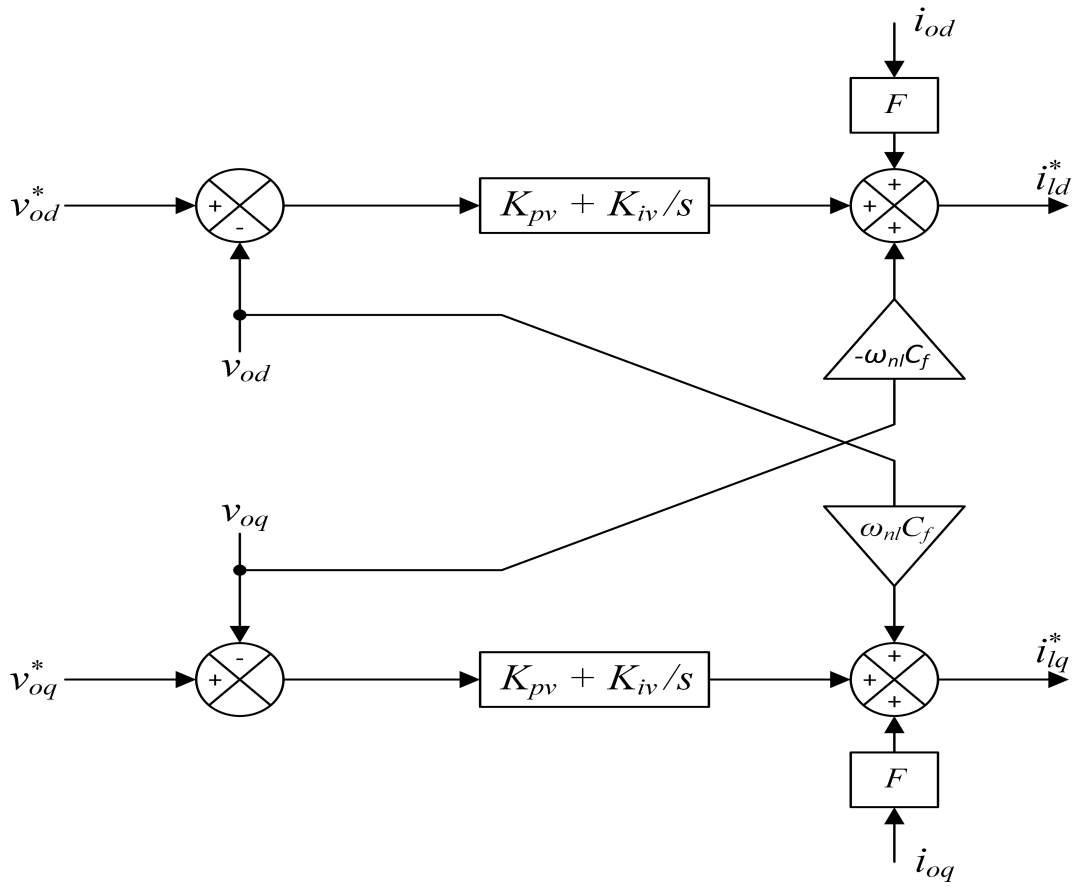


Figure 2.5: Voltage Controller of the Microgrid

$$\dot{\varphi_q} = v^*_{oq} - v_{oq} \tag{2.10}$$

$$i^*_{ld} = K_{pv}\dot{\varphi}_d + K_{iv}(\int \dot{\varphi}_d dt) - \omega_{nl}C_f v_{oq} + F i_{od} \tag{2.11}$$

$$i^*_{lq} = K_{pv}\dot{\varphi}_q + K_{iv}(\int \dot{\varphi}_q dt) + \omega_{nl}C_f v_{od} + F i_{oq} \tag{2.12}$$

Here, K_{pv} and K_{iv} are proportional and integral gains respectively for controller of d-q axis ; φ_d, φ_q are the voltage controller's integrator states and F is the feed forward gain which is used to minimize the disturbance effect of output grid current components i_{od} and i_{oq}.

2.3 Current Controller

The current controller shown in Figure 2.6 basically compares the measured filter currents with the reference inductor current (i^*_{ldq}) . It controls the voltage across the inductor in such a way that the current error becomes minimum. Standard PI regulators helps in controlling the output inductor current. The corresponding current controller dynamic equations are given in Eqs. (2.13) - (2.15).Here, γ_d and γ_q denotes state of

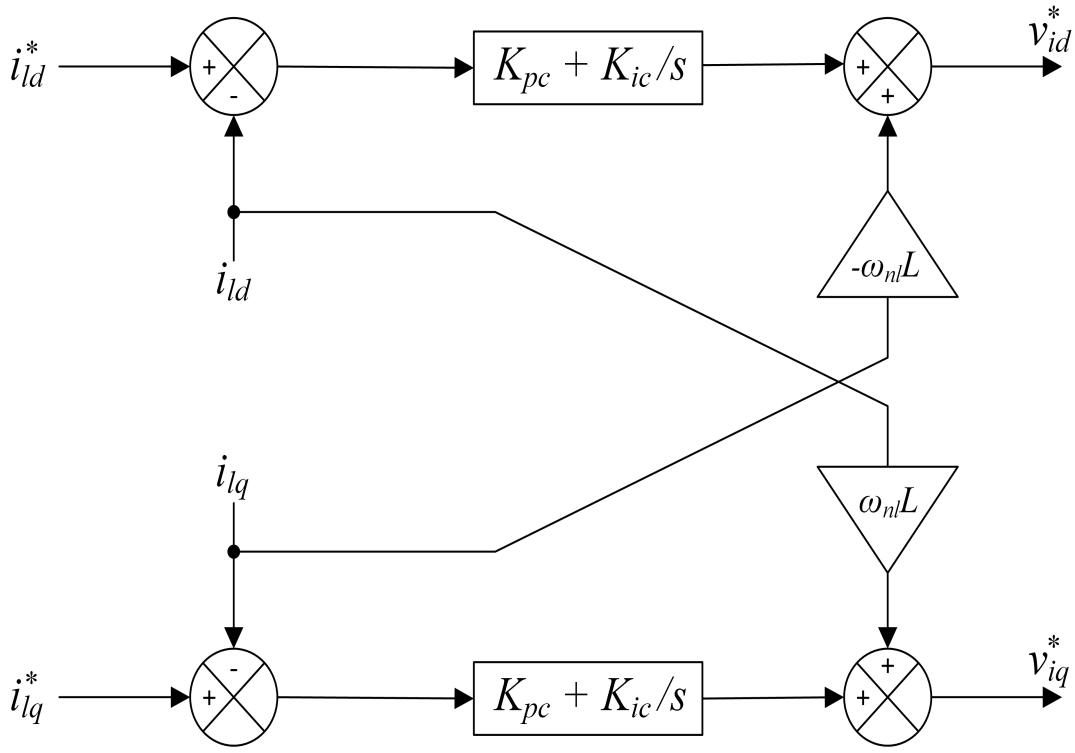


Figure 2.6: Current Controller of the Microgrid

current controller. K_{pc} and K_{ic} are proportional and integral gains of current controller respectively. v_{id}^* and v_{iq}^* are the reference values of firing voltage for $d-axis$ and $q-axis$ respectively which is used for firing the gate pulse of inverter after $d-q$ to abc transformation. Here, L_f is the inductance of the filter.

$$\dot{\gamma_d} = i_{ld}^* - i_{ld},\ \dot{\gamma_q} = i_{lq}^* - i_{lq} \tag{2.13}$$

$$v_{id}^* = K_{ic}\gamma_d + K_{pc}(i_{id}^* - i_{id}) - \omega_n L_f i_{lq} \tag{2.14}$$

$$v_{iq}^* = K_{ic}\gamma_q + K_{pc}(i_{iq}^* - i_{iq}) + \omega_n L_f i_{ld} \tag{2.15}$$

2.4 LCL Filter

The LCL filter in this work mainly includes the inductor in filter (L_f), capacitor in filter (C_f) along with the coupling inductor (L_c) and coupling resistance R_c of the transformer. R_f is a series resistance along with the LCL filter and v_{bd}, v_{bq} are the $d-q$ frame bus voltages on the grid side. Figure 2.7 depicts the circuit configuration of a LCL filter. Here it is assumed that voltage on input and output side of the inverter is equal i.e. $v_i = v_i^*$ considering zero loss in the diode and transistor of the inverter [11] [59]. The LCL filter dynamics states are shown in the following equations.

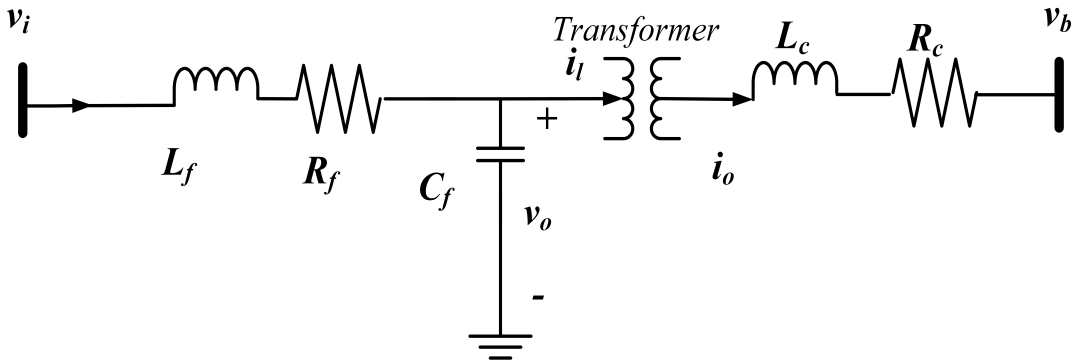


Figure 2.7: LCL filter of the Microgrid

$$\dot{i}_{ld} = \omega i_{lq} - \frac{1}{L_f}(v_{od} - v_{id} + R_f i_{ld}) \tag{2.16}$$

$$\dot{i}_{lq} = -\omega i_{ld} - \frac{1}{L_f}(v_{oq} - v_{iq} + R_f i_{lq}) \tag{2.17}$$

$$\dot{i}_{od} = \omega i_{oq} - \frac{1}{L_c}(v_{bd} - v_{od} + R_c i_{od}) \tag{2.18}$$

$$\dot{i}_{oq} = -\omega i_{od} - \frac{1}{L_c}(v_{bq} - v_{oq} + R_c i_{oq}) \tag{2.19}$$

$$\dot{v}_{od} = \omega v_{oq} + \frac{1}{C_f}(i_{ld} - i_{od}) \tag{2.20}$$

$$\dot{v}_{oq} = -\omega v_{od} + \frac{1}{C_f}(i_{lq} - i_{oq}) \tag{2.21}$$

2.5 Linearized Complete Inverter Model

There are two DG units along with an induction motor as dynamic load in the microgrid model. The model developed in the previous sections is a non-linear model which needs to be linearized for developing the linearized dynamic equation of the system as linearized system benefits in ease of study of small signal stability. The inverter model has been linearized using $Taylor's\ series\ expansion$. Now after linearizing the Eqs. (2.3) - (2.7) , linearized power controller model is shown in Eqs. (2.22)- (2.25).

$$\Delta\dot{\delta} = \Delta\omega \tag{2.22}$$

$$\Delta\dot{P} = 1.5(\omega_c \Delta v_{od} I_{od} + \omega_c V_{od} \Delta i_{od} + \omega_c \Delta v_{oq} I_{oq} + \omega_c V_{oq} \Delta i_{oq}) - \omega_c \Delta P \tag{2.23}$$

$$\Delta\dot{Q} = 1.5(\omega_c \Delta v_{od} I_{oq} + \omega_c V_{od} \Delta i_{oq} - \omega_c \Delta v_{oq} I_{od} - \omega_c V_{oq} \Delta i_{od}) - \omega_c \Delta Q \tag{2.24}$$

$$\Delta\omega = -m_p \Delta P \tag{2.25}$$

Linearizing Eqs. (2.9) - (2.12) of the voltage controller, the linearized equations of the controller are represented below in Eqs. (2.26) - (2.29)

$$\Delta\dot{\varphi}_d = -n_q\Delta Q - \Delta V_{od} \tag{2.26}$$

$$\Delta\dot{\varphi}_q = -\Delta V_{oq} \tag{2.27}$$

$$\Delta i_{ld}^* = F\Delta i_{od} - \omega_{nl}C_f\Delta v_{oq} + K_{pv}\Delta\dot{\varphi}_d + K_{iv}\Delta\varphi_d \tag{2.28}$$

$$\Delta i_{lq}^* = F\Delta i_{oq} + \omega_{nl}C_f\Delta v_{od} + K_{pv}\Delta\dot{\varphi}_q + K_{iv}\Delta\varphi_q \tag{2.29}$$

Current controller used in the inverter model is linearized by linearizing Eqs. (2.13) - (2.15). The linearized equations are shown in Eqs. (2.30) - (2.33)

$$\Delta\dot{\gamma}_d = F\Delta i_{od} - \omega_{nl}C_f\Delta v_{oq} - K_{pv}n_q\Delta Q - K_{pv}\Delta v_{od} + K_{iv}\Delta\varphi_d - \Delta i_{ld} \tag{2.30}$$

$$\Delta\dot{\gamma}_q = F\Delta i_{oq} + \omega_{nl}C_f\Delta v_{od} - K_{pv}\Delta v_{oq} + K_{iv}\Delta\varphi_q - \Delta i_{lq} \tag{2.31}$$

$$\Delta v_{id}^* = K_{ic}\Delta\gamma_d + K_{pc}\Delta\dot{\gamma}_d - \omega_{nl}L_f\Delta i_{lq} \tag{2.32}$$

$$\Delta v_{iq}^* = K_{ic}\Delta\gamma_q + K_{pc}\Delta\dot{\gamma}_q + \omega_{nl}L_f\Delta i_{ld} \tag{2.33}$$

Similarly, linearizing Eqs. (2.16) - (2.21) following linearized equations of LCL filter are obtained.

$$\Delta\dot{i}_{ld} = I_{lq}\Delta\omega + \omega\Delta i_{lq} - \frac{1}{L_f}(\Delta v_{od} - \Delta v_{id} + R_f\Delta i_{ld}) \tag{2.34}$$

$$\Delta\dot{i}_{lq} = -I_{ld}\Delta\omega - \omega\Delta i_{ld} - \frac{1}{L_f}(\Delta v_{oq} - \Delta v_{iq} + R_f\Delta i_{lq}) \tag{2.35}$$

$$\Delta\dot{i}_{od} = I_{oq}\Delta\omega + \omega\Delta i_{oq} - \frac{1}{L_c}(\Delta v_{bd} - \Delta v_{od} + R_c\Delta i_{od}) \tag{2.36}$$

$$\Delta\dot{i}_{oq} = -I_{od}\Delta\omega - \omega\Delta i_{od} - \frac{1}{L_c}(\Delta v_{bq} - \Delta v_{oq} + R_c\Delta i_{oq}) \tag{2.37}$$

$$\Delta\dot{v}_{od} = V_{oq}\Delta\omega + \omega\Delta v_{oq} + \frac{1}{C_f}(\Delta i_{ld} - \Delta i_{od}) \tag{2.38}$$

$$\Delta\dot{v}_{oq} = -V_{od}\Delta\omega - \omega\Delta v_{od} + \frac{1}{C_f}(\Delta i_{lq} - \Delta i_{oq}) \tag{2.39}$$

Here in Eqs. (2.36) and (2.37), v_{bd} and v_{bq} are the bus voltages which are input to the subsystem. As v_{bdq} is not a state variable so using Eqs. (2.1) and (2.2) v_{bdq} can be transferred to global $D-Q$ reference frame. Equations (2.40) and (2.41) represent v_{bD} and v_{bQ} respectively where i_{loadDQ} is load current and i_{lineDQ} is line current in $D-Q$

frame.

$$v_{bD} = R_n(i_{oD} + i_{LineD} - i_{LoadD}) \tag{2.40}$$

$$v_{bQ} = R_n(i_{oQ} + i_{LineQ} - i_{LoadQ}) \tag{2.41}$$

Equations (2.40) and (2.41) needs to be represented with respect to their state variable as i_{oDQ} is not a state variable. Now, i_{oDQ} is transformed to i_{odq} using Eqs. (2.1) and (2.2).

For completion of the whole microgrid state space model each of the states were transferred to a common global reference frame using Eqs. (2.1) and (2.2). The complete linearized inverter state space model is given in the Eqs. (2.42) - (2.44).

$$\Delta\dot{x}_{INVi} = A_{INVi}\Delta x_{INVi} + B_{INVi}\Delta v_{bDQi} + B_{com}\Delta\omega_{com} \tag{2.42}$$

$$\Delta i_{oDQi} = C_{INVi}\Delta x_{INVi} \tag{2.43}$$

$$\Delta x_{INVi} = [\Delta\delta_i\ \Delta P_i\ \Delta Q_i\ \Delta\varphi_d\ \Delta\varphi_q\ \Delta\gamma_d\ \Delta\gamma_q\ \Delta i_{ld}\ \Delta i_{lq}\ \Delta v_{od}\ \Delta v_{oq}\ \Delta i_{od}\ \Delta i_{oq}] \tag{2.44}$$

An individual inverter model basically consist of thirteen states here. The components of A_{INVi}, B_{INVi}, B_{com} and C_{INVi} are given in matrix form in Eqs. (2.45) - (2.47)

$$A_{INVi} = \begin{bmatrix}
0 & -m_p & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -\omega_c & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.5\omega_c I_{odi} & 1.5\omega_c I_{oqi} & 1.5\omega_c V_{odi} & 1.5\omega_c V_{oqi} \\
0 & 0 & -\omega_c & 0 & 0 & 0 & 0 & 0 & 0 & 1.5\omega_c I_{oqi} & -1.5\omega_c I_{odi} & -1.5\omega_c V_{oqi} & 1.5\omega_c V_{odi} \\
0 & 0 & -n_q & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & -n_q K_{pv} & K_{ic} & 0 & 0 & 0 & -1 & 0 & -K_{pv} & -\omega_{nl} & F & 0 \\
0 & 0 & 0 & 0 & K_{ic} & 0 & 0 & 0 & -1 & \omega_{nl} & -K_{pv} & 0 & F \\
0 & -m_p I_{lqi} & \frac{-n_q K_{pc} K_{pv}}{L_f} & \frac{K_{iv} K_{pc}}{L_f} & 0 & \frac{K_{ic}}{L_f} & 0 & \frac{-K_{pc}-R_f}{L_f} & \omega-\omega_{nl} & \frac{-K_{pc}K_{pv}-1}{L_f} & \frac{-\omega_{nl}C_f K_{pc}}{L_f} & \frac{K_{pc}F}{L_f} & 0 \\
0 & m_p I_{ldi} & 0 & 0 & \frac{K_{iv}K_{pc}}{L_f} & 0 & \frac{K_{ic}}{L_f} & \omega-\omega_{nl} & \frac{-K_{pc}-R_f}{L_f} & \frac{\omega_{nl}C_f K_{pc}}{L_f} & \frac{-K_{pc}K_{pv}-1}{L_f} & 0 & \frac{K_{pc}F}{L_f} \\
0 & -m_p V_{oqi} & 0 & 0 & 0 & 0 & 0 & \frac{1}{C_f} & 0 & 0 & \omega_o & \frac{-1}{C_f} & 0 \\
0 & -m_p V_{odi} & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{C_f} & \omega & 0 & 0 & \frac{-1}{C_f} \\
\frac{1}{L_c}\Big(V_{bDi}\sin\delta_i & & & & & & & & & & & & \\
-V_{bQi}\cos\delta_i\Big) & -m_p I_{oqi} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{L_c} & 0 & \frac{-R_c}{L_c} & \omega_o \\
\frac{1}{L_c}\Big(V_{bDi}\cos\delta_i & & & & & & & & & & & & \\
+V_{bQi}\sin\delta_i\Big) & m_p I_{odi} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{L_c} & -\omega_o & \frac{-R_c}{L_c}
\end{bmatrix} \tag{2.45}$$

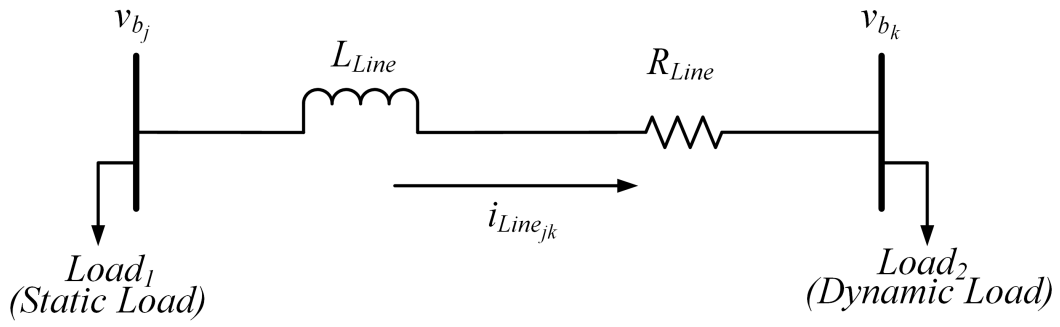


Figure 2.8: Line Network of the Microgrid

$$B_{INVi} = \begin{bmatrix} 0 & .... & 0 & \frac{-\cos\delta_i}{L_c} & \frac{-\sin\delta_i}{L_c} \\ 0 & .... & 0 & \frac{\sin\delta_i}{L_c} & \frac{-\cos\delta_i}{L_c} \end{bmatrix}^T_{13\times 2} \quad B_{com} = \begin{bmatrix} -1 & 0 & 0 & 0 & ..... & 0 \end{bmatrix}^T_{13\times 1} \tag{2.46}$$

$$C_{INVi} = \begin{bmatrix} -I_{odi}\sin\delta_i - I_{oqi}\cos\delta_i & 0 & 0 & .... & 0 & 0 & \cos\delta_i & -\sin\delta_i \\ I_{odi}\cos\delta_i - I_{oqi}\sin\delta_i & 0 & 0 & .... & 0 & 0 & \sin\delta_i & \cos\delta_i \end{bmatrix}_{2\times 13} \tag{2.47}$$

2.6 Line Network and Static Load Model

Line current flows from one node to other node of the line connecting the buses. As the system has one single line network so if two nodes of a line is j and k then the state equations of line network of the microgrid can be shown in Eqs. (2.48) and (2.49). Now after linearizing Eqs. (2.48) and (2.49), the obtained linearized state equations are given in Eqs. (2.50) and (2.51).

$$\dot{i}_{LineD} = \frac{v_{bD_j}}{L_{Line}} - \frac{v_{bD_k}}{L_{Line}} - \frac{R_{Line}}{L_{Line}} i_{LineD} + \omega i_{LineQ} \tag{2.48}$$

$$\dot{i}_{LineQ} = \frac{v_{bQ_j}}{L_{Line}} - \frac{v_{bQ_k}}{L_{Line}} - \frac{R_{Line}}{L_{Line}} i_{LineQ} - \omega i_{LineD} \tag{2.49}$$

$$\Delta\dot{i}_{LineD} = \frac{1}{L_{Line}}(\Delta v_{bD_j} - \Delta v_{bD_k}) - \frac{R_{Line}}{L_{Line}}\Delta i_{LineD} + \omega\Delta i_{LineQ} + I_{LineQ}\Delta\omega \tag{2.50}$$

$$\Delta\dot{i}_{LineQ} = \frac{1}{L_{Line}}(\Delta v_{bQ_j} - \Delta v_{bQ_k}) - \frac{R_{Line}}{L_{Line}}\Delta i_{LineQ} - \omega\Delta i_{LineD} - I_{LineD}\Delta\omega \tag{2.51}$$

Equation (2.52) shows the state space representation of the line network. A network model for two bus system is shown in Figure 2.8 where on one bus static load is connected and on other bus dynamic load is connected.

$$[\Delta\dot{i}_{LineDQ}] = A_{Line}[\Delta i_{LineDQ}] + B_{1Line}[\Delta v_{bDQ}] + B_{2Line}[\Delta\omega] \tag{2.52}$$

The load model of the microgrid is formulated in Eqs. (2.53) - (2.57) in a similar way like the line network. All state equations formulated for line network and load model are considered on the common global $D-Q$ reference frame.

$$\dot{i}_{LoadD} = \frac{v_{bD}}{L_{Load}} - \frac{R_{Load}}{L_{Load}} i_{LoadD} + \omega i_{LoadQ} \tag{2.53}$$

$$\dot{i}_{LoadQ} = \frac{v_{bQ}}{L_{Load}} - \frac{R_{Load}}{L_{Load}} i_{LoadQ} - \omega i_{LoadD} \tag{2.54}$$

$$\Delta\dot{i}_{LoadD} = \frac{1}{L_{Load}}\Delta v_{bD} - \frac{R_{Load}}{L_{Load}}\Delta i_{LoadD} + \omega\Delta i_{LoadQ} + I_{LoadQ}\Delta\omega \tag{2.55}$$

$$\Delta\dot{i}_{LoadQ} = \frac{1}{L_{Load}}\Delta v_{bQ} - \frac{R_{Load}}{L_{Load}}\Delta i_{LoadQ} - \omega\Delta i_{LoadD} - I_{LoadD}\Delta\omega \tag{2.56}$$

$$[\Delta\dot{i}_{LoadDQ}] = A_{Load}[\Delta i_{LoadDQ}] + B_{1Load}[\Delta v_{bDQ}] + B_{2Load}[\Delta\omega] \tag{2.57}$$

Now the components of A_{Line}, B_{1Line}, B_{2Line}, A_{Load}, B_{1Load} and B_{2Load} are given below in Eqs. (2.58) and(2.59).

$$A_{Line} = \begin{bmatrix} \frac{-R_{Line}}{L_{Line}} & \omega_o \\ -\omega_o & \frac{-R_{Line}}{L_{Line}} \end{bmatrix},\ B_{1Line} = \begin{bmatrix} \frac{1}{L_{Line}} & 0 & \frac{-1}{L_{Line}} & 0 \\ 0 & \frac{1}{L_{Line}} & 0 & \frac{-1}{L_{Line}} \end{bmatrix},\ B_{2Line} = \begin{bmatrix} I_{LineQ} \\ -I_{LineD} \end{bmatrix} \tag{2.58}$$

$$A_{Load} = \begin{bmatrix} \frac{-R_{Load}}{L_{Load}} & \omega_o \\ -\omega_o & \frac{-R_{Load}}{L_{Load}} \end{bmatrix},\ B_{1Load} = \begin{bmatrix} \frac{1}{L_{Load}} & 0 & \frac{-1}{L_{Load}} & 0 \\ 0 & \frac{1}{L_{Load}} & 0 & \frac{-1}{L_{Load}} \end{bmatrix},\ B_{2Load} = \begin{bmatrix} I_{LoadQ} \\ -I_{LoadD} \end{bmatrix} \tag{2.59}$$

2.7 State Space Model of Induction Motor as Dynamic Load

In our study an induction motor has been used as a dynamic load. The stator and rotor voltages of an induction motor developed in common reference frame is given in Eqs. (2.60) - (2.63) following ref. [31] where v_{DQs} and v_{DQr} are stator and rotor side voltage respectively; i_{DQs} and i_{DQr} are stator and rotor current respectively; L_s, R_s are the stator inductance and resistance respectively; L_r, R_r are the rotor inductance and resistance respectively; L_m is the mutual inductance; ω is stator angular frequency and s_l is the rotor slip.

$$v_{Ds} = R_s i_{Ds} + L_s\frac{d}{dt}i_{Ds} + L_m\frac{d}{dt}i_{Dr} - \omega L_s i_{Qs} - \omega L_m i_{Qr} \tag{2.60}$$

$$v_{Qs} = R_s i_{Qs} + L_s\frac{d}{dt}i_{Qs} + L_m\frac{d}{dt}i_{Qr} + \omega L_s i_{Ds} + \omega L_m i_{Dr} \tag{2.61}$$

$$v_{Dr} = R_r i_{Dr} + L_r \frac{d}{dt} i_{Dr} + L_m \frac{d}{dt} i_{Ds} - \omega s_l L_m i_{Qs} - \omega s_l L_r i_{Qr} \tag{2.62}$$

$$v_{Qr} = R_r i_{Qr} + L_r \frac{d}{dt} i_{Qr} + L_m \frac{d}{dt} i_{Qs} + \omega s_l L_m i_{Ds} + \omega s_l L_r i_{Dr} \tag{2.63}$$

The relationship between mechanical speed of motor and electro-mechanical torque developed by the motor is shown in Eq. (2.64) and (2.65) respectively where T_E is electromagnetic torque, T_L is load torque, J is the combined inertia of the rotating masses of motor and load and ρ is the number of poles of the motor.

$$T_E = \frac{3\rho L_m}{4}(i_{Qs} i_{Dr} - i_{Ds} i_{Qr}) \tag{2.64}$$

$$T_E - T_L = J \frac{d}{dt}((1 - s_l)\omega) \tag{2.65}$$

Now after linearizing the motor equations using $Taylor's\ series\ expansion$ and rearranging them, the state space model of induction motor is given in Eq. (2.66) where the components of ΔX_{IM}, ΔU_{IM}, A_{IM}, B_{IM}, C_{IM} and D_{IM} are given below in Eqs. (2.68) - (2.71). Here $\Delta\omega$ can be transferred and expressed according to the states of microgrid using equation (2.25). $\Delta\dot{\omega}$ is expressed according to Eq. (2.67)

$$\Delta \dot{X}_{IM} = (-A_{IM}^{-1} B_{IM})\Delta X_{IM} + A_{IM}^{-1}\Delta U_{IM} + (-A_{IM}^{-1} C_{IM})\Delta\omega + (-A_{IM}^{-1} D_{IM})\Delta\dot{\omega} \tag{2.66}$$

$$\Delta\dot{\omega} = -m_p \Delta\dot{P} \tag{2.67}$$

$$\Delta X_{IM} = \begin{bmatrix} \Delta i_{Qs} \\ \Delta i_{Ds} \\ \Delta i_{Qr} \\ \Delta i_{Dr} \\ \Delta s_l \end{bmatrix},\ \Delta U_{IM} = \begin{bmatrix} \Delta v_{Qs} \\ \Delta v_{Ds} \\ \Delta v_{Qr} \\ \Delta v_{Dr} \\ \Delta T_L \end{bmatrix} \tag{2.68}$$

$$A_{IM} = \begin{bmatrix} L_s & 0 & L_m & 0 & 0 \\ 0 & L_s & 0 & L_m & 0 \\ L_m & 0 & L_r & 0 & 0 \\ 0 & L_m & 0 & L_r & 0 \\ 0 & 0 & 0 & 0 & \frac{4J\omega}{3\rho} \end{bmatrix} \tag{2.69}$$

$$B_{IM} = \begin{bmatrix} R_s & \omega L_s & 0 & \omega L_m & 0 \\ -\omega L_s & R_s & -\omega L_m & 0 & 0 \\ 0 & s_l \omega L_m & R_r & s_l \omega L_r & -\omega(L_m i_{Ds0} + L_r i_{Dr0}) \\ -s_l \omega L_m & 0 & -s_l \omega L_r & R_r & \omega(L_m i_{Qso} + L_r i_{Qro}) \\ L_m i_{Dr0} & -L_m i_{Qr0} & -L_m i_{Ds0} & L_m i_{Qs0} & 0 \end{bmatrix} \tag{2.70}$$

$$C_{IM} = \begin{bmatrix} L_s i_{Ds0} + L_m i_{Dr0} \\ -L_s i_{Qs0} - L_m i_{Qr0} \\ s_l L_m i_{Ds0} + s_l L_r i_{Dr0} \\ -s_l L_m i_{Qs0} - s_l L_r i_{Qr0} \\ 0 \end{bmatrix},\ D_{IM} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{-4J(1-s_l)}{3\rho} \end{bmatrix} \tag{2.71}$$

For ease of calculation $(-A_{IM}^{-1}B_{IM})$, A_{IM}^{-1}, $(-A_{IM}^{-1}C_{IM})$ and $(-A_{IM}^{-1}D_{IM})$ were considered as E_{IM}, F_{IM}, G_{IM} and H_{IM} respectively.

2.8 Complete Microgrid Model

Complete microgrid model of the proposed system is developed by combining the individual inverter, line network, load and induction motor state space models from Eqs. (2.42), (2.52), (2.57) and (2.66) respectively. The complete microgrid model is depicted in Eq. (2.72).

$$\Delta \dot{X}_{MG} = A_{MG} \Delta X_{MG} \tag{2.72}$$

where ΔX_{MG} is the integrated states of the microgrid model. The components of ΔX_{MG} and A_{MG} are given in Eqs. (2.73) and (2.74)

$$\Delta X_{MG} = [\Delta x_{INV1}\ \Delta x_{INV2}\ \Delta i_{LineDQ}\ \Delta i_{LoadDQ}\ \Delta X_{IM}]_{1\times 35} \tag{2.73}$$

$$A_{MG} = \begin{bmatrix} [A_{INV} + B_{INV} R_N M_{INV} C_{INV}]_{26\times 26} & [B_{INV} R_N M_{NET}]_{26\times 4} & [B_{INV} R_N M_{IM}]_{26\times 5} \\ [B_{1Line} R_N M_{INV} C_{INV} + B_{2Line} C_{INV_\omega}]_{2\times 26} & [A_{INV} + B_{1Line} R_N M_{NET}]_{2\times 4} & [B_{1Line} R_N M_{IM}]_{2\times 5} \\ [B_{1Load} R_N M_{INV} C_{INV} + B_{2Load} C_{INV_\omega}]_{2\times 26} & [B_{1Load} R_N M_{NET}]_{2\times 4} & [A_{Load} + B_{1Load} R_N M_{IM}]_{2\times 5} \\ [[F_{IM}T][R_N M_{INV} C_{INV}] + G_{IM} C_{INV_\omega} + H_{IM} C_{INV_\omega} A_{INV}] & [F_{IM}T][R_N M_{NET}] & E_{IM} + [F_{IM}T][R_N M_{IM}] \end{bmatrix}_{35\times 35} \tag{2.74}$$

$$C_{INV_{\omega com}} = [0\ \ -m_p\ \ 0\ ....\ \ 0]_{1\times 13},\ C_{INV_\omega} = [C_{INV_{\omega com}}\ \ 0\ \ ....\ \ 0]_{1\times 26} \tag{2.75}$$

M_{INV}, M_{NET} and M_{IM} are used for mapping the connection points between nodes of the network and line. In this study the system has 2 nodes, 2 inverters a single line. The size of M_{INV} is $(2 \times no.\ of\ nodes) \times (2 \times no.\ of\ inverters)$. Similarly size of M_{IM} is (2 $\times no.\ of\ nodes) \times (no.\ of\ connection\ points\ of\ induction\ motor\ i.e.\ 5)$ as this indicates the induction motor connection points with the nodes. Size of M_{NET} is $(2 \times no.\ of\ nodes) \times (2 \times No.\ of\ Lines)$ and R_N is $(2 \times no.\ of\ nodes) \times (2 \times no.\ of\ nodes)$. The diagonal elements of R_N are equal to R_n. Components of M_{NET} are either +1 or -1 depending upon the direction of line current means if current is entering the node then +1 is considered and if leaving the node then -1 is considered. For linking the stator

voltage dynamics of induction motor with the microgrid system matrix T is used whose size is $(no.\ of\ connection\ points\ of\ induction\ motor\ i.e.\ 5)\ \times\ (2 \times no.\ of\ nodes)$.

The system parameters of the above mentioned mathematical model have been given in Table 2.1.

Table 2.1: System Parameters

Parameters	***Value***	***Parameters***	***Value***
ω_{nl}	377 rad/s	ω_c	69 rad/s
m_p	10000 W/rad/s	R_c	1.7 Ω
n_q	4000 VAR/V	L_c	4 mH
R_{Line}	3.3 Ω	C_f	50 μ F
L_{Line}	0.15 mH	L_f	4.2 mH
R_{Load}	24 Ω	R_f	0.1 Ω
L_{Load}	7.8 mH	R_s	3.5 Ω
s_l	0.1	R_r	0.3 Ω
L_m	30 mH	L_s	3 mH
J	63.67 kgm^{-2}	L_r	20 mH
F	105	T_L	7964.04 N.m
ρ	4		

2.9 Summary

This chapter gives an insight about the mathematical model of the microgrid consisting of static load and induction motor as dynamic load. The detailed linearized model of the system has been also mentioned in this chapter. This linearized model needs to be optimized for stable operation of the microgrid system. In the later subsequent chapters this linearized model of the system has been optimized to obtain a stable operating microgrid system including both static and dynamic load.

Chapter 3

Problem Statement and Proposed Solution

Stable operation of microgrid in islanded mode in the presence of both static and dynamic load is a challenge that is addressed in this work. In the aforementioned chapters as it is mentioned that for efficient, fast and stable operation of microgrid the role of controller parameters are noteworthy. In autonomous mode of operation of a microgrid controller parameter settings plays an important role in system stability. Moreover, the effect of controller parameters in system stability can be well understood from the root locus of the system.

3.1 Root Locus Analysis

Root locus analysis enables to study the effect of controller parameters in system stability. In this study, the controller parameters whose effect will be analyzed are the PI gains in voltage and current controller. Root locus analysis mainly gives idea about the damping properties and oscillatory properties of the system [59, 73].The root locus analysis of the controller gains are shown in Figure 3.1 - Figure 3.4 which gives the idea about the effect of controller gain in system stability. Eigen value nearer to the imaginary axes are of low frequency values. Now, if eigen value moves away from the real axes then the system becomes more oscillatory and unstable. On the other hand, if eigen value moves away from the imaginary axes and goes more on the left half plane then damping performance of the system gets improved.

The eigen value of the system is depicted in Table 3.1 where eigen value of each of the states has been mentioned.

Table 3.1: Eigen Value Of the System

Index	States	Eigen Value
1	$\Delta\delta_1$	-2909410.542 +12209388.904i
2	ΔP_1	-2909410.542 - 12209388.904i
3	ΔQ_1	-3261123.439 + 8309127.704i
4	$\Delta\varphi_{d1}$	-3261123.439 - 8309127.704i
5	$\Delta\varphi_{q1}$	-55.194 + 45315.607i
6	$\Delta\gamma_{d1}$	-55.194 - 45315.607i
7	$\Delta\gamma_{q1}$	-352.417 + 44476.812i
8	Δi_{ld1}	-352.417 - 44476.812i
9	Δi_{lq1}	-4107.969 + 31524.405i
10	Δv_{od1}	-4107.969 - 31524.405i
11	Δv_{oq1}	-5629.972 + 29764.098i
12	Δi_{od1}	-5629.972 - 29764.098i
13	Δi_{oq1}	-8720.541 + 8365.113i
14	$\Delta\delta_2$	-8720.541 - 8365.113i
15	ΔP_2	-6328.127 + 8624.675i
16	ΔQ_2	-6328.127 - 8624.675i
17	$\Delta\varphi_{d2}$	-1291.429 + 0i
18	$\Delta\varphi_{q2}$	213.426 + 784.754i
19	$\Delta\gamma_{d2}$	213.426 - 784.754i
20	$\Delta\gamma_{q2}$	-81.284 + 376.280i
21	Δi_{ld2}	-81.284 - 376.280i
22	Δi_{lq2}	-162.677 + 0i
23	Δv_{od2}	-70.227 + 1.578i
24	Δv_{oq2}	-70.227 - 1.578i
25	Δi_{od2}	-67.497 + 0i
26	Δi_{oq2}	22.647 + 0i
27	Δi_{LineD}	1.315 + 0i
28	Δi_{LineQ}	-2.394 + 0i
29	Δi_{LoadD}	-2.393 + 0i
30	Δi_{LoadQ}	-0.018 + 0.045i
31	Δi_{Qs}	-0.018 - 0.045i
32	Δi_{Ds}	-0.021 + 0i
33	Δi_{Qr}	-0.329 + 0i
34	Δi_{Dr}	-0.200 + 0i
35	Δs_l	-0.202 + 0i

In Figure 3.1 it is seen that as K_{pv} increases the most dominant eigen value (9, 10) moves away from the real axes making the system oscillatory but as it moves towards the left hand plane of the imaginary axes so this improves the damping performance of the system. Eigen value (15, 16, 19, 23, 24, 30) moves along the left half of s plane improving the damping performance. On the other hand, eigen value (17, 18) moves towards the real axes and also moves away from the imaginary axes towards left half of s plane improving system's damping and oscillatory performance.

Figure 3.2 shows that as K_{pc} increases the unstable eigen values (9, 10, 11, 12, 13, 14, 15, 16) moves towards the left hand plane of imaginary axes improving system stability but at the same time as they moves away from the real axes so this makes system more oscillatory. Now, eigen value (23, 24, 25, 26) which are dominant low frequency values moves towards the real axes making system less oscillatory.

Now, if K_{iv} increases as shown in Figure 3.3 then eigen value (21, 22, 29, 30, 31) which are dominant low frequency values move away from the imaginary axes towards left half of s plane improving damping performance of the system and improving system stability.

Similarly Figure 3.4 shows if K_{ic} is increased then eigen value (1, 2, 3, 4) moves away from the real axes and imaginary axes making system oscillatory and improving damping performance of the system respectively. On the other hand, eigen value (19, 20) moves towards real axes making system less oscillatory but as they are moving towards right half plane of imaginary axes so damping ratio of the system decreases making system less stable.

From the root locus analysis it is observed that the controller parameters have significant effect on system stability. So, for stable operation of the microgrid optimized controller parameters are inevitable.

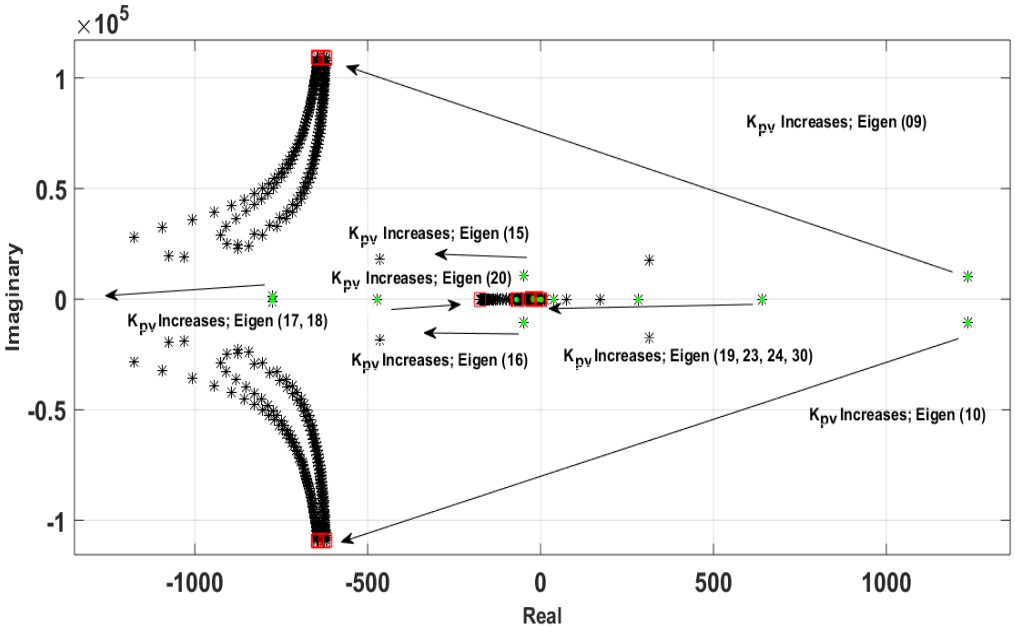


Figure 3.1: Root Locus of the system for $K_{pv} \ \epsilon \ [0, 500]$

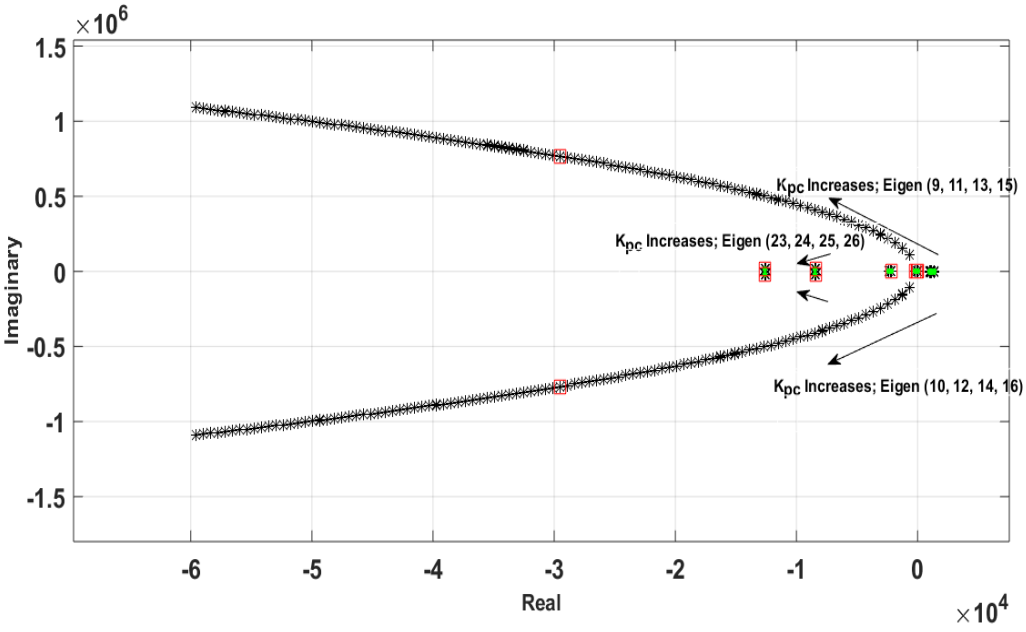


Figure 3.2: Root Locus of the system for $K_{pc} \ \epsilon \ [0, 500]$

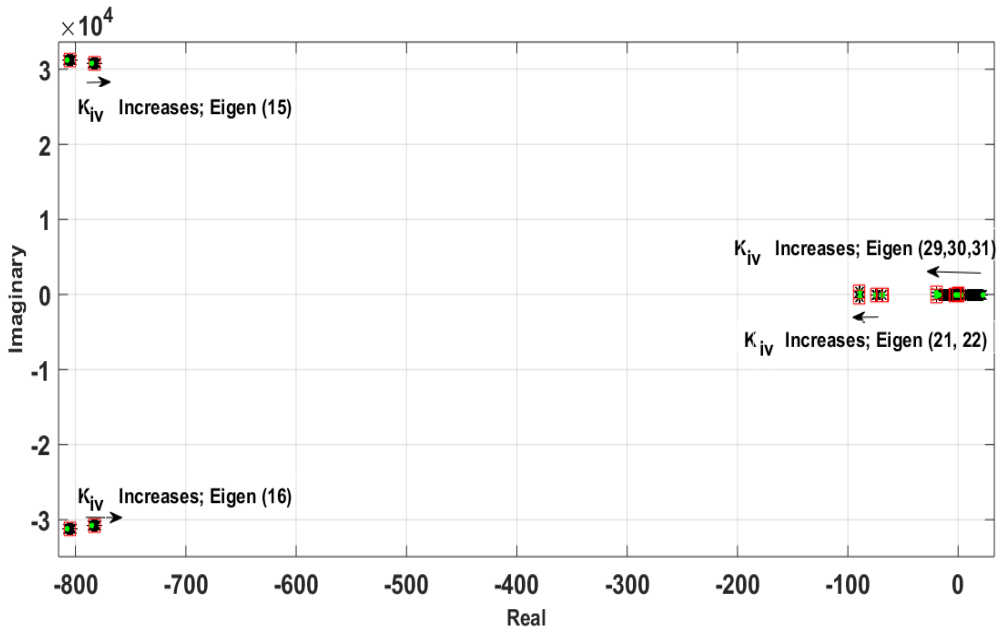


Figure 3.3: Root Locus of the system for $K_{iv} \epsilon [-10, 300]$

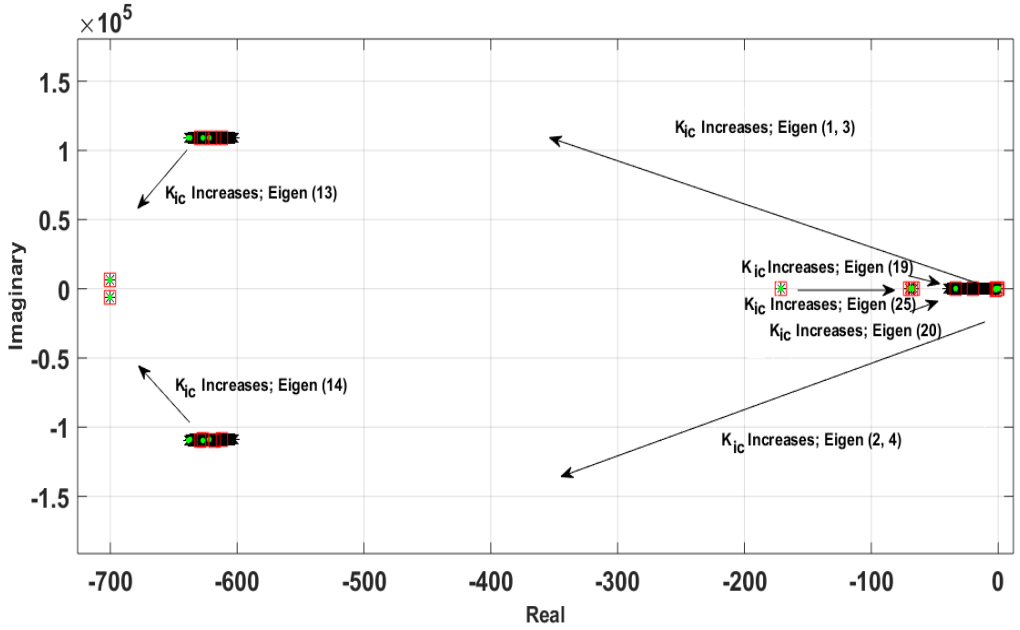


Figure 3.4: Root Locus of the system for $K_{ic} \epsilon [0, 200]$

3.2 Objective Functions

In Chapter 2, the mathematical model of the microgrid was developed where K_{pv1}, K_{iv1}, K_{pc1}, K_{ic1}, K_{pv2}, K_{iv2}, K_{pc2} and K_{ic2} are the controller parameters. Now, from the root locus analysis it is evident that these controller parameters plays a key role is damping and oscillatory characteristics of the system and eventually they controls

system stability. The main objective of this work is to optimize these controller parameters in order to obtain a stable microgrid consisting of both static and dynamic load. The objective functions based on the damping ratio and eigen value are stated below in Eq. (3.1).

$$J_1 = (-\sigma_{desired} - Minimum(\sigma_N)),\ J_2 = (\zeta_{desired} - Minimum(\zeta_N)) \tag{3.1}$$

Here σ and ζ represents real part of eigen values and damping ratio respectively. Here N represents the number of states. For this study $N = 1,\ 2,\ 3,\ .....\ 35$ as there are 35 states in the system as per the mathematical model developed in Chapter-2. This indicates that the σ and ζ value will be evaluated for each of the states and the objective of this study is to optimize the σ and ζ value for each of the states to obtain stable performance of the microgrid system.

$$\zeta_N = \frac{-\sigma_N}{\sqrt{{\sigma_N}^2 + {\omega_N}^2}} \tag{3.2}$$

$\sigma_{desired}$ and $\zeta_{desired}$ are taken as reference values in order to limit the objective functions within the desired boundary [74].

The problem constraints for the developed model are given below in Eq. (3.3). Now as seen from root locus analysis that as the controller gains increases from zero to positive value, the damping performance improves but at times makes system more oscillatory. So, in order to obtain a stable system with less oscillations and improved damping performance the controller gains are limited from $0\ to\ 500$.

$$0 < K_{pv},\ K_{pi},\ K_{iv},\ K_{ic} < 500 \tag{3.3}$$

Here, the controller parameters are aimed to be kept within the given range in order to obtain stable performance from the microgrid system. Thus, these are the objective functions which need to be optimized using optimizing algorithms in order to obtain optimized controller parameters for stable operation of microgrid.

3.3 Proposed Solution

In this study multi-objective based optimization algorithms have been proposed in order to develop an autonomous AC microgrid with optimal controller parameters. As mentioned in earlier sections that there are two objective functions that need to be optimized for obtaining a stable system. The objective functions mentioned in previous section are conflicting in nature as if σ increases then ζ becomes more negative i.e. it

decreases and vice versa condition if σ decreases. So, with a view to optimizing these functions multi-objective based non dominated sorting technique has been incorporated with nature inspired algorithms to form hybrid algorithm. Detailed methodology of the proposed algorithms has been included in later sections of this chapter.

3.4 Non-dominated Sorting Technique

In this technique basically pareto-dominance criterion is used to sort the solutions obtained from the generated population. Here, at first a set of solutions is created from the initial randomly generated population and each of these solutions is again compared with the other solutions generated from the population. Then the solutions are sorted amongst themselves from which two sets of solutions are developed, one is non-dominated solutions i.e. they are not dominated by any other solutions and other set is known is dominated solutions i.e. they are dominated by any one of the solution at least. Moreover, in this technique domination count of the solutions are kept in account which means if one solution suppose A dominates another solution B then domination count of A is 0 and domination count of B is 1. Now again if there is another solution C which is being dominated by A and B then domination count of C is 2. The first set of non-dominated solution is assigned to a particular front suppose, F_1 for instance in this case that solution is A which is is removed from the generated population. Now solution in front F_1 is assigned a rank r_1 which means rank value of the solution is 1. In the next step the remaining solutions i.e. B and C are again compared with each other and as solution A has been removed so domination count of B is reduced to 0 and that of C is reduced to 1. Similarly another set of non-dominated solution is obtained; for this case it is B which is assigned to another front suppose, F_2 and the solution in front F_2 is assigned to rank, r_2 which means the rank of the solution is 2. This process is continued in a loop till all the solutions generated from the population are assigned to any specific front. Each of these fronts are known as pareto fronts and each of those fronts are assigned a particular rank on the basis of the number of comparison required for those solutions to be included in a front.

Pareto-dominance method is based on two criterion [55] [75]. Suppose if there are two solutions X_1 and X_2 to be compared then X_1 will dominate X_2 if both of the given conditions are fulfilled.

Condition 1: For both the objectives J_1 and J_2 in our study, solution X_1 is not worse than solution X_2 .

Condition 2: For at least one objective J_1 or J_2, solution X_1 is definitely better

than solution X_2.

Following the above criterion non-dominated sorting is performed. Now in the above methodology the solutions having lower rank value are always preferred i.e. $r = 1$ is preferred over $r = 2$ as solutions with lower rank value have better fitness value [54]. Now, solutions in a particular front have same rank and same fitness value. So, in order to separate the solutions and determine better solution calculation of crowding distance is required. Solutions in a front having more crowding distance i.e. surrounding area is less crowded; is preferred over solutions with less crowding distance due high possibility of diversity preservation [76].

3.5 Crowding Distance Calculation

Crowding distance means the distance between two solutions in a front. It helps to determine the concentration of solutions around a specific solution in a front [55]. The methodology of crowding distance calculation is given below:

Step 1: Suppose F is a pareto front which contains X number of solutions. Now at first for all the individual solutions in the fronts, the crowding distance is initiated to zero. If $F_i(D_j)$ represents crowding distance of j^{th} solution in the i^{th} front then $F_i(D_j) = 0, \forall\, i,\ j$.

Step 2: Now for each objective function, the solutions in a front are sorted on the basis of their fitness value in ascending order. If m represent objective functions then for this study $m = [\ J_1,\ J_2\ ]$. Now at first for $m = J_1$ the solutions in the front F_i are sorted. Mathematically it can be represented as $S = sort(F_i,\ J_1)$.

Step 3: The boundary values for each objective functions in a front is assigned in this step. The solutions with highest and lowest fitness values in a front are assigned infinite crowding distance value i.e. for front F_i, $S(D_1) = S(D_X) = \infty$. Here, X represents final number of solution.

Step 4: Depending on the boundary limits calculated in the previous step, in this step the crowding distance for the intermediate solutions i.e. crowding distance $L = D_2$ ..... D_{X-1} for front F_i needs to be calculated. This calculation is done following eq. (4.1). Here f_m^{max} and $_m^{min}$ are maximum and minimum value of fitness values

respectively for $m = [\ J_1,\ J_2\ ]$ objectives.

$$S(D_L) = S(D_L) + \frac{S_{L+1}.m - S_{L-1}.m}{f_m^{max} - f_m^{min}} \tag{3.4}$$

Step 5: Now similar procedure is followed for $m = J_2$ i.e. same procedure will be repeated in a loop and crowding distance of a particular solution is summation of distance calculated for both the objectives.

Following the above procedure crowding distance is calculated.

3.6 Optimization Process of Non-dominated Sorting Whale Optimization Algorithm (NSWOA)

Non-dominated sorting based optimization technique has been incorporated with Whale optimization algorithm (WOA) to form a nature inspired hybridized algorithm. As multi-objective optimization problems often requires various conflicting objectives to be solved so it is challenging in satisfying each of the conflicting objectives and optimize them [15]. At times there exist different discordant objective functions so non-dominated sorting technique at first sorts a set pareto-optimal solutions which can be considered as the optimal solutions. In order to differentiate between dominated and non-dominated sets of data pareto-optimal value is computed from the pareto-optimal solutions. As mentioned earlier the sorting is done by moving the optimal solutions towards pareto-front by non-dominated sorting of their rank values. The obtained results are then again optimized using WOA technique in order to obtain a better, efficient result with less computational time. As non-dominated sorting technique is hybridized with WOA so this proposed technique is named as Non-dominated Sorting Whale Optimization Algorithm (NSWOA).

Whale optimization algorithm consists of three stages. At first the humpback whales encircle their preys and in the next step they use bubble hunt method to attack their prey. Next the whales again begin to search for their prey randomly depending upon the position of each other [39, 77]. A flowchart is given in Figure 3.5 showing detailed procedure of NSWOA. The step-wise procedure of NSWOA is described as follow:

Step 1: The first step is to initialize the system parameters as well as define the number of iteration and generate the initial population for the controller parameters which are to be taken into consideration for optimization. Size of initial population of whale i.e search agent was taken as 50.

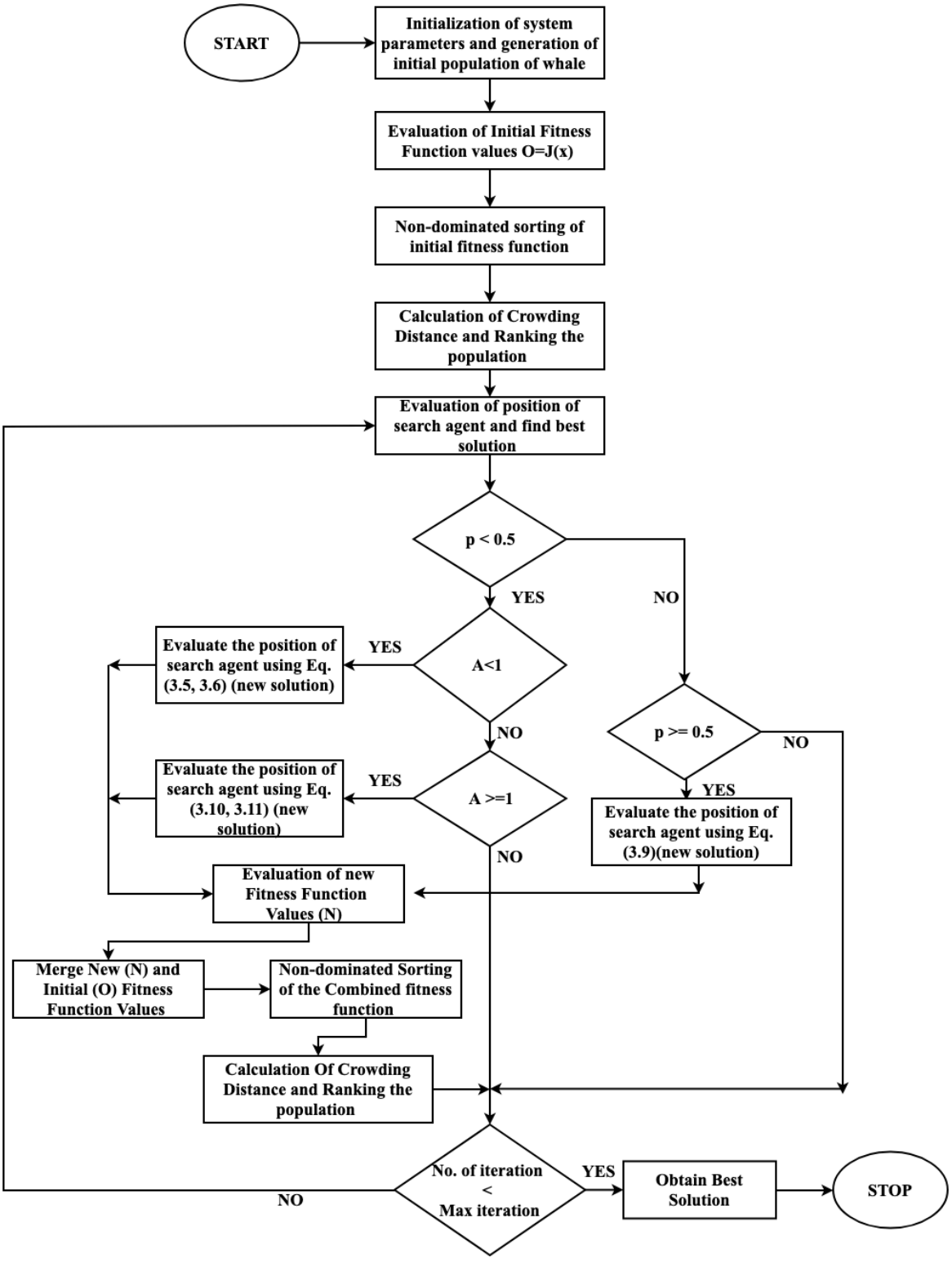


Figure 3.5: Flow chart depicting NSWOA

Step 2: In this step the fitness of each search agent is evaluated using fitness function i.e. the objective functions $J_1(x)$ and $J_2(x)$ of this study.

Step 3: In this step initially generated search agents are sorted according to their fitness function using non-dominated sorting technique as mentioned in the previous section.

Step 4: In this step a loop is run to evaluate the position of the search agent so that they can encircle their prey. Humpback whales considers current solution as the best solution and detects that as the target prey . Now if $\vec{X}^*$ as the position of current best solution which is updated on the basis of better solution being obtained then encircling behaviour of the whales can be represented by Eqs. (3.5) and (3.6) [39].

$$\vec{D} = |\vec{C}.\vec{X}^*(t) - \vec{X}(t)| \tag{3.5}$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A}.\vec{D} \tag{3.6}$$

Here $\vec{X}$ is the position vector, $|\ |$ represents absolute value, $'\ .\ '$ is used for element by element multiplication, t represent the number of current iteration, $\vec{A}$, $\vec{C}$ are coefficient vectors which are updated to adjust the latest position of prey around the search agent following Eqs. (3.7) and (3.8) [39].

$$\vec{A} = 2\vec{a}.\vec{r} - \vec{a} \tag{3.7}$$

$$\vec{C} = 2.\vec{r} \tag{3.8}$$

Here, $\vec{a}$ decreases linearly from 2 to 0 as the iteration progresses which make $\vec{A}$ vary in $[-a,\ a]$ range and $\vec{r}$ is a random vector between [0, 1].

Step 5: The next stage of WOA technique is spiral positioning update of the humpback whales which is done to replicate their helix-shaped movement. Position of the search agent are updated using Eq. (3.9) as given in [39].

$$\vec{X}(t+1) = \vec{D}'.e^{bl}.\cos(2\pi l) + \vec{X}^*(t) \tag{3.9}$$

where, b defines the shape of the logarithmic spiral movement of the whales and it is a constant value, l is a random number between [-1, 1] and $\vec{D}' = |\vec{X}^*(t) - \vec{X}(t)|$ indicates the distance between the position of prey ($\vec{X}^*$) from position ($\vec{X}$) of search agent .

Step 6: Now, there are two cases for updating the position of the whales: one

is by waning or reducing the encircling procedure and the other one is using spiral model [39]. So probability is indicated by p which is a random number between [0, 1]. In this step an if else condition is run to satisfy the objective. If random number $p < 0.5$ and $|\vec{A}| < 1$ then position of search agent is updated using Eqs. (3.5) and (3.6).

Alternately, if $p < 0.5$ and $|\vec{A}| \geq 1$ then a random search agent is selected and $\vec{X_r}$ denotes the position of that search agent in the current population. For that random search agent the distance between prey and search agent is determined using Eq. (3.10) and position of search agent is updated using Eq. (3.11).

$$\vec{D} = |\vec{C}.\vec{X_r} - \vec{X}| \tag{3.10}$$

$$\vec{X}(t+1) = \vec{X_r} - \vec{A}.\vec{D} \tag{3.11}$$

Step 7: If the conditions of step 6 are not satisfied and if $p \geq 0.5$ then position of search agent is updated using Eq. (3.9)

Step 8: Fitness function values of the search agent of the whales are evaluated in this step which is the new fitness value N.

Step 9: Initial generation of fitness function values is merged with the WOA optimized new fitness values generated in step 8.

Step 10: Here, non dominated sorting of the combined fitness function values is carried out. A loop is run from step 4 to step 10 till the maximum number of iteration is reached and stopping criterion is fulfilled.

Step 11: The best possible position of the search agent is determined after the loop ends and that is the optimal solution.

Thus, in this way NSWOA optimization technique is being implemented.

3.7 Optimization Methodology of Non-dominated Sorting Firefly Algorithm (NSFA)

The technique used for optimization of the controller parameters in the proposed system is non-dominated sorting which is used mostly for multi-objective problems. This non-dominated sorting is hybridized with firefly algorithm to form Non-dominated sorting Firefly Algorithm (NSFA). The detailed step wise procedure of non-dominated Sorting Firefly Algorithm (NSFA) is explained through a flowchart shown in Fig-

ure 3.6.

Step 1: At first the system parameters are defined and initial population of fireflies are generated for the controller parameters to be optimized. Initial population size was 50 for making the iteration quicker.

Step 2: Fitness functions $J_1(x)$ and $J_2(x)$ of the fireflies are evaluated where basically the light intensity I of the fireflies were determined as attractiveness of a firefly depends on its brightness [26]. This light intensity is the initial fitness function value.

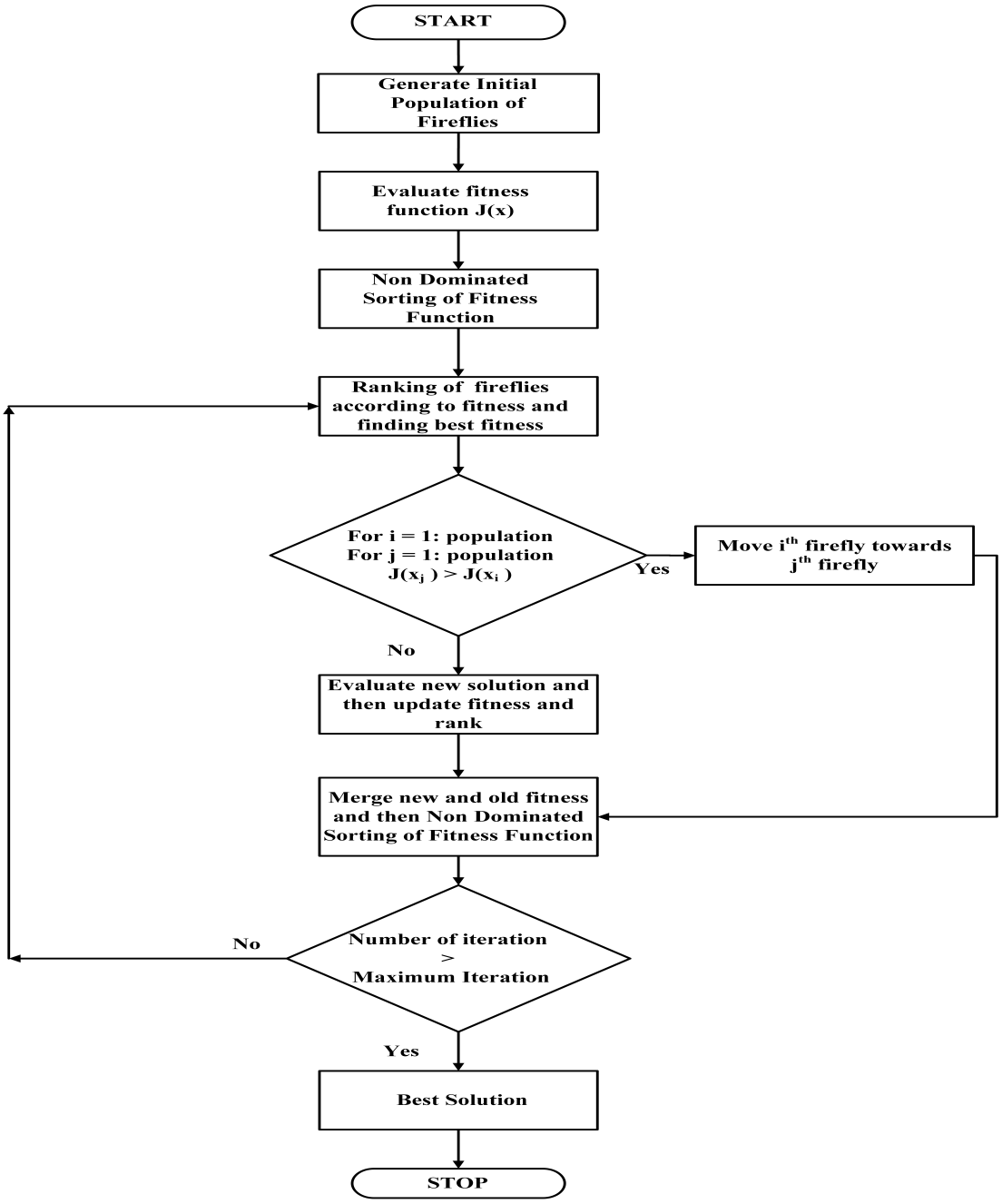


Figure 3.6: Flow chart for NSFA

Step 3: In this step non-dominated sorting of the initial fitness function is performed for sorting the fireflies of initial generation.

Step 4: In the next step, the rank of fireflies are determined according to their fitness function and the best fitness amongst them is noted.

Step 5: Now, two $for\ loops$ are run as there are two fireflies i and j respectively and their fitness values are compared. If j^{th} firefly is brighter than i^{th} firefly then i^{th} firefly is moved towards the j^{th} firefly using the formula shown in Eq. (3.12) [26].

$$x_i(t+1) = x_i(t) + \beta \exp^{-\gamma r^2}(x_i - x_j) + \alpha\epsilon_i \tag{3.12}$$

where β denotes attractiveness of fireflies, α denotes mutation coefficient, γ is light absorption coefficient and r represents distance between i^{th} and j^{th} firefly respectively. The values of β, α and γ were 2, 0.2 and 1 respectively [26, 30]. $\beta \exp^{-\gamma r^2}(x_i - x_j)$ is used for the attraction of j^{th} firefly and $\alpha\epsilon_i$ is used as a randomization parameter.

Step 6: If previous step is not satisfied then again fitness function of the fireflies is evaluated. Rank and the best fitness are updated accordingly. From this step we determine the new fitness value of the fireflies.

Step 7: In this step initial and new fitness values of the fireflies are merged and non-dominated sorting of the fitness function is carried out. If the condition given in step 5 is satisfied then step 6 is skipped and result is moved directly from step 5 to this step.

Step 8: The loop from step 4 is continued to run till maximum number of iteration defined at the beginning and finally the best fitness is determined after completion of all the iteration. Thus, by this methodology NSFA algorithm is implemented.

3.8 Summary

In this chapter the problem statement of this study has been described along with the proposed solution for that problem statement. At the beginning a root locus analysis using the mathematical model of the system was described in order to justify the objective functions for this study. In the later part proposed solution for the objective functions has been mentioned where a detailed overview, methodology and implementation of two proposed hybridized algorithms namely : NSWOA and NSFA has been

discussed. Moreover the detailed procedure of non-dominated sorting and crowding distance calculation were also discussed in this chapter.

Chapter 4

Performance Analysis of NSWOA

The mathematical model of the aforementioned microgrid is being designed using the software MATLAB. In this chapter the model developed is used for stability and performance analysis of the system using the proposed NSWOA algorithm. The comparative study between NSWOA, NSGA-II and SPEA algorithms based on the developed model is discussed in three segments namely - (i) Eigen value Analysis, (ii) Time Domain Simulation and (iii) Statistical Tests where performance of each of the algorithms has been analysed.

4.1 Simulation Setup

The simulating parameters for all the optimizing algorithms namely: NSWOA, NSGA-II and SPEA are kept same due to the comparative study that is to be made amongst these algorithms in order to compare their performance. The initial population size for NSWOA, NSGA-II and SPEA are taken as 50 and maximum number of iterations for each case is 100. As all these algorithms consists of a common trait that is randomness so each algorithm was run 30 times to find the average performance of each of the algorithms and compare their performance.

Now for NSWOA, the random vector $\vec{r}$ is defined between $[0,\ 1]$, $\vec{a}$ is also defined so that it linearly decrease from 2 to 0 which results in $\vec{A}$ to limited in range of $[-a,\ a]$ as per Eq. (3.7) and Eq. (3.8). b is taken as 1 to define the logarithmic shape of spiral movement and p is defined as a random number between $[0,\ 1]$. All these parameters are set following ref. [39].

Similarly for NSGA-II, rate of mutation was defined as 0.02. Percentage of Crossover and percentage of mutation was taken as 0.7 and 0.4 respectively.

Finally for SPEA, the archive size was defined as 50, percentage of crossover is

same as NSGA-II i.e. 0.7 and percentage of mutation is 0.3.

4.2 Eigen Value Analysis

Eigen value analysis is carried out for determination of the capability of the proposed NSWOA algorithm in stabilization of the proposed system containing both static and dynamic loads. The eigen value state of the proposed system has been reflected on Figure 4.1 which shows the initial state i.e. before optimization condition. It is notable from Figure 4.1 and Table 4.1 that there exists instability in the system as the eigen values of states $\Delta\varphi_{q2}$, $\Delta\gamma_{d2}$, Δi_{oq2} and Δi_{LineD} are situated on the right half portion of s plane which means the values are positive. As a result, for making the system stable, the proposed NSWOA algorithm is being used for optimization of the system which results in making the unstable positive values of the states negative. The parameters optimized are $K_{pv1} = 60.0229$, $K_{iv1} = 339.6462$, $K_{pc1} = 105.3193$, $K_{ic1} = 155.0871$, $K_{pv2} = 129.8271$, $K_{iv2} = 227.7560$, $K_{pc2} = 240.2978$, and $K_{ic2} = 97.3373$.

Figure 4.2 shows the eigen value of the states after optimizing the system using NSWOA algorithm and it is observable that the unstable positive eigen values of the states $\Delta\varphi_{q2}$, $\Delta\gamma_{d2}$, Δi_{oq2} and Δi_{LineD} have moved from positive axis to negative axis i.e. from right half of s plane to the left half of s plane. The eigen values of the system before and after optimization are mentioned in Table 4.1. Moreover the eigen values nearer to the imaginary axes i.e of states Δi_{LoadQ}, Δi_{Qs}, Δi_{Ds}, Δi_{Qr}, Δi_{Dr} and Δs_l have moved further away from the imaginary axes towards in the left hand portion of s plane which indicates that system has become more stable after using NSWOA technique.

The above discussion indicates the ability of the proposed NSWOA algorithm in stabilization of the unstable system using optimized setting for the controllers.

Table 4.1: Eigen Value Analysis

Index	States	Eigen Value of the System Before Optimization	Eigen Value of the System After Using NSWOA Optimization
1	$\Delta\delta_1$	-2909410.542 +12209388.904i	-2909410.508 + 12209388.839i
2	ΔP_1	-2909410.542 - 12209388.904i	-2909410.508 - 12209388.839i
3	ΔQ_1	-3261123.439 + 8309127.704i	-3261123.425 + 8309127.501i
4	$\Delta\varphi_{d1}$	-3261123.439 - 8309127.704i	-3261123.425 - 8309127.501i
5	$\Delta\varphi_{q1}$	-55.194 + 45315.607i	-28863.554 + 384659.375i
6	$\Delta\gamma_{d1}$	-55.194 - 45315.607i	-28863.554 - 384659.375i
7	$\Delta\gamma_{q1}$	-352.417 + 44476.812i	-28820.165 + 384322.746i
8	Δi_{ld1}	-352.417 - 44476.812i	-28820.165 - 384322.746i
9	Δi_{lq1}	-4107.969 + 31524.405i	-12450.226 + 172830.502i
10	Δv_{od1}	-4107.969 - 31524.405i	-12450.226 - 172830.502i
11	Δv_{oq1}	-5629.972 + 29764.098i	-12497.721 + 172390.083i
12	Δi_{od1}	-5629.972 - 29764.098i	-12497.721 - 172390.083i
13	Δi_{oq1}	-8720.541 + 8365.113i	-8197.709 + 21329.295i
14	$\Delta\delta_2$	-8720.541 - 8365.113i	-8197.709 - 21329.295i
15	ΔP_2	-6328.127 + 8624.675i	-12535.996 + 19463.657i
16	ΔQ_2	-6328.127 - 8624.675i	-12535.996 - 19463.657i
17	$\Delta\varphi_{d2}$	-1291.429 + 0i	-2217.887 + 515.833i
18	$\Delta\varphi_{q2}$	213.426 + 784.754i	-2217.887 - 515.833i
19	$\Delta\gamma_{d2}$	213.426 - 784.754i	-22.813 + 194.338i
20	$\Delta\gamma_{q2}$	-81.284 + 376.280i	-22.813 - 194.338i
21	Δi_{ld2}	-81.284 - 376.280i	-138.088 + 0i
22	Δi_{lq2}	-162.677 + 0i	-4.848 + 52.592i
23	Δv_{od2}	-70.227 + 1.578i	-4.848 - 52.592i
24	Δv_{oq2}	-70.227 - 1.578i	-70.683 + 0i
25	Δi_{od2}	-67.497 + 0i	-67.556 + 2.059i
26	Δi_{oq2}	22.647 + 0i	-67.556 - 2.059i
27	Δi_{LineD}	1.315 + 0i	-5.740 + 0.974i
28	Δi_{LineQ}	-2.394 + 0i	-5.740 - 0.974i
29	Δi_{LoadD}	-2.393 + 0i	-2.801 + 0i
30	Δi_{LoadQ}	-0.018 + 0.045i	-1.715+ 0i
31	Δi_{Qs}	-0.018 - 0.045i	-0.188 + 0i
32	Δi_{Ds}	-0.021 + 0i	-1.472 + 1.870e-06i
33	Δi_{Qr}	-0.329 + 0i	-1.472 - 1.870e-06i
34	Δi_{Dr}	-0.200 + 0i	-0.405 + 0i
35	Δs_l	-0.202 + 0i	-0.405 + 0i

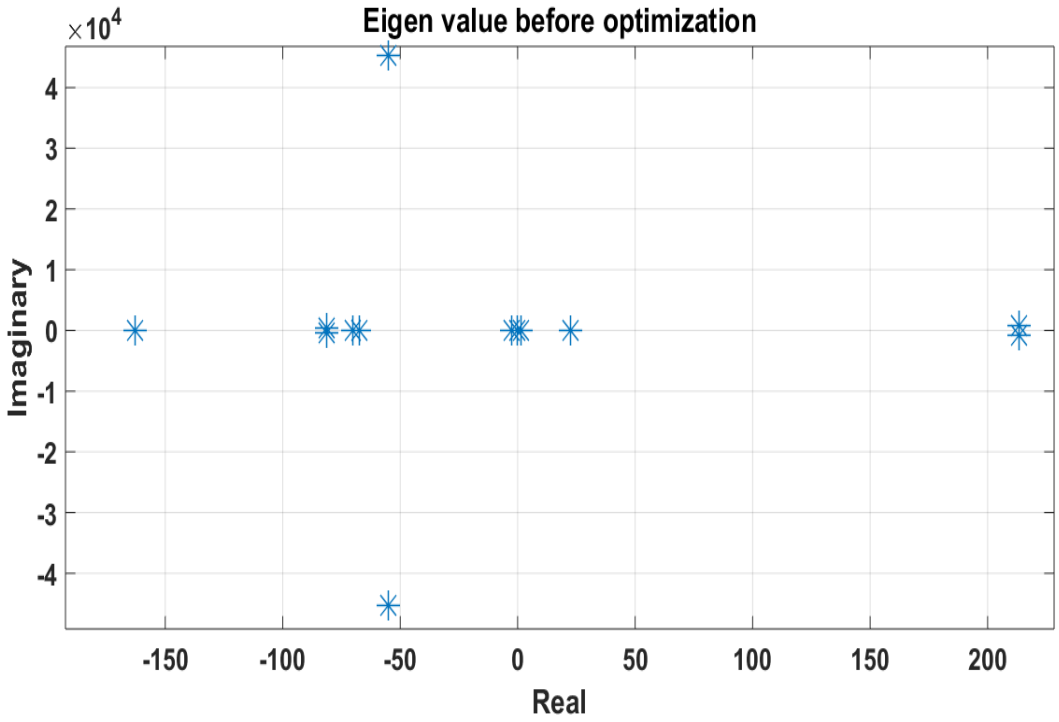


Figure 4.1: Eigen Value of the system before optimization

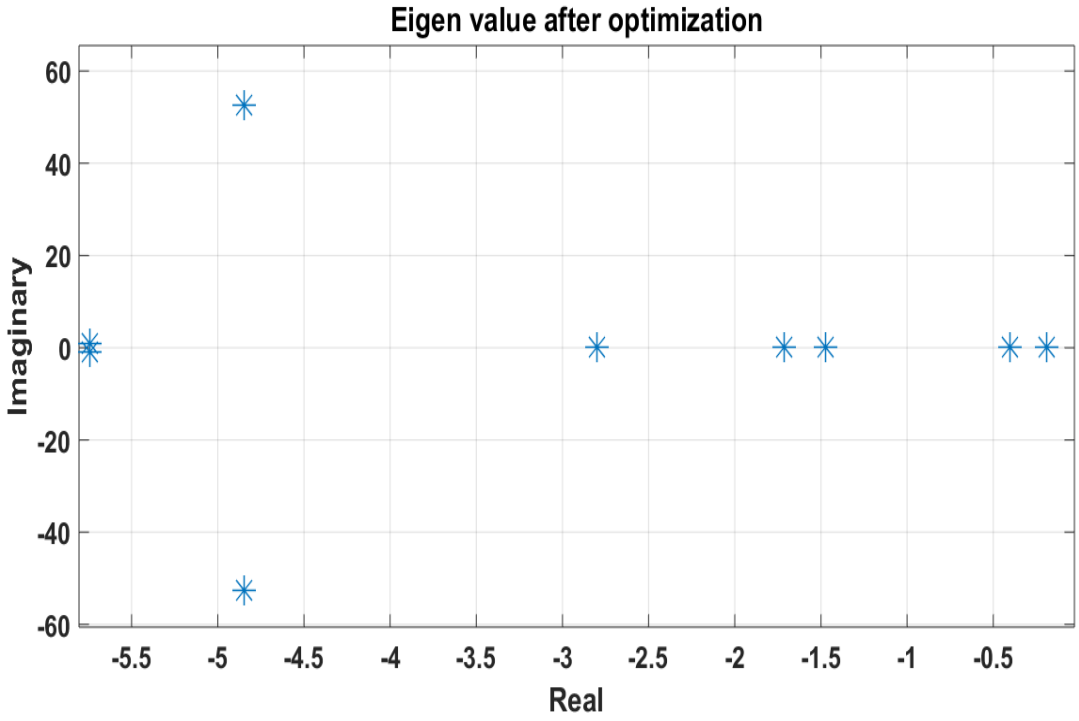


Figure 4.2: Eigen Value of the system after optimization

4.3 Time Domain Simulation Analysis

In this section, results of proposed NSWOA algorithms is being compared with both NSGA-II and SPEA algorithms to differentiate their performance. The comparison of performance was done on the basis of amount of overshoot and oscillation frequency of real power, reactive power, inductor current ($d-q$) and output voltage ($d-q$) of DG-1 and DG-2. The comparison is presented in Table 4.2. The step response of the above mentioned parameters for DG-1 and DG-2 are represented in Figure 4.3 - Figure 4.14. For real power of DG-1 and DG-2 it is visible from Table 4.2 and Figure 4.3, Figure 4.4 that SPEA has overshoot percentage of 75.1 compared to zero overshoot in NSGA-II and NSWOA though oscillation frequency is comparable for all these algorithms. Similarly from Figure 4.5, Figure 4.6 and tabulated data in Table 4.2 it is seen that NSWOA has zero overshoot for both the DGs. But both NSGA-II and SPEA have very high overshoot percentage of $7.54e\ +13$ and $5.73e\ +13$ respectively for DG-1. For DG-2 only SPEA produces overshoot percentage of 15. The oscillation frequency of reactive power is comparable for all the algorithms in case of DG-1; whereas for DG-2, SPEA provides more oscillation frequency i.e. $5025.729\ Hz$ compared to $3093.626\ Hz$ of NSGA-II and $3097.737\ Hz$ of NSWOA.

Now, for inductor current ($d-axis$) it is observed from Figure 4.7 and Figure 4.8 that in case of DG-1 all optimizing algorithms provide zero overshoot among which NSWOA has the least oscillation frequency of $61166.864\ Hz$ compared to $64349.572\ Hz$ of NSGA-II and $75598.184\ Hz$ of SPEA. Again for DG-2, SPEA gives largest overshoot value of 15% whereas NSWOA and NSGA-II provides the least zero overshoot. Here, oscillation frequency of NSGA-II and NSWOA is zero compared to larger oscillation frequency of SPEA which is $6.785\ Hz$. Figure 4.9, Figure 4.10 and tabular data in Table 4.2 shows that, for DG-1, SPEA has the largest overshoot of $2.31e\ +3\%$ where as NSGA-II provides overshoot percentage of 132% and NSWOA provides zero overshoot which is the least overshoot whereas SPEA has the least oscillation frequency of $18305.6577\ Hz$ compared to NSGA-II and NSWOA. For DG-2 the oscillation frequency and overshoot values of NSGA-II and NSWOA are comparable and less compared to SPEA where SPEA produces largest overshoot of 55.8

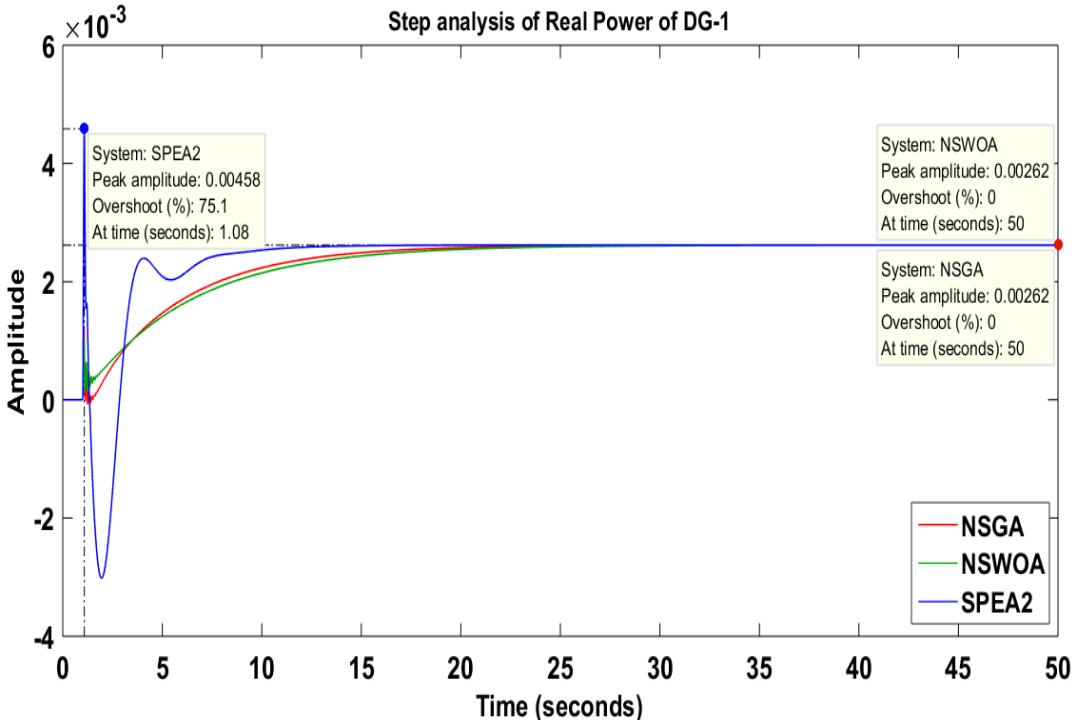
In case of output voltage ($d-axis$) of DG-1 as shown in Figure 4.11 it is seen that all algorithms produces zero overshoot whereas from Figure 4.12 it can be observed that for DG-2 though SPEA gives small overshoot of $2.03e\ -7$ but other algorithms have zero overshoot. Now, SPEA and NSWOA have relatively lower oscillation frequency for the output voltage ($d-axis$) for both DGs. Similarly for output voltage ($q-axis$) represented in Figure 4.13, Figure 4.14 and Table 4.2, NSWOA provides

zero overshoot for both DGs with better oscillation frequency response.

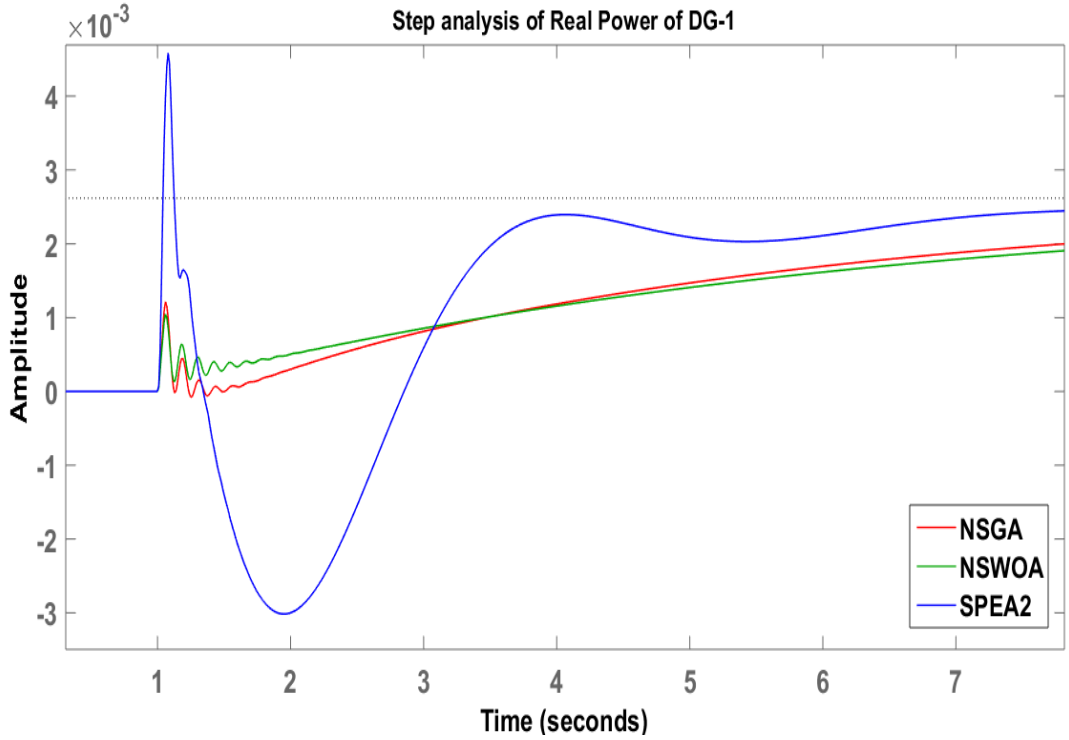
The time domain analysis of the system using the algorithms namely NSGA-II, SPEA and NSWOA shows that though all the optimizing techniques have the capability to optimize and improve damping characteristics of the system but from comparison it is clearly observed that the proposed NSWOA algorithm provides better step response compared to the other mentioned algorithms.

Table 4.2: NSGA-II, SPEA and NSWOA comparison on the basis of overshoot and oscillation frequency

Criterion	Parameters	NSGA-II	SPEA	NSWOA
Real Power of DG-1	*Overshoot (%)*	0	75.1	0
	Oscillation Frequency (Hz)	1943184.58924875	1943184.58908575	1943184.58590787
Real Power of DG-2	*Overshoot (%)*	0	1.83e -07	0
	Oscillation Frequency (Hz)	3093.62629126010	5025.72911114868	3097.73723900745
Reactive Power of DG-1	*Overshoot (%)*	7.54e +13	5.73e +13	0
	Oscillation Frequency (Hz)	1322438.73204884	1322438.73035667	1322438.71460419
Reactive Power of DG-2	*Overshoot (%)*	0	15	0
	Oscillation Frequency (Hz)	3093.62629126010	5025.72911114868	3097.73723900745
Output Voltage of DG-1 (d-axis)	*Overshoot (%)*	0	0	0
	Oscillation Frequency (Hz)	42789.3365339348	18305.6577760513	27506.8287713625
Output Voltage of DG-2 (d-axis)	*Overshoot (%)*	0	2.03e -07	0
	Oscillation Frequency (Hz)	8.10924153674564	2.04792679461399	8.37029981802671
Output Voltage of DG-1 (q-axis)	*Overshoot (%)*	8.56	612	0
	Oscillation Frequency (Hz)	42858.5317697674	18218.3019399697	27436.7338495822
Output Voltage of DG-2 (q-axis)	*Overshoot (%)*	0	15.1	0
	Oscillation Frequency (Hz)	0.931166398517253	2.04792679461399	0
Inductor Current of DG-1 (d-axis)	*Overshoot (%)*	0	0	0
	Oscillation Frequency (Hz)	64349.5728503008	75598.1848439340	61166.8648946795
Inductor Current of DG-2 (d-axis)	*Overshoot (%)*	0	15	0
	Oscillation Frequency (Hz)	0	6.78536409154396	0
Inductor Current of DG-1 (q-axis)	*Overshoot (%)*	132	2.31e +3	0
	Oscillation Frequency (Hz)	42789.3365339348	18305.6577760513	27506.8287713625
Inductor Current of DG-2 (q-axis)	*Overshoot (%)*	24.6	55.8	20.9
	Oscillation Frequency (Hz)	8.10924153674564	6.78536409154396	8.37029981802671

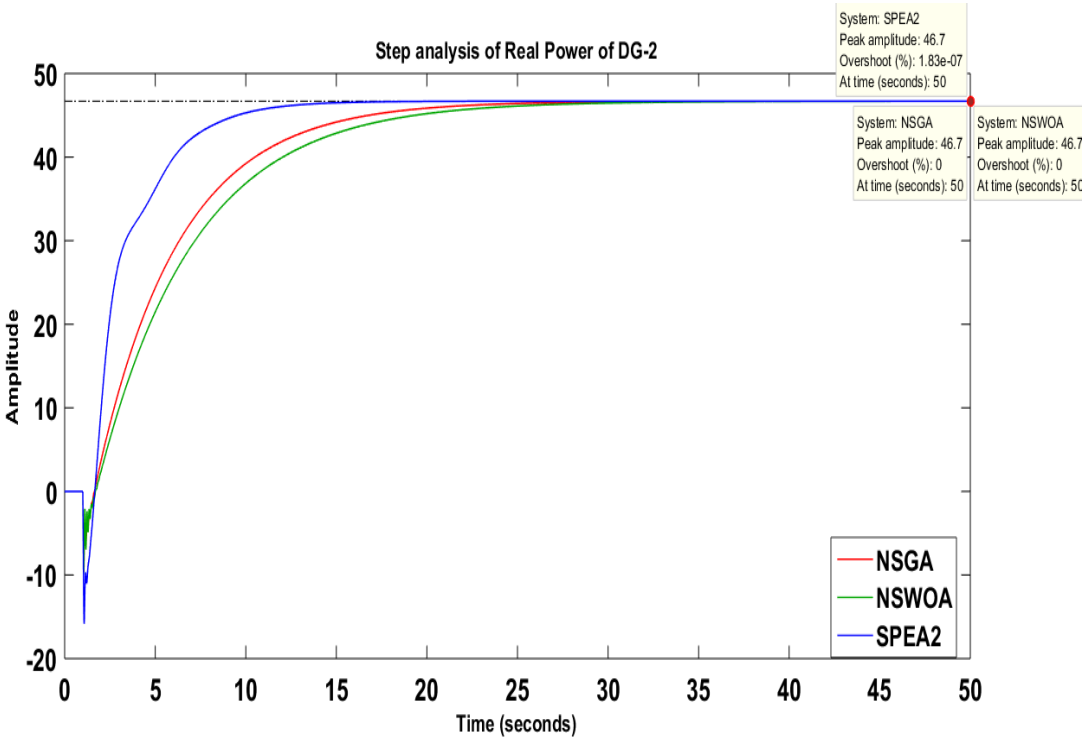


(a) Real Power of DG-1

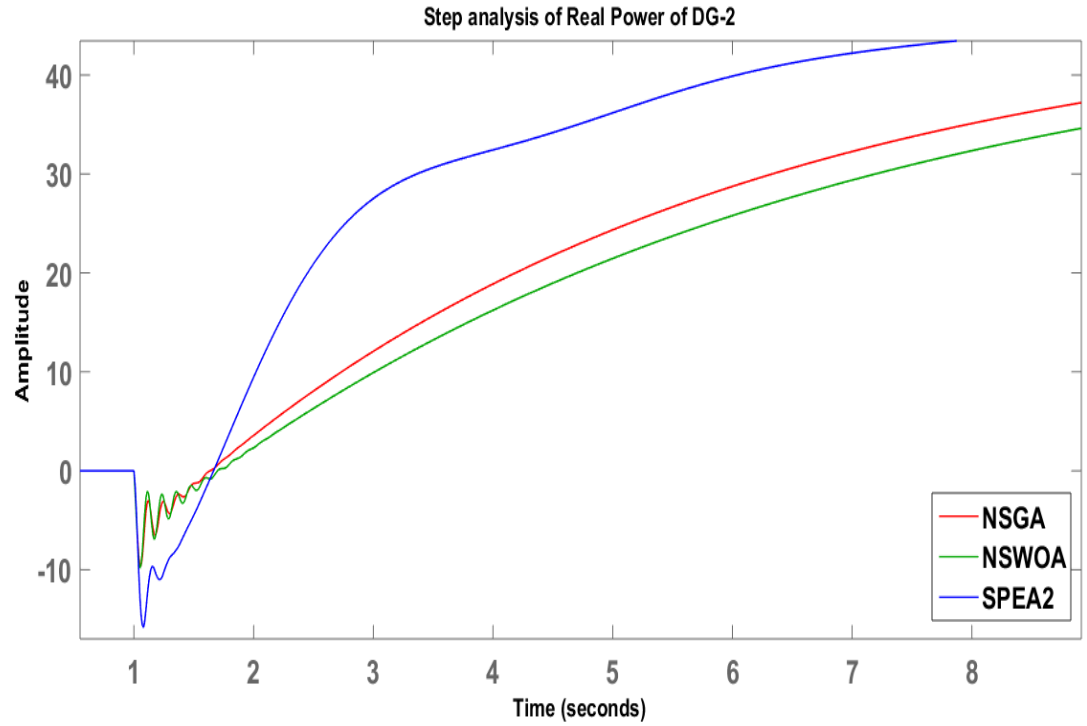


(b)Real Power of DG-1 (Zoomed View)

Figure 4.3: Step Analysis of Real Power of DG-1

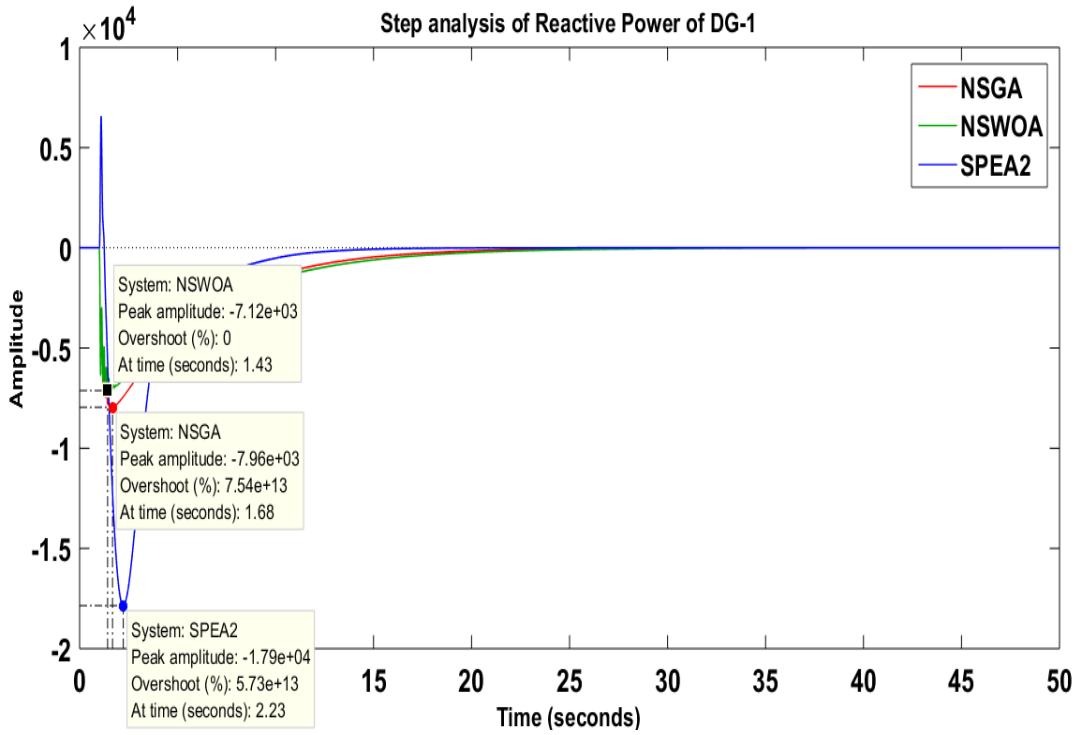


(a) Real Power of DG-2

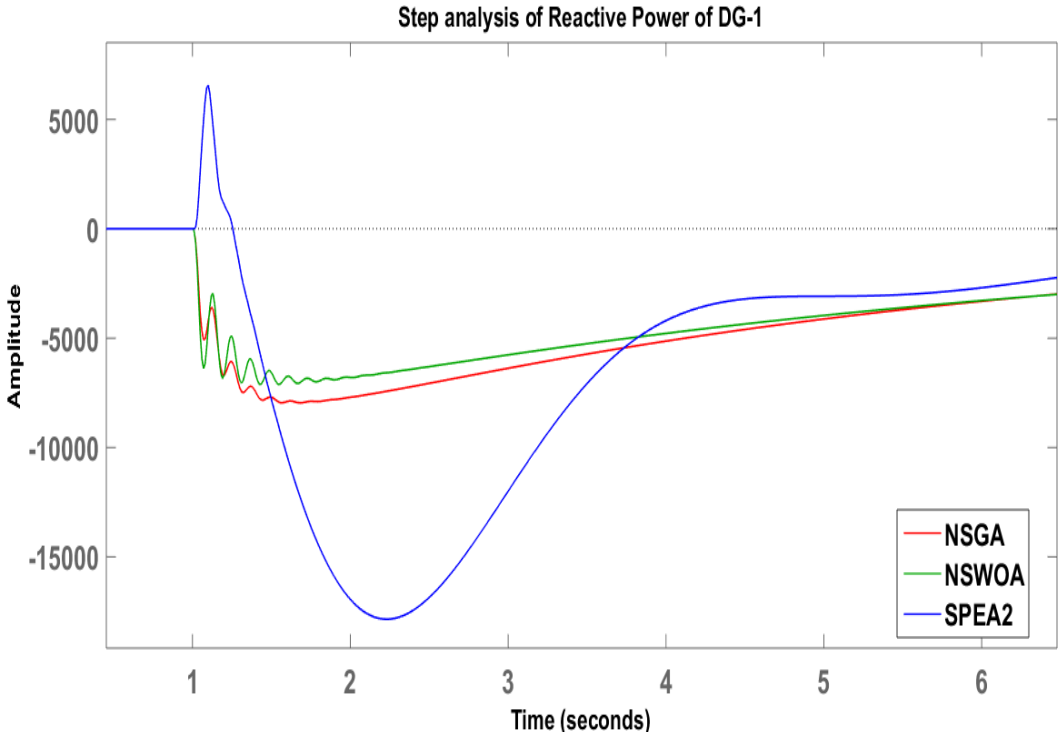


(b)Real Power of DG-2 (Zoomed View)

Figure 4.4: Step Analysis of Real Power of DG-2

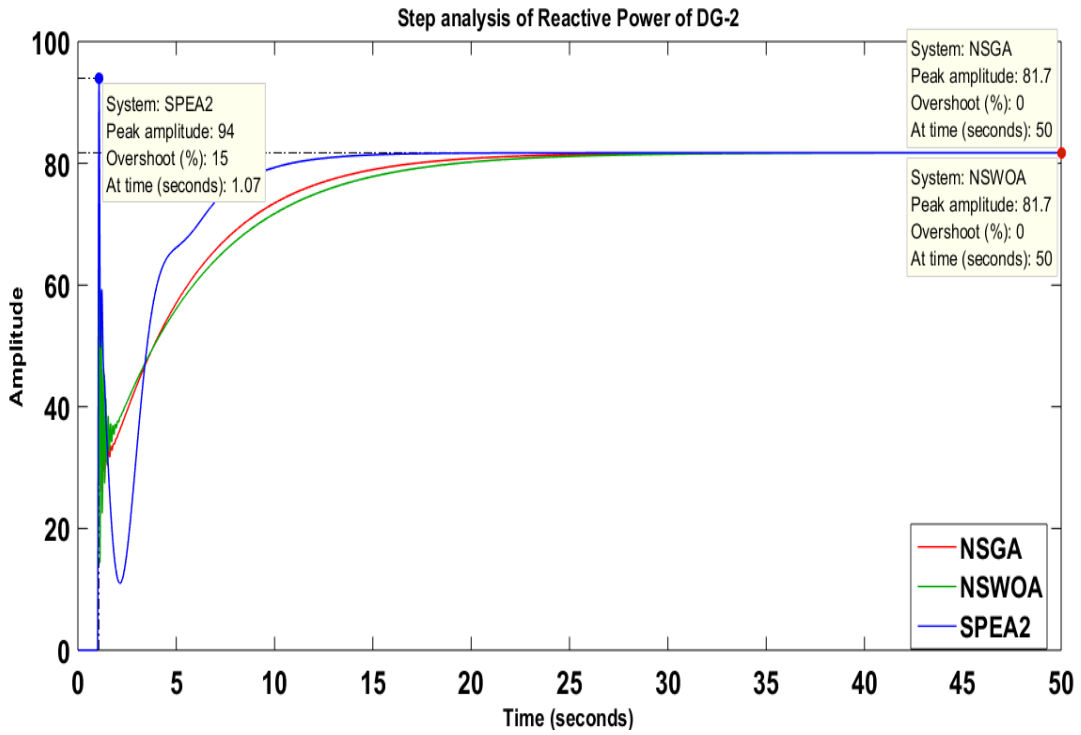


(a) Reactive Power of DG-1

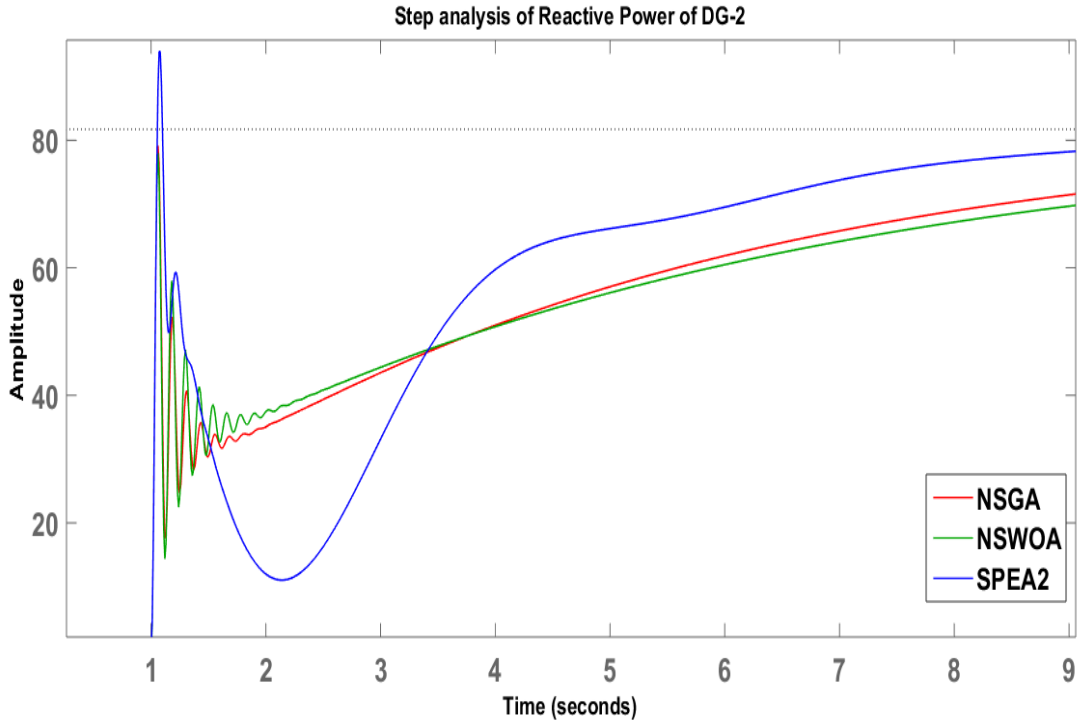


(b) Reactive Power of DG-1 (Zoomed View)

Figure 4.5: Step Analysis of Reactive Power of DG-1

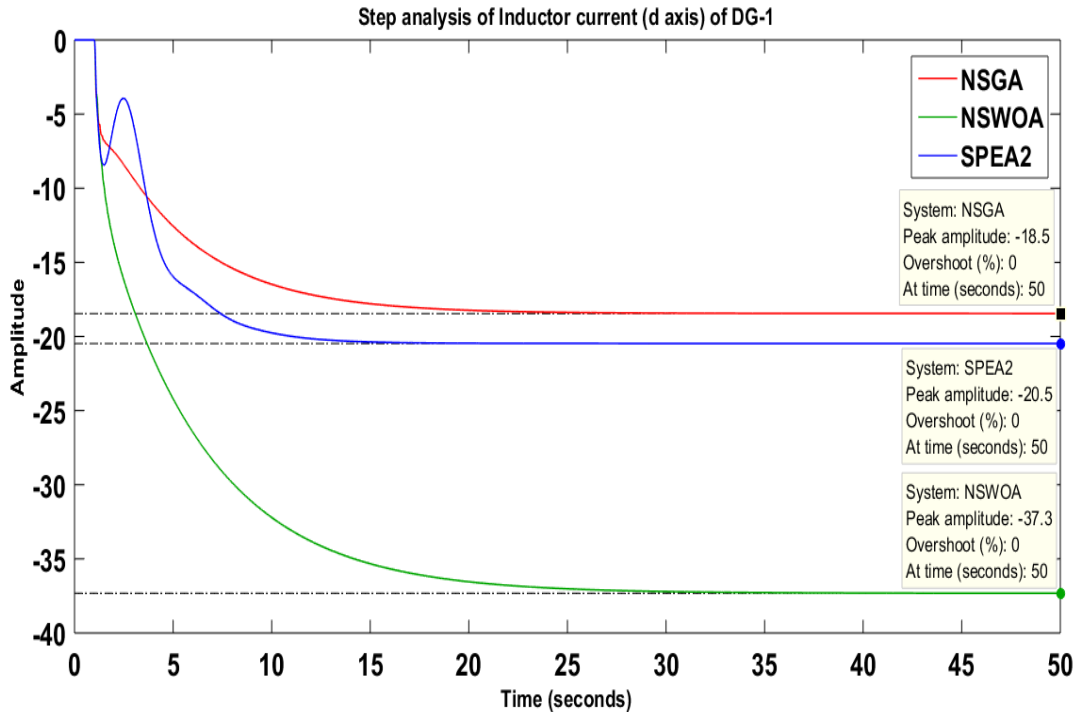


(a) Reactive Power of DG-2

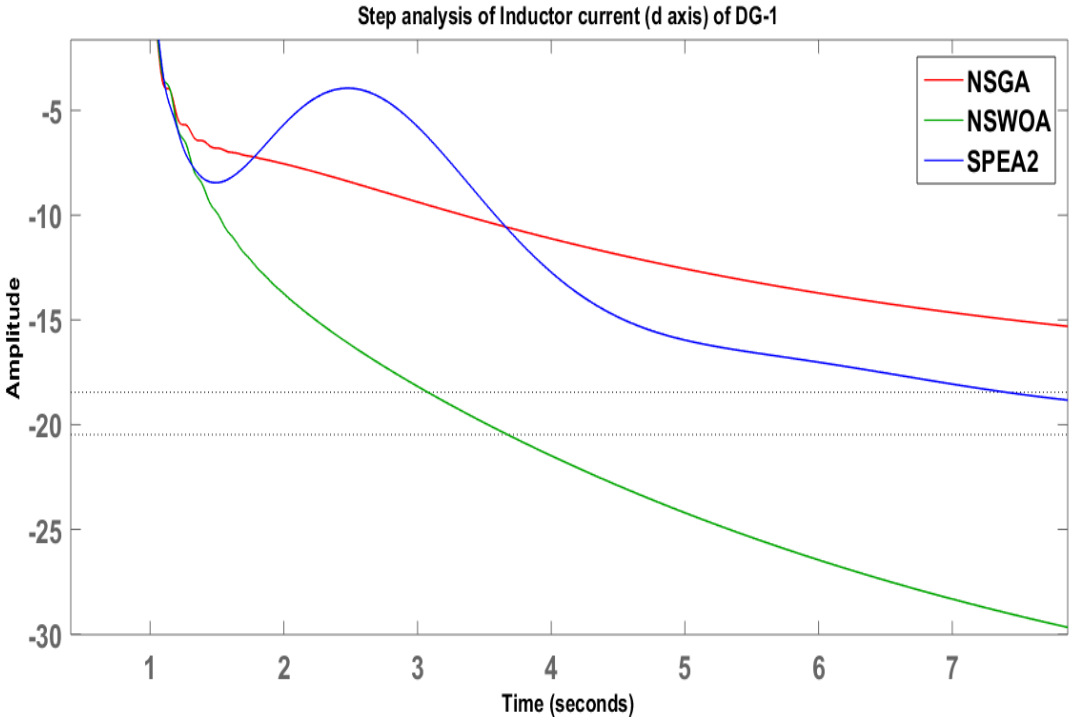


(b) Reactive Power of DG-2 (Zoomed View)

Figure 4.6: Step Analysis of Reactive Power of DG-2

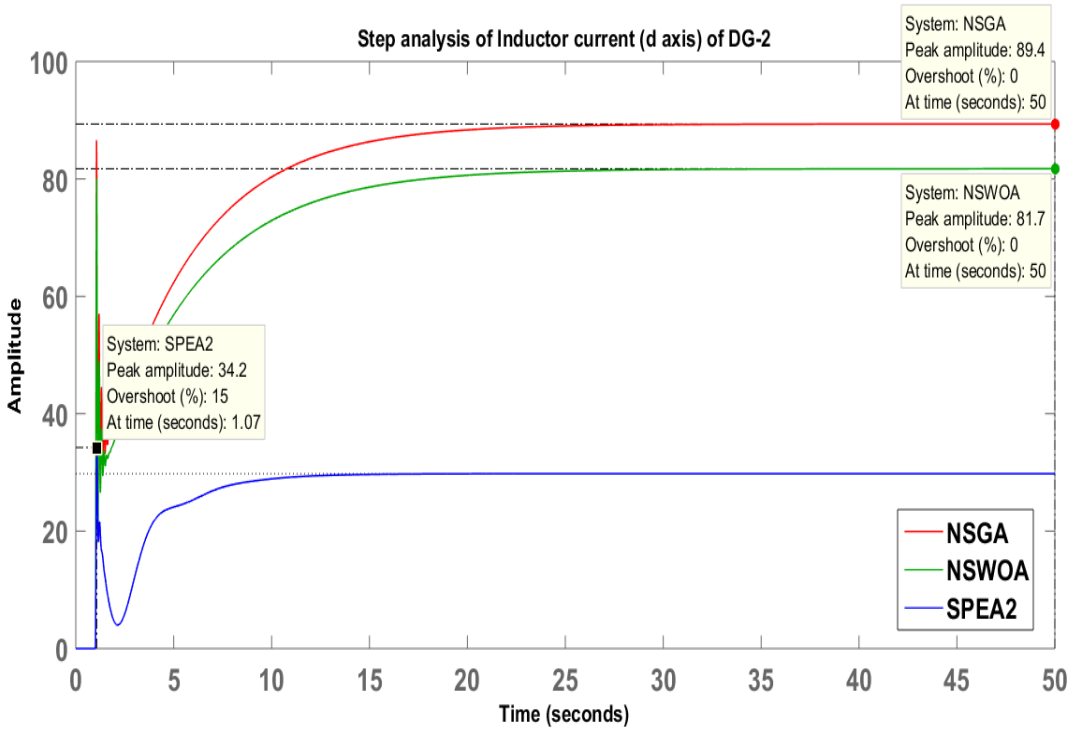


(a) Inductor Current (d axis) of DG-1

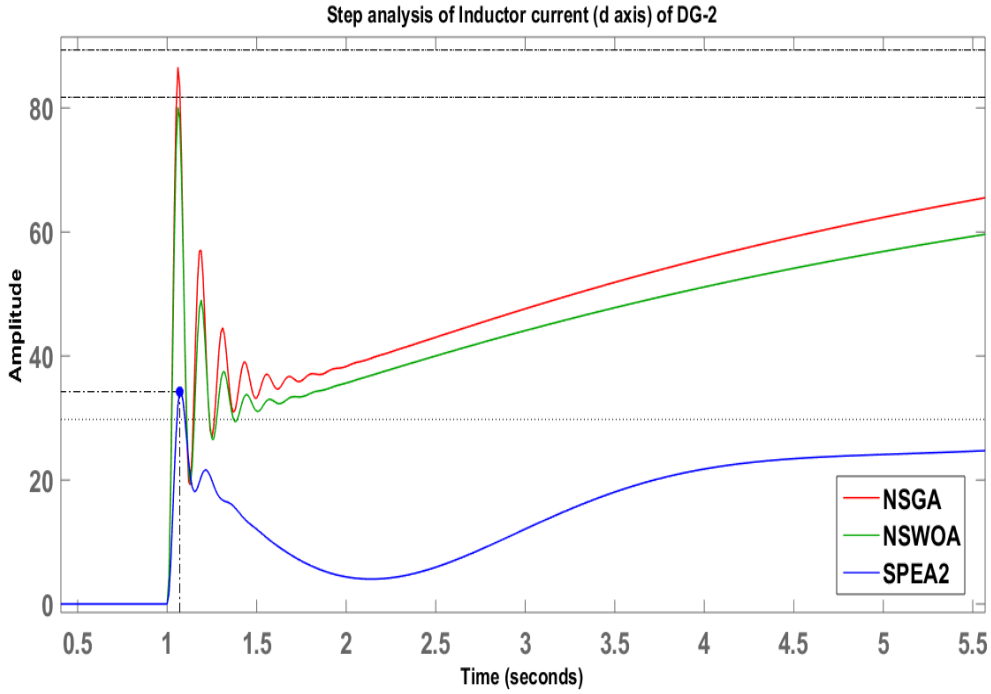


(b) Inductor Current (d axis) of DG-1 (Zoomed View)

Figure 4.7: Step Analysis of Inductor Current (d axis) of DG-1

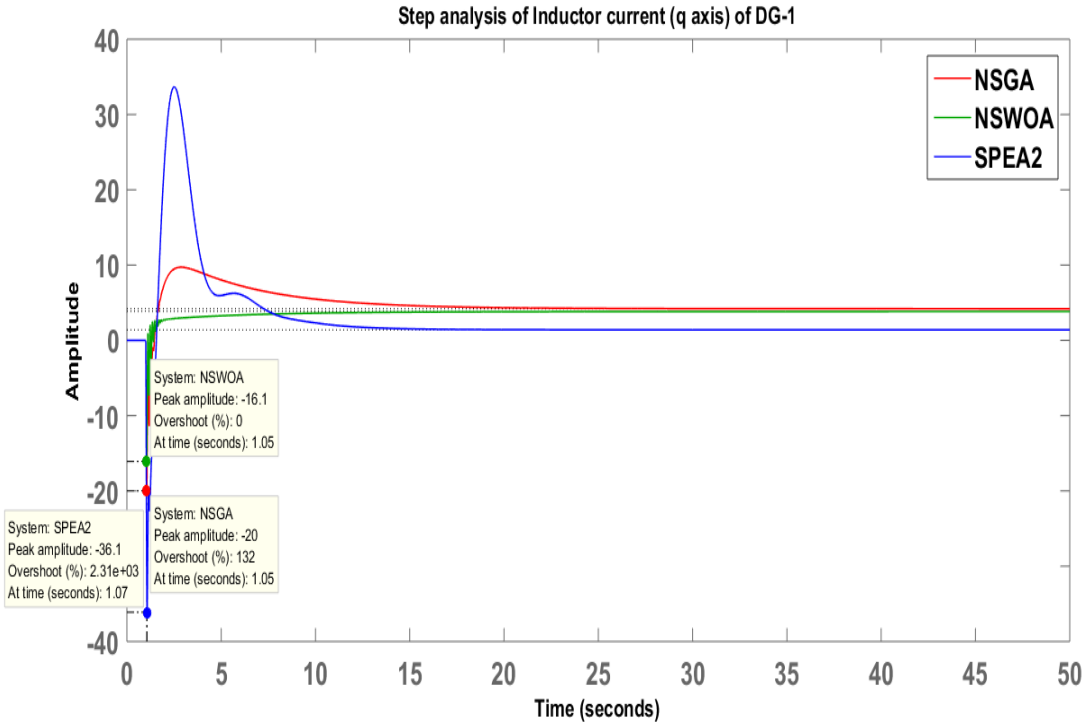


(a) Inductor Current (d axis) of DG-2

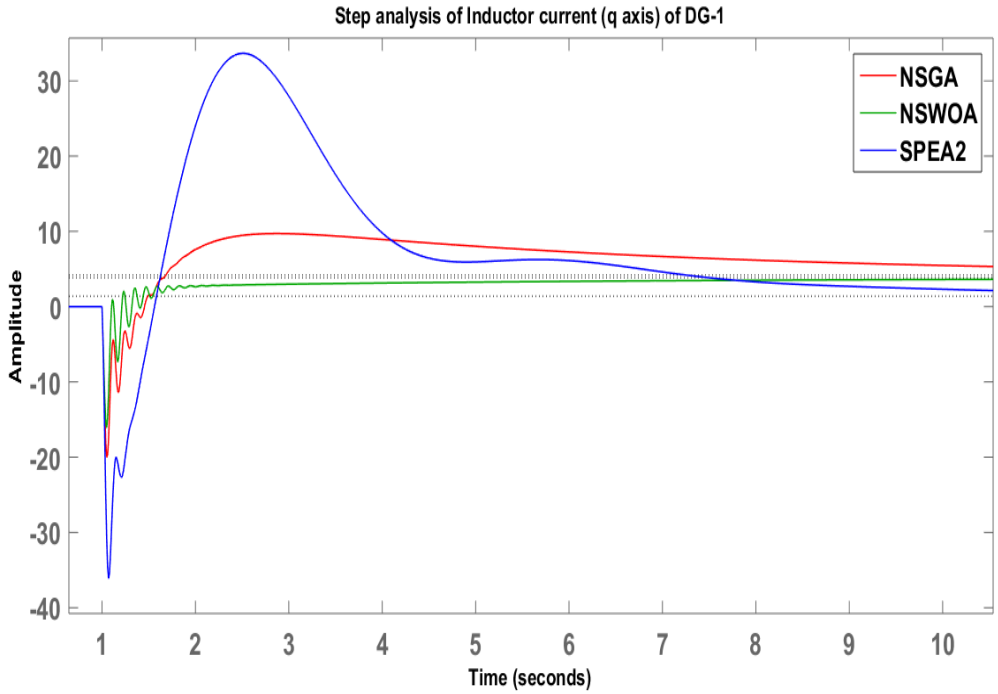


(b) Inductor Current (d axis) of DG-2 (Zoomed View)

Figure 4.8: Step Analysis of Inductor Current (d axis) of DG-2

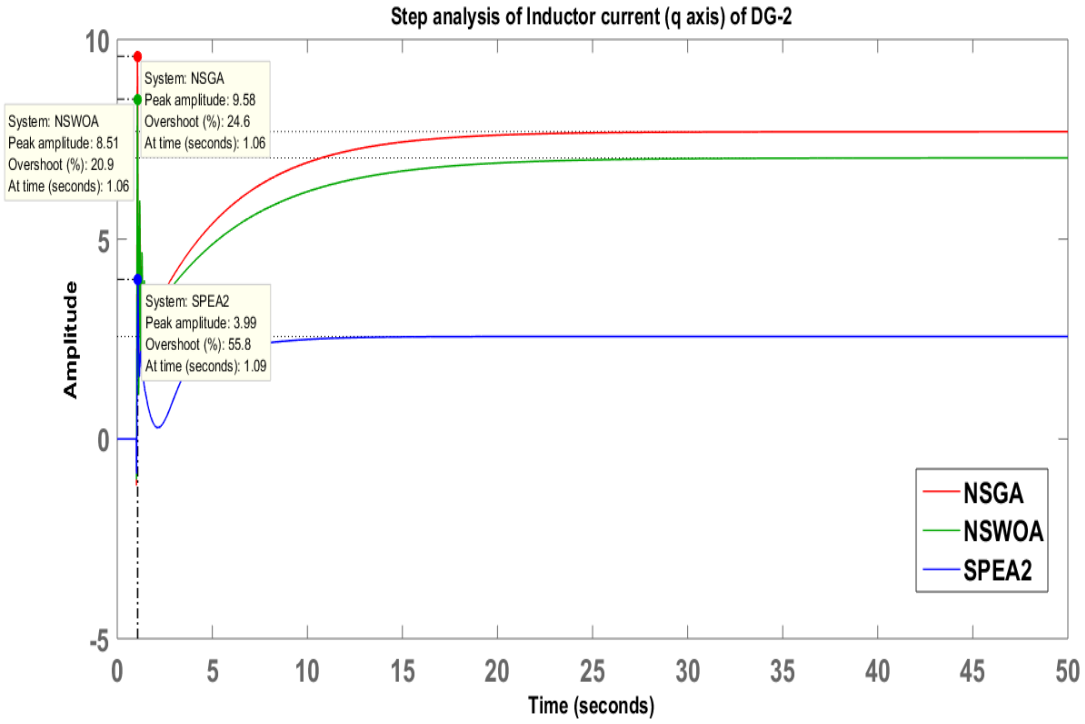


(a) Inductor Current (q axis) of DG-1

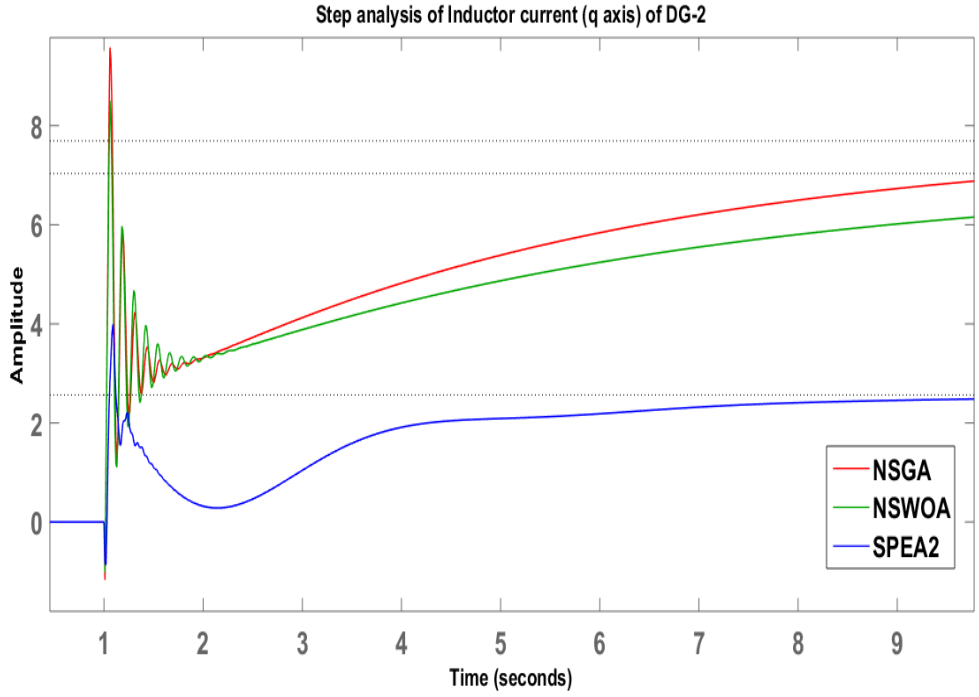


(b) Inductor Current (q axis) of DG-1 (Zoomed View)

Figure 4.9: Step Analysis of Inductor Current (q axis) of DG-1

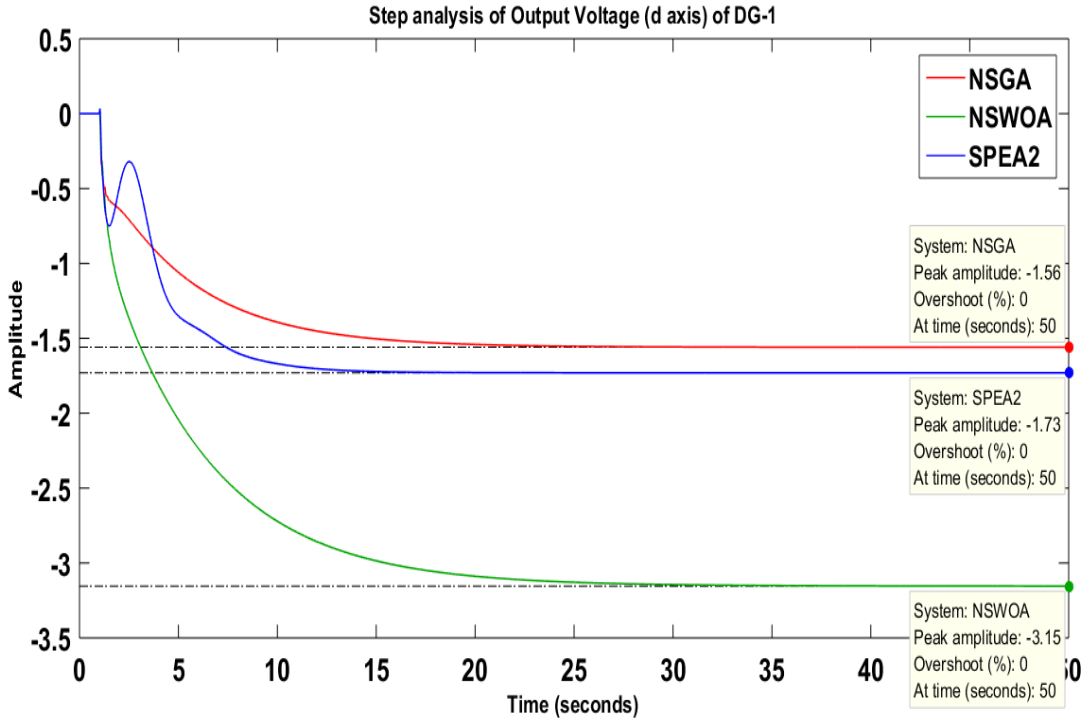


(a) Inductor Current (q axis) of DG-2

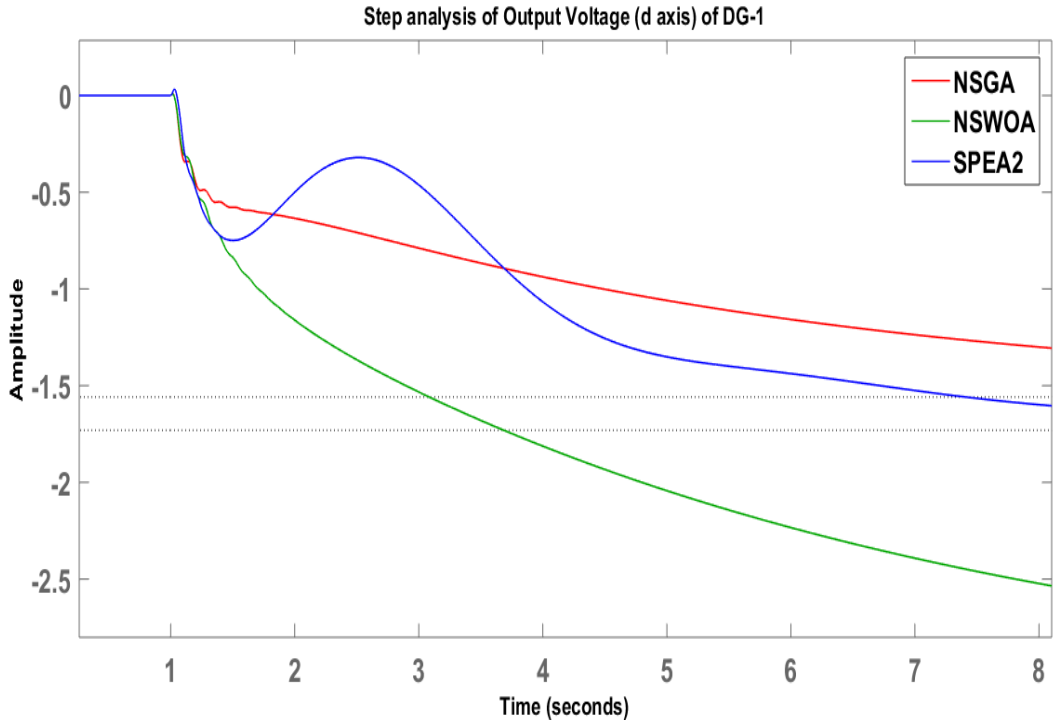


(b) Inductor Current (q axis) of DG-2 (Zoomed View)

Figure 4.10: Step Analysis of Inductor Current (q axis) of DG-2

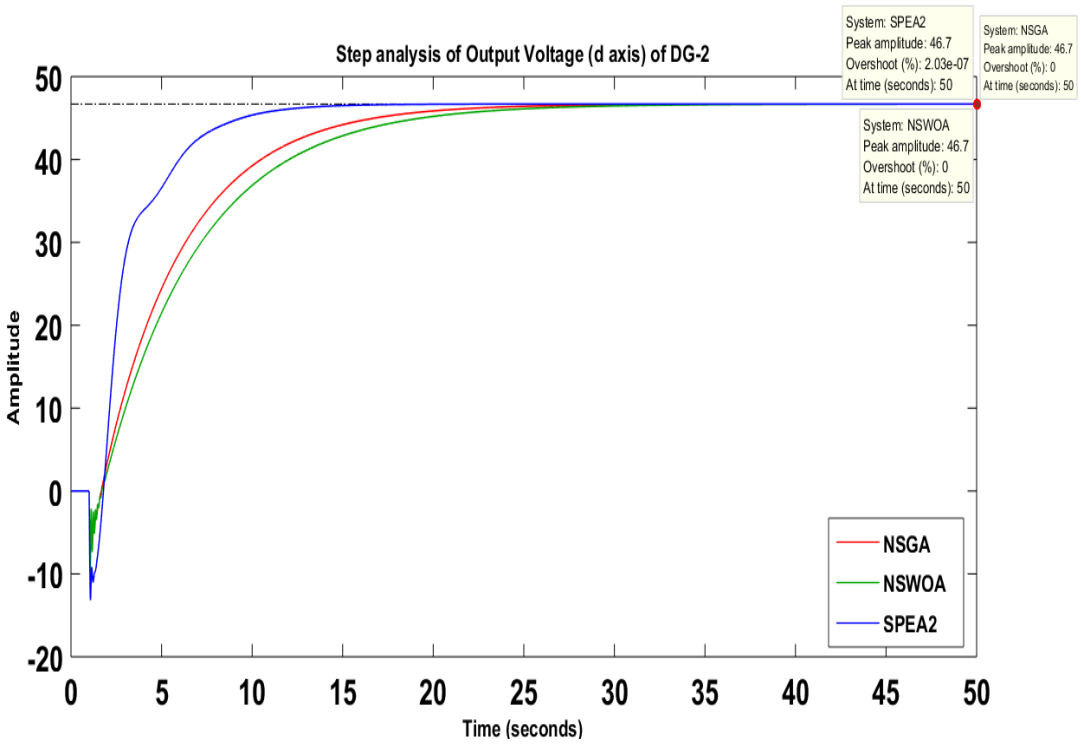


(a) Output Voltage (d axis) of DG-1

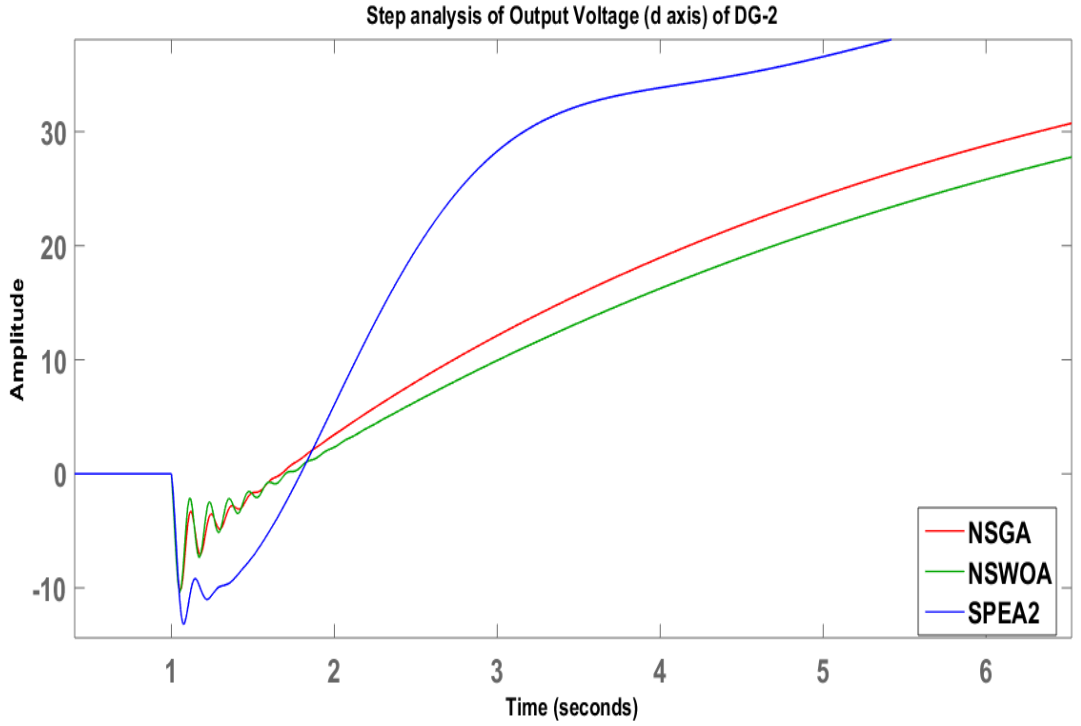


(b) Output Voltage (d axis) of DG-1 (Zoomed View)

Figure 4.11: Step Analysis of Output Voltage (d axis) of DG-1

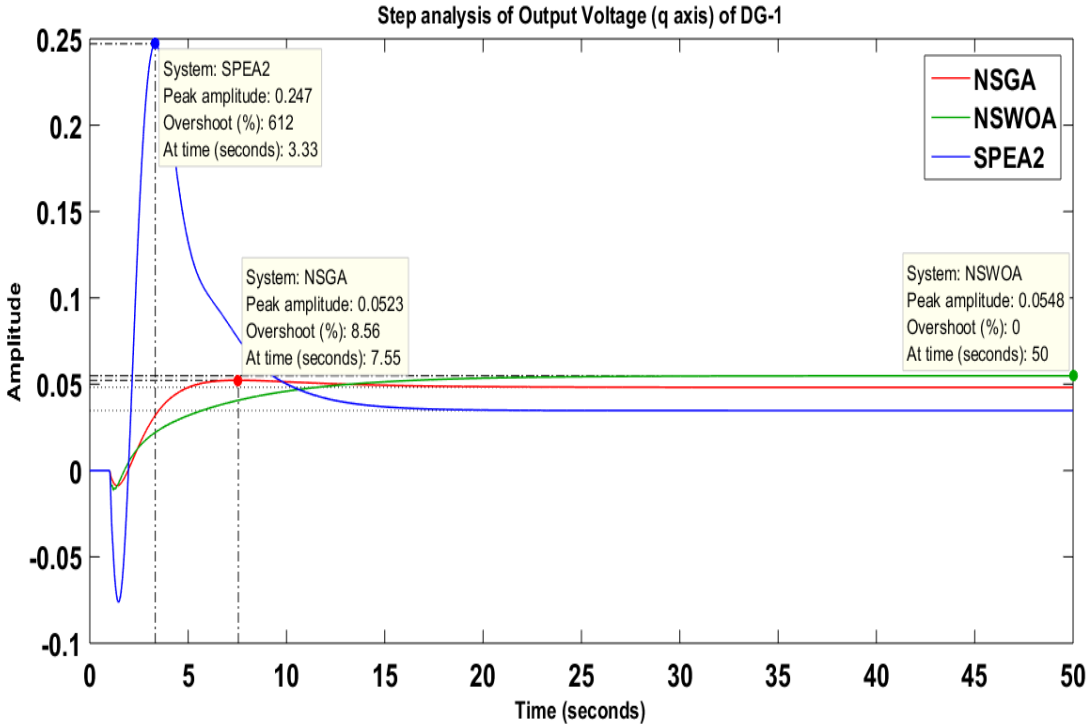


(a) Output Voltage (d axis) of DG-2

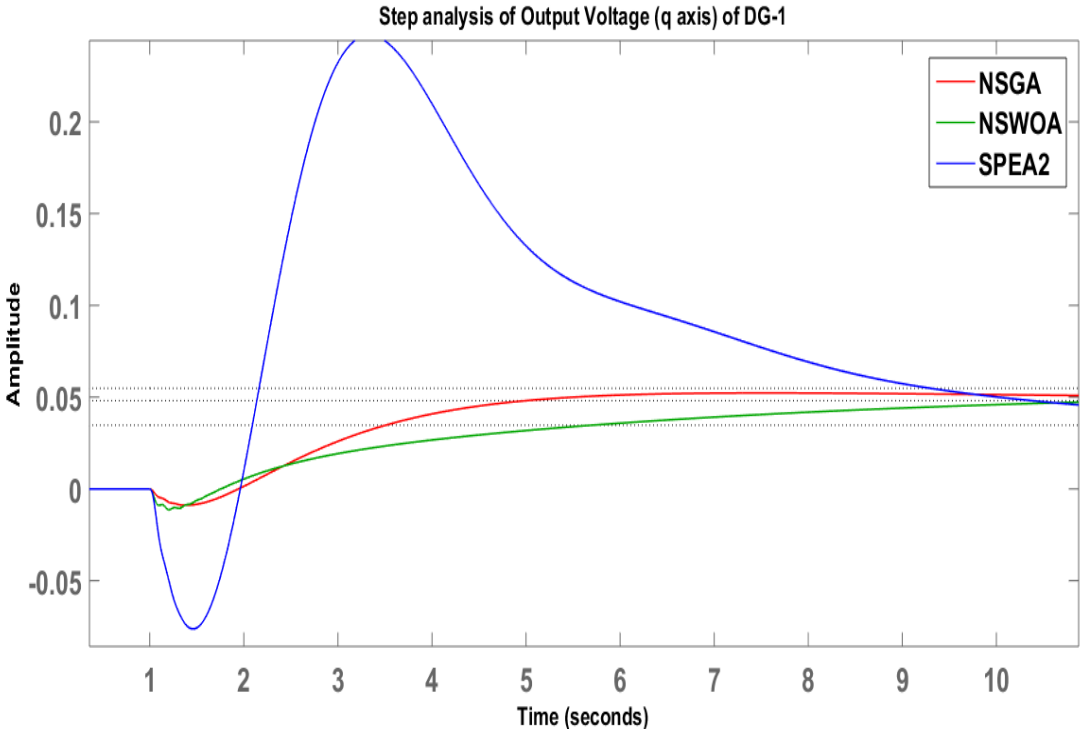


(b) Output Voltage (d axis) of DG-2 (Zoomed View)

Figure 4.12: Step Analysis of Output Voltage (d axis) of DG-2

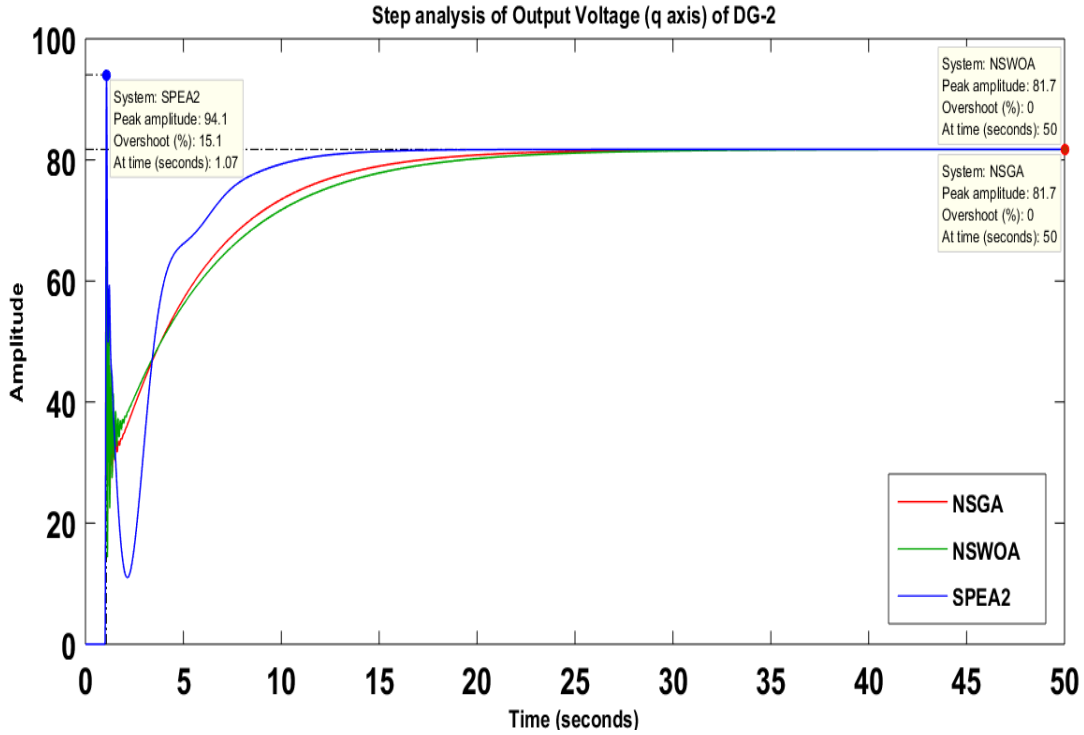


(a) Output Voltage (q axis) of DG-1

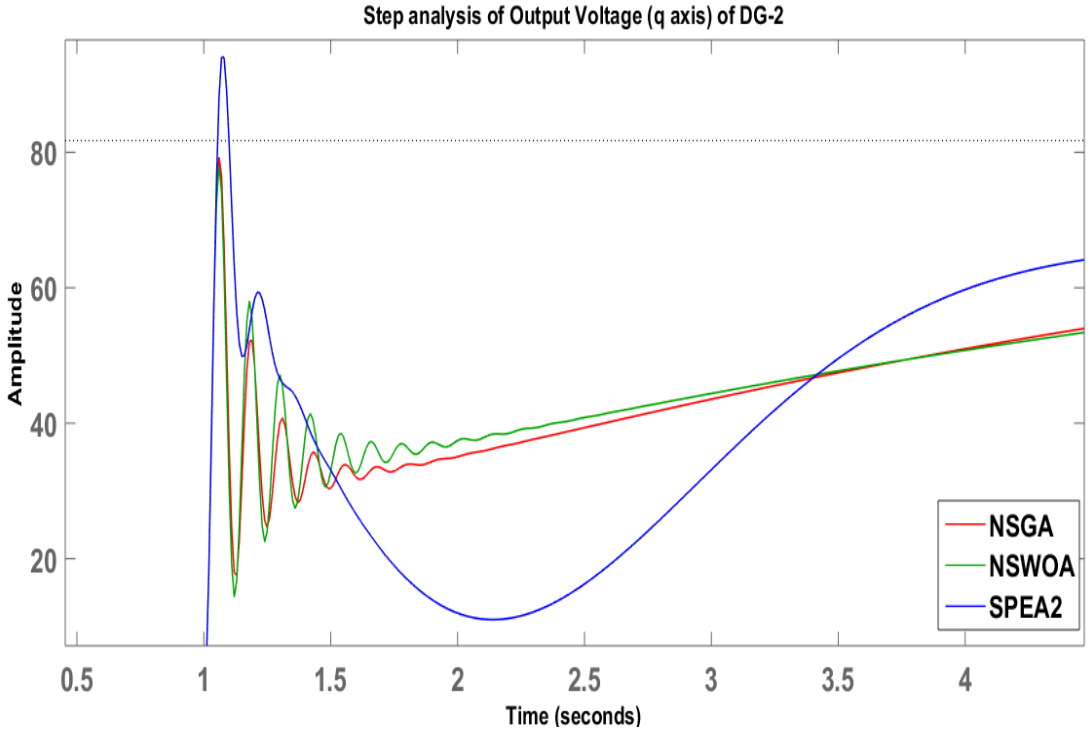


(b) Output Voltage (q axis) of DG-1 (Zoomed View)

Figure 4.13: Step Analysis of Output Voltage (q axis) of DG-1



(a) Output Voltage (q axis) of DG-2



(b) Output Voltage (q axis) of DG-2 (Zoomed View)

Figure 4.14: Step Analysis of Output Voltage (q axis) of DG-2

4.4 Statistical Tests

In order to ratify the competence of the aforementioned algorithms a statistical assessment was enacted using SPSS software where independent sample $t-test$ was carried out for determination of the uniqueness of each of the algorithms. In this test the parameters taken under consideration were total number of iterations needed by each of the algorithms to satisfy the objective functions, total time required for completion of the iteration and total summation of real part of eigen values for determination of the algorithm providing most negative eigen values. The data sets included here are for 30 independent runs from which the average data is being tabulated. $Levene's test for equality of variances$ [78] i.e. $F-test$ was also carried out alongside the $t-test$. $F-test$ is used for determination of the variances in the data sets of the algorithms. $F-test$ determines whether there exist any equal variances amongst the algorithm depending upon the condition that the $Sigma(p)$ value is greater than 0.05 or not. If the value is greater than 0.05 then equal variances between the data sets of the algorithms are assumed. Similarly, in $t-test$ assumption of null hypothesis (H_o) and alternative hypothesis H_1 are determined on the basis of $Sigma(2-tailed)(p)$ value. If the value is greater than 0.05 then null hypothesis (H_o) needs to be considered which indicates the mean values of the algorithms' group data are equal. But if $Sigma(2-tailed)(p)$ is less than 0.05 then alternative hypothesis H_1 can be taken into consideration which indicates mean value of group data is not equal. This independent sample $t-test$ basically helps in determining the homogeneity of the algorithms.

Table 4.3 shows the $F-test$ and $t-test$ analysis of NSWOA and NSGA-II. It is observable from the above mentioned table that in $F-test$, $Sigma(p)$ value is greater than 0.05 for total time required for completion of iteration only but for all other cases it is less than 0.05. So, it concludes that the population variance of NSWOA and NSGA-II are not spread equally. Now, for equality means $t-test$ it is seen from Table 4.3 that equality means exist for total summation of eigen values only where $Sigma(2-tailed)(p)$ is higher than 0.05. For total number of iterations and total summation of real eigen values it is less than 0.05 which indicates that null hypothesis is rejected and the difference in means between NSGA-II and NSWOA is statistically significant.

Similarly Table 4.4 shows the analysis of NSWOA with SPEA. From Table 4.4 it is seen that $Sigma(p)$ value is less than 0.05 for total summation of eigen values though for total number iterations it is slightly greater than 0.05. Moreover, from $t-test$ it is

observed that $Sigma(2-tailed)(p)$ is significantly lower than 0.05 for total number of iterations and total summation of real eigen values. Thus from Table 4.4 it can be analyzed that there exist significant difference in means between NSWOA and SPEA statistically.

Table 4.5 shows the group data of NSGA-II, SPEA and NSWOA from which it is clearly visible that for NSWOA, even though total summation of eigen values is least negative but NSWOA takes significant lesser number of iterations and less time of convergence compared to NSGA-II and SPEA. Eigen value analysis proved the ability of NSWOA in obtaining a much stable system which ensures that despite of having least negative total summation of eigen values, NSWOA is much efficient and much faster in obtaining optimized values for the controller.

Table 4.3: NSGA-II and NSWOA comparison on the basis of F-test and t-test

Parameters	**Levene's Test for Equality of Variances (F- test)**		**Equality Means Test (t - test)**			
	F	*Sig (p)*	*t*	*df*	*Sig. (2- tailed)*	*Mean Difference*
Total Number of iterations	5.069276	.028	4.040	52.273	.000	7.50000
Total summation of Eigen Values (Real)	8.947109	.004	-1.795	42.743	.078	-45741.56900
Total time required	2.758497	.102	3.473	58	.001	3.80446

Table 4.4: SPEA and NSWOA comparison on the basis of F-test and t-test

Parameters	**Levene's Test for Equality of Variances (F- test)**		**Equality Means Test (t - test)**			
	F	*Sig(p)*	*t*	*df*	*Sig. (2- tailed)*	*Mean Difference*
Total Number of iterations	3.092900	.084	2.287	58	.029	2.56667
Total summation of Eigen Values (Real)	8.527837	.005	-.942	42.896	.350	-24030.48633
Total time required	1.622513	.208	3.603	58	.001	2.74787

Table 4.5: Group Statistical Data of NSGA-II, SPEA and NSWOA

Parameters	**Algorithm**	**Mean**	**Standard Deviation**	**Standard Error Mean**
Total Number of iterations	NSGA-II	11.5333	8.29513	1.51448
	SPEA	10.1333	4.24047	.77420
	NSWOA	4.0333	5.88091	1.07370
Total Summation of Eigen Values (Real)	NSGA-II	-12734141.7097	62613.54069	11431.61621
	SPEA	-12709722.4533	60017.17312	10957.58652
	NSWOA	-12688400.1407	124729.01128	22772.29768
Total Time Required(sec)	NSGA-II	6.7245	4.57376	.83505
	SPEA	4.2842	1.88320	.34382
	NSWOA	2.9201	3.88320	.70897

4.5 Summary

In this chapter at first the eigen value analysis of NSWOA was shown which proved the ability of NSWOA in optimizing the system and ensured stabilized operation of the system. In the later part time domain simulation analysis and statistical test was performed to compare the performance of NSWOA with existing NSGA-II and SPEA algorithms. Results suggest than NSWOA has better damping performance compared to NSGA-II and SPEA as well as the convergence speed of NSWOA is better than NSGA-II and SPEA as it takes lesser number of iteration and less time to converge to the best solution.

Chapter 5

Performance Analysis of NSFA

This chapter discusses about performance analysis of another proposed hybridized optimization algorithm namely: NSFA. The detailed study is carried out using the mathematical model of the microgrid developed in Chapter 2. MATLAB software is used here to develop the aforementioned mathematical model. The performance analysis is subdivided into three parts namely - (i) Eigen value Analysis, (ii) Time Domain Simulation and (iii) Statistical Tests. Eigen value analysis gives an idea about the ability of the NSFA algorithm in providing stability to the system by optimizing the required parameters. Later on, time domain simulation and statistical test were performed in order to carry out a comparative study between NSFA and NSGA-II.

5.1 Simulation Setup

The simulating parameters for all the optimizing algorithms namely: NSFA and NSGA-II are kept same as like as in Chapter 4 due to the comparative study that is to be made amongst these algorithms in order to compare their performance. The initial population size for NSFA and NSGA-II are taken as 50 and maximum number of iterations for each case is 100. As randomness is the common behaviour of these algorithms, so each algorithm was run 30 times to find the average performance of each of the algorithms and compare their performance.

The simulating parameters of NSGA-II is kept as like as mentioned in section 4.1 of Chapter 4. But for NSFA, there are several parameters to be defined. For NSFA, the light absorption co-efficient is taken as 1, base value of attraction co-efficient is taken as 2, mutation co-efficient is 0.2 and mutation co-efficient damping ration is taken as 0.98.

5.2 Eigen Value Analysis

At first eigen value analysis was carried out for determination of the ability of the proposed algorithm in obtaining stability of the proposed system. Figure 4.1 mentioned in Chapter 5 showed the condition of the eigen values of the system before optimization, where it can be seen that there exist some eigen values at the right half plane i.e. on the positive axis which proved that the system is not stable. Now using the proposed NSFA algorithm, optimization of the system is performed and from the Figure 5.1 it is noticeable that the system has become stable as the unstable eigen values have moved from positive axis to negative axis i.e. on the left half of s plane. This indicates the ability of NSFA algorithm in obtaining a stable system by optimizing the controller parameters. The optimized values of the controller parameters are $K_{pv1} = 60.4500$, $K_{iv1} = 155.1168$, $K_{pc1} = 354.7484$, $K_{ic1} = 490.0740$, $K_{pv2} = 100.7187$, $K_{iv2} = 439.9541$, $K_{pc2} = 499.6512$, and $K_{ic2} = 436.8814$.

Table 5.1 shows the detailed values of the eigen values of the system before optimization as well as after optimization of the system using NSFA. From the table it is seen that eigen value of the states $\Delta\varphi_{q2}$, $\Delta\gamma_{d2}$, Δi_{oq2} and Δi_{LineD} which were previously lying on the right half of s plane i.e on the right side of imaginary axis have now moved to the left half of s plane i.e on the left side of imaginary axis. Moreover, the eigen values of the states Δi_{LoadQ}, Δi_{Qs}, Δi_{Ds}, Δi_{Qr}, Δi_{Dr} and Δs_l which were closer to the imaginary axis have now moved further away from the imaginary axis on the left hand side.

This proves the ability of the proposed NSFA algorithm to stabilize an unstable system with optimized control parameter settings.

Table 5.1: Eigen Value Analysis

Index	States	Eigen Value of the System Before Optimization	Eigen Value of the System After Using NSFA Optimization
1	$\Delta\delta_1$	-2909410.542 +12209388.904i	-2909410.529 + 12209388.805i
2	ΔP_1	-2909410.542 - 12209388.904i	-2909410.529 - 12209388.805i
3	ΔQ_1	-3261123.439 + 8309127.704i	-3261123.517 + 8309127.438i
4	$\Delta\varphi_{d1}$	-3261123.439 - 8309127.704i	-3261123.517 - 8309127.438i
5	$\Delta\varphi_{q1}$	-55.194 + 45315.607i	-59787.105 + 486045.795i
6	$\Delta\gamma_{d1}$	-55.194 - 45315.607i	-59787.105 - 486045.795i
7	$\Delta\gamma_{q1}$	-352.417 + 44476.812i	-59850.414 + 486371.709i
8	Δi_{ld1}	-352.417 - 44476.812i	-59850.414 - 486371.709i
9	Δi_{lq1}	-4107.969 + 31524.405i	-42180.718 + 316294.548i
10	Δv_{od1}	-4107.969 - 31524.405i	-42180.718 - 316294.548i
11	Δv_{oq1}	-5629.972 + 29764.098i	-42190.874 + 316724.711i
12	Δi_{od1}	-5629.972 - 29764.098i	-42190.874 - 316724.711i
13	Δi_{oq1}	-8720.541 + 8365.113i	-8158.994 + 20798.414i
14	$\Delta\delta_2$	-8720.541 - 8365.113i	-8158.994 - 20798.414i
15	ΔP_2	-6328.127 + 8624.675i	-12454.773 + 18925.811i
16	ΔQ_2	-6328.127 - 8624.675i	-12454.773 - 18925.811ii
17	$\Delta\varphi_{d2}$	-1291.429 + 0i	-2178.097 + 516.632i
18	$\Delta\varphi_{q2}$	213.426 + 784.754i	-2178.097 - 516.632i
19	$\Delta\gamma_{d2}$	213.426 - 784.754i	-23.436 + 200.800i
20	$\Delta\gamma_{q2}$	-81.284 + 376.280i	-23.436 - 200.800i
21	Δi_{ld2}	-81.284 - 376.280i	-137.294 + 0i
22	Δi_{lq2}	-162.677 + 0i	-7.810 + 49.700i
23	Δv_{od2}	-70.227 + 1.578i	-7.810 - 49.700i
24	Δv_{oq2}	-70.227 - 1.578i	-70.767 + 0i
25	Δi_{od2}	-67.497 + 0i	-67.341 + 2.142i
26	Δi_{oq2}	22.647 + 0i	-67.341 - 2.142i
27	Δi_{LineD}	1.315 + 0i	-7.426 + 0i
28	Δi_{LineQ}	-2.394 + 0i	-4.239 + 0i
29	Δi_{LoadD}	-2.393 + 0i	-2.637 + 0.2744i
30	Δi_{LoadQ}	-0.018 + 0.045i	-2.637 - 0.2744i
31	Δi_{Qs}	-0.018 - 0.045i	-0.207 + 0i
32	Δi_{Ds}	-0.021 + 0i	-1.381 + 1.122e-05i
33	Δi_{Qr}	-0.329 + 0i	-1.381 - 1.122e-05i
34	Δi_{Dr}	-0.200 + 0i	-0.873 + 0i
35	Δs_l	-0.202 + 0i	-0.874 + 0i

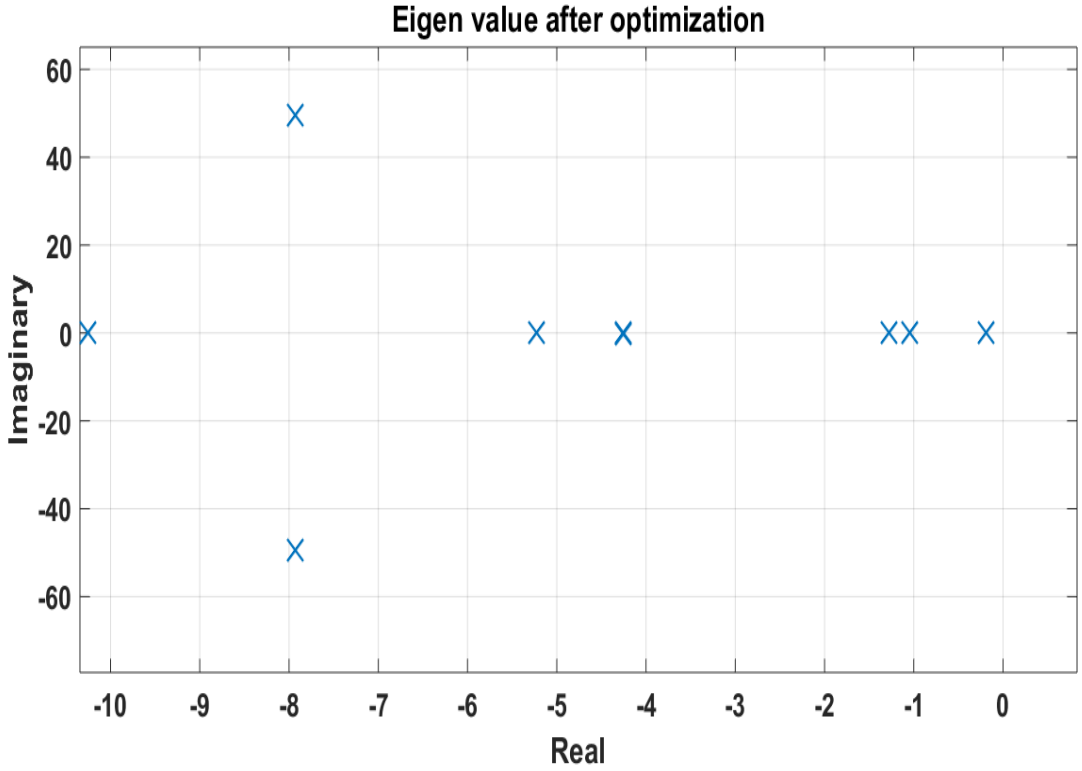


Figure 5.1: Eigen Value of the system after optimization

5.3 Time Domain Simulation Analysis

The results obtained using NSFA optimization techniques is compared with NSGA-II optimization technique to determine their differences in performance. The overshoot and oscillation frequency of real power, reactive power, inductor current and output current of DG-1 and DG-2 for both NSFA and NSGA-II algorithms have been represented in a tabulated form in Table 5.2. Along with the tabulated data, the unit step response of real power, reactive power, inductor current $(d-q)$ and output current $(d-q)$ are shown in Figure 5.2 - Figure 5.13.

In Figure 5.2 and Figure 5.3 it is seen that for real power in both DG-1 and DG-2; zero overshoot is given by NSFA whereas NSGA-II provides overshoot of 0.658% for DG-1 and $2.89e \ -7\%$ for DG-2. Again, for DG-1 both NSFA and NSGA-II have similar oscillation frequency but for DG-2, NSGA-II has oscillation frequency of $3530.539\ Hz$ whereas NSFA has $3012.1365\ Hz$ which is less than that of NSGA-II.

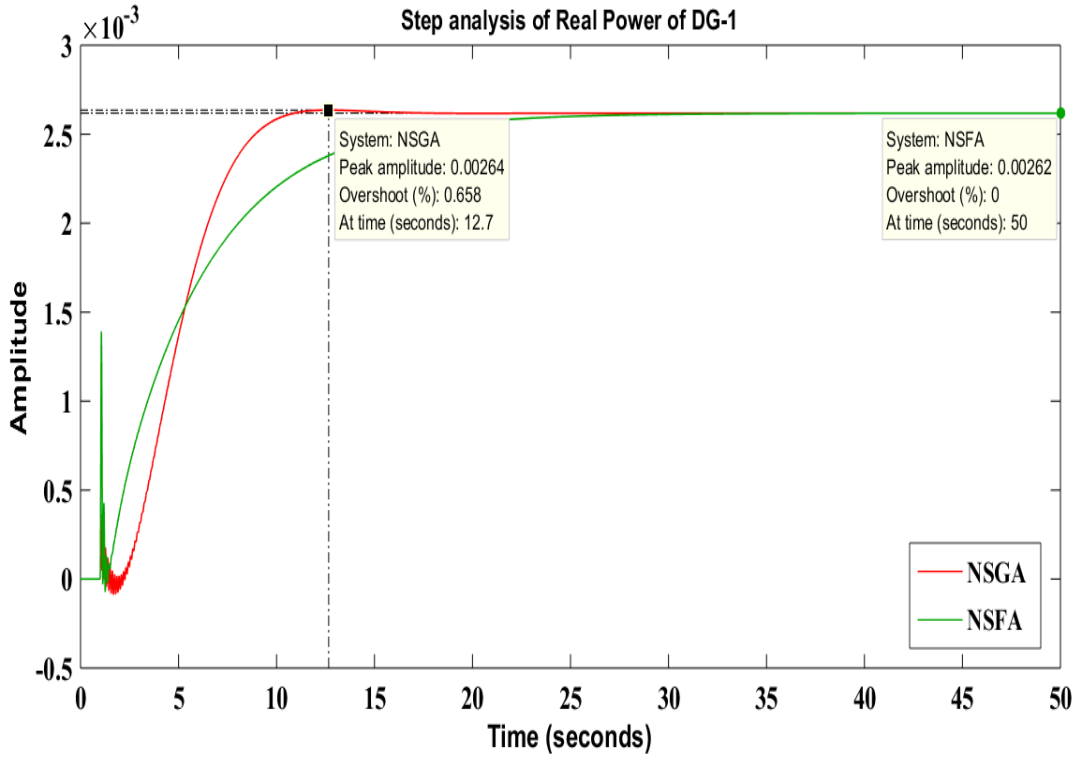
From Figure 5.4, Figure 5.5 and Table 5.2 in case of reactive power it is seen that for DG-1 both NSGA-II and NSFA algorithm provides similar oscillation frequency. For DG-2, NSFA gives oscillation frequency of $3012.136\ Hz$ which is less

Table 5.2: NSGA-II and NSFA comparison on the basis of overshoot and oscillation frequency

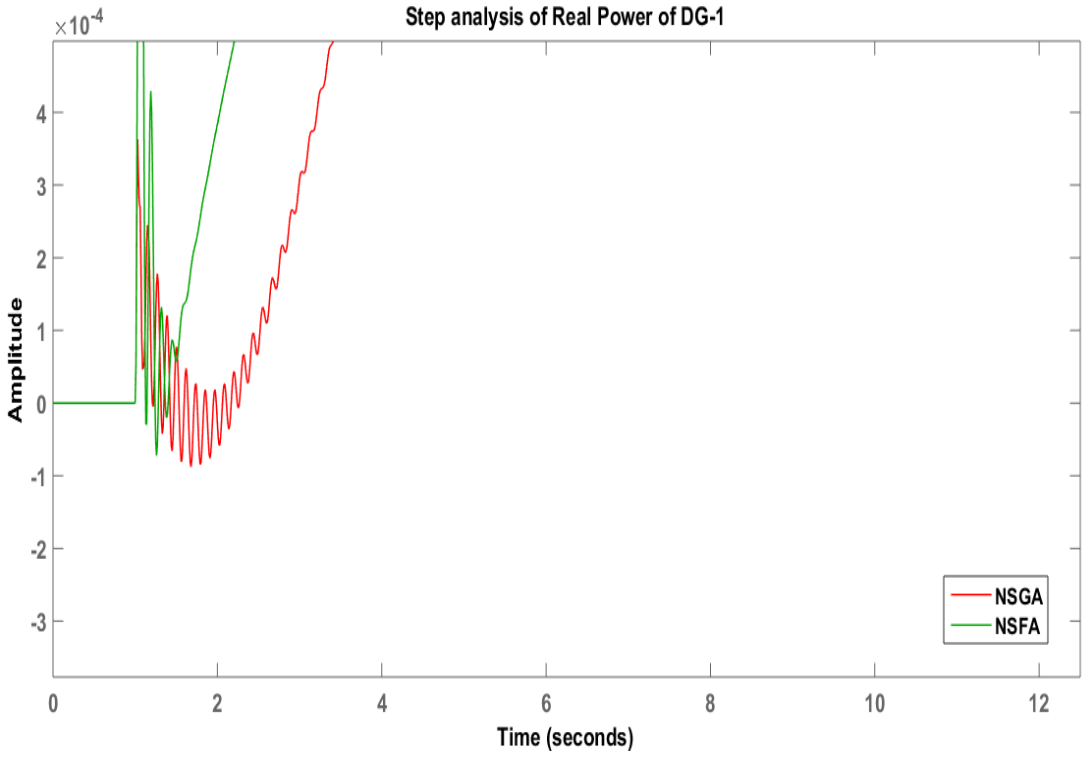
Criterion	Parameters	NSGA-II	NSFA
Real Power of DG-1	*Overshoot (%)*	0.658	0
	Oscillation Frequency (Hz)	1943184.60945872	1943184.58052368
Real Power of DG-2	*Overshoot (%)*	2.89e -7	0
	Oscillation Frequency (Hz)	3530.53971016327	3012.13652730558
Reactive Power of DG-1	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	1322438.80856331	1322438.70465282
Reactive Power of DG-2	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	3530.53971016327	3012.13652730558
Inductor Current of DG-1 (d-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	66757.6167391984	77408.4617388257
Inductor Current of DG-2 (d-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	0	0
Inductor Current of DG-1 (q-axis)	*Overshoot (%)*	132	63.5
	Oscillation Frequency (Hz)	46067.9986622605	50339.8409294205
Inductor Current of DG-2 (q-axis)	*Overshoot (%)*	24.6	27.5
	Oscillation Frequency (Hz)	8.62138639160466	7.91008749588845
Output Voltage of DG-1 (d-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	46067.9986622605	50339.8409294205
Output Voltage of DG-2 (d-axis)	*Overshoot (%)*	0.077	0
	Oscillation Frequency (Hz)	8.62138639160466	7.91008749588845
Output Voltage of DG-1 (q-axis)	*Overshoot (%)*	211	0
	Oscillation Frequency (Hz)	46139.9805662976	50408.3033958121
Output Voltage of DG-2 (q-axis)	*Overshoot (%)*	1.09e -8	0
	Oscillation Frequency (Hz)	0.358162462584608	0

than $3530.539\ Hz$ given by NSGA-II. Now for both the DGs there is zero overshoot provided bt NSGA-II and NSFA.

From Figure 5.6 and tabulated data it is observable that for inductor current of DG-1 ($d-axis$), both algorithms provides zero overshoot but NSFA has $77408.46\ Hz$ oscillation frequency which is more compared to $66757.616\ Hz$ of NSGA-II. Again, for DG-2 ($d-axis$) from Figure 5.7 it is seen that both NSFA and NSGA-II gives zero oscillation frequency and zero overshoot.

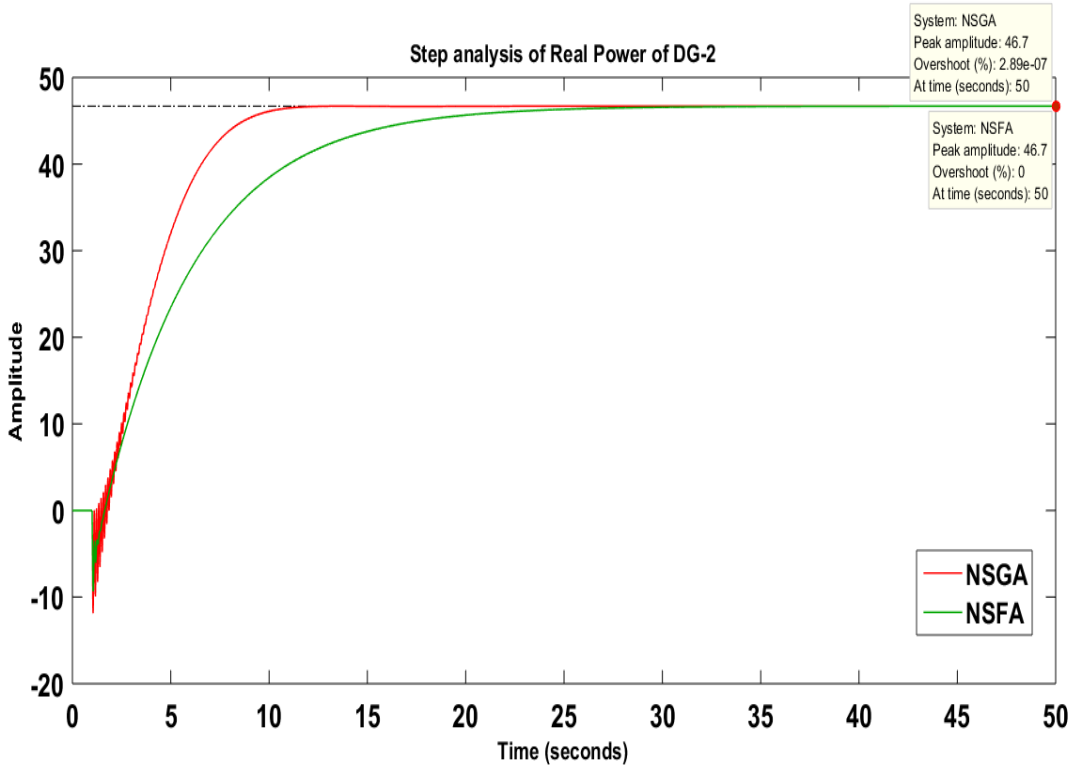


(a) Real Power of DG-1

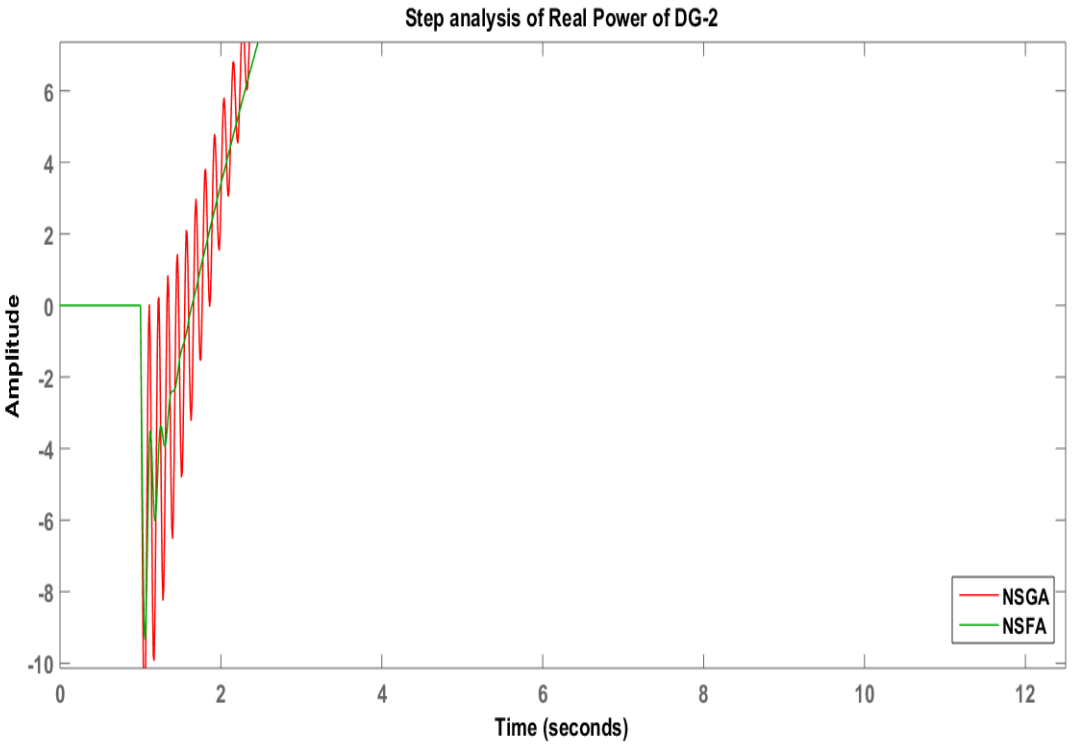


(b) Real Power of DG-1 (Zoomed View)

Figure 5.2: Step Analysis of Real Power of DG-1

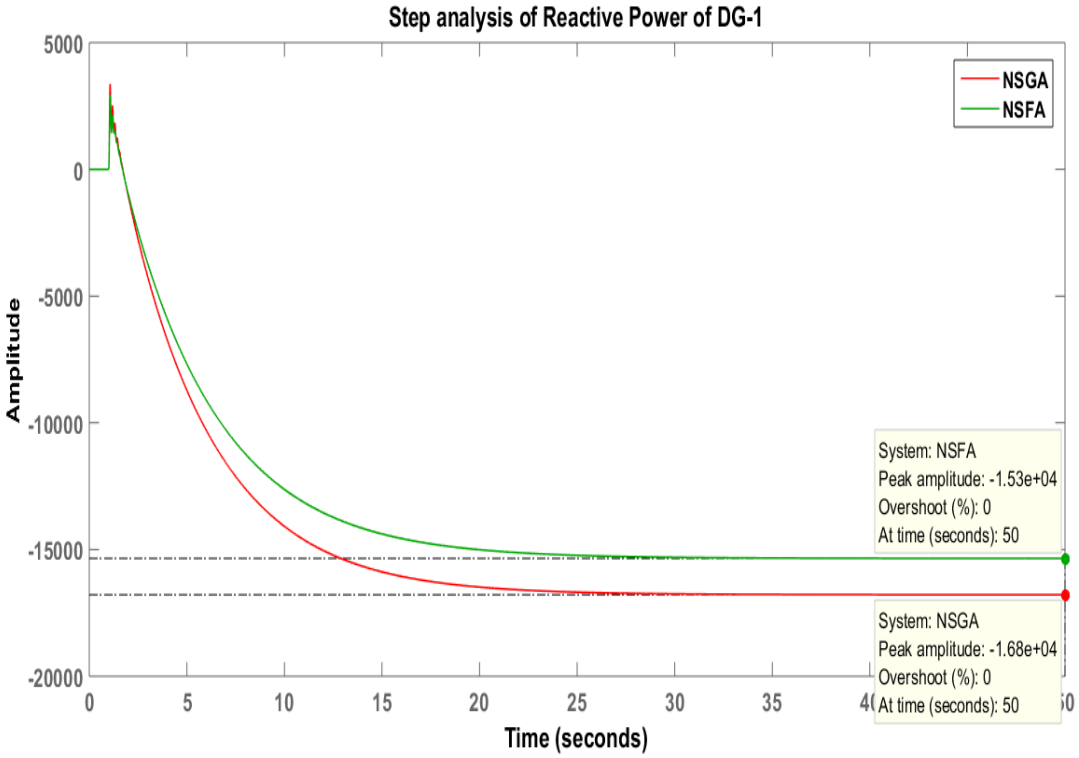


(a) Real Power of DG-2

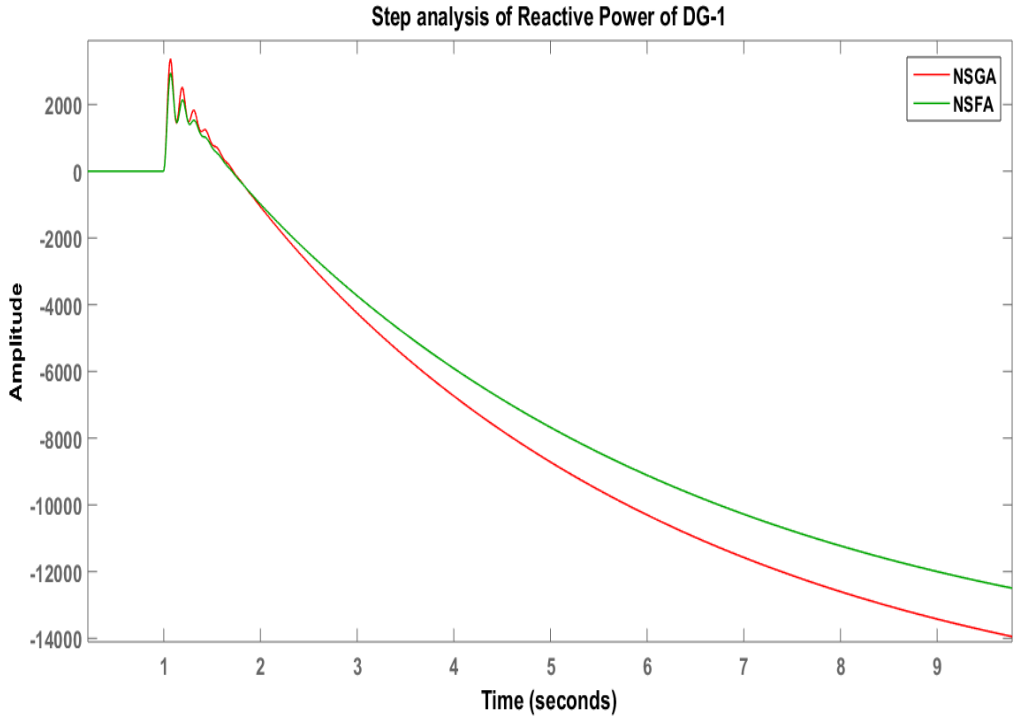


(b) Real Power of DG-2 (Zoomed View)

Figure 5.3: Step Analysis of Real Power of DG-2

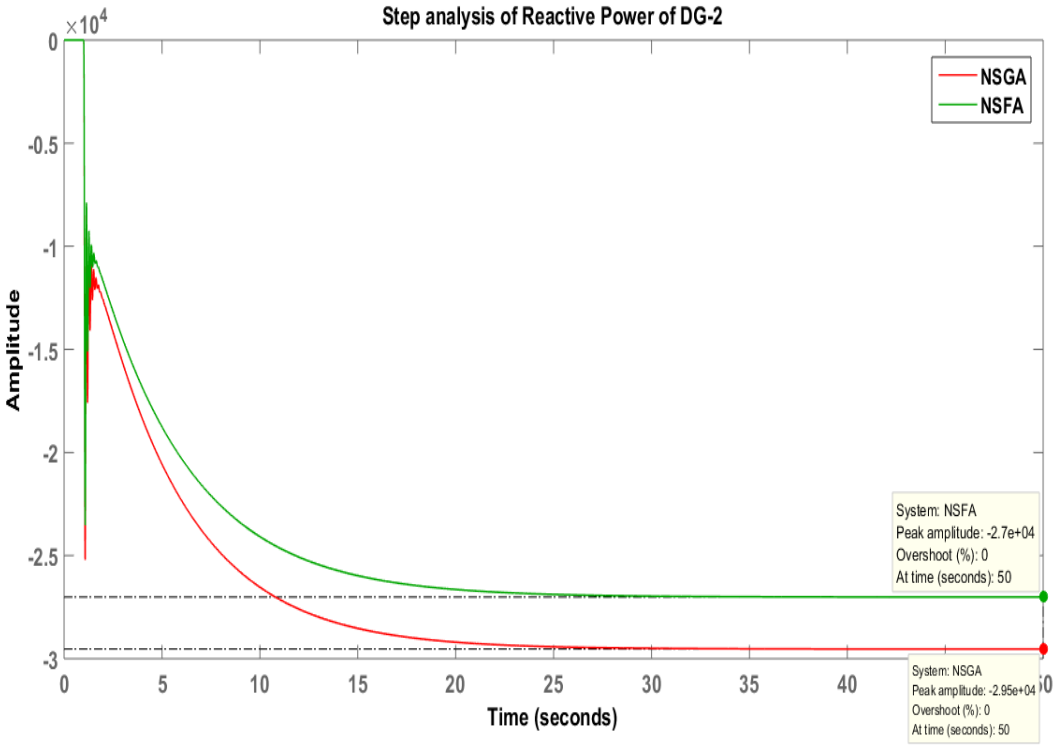


(a) Reactive Power of DG-1

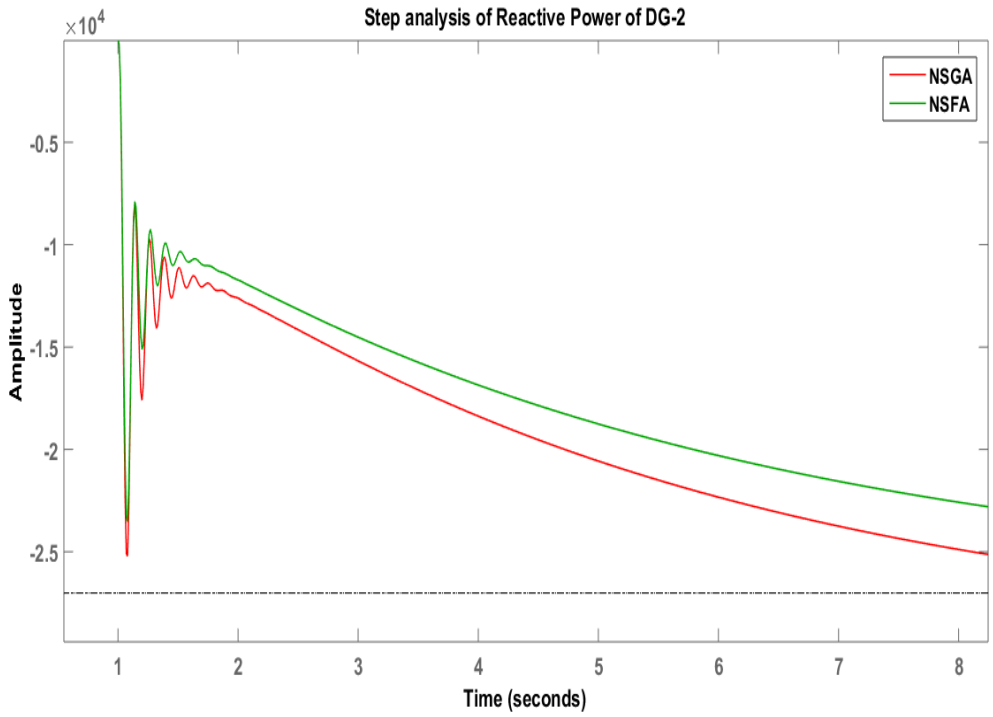


(b) Reactive Power of DG-1 (Zoomed View)

Figure 5.4: Step Analysis of Reactive Power of DG-1

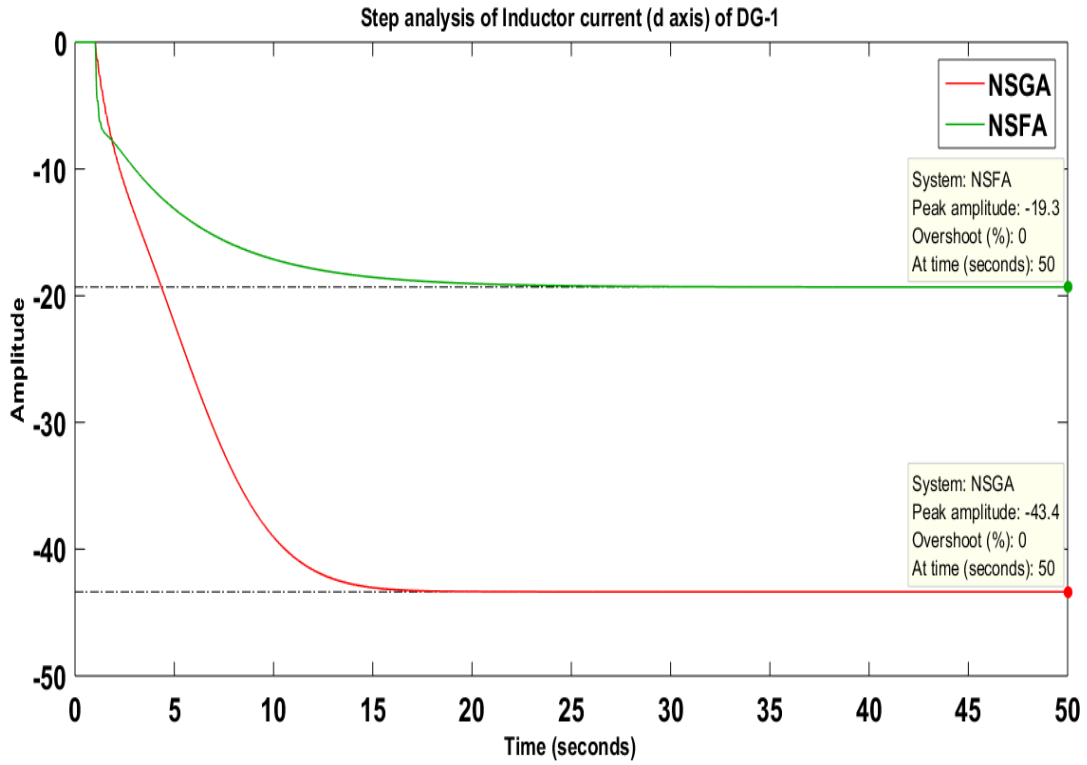


(a) Reactive Power of DG-2

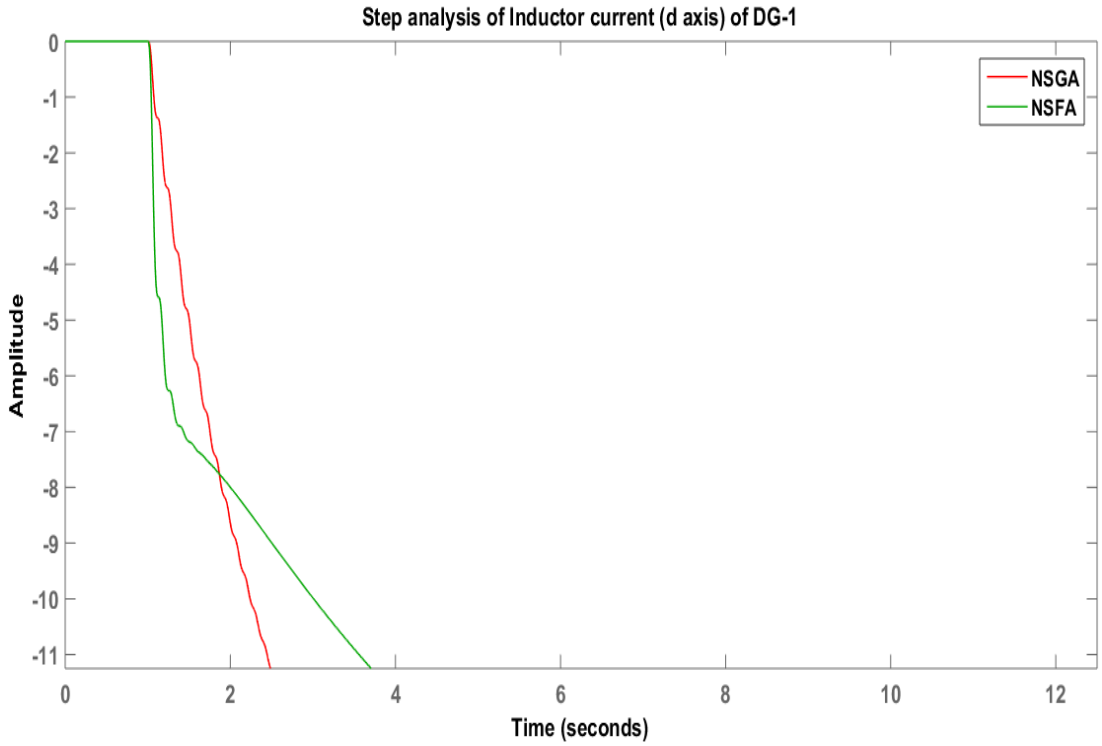


(b) Reactive Power of DG-2 (Zoomed View)

Figure 5.5: Step Analysis of Reactive Power of DG-2

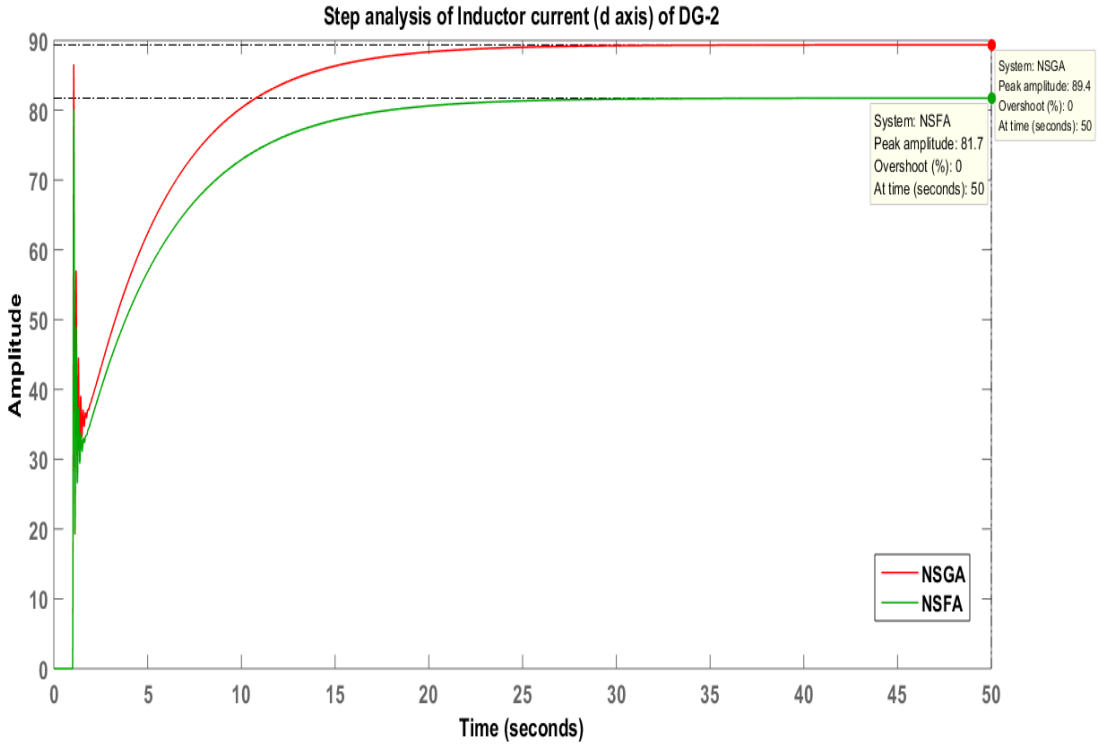


(a) Inductor Current (d axis) of DG-1

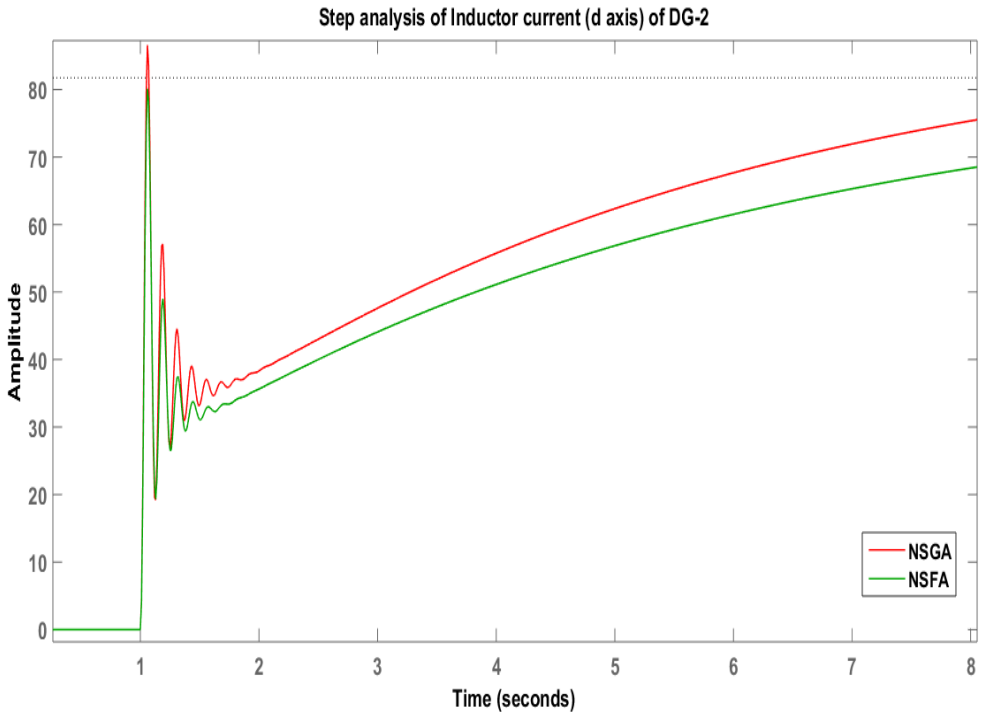


(b) Inductor Current (d axis) of DG-1 (Zoomed View)

Figure 5.6: Step Analysis of Inductor Current (d axis) of DG-1

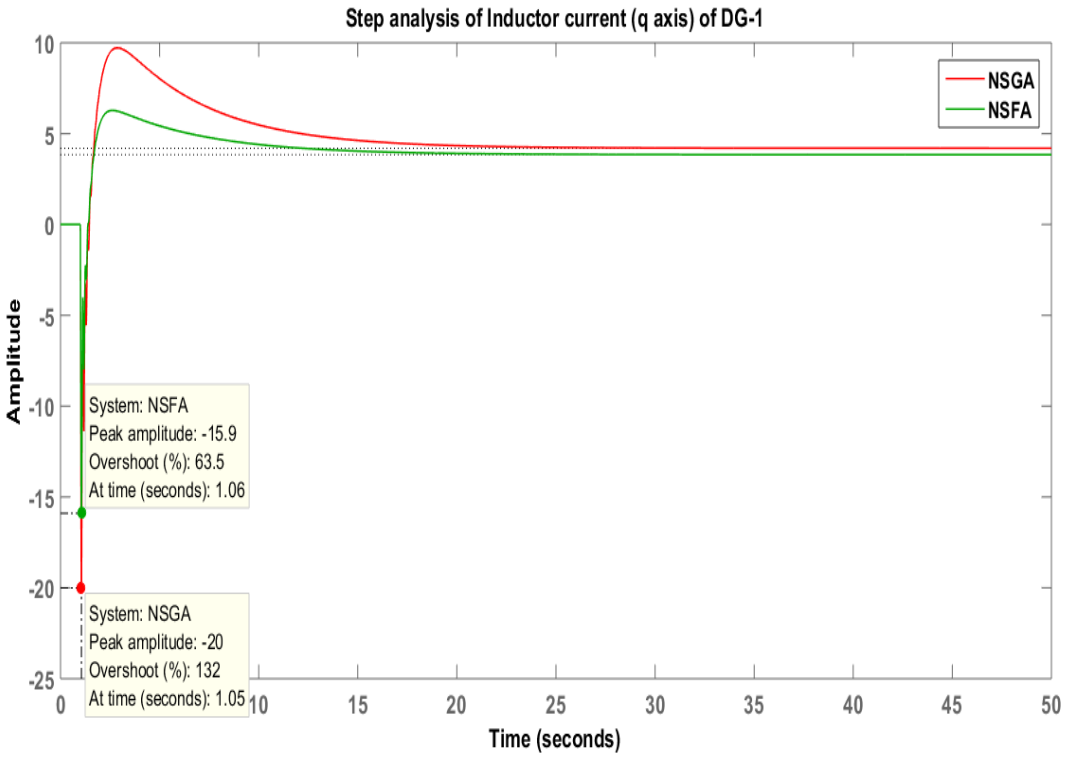


(a) Inductor Current (d axis) of DG-2

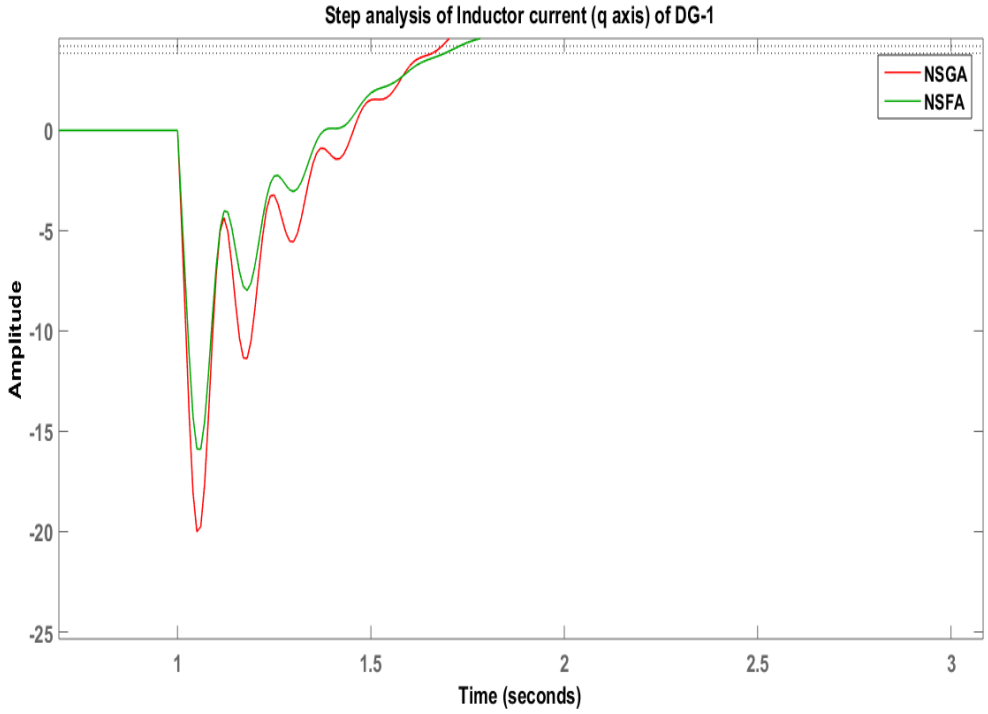


(b) Inductor Current (d axis) of DG-2 (Zoomed View)

Figure 5.7: Step Analysis of Inductor Current (d axis) of DG-2

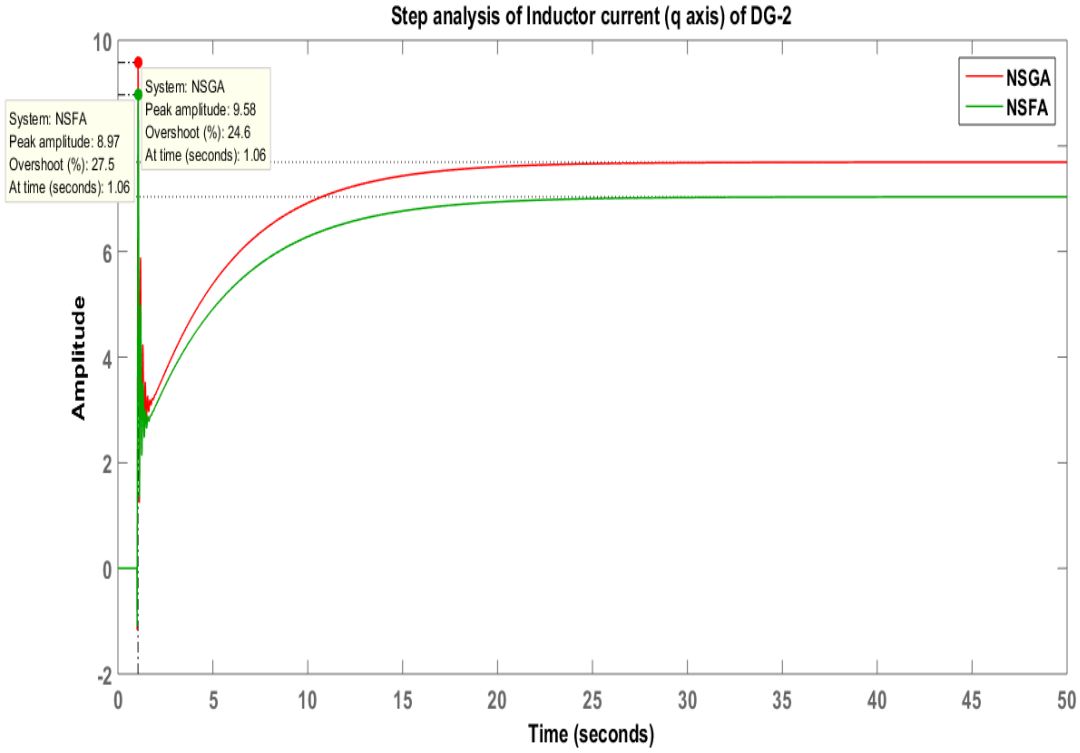


(a) Inductor Current (q axis) of DG-1

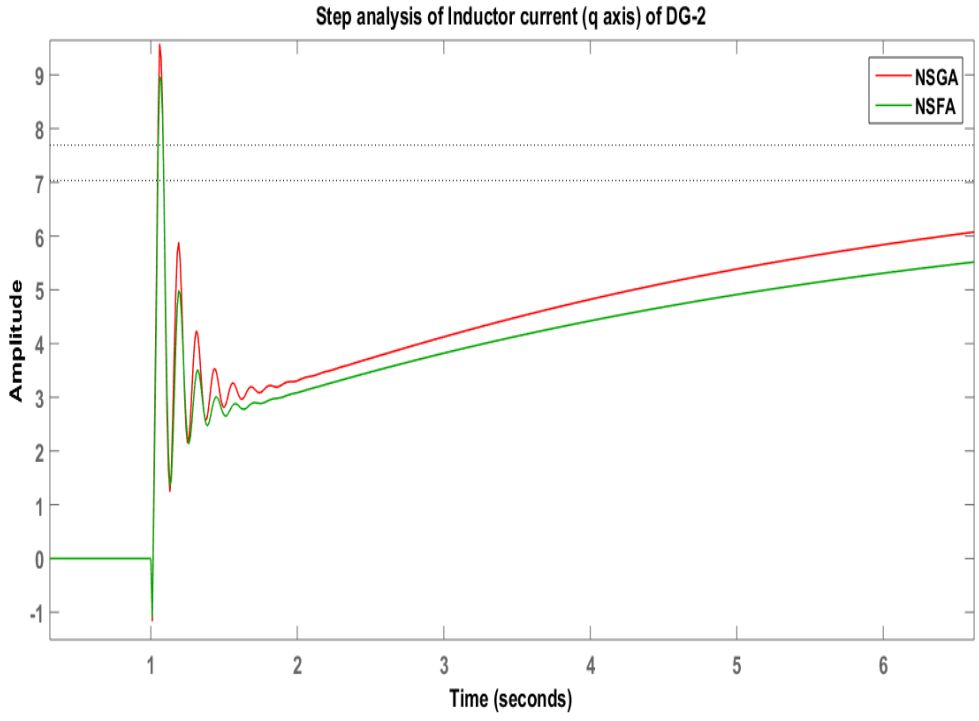


(b) Inductor Current (q axis) of DG-1 (Zoomed View)

Figure 5.8: Step Analysis of Inductor Current (q axis) of DG-1

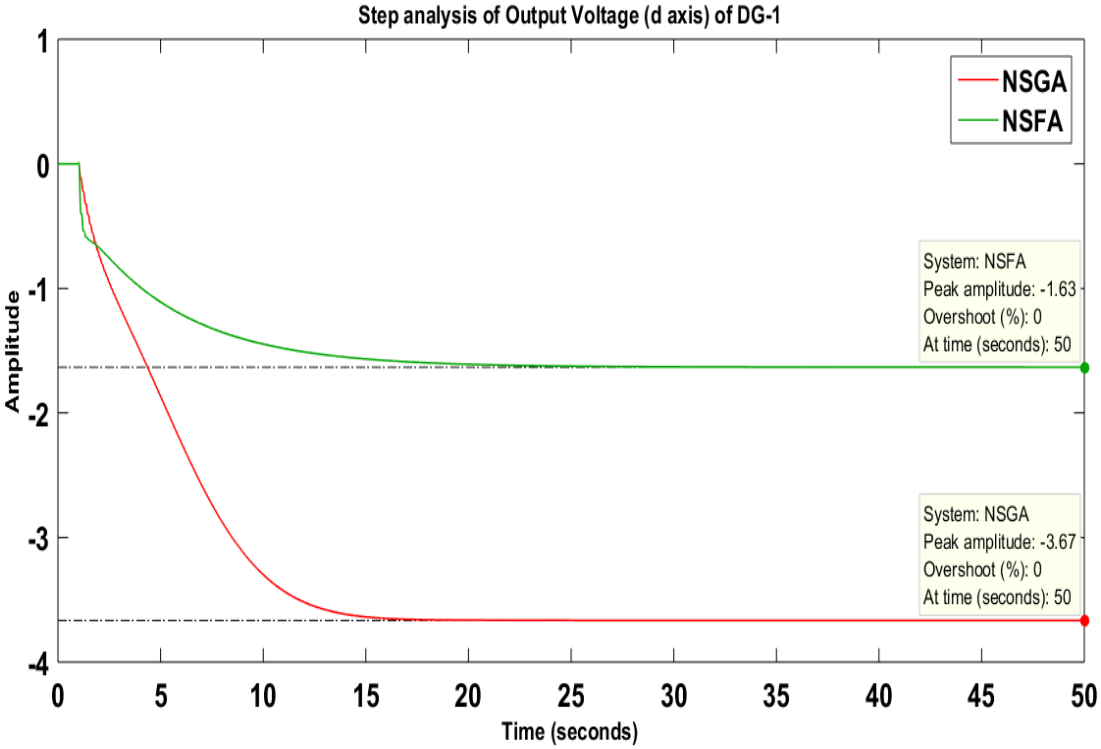


(a) Inductor Current (q axis) of DG-2

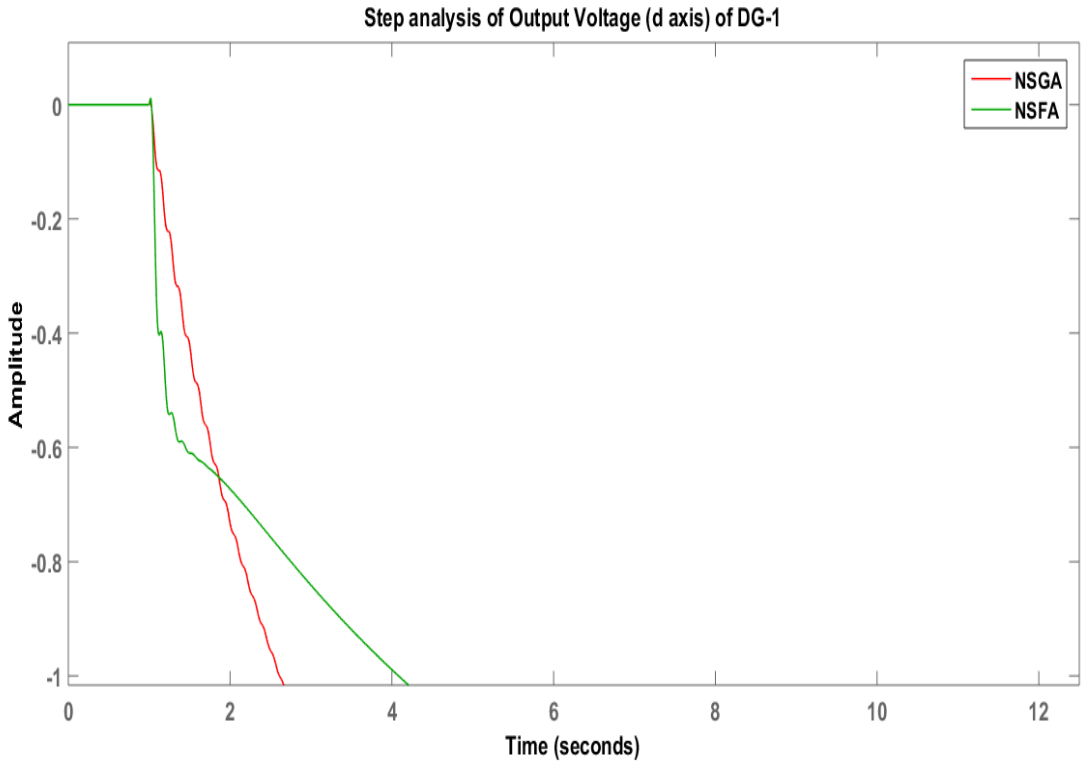


(b) Inductor Current (q axis) of DG-2 (Zoomed View)

Figure 5.9: Step Analysis of Inductor Current (q axis) of DG-2

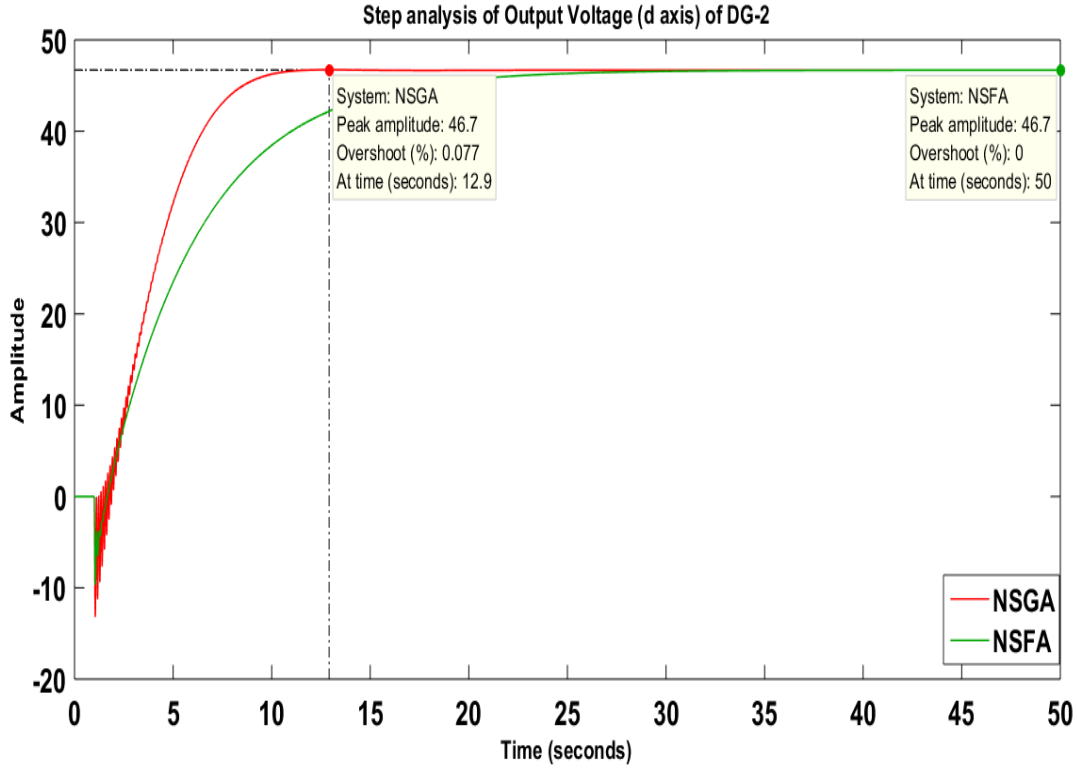


(a) Output Voltage (d axis) of DG-1

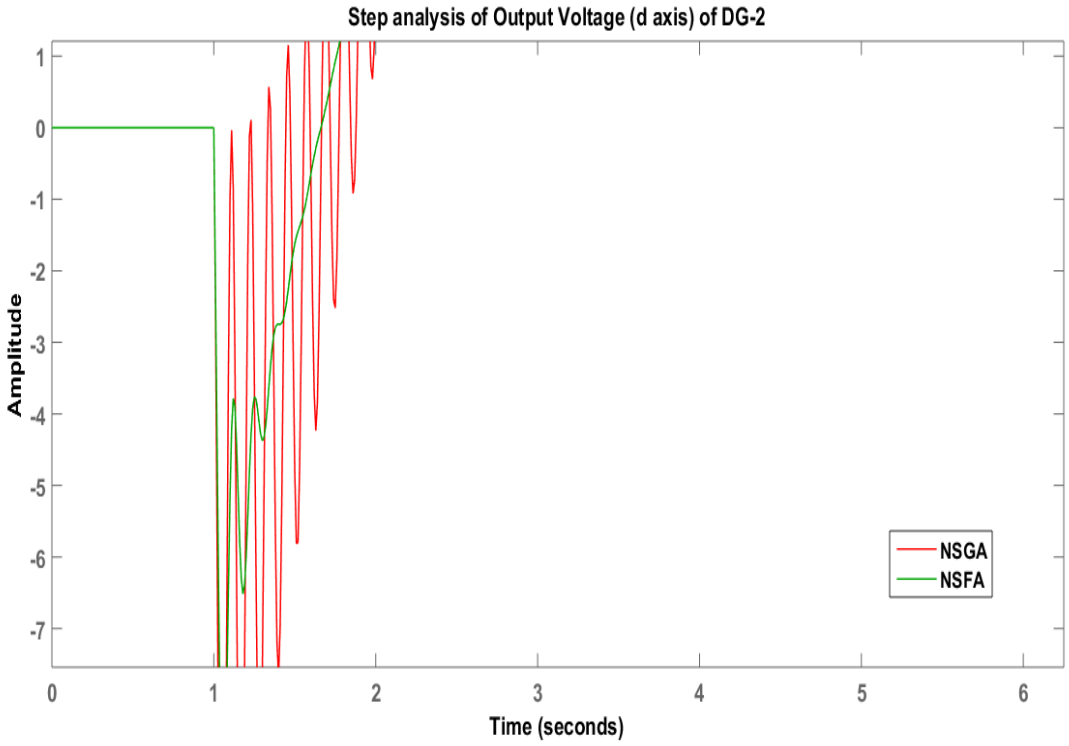


(b) Output Voltage (d axis) of DG-1 (Zoomed View)

Figure 5.10: Step Analysis of Output Voltage (d axis) of DG-1

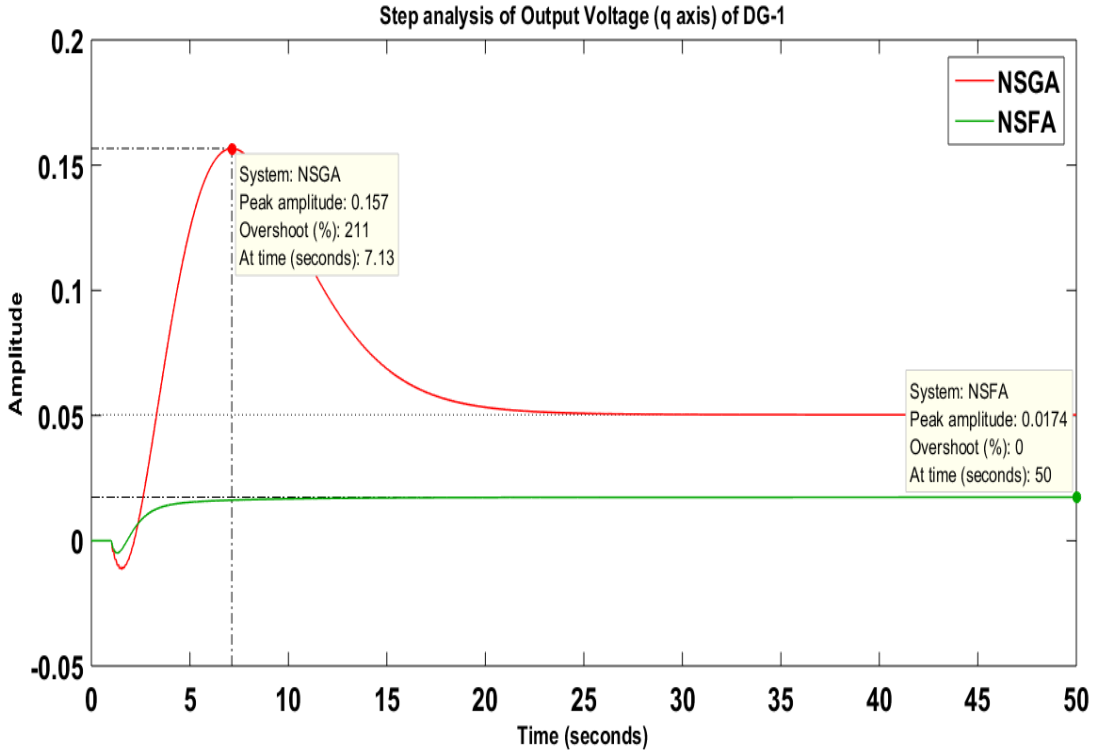


(a) Output Voltage (d axis) of DG-2

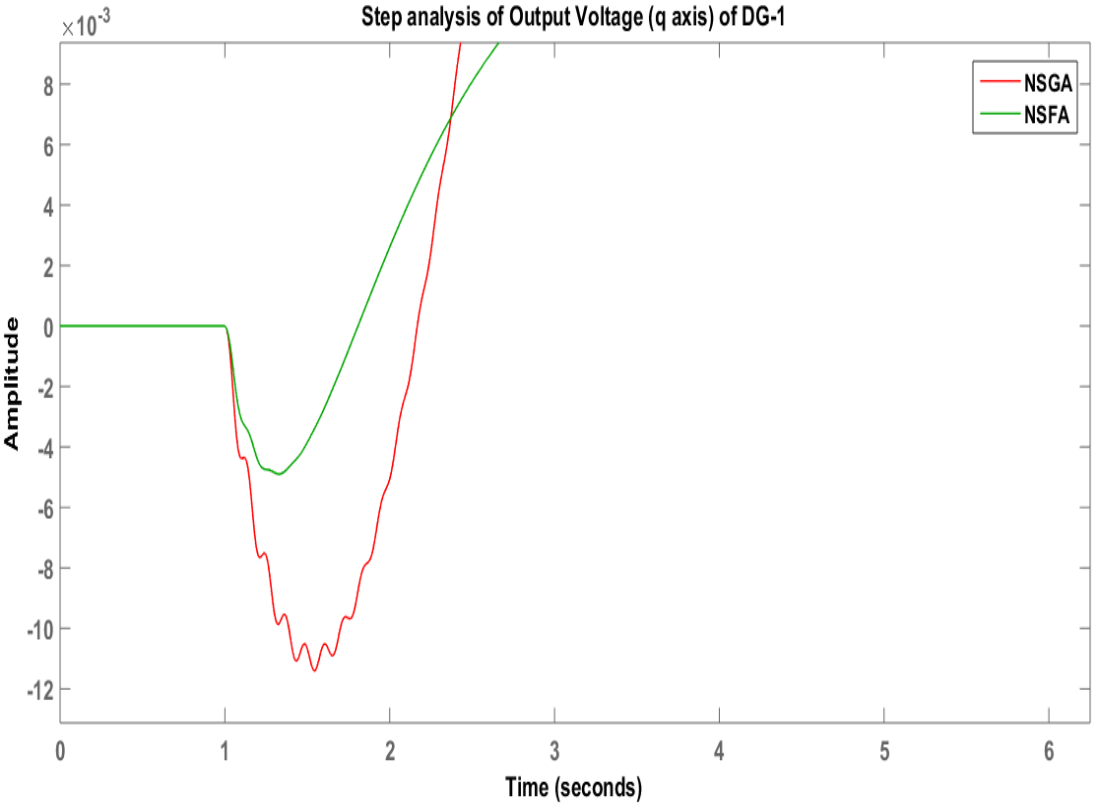


(b) Output Voltage (d axis) of DG-2 (Zoomed View)

Figure 5.11: Step Analysis of Output Voltage (d axis) of DG-2

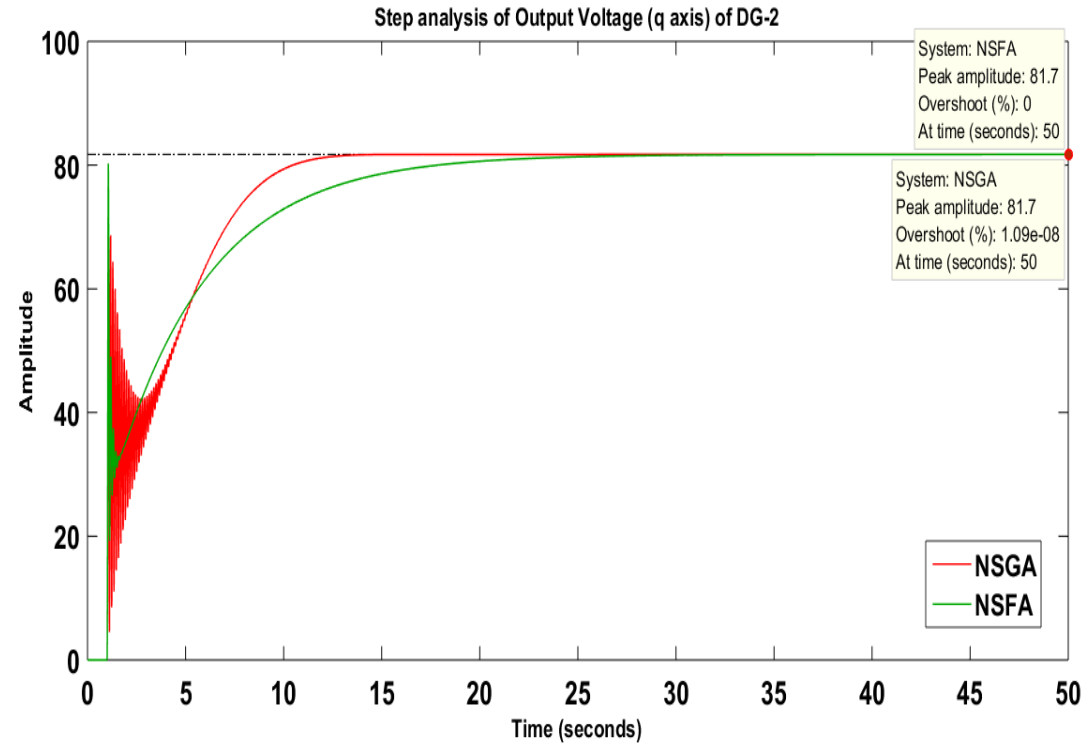


(a) Output Voltage (q axis) of DG-1

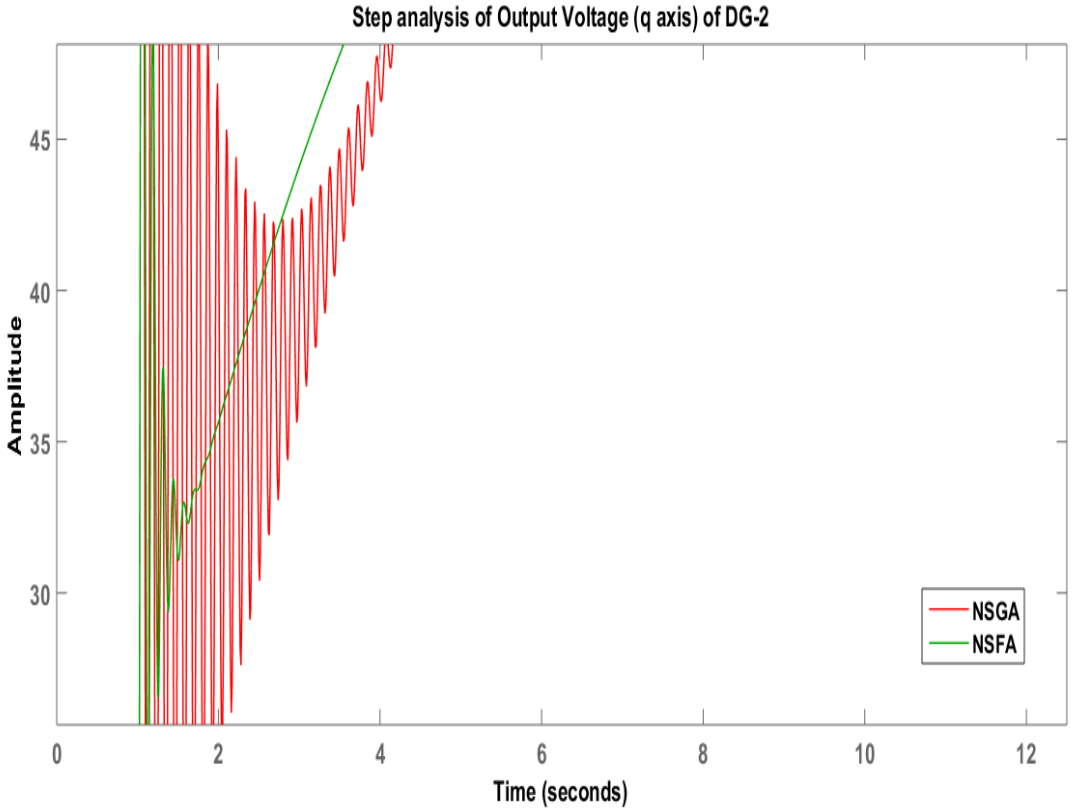


(b) Output Voltage (q axis) of DG-1 (Zoomed View)

Figure 5.12: Step Analysis of Output Voltage (q axis) of DG-1



(a) Output Voltage (q axis) of DG-2



(b) Output Voltage (q axis) of DG-2 (Zoomed View)

Figure 5.13: Step Analysis of Output Voltage (q axis) of DG-2

Figure 5.8 and tabulated data in Table 5.2 show that for inductor current of DG-1 ($q-axis$), NSFA's overshoot is 63.5 which is lower compared to NSGA-II's overshoot of 132% but oscillation frequency of NSFA is $50339.840\ Hz$ which is higher than oscillation frequency of NSGA-II i.e. $46067.998\ Hz$. Vice-versa performance was found for overshoot of inductor current of DG-2 ($q-axis$) as shown in Figure 5.9 where NSFA produces slightly higher overshoot than NSGA-II. But in this case of DG-2, oscillation frequency of NSFA is $7.910\ Hz$ which is less than $8.621\ Hz$ of NSGA.

Finally, for output voltage it is clearly seen from Figure 5.10 - Figure 5.13 and tabulated data in Table 5.2 that the performance of NSFA is better than that of NSGA as overshoot and oscillation frequency is higher with NSGA algorithm. For both the DGs NSFA provides zero overshoot whereas there exist negligible overshoot for NSGA. From the performance analysis it can be seen that damping characteristics of the system has better response by using NSFA compared to NSGA.

5.4 Statistical Tests

An statistical analysis was performed using SPSS software to validate the performance analysis of the algorithms on the basis of obtained values as like as described in Chapter 5. Independent sample $t-test$ was performed to determine how the algorithms are unique from each other. The test was performed on the basis of total number of iteration required by each algorithm for convergence, total time for convergence and total summation of real part of eigen values. $F-test$ was also obtained along with the $t-test$. $F-test$ basically finds out the variances of the sample data sets of NSFA and NSGA and determines whether equal variances prevails between them or not. If the $Sigma(p)$ value from $F-test$ is greater than 0.05 then the group variances of the data sets are said to have equal variances or-else group variances of the data sets are not taken as equal. Similarly, in $t-test$ if $Sigma(2-tailed)$ value is greater than 0.05 then null hypothesis (H_o) cannot be rejected i.e. it is considered that mean values of the group data are equal but if is less than 0.05 then alternative hypothesis H_1 can be considered where mean values of group data are not equal. From Table 5.3 it can be seen that except for total summation of eigen values in $t-test$, all other cases have $Sigma(2-tailed)$ value less than 0.05 which indicates that null hypothesis can be rejected. For $F-test$, $Sigma(p)$ is less than 0.05 for all the cases which means there exist no equal variances between NSFA and NSGA. So, it can be said that there is significant differences in the means of NSGA and NSFA algorithms ensuring both are unique in their own way.

Table 5.3: Results of $F - test$ and $t - Test$ for NSGA-II and NSFA

Parameters for Test	Levene's Test for equality of Variances (F-Test)		Equality Means Test (t-Test)			
	F	*Sig.*	*t*	*df*	*Sig.(2-talied)*	*Mean Difference*
Total number of iterations	62.115019	.000	5.383	30.164	.000	6.50000
Total time required	45.776540	.000	4.354	29.841	.000	7.99043
Total summation of Eigen Values(Real)	6.342609	.015	.923	58	.360	21204.25933

Table 5.4: Group Statistical Data

Parameters for Test	Algorithm	Mean	Standard Deviation	Standard Error Mean
Total number of iterations	NSGA-II	8.8667	6.54814	1.19552
	NSFA	2.3667	.92786	.16940
Total Time Required(sec)	NSGA-II	12.5364	9.97964	1.82203
	NSFA	4.5459	1.20166	.21939
Total summation of Eigen values (Real)	NSGA-II	-12683862.1353	66179.19802	12082.61320
	NSFA	-12705066.3947	107004.22337	19536.20896

Now, Table 5.4 shows the mean, standard deviation and standard mean error in total number of iteration, total time and total summation of eigen values for NSGA-II and NSFA from which it is seen that NSFA requires less number of iteration and less time to reach convergence compared to NSGA-II and also have more negative eigen values compared to NSGA-II. The above data includes 30 independent runs from which the best data is being tabulated.

5.5 Comparison between NSWOA and NSFA

In this study as two novel algorithms namely NSWOA and NSFA have been proposed and as both of these algorithms shows better performance than NSGA-II so a comparative study between NSWOA and NSFA is discussed in this section.

Table 5.5 shows the comparison of NSWOA and NSFA on the basis of overshoot and oscillation frequency where it is seen that NSWOA and NSFA both provides zero overshoot for real and reactive power of both DG-1 and DG-2, inductor current (d-axis) of DG-1 and DG-2, output voltage (d-q axes) of both DG-1 and DG-2. Moreover for inductor current (q- axis) of DG-1 the overshoot percentage of NSWOA zero which is lower than that of NSFA. Similarly for inductor current (q- axis) of DG-2 the overshoot of NSWOA is 20.9% which is less than 27.5% overshoot of NSFA. The

Table 5.5: NSWOA and NSFA comparison on the basis of overshoot and oscillation frequency

Criterion	Parameters	NSWOA	NSFA
Real Power of DG-1	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	1943184.58590787	1943184.58052368
Real Power of DG-2	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	3097.73723900745	3012.13652730558
Reactive Power of DG-1	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	1322438.71460419	1322438.70465282
Reactive Power of DG-2	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	3097.73723900745	3012.13652730558
Inductor Current of DG-1 (d-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	61166.8648946795	77408.4617388257
Inductor Current of DG-2 (d-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	0	0
Inductor Current of DG-1 (q-axis)	*Overshoot (%)*	0	63.5
	Oscillation Frequency (Hz)	27506.8287713625	50339.8409294205
Inductor Current of DG-2 (q-axis)	*Overshoot (%)*	20.9	27.5
	Oscillation Frequency (Hz)	8.37039981802671	7.91008749588845
Output Voltage of DG-1 (d-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	27506.8287713625	50339.8409294205
Output Voltage of DG-2 (d-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	8.37029981802671	7.91008749588845
Output Voltage of DG-1 (q-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	27436.7338495822	50408.3033958121
Output Voltage of DG-2 (q-axis)	*Overshoot (%)*	0	0
	Oscillation Frequency (Hz)	0	0

oscillation frequency of NSWOA is also lower than that of NSFA for inductor current (d-q axes) of DG-1 and output voltage (d-q axes) of DG-1.

Similarly, Table 5.6 shows the group statistical data of both NSWOA and NSFA. From the table it is seen that NSWOA requires less time to converge to the best solution even though the number of iteration required for that purpose is more. So from this tabular analysis it can be concluded that NSWOA performs slightly better than NSFA in optimizing the system and obtains a stable system in less amount of time.

Table 5.6: Group Statistical Data

Parameters for Test	Algorithm	Mean	Standard Deviation	Standard Error Mean
Total number of iterations	NSWOA	4.033	5.88091	1.07370
	NSFA	2.3667	.92786	.16940
Total Time Required (sec)	NSWOA	2.9201	3.88320	.70897
	NSFA	4.5459	1.20166	.21939
Total summation of Eigen values (Real)	NSWOA	-12688400.147	124729.01128	22772.29768
	NSFA	-12705066.3947	107004.22337	19536.20896

5.6 Summary

In the beginning of this chapter the eigen value analysis of NSFA was shown which proved the ability of NSFA in optimizing the system and ensured stabilized operation of the system. Subsequently, time domain simulation analysis and statistical test was performed to compare the performance of NSFA with existing NSGA-II algorithm. Results suggest than NSFA has better damping performance compared to NSGA-II and also the convergence rate of NSFA is better than that of NSGA-II since NSFA requires lesser number of iteration and less time to converge to the best solution.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this work two new hybridized optimization algorithm namely, Non-dominated Sorting Whale Optimization Algorithm (NSWOA) and Non-dominated Sorting Firefly Algorithm (NSFA) have been developed by combining nature inspired swarm intelligence based Whale Optimization Algorithm (WOA) with multi-objective based non-dominated sorting technique to form NSWOA and for NSFA, nature inspired Firefly Algorithm (FA) was combined with non-dominated sorting technique. The total work was divided into six (6) chapters where detailed demonstration about the literature review, development of mathematical model, development of the proposed algorithms and finally result analysis was performed.

At first, a brief introduction about the microgrid was given where an insight about the importance of microgrid as well as the importance of controller optimization was also mentioned including the detailed literature review about the optimization techniques and controller optimization. In the next part, a linearized mathematical model of the microgrid was developed which consisted of static load as well as induction motor as dynamic load. This model is being used in our study for controller optimization. Root locus analysis was performed with this system to justify the significance of controller parameter optimization as well as to observe the effect of the controller parameters on system stability. On the basis of root locus analysis the objective function was developed. In order to satisfy the objective functions two novel hybridized algorithms has been formed.

The performance analysis of NSWOA and NSFA mentioned in Chapter 4 and Chapter 5 justifies the ability of these two algorithms in obtaining stable operation of the microgrid system by optimizing the controller parameters. In order to compare

the performance of the proposed NSWOA and NSFA with existing well established NSGA-II and SPEA algorithm a time domain analysis and statistical analysis was carried out.

From the result analysis it can be concluded to the fact that the proposed NSWOA technique is able to obtain a stable system with less overshoot and oscillation frequency in the mentioned cases compared to the well recognized NSGA-II and SPEA technique. Moreover, the computation speed of NSWOA is much faster than NSGA-II and SPEA as well it takes less number of iteration for reaching convergence which adds to the beneficial characteristics of NSWOA. For instance the total number of iteration required by NSWOA to reach convergence is 4.033 compared to 11.533 iterations of NSGA-II and 10.13 iterations of SPEA. Similarly, the time to reach convergence is 2.92 for NSWOA which is much faster than $6.72\ s$ of NSGA-II and $4.28\ s$ of SPEA. From the statistical $F-test$ and $t-test$ it is also proved that all NSWOA, NSGA and SPEA techniques are different from each other and there are unique features in each of these algorithms. Similarly for NSFA the analysis done in the result and discussion section indicates that NSFA is able to stabilize a system with improved damping characteristics requiring much lesser time than NSGA-II. The results shows that overshoot and oscillation frequency response is much better with NSFA algorithm and it also requires significantly less number of iteration and less amount of time to reach convergence to obtain the best value. NSFA requires 2.366 number of iteration and takes $4.54\ s$ to reach convergence which proves NSFA requires significantly lesser number of iterations and time to reach to convergence.

So, both NSWOA and NSFA justifies their ability in obtaining a stable microgrid system where NSWOA requires less time to reach convergence compared to NSFA but on the other hand NSFA requires less number of iterations to reach convergence compared to NSWOA.

6.2 Future Scope of this work

Whale optimization algorithm itself gained much appreciation due to its better searching capability and convergence rate as well as evolved as an efficient technique. Now, combining this whale optimization technique with multi objective based non-dominated sorting technique opens a new door in the field of optimization as it is known to all that multi-objective optimizer are capable in solving many pragmatic problems. In the field of controller optimization of microgrid this NSWOA technique will enlighten the re-

searchers in further improvement of this field as NSWOA provides better response for multi-objective problems compared to NSGA and SPEA. Similarly for NSFA it can be added that Firefly Algorithm itself evolved as an efficient optimization technique and combining it with non-dominated sorting method to optimize multi objective functions opened a new opportunity for the researchers as well.

Moreover, researchers can also implement the proposed algorithms in optimizing the load flow as well as cost optimization of the microgrid.

REFERENCES

[1] C. Colson and M. Nehrir, "A review of challenges to real-time power management of microgrids," in *2009 IEEE Power & Energy Society General Meeting*. IEEE, 2009, pp. 1–8.

[2] A. K. Basu, S. Chowdhury, S. Chowdhury, and S. Paul, "Microgrids: Energy management by strategic deployment of ders—a comprehensive survey," *Renewable and Sustainable Energy Reviews*, vol. 15, no. 9, pp. 4348–4356, 2011.

[3] F. Katiraei, R. Iravani, N. Hatziargyriou, and A. Dimeas, "Microgrids management," *IEEE power and energy magazine*, vol. 6, no. 3, pp. 54–65, 2008.

[4] A. Al-Awami, E. Sortomme, and M. El-Sharkawi, "Optimizing economic/environmental dispatch with wind and thermal units," in *2009 IEEE Power & Energy Society General Meeting*. IEEE, 2009, pp. 1–6.

[5] C.-T. Hsu, R. Korimara, T.-J. Cheng, L.-J. Tsai, and H.-M. Huang, "Cost power curtailment analysis for optimum pv size and the energy potential for the desalination plant on the island distribution system," *Journal of Clean Energy Technologies*, vol. 5, no. 3, 2017.

[6] Y. Agarwal, T. Weng, and R. K. Gupta, "Understanding the role of buildings in a smart microgrid," in *2011 Design, Automation & Test in Europe*. IEEE, 2011, pp. 1–6.

[7] M. Agrawal and A. Mittal, "Micro grid technological activities across the globe: A review," *Int. J. Res. Rev. Appl. Sci*, vol. 7, no. 2, pp. 147–152, 2011.

[8] H.-C. Chen, P.-H. Chen, L.-Y. Chang, and W.-X. Bai, "Stand-alone hybrid generation system based on renewable energy," *International Journal of Environmental Science and Development*, vol. 4, no. 5, p. 514, 2013.

[9] X. Li, D. Hui, and X. Lai, "Battery energy storage station (bess)-based smoothing control of photovoltaic (pv) and wind power generation fluctuations," *IEEE transactions on sustainable energy*, vol. 4, no. 2, pp. 464–473, 2013.

[10] S. V. Iyer, M. N. Belur, and M. C. Chandorkar, "A generalized computational method to determine stability of a multi-inverter microgrid," *IEEE Transactions on Power Electronics*, vol. 25, no. 9, pp. 2420–2432, 2010.

[11] N. Pogaku, M. Prodanovic, and T. C. Green, "Modeling, analysis and testing of autonomous operation of an inverter-based microgrid," *IEEE Transactions on power electronics*, vol. 22, no. 2, pp. 613–625, 2007.

[12] S. Hemamalini and S. Simon, "Maclaurin series-based lagrangian method for economic dispatch with valve-point effect," *IET generation, transmission & distribution*, vol. 3, no. 9, pp. 859–871, 2009.

[13] H. Farhangi, "The path of the smart grid," *IEEE power and energy magazine*, vol. 8, no. 1, pp. 18–28, 2009.

[14] J. Hetzer, C. Y. David, and K. Bhattarai, "An economic dispatch model incorporating wind power," *IEEE Transactions on energy conversion*, vol. 23, no. 2, pp. 603–611, 2008.

[15] K. Prabakar, F. Li, and B. Xiao, "Controller hardware-in-loop testbed setup for multi-objective optimization based tuning of inverter controller parameters in a microgrid setting," in *2016 Clemson University Power Systems Conference (PSC)*. IEEE, 2016, pp. 1–8.

[16] H. Ishibuchi, Y. Nojima, and T. Doi, "Comparison between single-objective and multi-objective genetic algorithms: Performance comparison and performance measures," in *2006 IEEE International Conference on Evolutionary Computation*. IEEE, 2006, pp. 1143–1150.

[17] A. A. El-Fergany and H. M. Hasanien, "Single and multi-objective optimal power flow using grey wolf optimizer and differential evolution algorithms," *Electric Power Components and Systems*, vol. 43, no. 13, pp. 1548–1559, 2015.

[18] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," *Machine learning*, vol. 3, no. 2, pp. 95–99, 1988.

[19] J. Kennedy and R. Eberhart, "Particle swarm optimization (pso)," in *Proc. IEEE International Conference on Neural Networks, Perth, Australia*, 1995, pp. 1942–1948.

[20] L. J. Fogel, "Aj owens, and mj walsh," *Artificial Intelligence through Simulated Evolution. Wiley, New York*, 1966.

[21] M. Dorigo and M. Birattari, *Ant colony optimization*. Springer, 2010.

[22] J. Kennedy, "Particle swarm optimization," *Encyclopedia of machine learning*, pp. 760–766, 2010.

[23] Y. Gao, W. Du, and G. Yan, "Selectively-informed particle swarm optimization," *Scientific Reports*, vol. 5, no. 1, pp. 9295–9295, 2015.

[24] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm," *Journal of global optimization*, vol. 39, no. 3, pp. 459–471, 2007.

[25] X.-S. Yang and N.-I. M. Algorithms, "Luniver press," *Beckington, UK*, pp. 242–246, 2008.

[26] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *International symposium on stochastic algorithms*. Springer, 2009, pp. 169–178.

[27] T. Apostolopoulos and A. Vlachos, "Application of the firefly algorithm for solving the economic emissions load dispatch problem," *International journal of combinatorics*, vol. 2011, 2010.

[28] A. Chatterjee, G. K. Mahanti, and A. Chatterjee, "Design of a fully digital controlled reconfigurable switched beam concentric ring array antenna using firefly and particle swarm optimization algorithm," *Progress in Electromagnetics Research*, vol. 36, pp. 113–131, 2012.

[29] M. Sayadi, R. Ramezanian, and N. Ghaffari-Nasab, "A discrete firefly metaheuristic with local search for makespan minimization in permutation flow shop scheduling problems," *International Journal of Industrial Engineering Computations*, vol. 1, no. 1, pp. 1–10, 2010.

[30] M.-H. Horng, Y.-X. Lee, M.-C. Lee, and R.-J. Liou, "Firefly metaheuristic algorithm for training the radial basis function network for data classification and disease diagnosis," *Theory and new applications of swarm intelligence*, vol. 4, no. 7, pp. 115–132, 2012.

[31] M.-H. Horng, "Vector quantization using the firefly algorithm for image compression," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1078–1091, 2012.

[32] B. Basu and G. K. Mahanti, "Fire fly and artificial bees colony algorithm for synthesis of scanned and broadside linear array antenna," *Progress In Electromagnetics Research*, vol. 32, pp. 169–190, 2011.

[33] M. A. Zaman, A. Matin *et al.*, "Nonuniformly spaced linear antenna array design using firefly algorithm," *International Journal of Microwave Science and Technology*, vol. 2012, 2012.

[34] S. Tilahun and H. C. Ong, "Modified firefly algorithm," *Journal of Applied Mathematics*, vol. 467631, 11 2012.

[35] S. Palit, S. N. Sinha, M. A. Molla, A. Khanra, and M. Kule, "A cryptanalytic attack on the knapsack cryptosystem using binary firefly algorithm," in *2011 2nd International conference on computer and communication technology (ICCCT-2011)*. IEEE, 2011, pp. 428–432.

[36] S. M. Farahani, A. Abshouri, B. Nasiri, and M. Meybodi, "A gaussian firefly algorithm," *International Journal of Machine Learning and Computing*, vol. 1, no. 5, p. 448, 2011.

[37] L. dos Santos Coelho, D. L. de Andrade Bernert, and V. C. Mariani, "A chaotic firefly algorithm applied to reliability-redundancy optimization," in *2011 IEEE congress of evolutionary computation (CEC)*. Ieee, 2011, pp. 517–521.

[38] A. Abdullah, S. Deris, M. S. Mohamad, and S. Z. M. Hashim, "A new hybrid firefly algorithm for complex and nonlinear problem," in *Distributed Computing and Artificial Intelligence*. Springer, 2012, pp. 673–680.

[39] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in engineering software*, vol. 95, pp. 51–67, 2016.

[40] N. M. Laskar, K. Guha, I. Chatterjee, S. Chanda, K. L. Baishnab, and P. K. Paul, "Hwpso: A new hybrid whale-particle swarm optimization algorithm and its application in electronic design optimization problems," *Applied Intelligence*, vol. 49, no. 1, pp. 265–291, 2019.

[41] I. N. Trivedi, P. Jangir, A. Kumar, N. Jangir, and R. Totlani, "A novel hybrid pso–woa algorithm for global numerical functions optimization," in *Advances in Computer and Computational Sciences*. Springer, 2018, pp. 53–60.

[42] M. Abdel-Basset, G. Manogaran, D. El-Shahat, and S. Mirjalili, "A hybrid whale optimization algorithm based on local search strategy for the permutation flow shop scheduling problem," *Future Generation Computer Systems*, vol. 85, pp. 129–145, 2018.

[43] N. Singh and H. Hachimi, "A new hybrid whale optimizer algorithm with mean strategy of grey wolf optimizer for global optimization," *Mathematical and Computational Applications*, vol. 23, no. 1, p. 14, 2018.

[44] N. Singh and S. Singh, "A modified mean gray wolf optimization approach for benchmark and biomedical problems," *Evolutionary Bioinformatics*, vol. 13, 2017.

[45] Z. Xu, Y. Yu, H. Yachi, J. Ji, Y. Todo, and S. Gao, "A novel memetic whale optimization algorithm for optimization," in *International Conference on Swarm Intelligence*. Springer, 2018, pp. 384–396.

[46] M. M. Mafarja and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302–312, 2017.

[47] M. A. El Aziz, A. A. Ewees, and A. E. Hassanien, "Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation," *Expert Systems with Applications*, vol. 83, pp. 242–256, 2017.

[48] A. N. Jadhav and N. Gomathi, "Wgc: hybridization of exponential grey wolf optimizer with whale optimization for data clustering," *Alexandria engineering journal*, vol. 57, no. 3, pp. 1569–1584, 2018.

[49] S. Khalilpourazari and S. Khalilpourazary, "Scwoa: an efficient hybrid algorithm for parameter optimization of multi-pass milling process," *Journal of Industrial and Production Engineering*, vol. 35, no. 3, pp. 135–147, 2018.

[50] S. T. Revathi, N. Ramaraj, and S. Chithra, "Brain storm-based whale optimization algorithm for privacy-protected data publishing in cloud computing," *Cluster Computing*, vol. 22, no. 2, pp. 3521–3530, 2019.

[51] A. Kaveh and M. Rastegar Moghaddam, "A hybrid woa-cbo algorithm for construction site layout planning problem," *Scientia Iranica*, vol. 25, no. 3, pp. 1094–1104, 2018.

[52] A. Kaveh and V. R. Mahdavi, "Colliding bodies optimization: a novel metaheuristic method," *Computers & Structures*, vol. 139, pp. 18–27, 2014.

[53] N. Srinivas and K. Deb, "Muiltiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary computation*, vol. 2, no. 3, pp. 221–248, 1994.

[54] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[55] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii," in *International conference on parallel problem solving from nature*. Springer, 2000, pp. 849–858.

[56] Y. A.-R. I. Mohamed and E. F. El-Saadany, "Adaptive decentralized droop controller to preserve power sharing stability of paralleled inverters in distributed generation microgrids," *IEEE Transactions on Power Electronics*, vol. 23, no. 6, pp. 2806–2816, 2008.

[57] M. Ahmed, A. Vahidnia, L. Meegahapola, and M. Datta, "Small signal stability analysis of a hybrid ac/dc microgrid with static and dynamic loads," in *2017 Australasian Universities Power Engineering Conference (AUPEC)*. IEEE, 2017, pp. 1–6.

[58] A. A. A. Radwan and Y. A.-R. I. Mohamed, "Stabilization of medium-frequency modes in isolated microgrids supplying direct online induction motor loads," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 358–370, 2013.

[59] A. Kahrobaeian and Y. A.-R. I. Mohamed, "Analysis and mitigation of low-frequency instabilities in autonomous medium-voltage converter-based microgrids with dynamic loads," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 4, pp. 1643–1658, 2013.

[60] T. Jain *et al.*, "Impact of load dynamics and load sharing among distributed generations on stability and dynamic performance of islanded ac microgrids," *Electric Power Systems Research*, vol. 157, pp. 200–210, 2018.

[61] I.-Y. Chung, W. Liu, D. A. Cartes, and K. Schoder, "Control parameter optimization for a microgrid system using particle swarm optimization," in *2008 IEEE International Conference on Sustainable Energy Technologies*. IEEE, 2008, pp. 837–842.

[62] I.-Y. Chung, W. Liu, D. A. Cartes, and S.-I. Moon, "Control parameter optimization for multiple distributed generators in a microgrid using particle swarm optimization," *European Transactions on Electrical Power*, vol. 21, no. 2, pp. 1200–1216, 2011.

[63] M. Hassan and M. Abido, "Optimal design of microgrids in autonomous and grid-connected modes using particle swarm optimization," *IEEE Transactions on power electronics*, vol. 26, no. 3, pp. 755–769, 2010.

[64] K. Yu, Q. Ai, S. Wang, J. Ni, and T. Lv, "Analysis and optimization of droop controller for microgrid system based on small-signal dynamic model," *IEEE Transactions on Smart Grid*, vol. 7, no. 2, pp. 695–705, 2015.

[65] I. Abdulwahab, Y. Jibril, and Y. Shu'aibu, "Determination of optimal droop controller parameters for an islanded microgrid system using artificial fish swarm algorithm (afsa)," 03 2017.

[66] R. Wang, S. Wu, C. Wang, S. An, Z. Sun, W. Li, W. Xu, S. Mu, and M. Fu, "optimized operation and control of microgrid based on multi-objective genetic algorithm," in *2018 International Conference on Power System Technology (POWERCON)*. IEEE, 2018, pp. 1539–1544.

[67] R.-F. Yuan, Q. Ai, and X. He, "Research on dynamic load modelling based on power quality monitoring system," *IET Generation, Transmission & Distribution*, vol. 7, no. 1, pp. 46–51, 2013.

[68] Y. Li and Y. W. Li, "Power management of inverter interfaced autonomous microgrid based on virtual frequency-voltage frame," *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 30–40, 2011.

[69] J. M. Uudrill, "Dynamic stability calculations for an arbitrary number of interconnected synchronous machines," *IEEE Transactions on Power Apparatus and Systems*, no. 3, pp. 835–844, 1968.

[70] M. N. Marwali, J.-W. Jung, and A. Keyhani, "Control of distributed generation systems-part ii: Load sharing control," *IEEE Transactions on power electronics*, vol. 19, no. 6, pp. 1551–1561, 2004.

[71] M. Prodanovic, "Power quality and control aspects of parallel connected inverters in distributed generation," Ph.D. dissertation, Imperial College London (University of London), 2004.

[72] M. N. Marwali and A. Keyhani, "Control of distributed generation systems-part i: Voltages and currents control," *IEEE Transactions on power electronics*, vol. 19, no. 6, pp. 1541–1550, 2004.

[73] S. Anand and B. Fernandes, "Reduced-order model and stability analysis of low-voltage dc microgrid," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 11, pp. 5040–5049, 2012.

[74] S. R. Mudaliyar and S. S. Sahoo, "Comparison of different eigenvalue based multi-objective functions for robust design of power system stabilizers," 2015.

[75] G. Verma, A. Kumar, and K. K. Mishra, "A novel non-dominated sorting algorithm," in *International Conference on Swarm, Evolutionary, and Memetic Computing*. Springer, 2011, pp. 274–281.

[76] C. R. Raquel and P. C. Naval Jr, "An effective use of crowding distance in multi-objective particle swarm optimization," in *Proceedings of the 7th annual conference on Genetic and evolutionary computation*. ACM, 2005, pp. 257–264.

[77] P. D. P. Reddy, V. V. Reddy, and T. G. Manohar, "Optimal renewable resources placement in distribution networks by combined power loss index and whale optimization algorithms," *J. Electr. Syst. Inf. Technol*, vol. 28, pp. 669–678, 2017.

[78] M. B. Brown and A. B. Forsythe, "Robust tests for the equality of variances," *Journal of the American Statistical Association*, vol. 69, no. 346, pp. 364–367, 1974.

List of Publications

1. **Quazi Nafees Ul Islam** and Ashik Ahmed, "Optimized Controller Design for Islanded Microgrid Employing Non-Dominated Sorting Firefly Algorithm (NSFA)," **Book Chapter :** Nature Inspired Computation and Swarm Intelligence: Algorithms, Theory and Applications, Elsevier **(Accepted)**.