Hybrid Algorithm-Based Approach For Optimized PID Controller for Stability Analysis of DC-DC Interleaved-Buck Converter

by

Kazi Asif Rahman (170021048)
Sadman Sakib (170021092)
Moshiur Ahmed (170021149)

A Thesis Submitted to the Academic Faculty in Partial Fulfillment of the Requirements for the Degree of

BACHELOR OF SCIENCE IN ELECTRICAL AND ELECTRONIC ENGINEERING

Department of Electrical and Electronic Engineering
Islamic University of Technology (IUT)
Gazipur, Bangladesh

May 2022

CERTIFICATE OF APPROVAL

The thesis titled "A Novel Hybrid Algorithm-Based Optimised PID Controller for Stability Analysis of DC-DC Interleaved-Buck Converter" submitted by Kazi Asif Rahman (170021048), Sadman Sakib (170021092) and Moshiur Ahmed (170021149) has been found as satisfactory and accepted as partial fulfilment of the requirement for the degree of Bachelor of Science in Electrical and Electronic Engineering on 10th May, 2022.

Approved by:

(Signature of the Supervisor)
Mirza Muntasir Nishat
Assistant Professor,
Department of Electrical and Electronic Engineering,
Islamic University of Technology (IUT),
Board Bazar, Gazipur-1704.

(Signature of Co Supervisor)
Fahim Faisal
Assistant Professor,
Department of Electrical and Electronic Engineering,
Islamic University of Technology (IUT),
Boardbazar, Gazipur-1704.

DECLARATION OF CANDIDATES

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

Kazi Asif Rahman

(Signature of Candidate)

Kazi Asif Rahman

Student ID: 170021048

Sadman Sakib

(Signature of Candidate)

Sadman Sakib

Student ID: 170021092

Moshiur Ahmed

(Signature of Candidate)

Moshiur Ahmed

Student ID: 170021149

Table of Contents

List of Tables

No. Title

List of Figures

No. Title

List of Acronyms

Abbreviated Form	Description
IBC	Interleaved Buck Converter
WOA	Whale optimization algorithm
PSO	Particle swarm optimization
SI	Swarm Intelligence
GWO	Grey wolf Optimization
NIA	Nature inspired algorithm
CSA	Cuckoo Search Algorithm
FA	Firefly Algorithm
EMI	Electro Magnetic Induction
PWM	Pulse Width Modulation
SS	State Space
ABC	Ant Bee Colony
IAE	Integral Absolute Error
ITAE	Integral Time-weighted Absolute Error
ISE	Integral Squared Error
ITSE	Integral Time Squared Error

ACKNOWLEDGEMENTS

First, we thank Allah wholeheartedly for giving us the strength and goodwill to perform the research properly and complete our thesis. We are highly grateful to our honourable supervisor, **Mr Mirza Muntasir Nishat**, Assistant Professor, Department of EEE, IUT, for his contribution to our thesis research He devoted his considerable time to guiding and motivating us to finish the assignment. We conducted a study and collected numerous useful analyses under his supervision in order to get positive results. He demonstrated the most beneficial method for gaining a better understanding of our research. We feel that without his assistance, we would be lost and unable to pick the best course of action.

We would like to express our deep and sincere gratitude to our co-supervisor, **Mr Fahim Faisal**, Assistant Professor, Department of EEE, IUT, for mentoring us in performing the research. He has always been supportive and has given us motivation in completing our work properly. He has inspired us to learn the main objective of our work which made us more confident with developing skills while doing the thesis.

Finally, our deepest appreciation goes to our family who has nurtured and helped us to overcome the hurdles in life. Last but not least, we would like to thank our friends who have always supported us and kept our minds joyful through this journey.

Abstract

This thesis is an investigation of the stability analysis of the Interleaved Buck converter by inducing Nature-Inspired Algorithms to create an optimal PID controller. The applicability and compatibility of some algorithms such as PSO, Firefly Algorithm (FA), GWO and Cuckoo Search are used for optimizing the control mechanism of power converters. Improvements in performance parameters are noted, and the results are compared using multiple fitness functions. The thesis focuses on Interleaved converter which has the benefit of lowering ripple currents, simplifying EMI transmission, and faster transient response. The converters are constructed using the State Space Averaging (SSA) technique to provide promising feedback control and to evaluate the transfer functions. The Nature-Inspired Algorithms are non-linear optimization methods based on artificial intelligence. The above algorithms are based on swarm intelligence and operate in accordance with swarm creature customs. Swarm intelligence ensures better data exploitation, which concentrates the search method within the vicinity of optimal solutions while also assisting the procedure to escape the restriction of the local minima, resulting in successful exploration of the search space. With the help of these algorithms, we approached a hybrid algorithm named Hybrid Firefly Particle Swarm Algorithm (HFPSO). We also compared the performance according to the standard. As a result, the algorithms are evaluated for improved system performance using two fitness functions (ITAE, and ITSE) and performance parameters such as percentage of overshoot, rise time, settling time, and peak amplitude are all variables to consider. The procedure is carried using MATLAB. After studying the results, It is discovered that the percentage of HFPSO overshoot (ITSE) offers a lower for each of the error functions than PSO, CSA, FA, GWO, WOA.

Chapter 1

Introduction

Technological progress has dramatically impacted our societies in various fields of power electronics. The most common request for power electronics professionals is to design a high-performance, cost-effective, and highly integrated converter. Due to its non-linear characteristics, the control is now required to be more optimized. This thesis deals with the issue of providing optimized control over the converters with the help of Bio-inspired algorithms induced to the PID controller.

1.1 Background and Motivation

There has been an upsurge in recent years in demand for efficient power consumption systems in various industries, such as PV power systems, hybrid electrical vehicles, and thermoelectric generators, aircraft, battery-powered gadgets, and so on. Engineers have developed efficient conversion processes and have also been able to construct circuits with great efficiency to keep up with these demands. In this profession, however, new challenges arise every day.

In our daily lives, we come across several products such as mobile phones, laptops, cameras, navigation systems, and medical equipment that incorporate multiple IC produced using various semiconductor methods. These devices normally require a number of different supply voltages, each of which is usually different from the voltage given by the external AC - DC power supply. For various purposes of power consumption electronics devices, low voltages from a high source voltage or high voltage from a low source voltage may be required. So we use converters namely Buck or Boost converters to either step up or step down our required voltage.

The introduction of DC-DC power converters capable of producing a controlled and consistent DC power supply was credited with revolutionizing electrical energy control. The development of switching power converters allowed for better performance and improved thermal temperaments while reducing the size, weight, and cost of power supply. Switching converters also have the benefit of reducing power loss. Traditional Buck, Boost, and Buck-Boost converters are lower-order converters, while Cuk, SEPIC, and Zeta converters are higher-order converters. There is also Interleaved converters such as Interleaved Buck, Interleaved Boost etc.

Our thesis methodology utilizes an Interleaved Buck converter to generate a reliable output with acceptable performance standards. In 1976, Lilienstein and Miller initiated the idea of the interleaved converter [1]. The Interleaved Buck converter (IBC) is an updated form of Buck converter, which is basically two parallelly connected Buck converters. They are switched by interleaved gate pulse. An IBC has gained numerous attention due to the simplicity in structure and lower control complexity [2][46]

It also eliminates hot spots on a printed circuit board or a specific component. A buck converter effectively cuts RMS-current power dissipation in FETs and inductors in half.

Interleaving decreases transitory losses as well. Interleaved converters can be viewed as a topological process for EMI reduction that does not alter the converter's working principle. In fact, this is a cost-effective method of increasing system capability while lowering costs and EMI interferences. The peak-to-peak ripple of the interleaved power converter is reduced. This property greatly lowers the creation of EMI. [3][14] For the control mechanism of IBC to maintain steady-state error and transient response, Proportional Integral Derivative (PID) was induced in our thesis work. The PID controller employs a control loop to ensure system stability by adjusting the proportional, integral, and derivative gains. The notion of this controller was introduced in the 1900s, and Elmer Sperry constructed the first PID-like device in 1911 [4]. Nicolas Minorsky published his theoretical research of the PID controller in 1922. The Ziegler Nichols tuning method was used in this research project. However, such a tuning strategy has drawbacks and frequently fails to provide optimal regulation. This control method is also highly based on trial and error, which takes a long time to give controller gains, making the system less effective [5]

To enhance the control over the PID controller for optimizing the output of the Interleaved-Buck converter Particle Swarm Optimization (PSO), Firefly Algorithm (FA), Cuckoo Search Algorithm (CSA), Grey Wolf Algorithm (GWO), and Whale Optimization Algorithm (WOA) are implemented. These algorithms are metaheuristic and stochastic, which means they use random steps to get to the best solution and was influenced by biology. When constructing or using a metaheuristic, it's important to remember that the exploration of the search space and the exploitation of the best solutions must be balanced well. However, there are situations when a given algorithm fails to do so. In this case, one viable option is to create an algorithm in which multiple (at least two) algorithms are combined with the goal of improving overall performance.

This thesis discusses how to optimize the PID controller by using a hybrid optimization technique to provide better responses and construct fitness functions for the Interleaved-Buck converter. With the aid of several types of optimized controllers, a graphical representation of the results was provided, and a comparative analysis was tabulated. The comparative analysis table is created after seeing the graphical representation of all the step responses in order to have a better knowledge of the compatibility of Nature-inspired algorithms in the development of the PID controller. The gain levels and performance characteristics of the standard PID controller have been observed, as well as the parameters of the Interleaved-Buck converter. Following the conventional PID controller, the PID controller is tuned using several optimization approaches in order to reach a better outcome.

1.2 Literature Review

There are a bunch of research that exists on the implementations of Nature-inspired Algorithms to optimize the controls to design efficient systems of converters. In a paper Efendi et al. (2017) demonstrated the optimization of an Sepic converter using Modified-PSO (MPSO) to locate the GMPP without becoming trapped in the LMPP for partially shaded conditions with a quick convergence of roughly 0.5 to 1 second in order to produce an MPP

[6] and better than 95% certainty**.** Ragavendra et al. optimized the Cuk converter feeding DC motor by combining the Artificial Bee Colony (ABC) algorithm with the PI controller and achieved increasing efficiencies ranging from 93 per cent to 97.5 per cent for increasing input voltages. [7] Dr.T.Govindaraj and M.Senthamil et al has proposed a system of the interleaved boost converter (IBC) where the induction motor is controlled by swarm Intelligence - PSO algorithm. [8] Furthermore, Nishat et al. reported the increased performance of the Buck converter, governed by the Genetic Algorithm operated PID controller, with consistent and improved percentage overshoot, settling time, and rising time [9] Yaqoob et al. (2014) optimized a PID controller-regulated Buck converter using the PSO algorithm, which delivered enhanced transient responses using the performance criterion PSO-ITSE [10]Zhang, Y. Liu et al In this work provides an efficiency optimization control approach that uses the differential evolution (DE) algorithm to discover the ideal load current distribution scheme and picks the suitable operation mode based on the system state of a Interleaved Buck Converter. **[11]** Arora et al. (2020) optimized a hybrid renewable energy system utilizing the PSO algorithm on a PID controller, reducing voltage THD from 5.13 percent in conventional to 1.95 percent (with PSO-PID) and current THD from 9.80 percent (with traditional PID) to 0.12 percent (with PSO-PID). [12] All of this research work has assisted scholars and non-experts in comprehending the field of intelligent search algorithms and exercising them appropriately.

1.3 Problem Statement

Interleaved-Buck Converter has gained popularity in real-life scenarios requiring non-isolation, step-down conversion ratio, and high output current with minimum ripple due to its easy construction and low control complexity. [13] [15] [16] However, because all semiconductor devices in the typical IBC are affected by the input voltage, high-voltage devices rated above the input voltage should be used. High-voltage-rated devices are notorious for having undesirable traits such as higher cost, higher on-resistance, higher forward voltage drop, severe reverse recovery, and so on. Interleaved Buck Converter is a Due to these inherent properties controller designing of this system is a challenging task. In such instances, using a PID controller provides improved regulation and stability at the expense of time and efficiency. Thus, obtaining the appropriate controller gains is performed by the use of intelligent search algorithms that induce learning skills to analyze the situation and change the controller accordingly. The Bio-inspired-algorithms used in this thesis work are swarm intelligence exercises that use population, memory, competition, and cooperative interaction to get superior results. When PSO, FA, GWO, and WOA work together with the PID controller to improve the converter system, the whole system becomes more secure, stable, and competitive.

1.4 Thesis Objectives

The following are the objectives of this thesis work:

- Stability Analysis of Interleaved-Buck Converter.
- Apply different optimization algorithms to compare their accuracies and other performance parameters for the selected converter.
- Further, establish a more accurate model by hybridizing an algorithm with another optimization algorithm for improvement of the results.

1.5 Limitation of Study

The Interleaved Buck converter with constant input voltage was simulated in this thesis study. All simulations were carried out using MATLAB software. However, real-world applications will have more complicated and demanding conditions, which will influence the system's output. As a result, the control system designed for this scenario may not be suitable for more difficult situations. Furthermore, hardware implementations create inevitable faults and may diminish efficiencies. To retain the optimization level, the algorithms must be adjusted to accommodate the imposed difficulties.

Chapter 2

DC-DC CONVERTERS

Power electronics circuits that convert one DC voltage level to another, typically with controlled output are dc-dc converters. These are used extensively in renewable energy applications. Because of having low-voltage sources like fuel cells, PV panels, and batteries require dc/dc converters. As a result, a large step-up conversion is required. Unfortunately, when scaling up the voltage, dc/dc converters have limitations. High-step-up dc/dc converters can theoretically approach infinity voltage, however losses in the converter limit this. DC-DC converters are frequently used to convert direct current voltage between levels, with the output voltage controlled to avoid interference. The term "dc-dc converter" refers to a device with a DC input and a DC output. It is projected to produce constant output voltage over a wide variety of resistances and input voltages. Variable resistance is formed by different consumers in the electric circuit and varying input voltage, which is frequently a result of discharging when batteries are used as a source of electric current. Buck and boost converters are the two most common DC-DC converters. The other converters are generated from these two converters. The converter functions as a filter circuit thanks to the various combinations of inductors and capacitors. The uncontrolled DC input is converted into a controlled DC output at a chosen level using a switch-mode DC-DC converter. Due to lower losses in the electronic switch, a switch-mode DC-DC converter is substantially more efficient than a linear converter. Because of their benefits, these converters have become an essential component of renewable energy power plants, portable devices, and industrial systems.

2.1. Controls of DC-DC Converters

Two control objectives are required in the control of DC-DC converters: performance and efficiency. Switching mood in DC-DC converters necessitates the use of one or more switches to shift DC from one level to another. For example, at low loads, the switching frequency can be lowered, or the control circuit can be switched between a constant on-time and a constant off-time control circuit based on the load conditions. Though the output load and input voltages may fluctuate, to get the desired level, the average DC output voltage must be changed. Switch–mode to convert DC from one level to another, DC-DC converters use one or more switches. The average output voltages of a dc-dc converter with a particular input voltage are regulated by adjusting the duration of the switches on or off. We may regulate the converters simply by adjusting the duty cycle, where the duty cycle is the percentage of output and input. We can also change the components in the electrical circuits to control the efficiency and stability of the converter.

2.2 Topologies of Converter Circuit

DC-DC Converters are gaining popularity these days due to the increasing applications of renewable sources such as PV cells, Solar cells, Wind turbines etc. [17] Practically, switching mode dc-dc converters have impacted more on the conversion process for its lower ripple and faster transient response. Different types of dc-dc converters have low cost, compact size and high efficiency.

2.2.1 Buck Converter

This converter, commonly known as a step-down converter, delivers a lower average output voltage than the DC input voltage [18]. Conceptually, the fundamental circuits are a completely resistive load, an ideal switch, and a constant momentary input voltage, a diode, a capacitor, and an inductor. The buck converter circuit works by adjusting the amount of time that the inductor gets energy from the source. The converter's basic configuration is depicted in Fig. 2.2(a), which includes one switching element, a rectifier, and filter components. The converter is assumed to be in CCM, with the inductor current never reaching zero. The duty ratio is used to compute the average output voltage. The output voltage can be changed by adjusting the duty cycle. Another key consideration is that by adjusting the duty cycle, the output voltages change linearly with the input voltage. When the switch is turned on, the diode is reverse biased, and the input voltage charges the inductor. When the switch is turned off, the diode becomes forward biased and the inductor discharges to the load [19] [20]The voltage gain, G = D, where D denotes duty cycle. Traditional step down converters were diode-rectified devices, and diode-rectified step down converters were referred to as buck converters. Whatever nomenclature are used, there are a variety of step-down processes employed in step-down converters, and the step down converter in this case is the previously stated diode-rectified device.

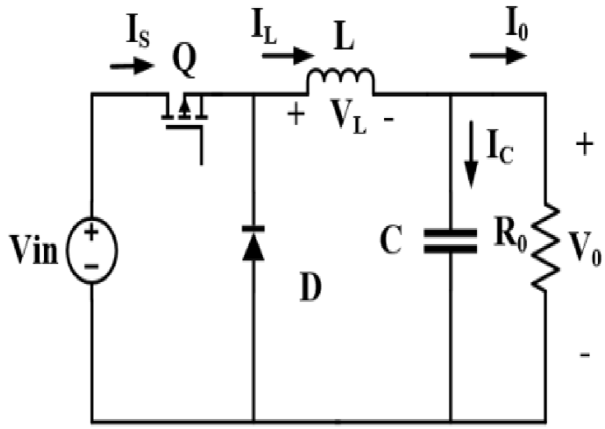


fig:2.2.1(a) Buck Converter Circuit

2.2.2 Boost Converter

A boost converter is a type of step-up converter that converts a lower dc input voltage to a greater dc output voltage. Because the output voltage is much higher than the input voltages, this is also used in solar photovoltaic systems, trolley cars, forklift trucks, and electric automobiles. It offers excellent efficiency, tight controls, and quick responses. To achieve the voltages needed to sustain the high gain, a power diode is used. Because the boost converter is a popular converter in power electronics circuits, some modified converters are derived from it. The basic converter arrangement is shown in Fig. 2.3. (a). As shown in (a), the circuit has a voltage source, an inductor, a capacitor, a switching device, a diode, and a resistance. There are 9 loading elements. When the switch is turned on, the diode is reverse biased. The current flows from the voltage source to the switching device, storing energy in the inductor. Changes in the input voltages, as well as the duty cycle of this converter, are used to examine the output voltages. The diode is forward biased and the switch is turned off in the OFF state. The inductor transfers the stored energy to the output load while maintaining a constant current flow and increasing the output voltage. Converter's gain, G = 1/(1 D) where D is the duty cycle.

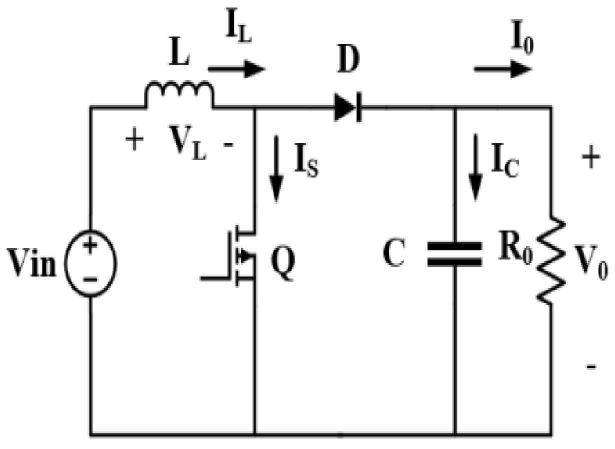


Fig: 2.2.2(a) Boost Converter Circuit

2.2.3 Buck-Boost Converter

A buck-boost converter is an inverting converter that can function as both a step-up and a step-down converter. As a result, depending on the duty cycle, the output voltage is either higher or lower than the input voltage. Figure 2.2.3 depicts the buck-boost converter topology (a). The converter's components are a dc input voltage source, an inductor, a capacitor, a diode, a controlled switch, and load resistance. When the switch is turned on, the diode is turned off and the inductor is charged. The current will grow as it passes through the voltage, switch, and inductor. When the switch is turned off, the diode becomes forward biased, and the inductor stores the energy to the capacitor and load resistance. As a result, the voltage gain of the buck-boost circuit is G = D/ (1 D).

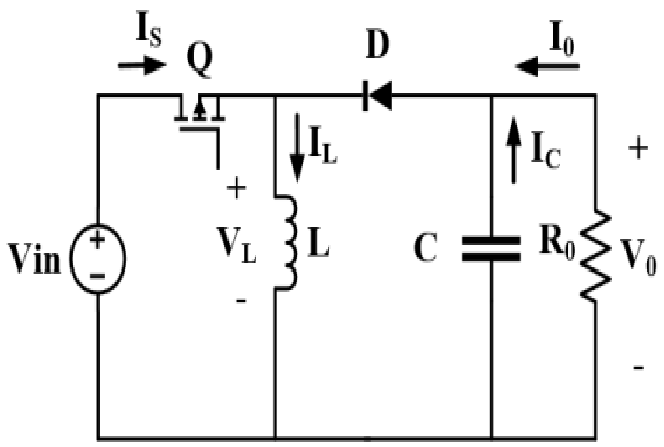


Fig: 2.2.3(a) Buck-Boost Converter Circuit

2.3 Interleaved Buck Converter

2.3.1 Outline of Interleaved Buck Converter

This circuit connects an interleaved buck converter, a drive circuit, and a microcontroller. The interleaved buck converter is formed by connecting two single-phase buck converters in parallel. When each switch is controlled via interleaving, the difference between the two PWM signals is 180 degrees. When the input current is passed through two-buck inductors, the inductor size and inductance can be lowered since the magnitude of each inductor current is reduced to one per phase. Because the input current is the sum of the currents of inductors L1 and L2, the input current ripple is decreased. The controller sends a pulse to the driver circuit with the necessary output, and the mode in the controller is ON/OFF.

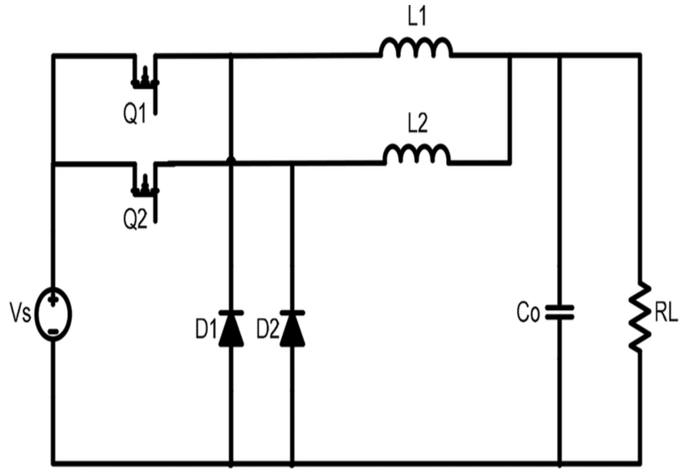


Fig: 2.3(a) Interleaved Buck Converter

2.3.2 State Space average method

State Space Average Modeling

1) Mode 1: Two diodes are not conducting and both switches are turned on. For mode 1, the differential equation is as follows:

$$L_a \frac{dI_a}{dt} + R_a I_a = V_i - V_0$$

$$L_b \frac{dI_a}{dt} + R_b I_b = V_i - V_0$$

$$A = d_{ma}.A_a + d_{mb}.A_b + d_{mc}.A_c + d_{md}.A_d$$

$$B = d_{ma}.B_a + d_{mb}.B_b + d_{mc}.B_c + d_{md}.B_d$$

2) Mode:2 In the second mode S1 is ON and S2 is OFF. D2 conducts the current in phase 2. The differential equations for this mode are expressed as

$$L_a \frac{dI_a}{dt} + R_a I_a = V_i - V_0$$

$$L_b \frac{dI_a}{dt} + R_b I_b = -V_0$$

$$C \frac{dV_0}{dt} + \frac{V_0}{R} = I_a + I_b$$

3) Mode 3: For the operation mode 3, S1 is OFF and S2 is ON. In this mode D1 conducts the current in phase 1. The differential equations for this mode are expressed as

$$L_a \frac{dI_a}{dt} + R_a I_a = -V_0$$

$$L_a \frac{dI_a}{dt} + R_a I_a = V_i - V_0$$

$$C \frac{dV_0}{dt} + \frac{V_0}{R} = I_a + I_b$$

4) Mode:4 In the last mode, both switches are OFF and both diodes are conducting the currents for both phases. The differential equations for this mode are expressed as

$$L_a \frac{dI_a}{dt} + R_a I_a = -V_0$$

$$L_b \frac{dI_a}{dt} + R_b I_b = -V_0$$

$$C \frac{dV_0}{dt} + \frac{V_0}{R} = I_a + I_b$$

Averaged SS Model

The differential equations of the two-phase interleaved buck converter are represented by four separate SS equations. The SS representations for each mode are provided by for the system with the state vector x and the input vector u.

$$\widehat{x} = A_p.\dot{x} + B_p.u.$$

$$\mathrm{y} = \mathrm{C}_p.\dot{x} + D_p.u.$$

By rewriting (1-4) as SS equations, the behavior of interleaved buck converter in mode 1 to 4 can be expressed as follows respectively

$$\frac{d}{dt}\begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix} = \begin{bmatrix} \frac{-R_a}{L_a} & 0 & \frac{-1}{L_a} \\ 0 & \frac{-R_b}{L_b} & \frac{-1}{L_b} \\ \frac{1}{C} & \frac{1}{C} & \frac{-1}{R_c} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} \\ \frac{1}{L_b} \\ 0 \end{bmatrix} V_i$$

$$\frac{d}{dt}\begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix} = \begin{bmatrix} \frac{-R_a}{L_a} & 0 & \frac{-1}{L_a} \\ 0 & \frac{-R_b}{L_b} & \frac{-1}{L_b} \\ \frac{1}{C} & \frac{1}{C} & \frac{-1}{R_c} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} \\ 0 \\ 0 \end{bmatrix} V_i$$

$$\frac{d}{dt}\begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix} = \begin{bmatrix} \frac{-R_a}{L_a} & 0 & \frac{-1}{L_a} \\ 0 & \frac{-R_b}{L_b} & \frac{-1}{L_b} \\ \frac{1}{C} & \frac{1}{C} & \frac{-1}{R_c} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L_b} \\ 0 \end{bmatrix} V_i$$

$$\frac{d}{dt}\begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix} = \begin{bmatrix} \frac{-R_a}{L_a} & 0 & \frac{-1}{L_a} \\ 0 & \frac{-R_b}{L_b} & \frac{-1}{L_b} \\ \frac{1}{C} & \frac{1}{C} & \frac{-1}{R_c} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} V_i$$

Output voltage vo and currents difference i12 = i1 − i2 are selected as output variables of the averaged SS model.

$$\begin{bmatrix} V_0 \\ I_{ba} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ V_0 \end{bmatrix}$$

The averaged SS model can be obtained from the above as follows.

$$A = d_{ma}.A_a + d_{mb}.A_b + d_{mc}.A_c + d_{md}.A_d$$
$$B = d_{ma}.B_a + d_{mb}.B_b + d_{mc}.B_c + d_{md}.B_d$$

Chapter 3
NATURE-ISPIRED ALGORITHMS

3.1 Particle Swarm Optimization (PSO)

Various swarm intelligence (SI) techniques are currently being used to optimize control parameters. PSO (Particle Swarm Optimization) is an important component of a broad family of SI techniques created by Eberhart and Kennedy in 1995, inspired by social, behavioral tendencies observed in biological swarm creatures [22]**.** It is a population-based meta-heuristic stochastic optimization technique that can solve difficult, nonlinear real-world optimization problems. This algorithm's premise is comparable to bird flocking and fish schooling, where each bird or fish is technically a 'Particle' and the flocks constitute a 'Particle population.' Based on their personal and societal influence, all particles will converge on the historical best place. The algorithm is modeled after a flock of animals without a leader who seek food by following the closest individual to the food source. By interacting with their members and remembering their previous situation, the flocks achieve their worldwide optimal state. Other flocks will be informed, and they will all work together to achieve the best worldwide position.

The position and velocity of the particles were randomly initialized in the PSO method due to mathematical constraints. Particles receive personal best-optimized solutions (Pbest) and wander around a pre-defined search space in order to find the global best (gbest). Equations such as weight value, coefficient of personal best value, coefficient of global best value, and others aid position and velocity.
The following equations control the velocity and position of particles,

$$V_{ij}^{k+1} = W \times V_{ij}^{k} + C_1 \times rand() \times [pbest_{ij}^{k} - X_{ij}^{k}] + C_2 \times rand() \times [gbest_{j}^{k} - X_{ij}^{k}]$$
$$X_{ij}^{k+1} = X_{ij}^{k} + V_{ij}^{k+1} \qquad (3.1)\ (3.2)$$

Originally velocity of each particle is bounded within the range of [−Vmax, +Vmax]. Although being a vital parameter creates problems with the oscillation of the particles [23].

The subsequent steps of PSO algorithm portrayed below,

Step 1. Initialization of Parameters

Step 2. Determining Fitness value for each particle

Step 3. If current > previous, update the Current as the Best or Keep the previous as Best

Step 4. Assign local best as Global Best

Step 5. Update the position and the velocity

Step 6. If Maximum Iteration or Minimum Error is found, then Global best is the optimum solution determine or repeat step 2

3.2 Cuckoo Search

The productivity of optimization techniques can be attributed to the way they replicate nature's best features, particularly the survival of the fittest in biological frameworks that have evolved over time through regular selection. Metaheuristics have two important characteristics: one ensures that the calculation can study the search space more effectively, often by randomization, and the other ensures that the computation can look around the existing best arrangements and find the optimal answer. The Cuckoo search (CS) algorithm is a metaheuristic that outperforms existing algorithms such as PSO.

Although certain cuckoo species, such as the ani and guira, deposit their eggs in communal nests, they may remove the eggs of other cuckoos to increase the chances of their own eggs hatching. Several species are forced to lay their eggs in the nests of other host birds. Brood parasitism is classified into three types: intraspecific brood parasitism, cooperative breeding, and nest takeover. Invading cuckoos can create a direct conflict with some host birds. If a host bird recognizes the eggs are not its own, it will either throw them away or abandon the nest and build a new one somewhere else. Some cuckoo species, such as the global brood parasitic Tapera, have evolved to the point where female parasitic cuckoos are exceptionally competent at imitating the colour and pattern of a few select hosts.[55]

3.3 Firefly Algorithm

Fireflies are among the most endearing of all insects, and their dazzling courting rituals have inspired both poets and scientists. Fireflies dwell in a range of heated habitats and are most active at night throughout the summer. FA is a modern swarm intelligence approach invented by Yang in 2008, and it is a type of stochastic, nature-inspired, meta-heuristic algorithm that may be used to solve the most difficult optimization problems.

When opposed to single-point search algorithms, population-based algorithms provide the following benefits.

- Building pieces are created by combining multiple solutions.
- Crossover demonstrates that if both parents have the same value for a variable, the offspring will also have the same value.
- Low-pass filtering eliminates environmental distractions.

· Early holdings or choices might be used to hedge against bad luck.

· In order to balance exploration and exploitation, the software may determine ideal parameter values through parameter tweaking**[24][25][26]**

3.3.1. Methodology

This approach is based on a physical formula for light intensity I, which diminishes as the square of distance r increases. As the distance between the light source and the observer grows, light absorption causes the light to weaken and fade. These phenomena are linked to the goal function, which has to be enhanced. In the algorithm, the issue to be solved is turned into an objective function. The firefly's brightness is controlled by the target function, which is set as low as possible to achieve our goal. Because brighter fireflies imply optimal responses, this verifies the acquisition of superior solutions. The free movement of the firefly, which is governed by the attraction factor of relevant fireflies, their distance from one another, random functions, and their capacity to absorb light, results in reasonable solutions.**[27]**

The approach begins with a population of fireflies prepared to survey the search space set by the programmer. Each agent advances to a place in the search area based on the attraction of the other fireflies to generate a solution. When following the enticing firefly, both brightness and distance are taken into account. Position acquisition is further aided by the light absorption coefficient and the randomization factor. The beautiful firefly's present position is then appraised in terms of attractiveness. If the new job's attractiveness exceeds that of the prior position, the associated position is used to update the new position. If the attraction is not stronger, the firefly will remain in the same place. When the stated number of iterations have been finished, the procedure is terminated. The firefly's positions are being revised as better possibilities are explored.

3.3.2. Applications

In reality, FA and its variations have been utilized to handle a wide variety of optimization and classification problems, as well as engineering obstacles. As indicated in the graphic, FA has been applied to the following types of optimization problems: continuous, combinatorial, constrained, multi-objective, dynamic, and noisy optimization. It has also been used in machine learning, data mining, and neural networks to tackle categorization problems. Finally, firefly algorithms are used in nearly every technological industry. In this review, we emphasized image processing, industrial optimization, wireless sensor networks, antenna design, business optimization, robotics, semantic web, chemistry, and civil engineering **[28].**

3.3.3 Mathematical Model of Firefly Algorithm

The movement of each firefly in the search space can be evaluated with the help of the following equations

$$x_{cn}^{(t+1)} = x_{cn}^{t} + \rho_a (x_b^t - x_n^t) + ar \qquad (3.3)$$

The equation below is used to calculate the co-efficient of attraction

$$\rho_a = \rho_0 \times e^{(-\eta \times h_{bn}^{2})} \qquad (3.4)$$

The brightness of the firefly n on the brighter firefly b is computed using light intensity physics in the form of the equation below.

$$i_i(h_{bn}) = \frac{i_q}{h_{bn}^{2}} \qquad (3.5)$$

3.3.3. Flowchart of Firefly Algorithm

The algorithm's method is shown chronologically in the form of a simple pseudo code.

The following are the steps in the FA:

Step 1. To begin the procedure, create a random population of fireflies.

Step 2: Determine which of the target functions should be lowered.

Step 3: Calculate the coefficient of attraction.

Step 4: Determine the location of each firefly.

Step 5: Evaluate the firefly's beauty in comparison to its prior place.

Step 6: If the new job's attractiveness grows, it should be replaced with the comparable position.

Step 7: If the position's appeal does not increase, it will not be updated.

Step 8. Return to step 3 if the total number of iterations has not been achieved.

Step 9: The algorithm should be halted after the entire number of iterations has been achieved

3.4 Grey Wolf Optimizer Algorithm

In 2014, Mirjalili et al. presented the Grey Wolf Optimizer (GWO). It has hunting and leadership qualities similar to grey wolves. Grey wolves are members of the Canidae family and have a strict social hierarchy. They prefer to hunt in groups of 5 to 12 wolves. There are many different types of wolves in a pack, depending on their level of dominance. As the leader of the wolf pack, the wolf holds the greatest status. It could be a male or female wolf. It is in charge of making all decisions for the entire pack, such as hunting, discipline, and sleeping and waking times. [30] Subordinate wolves aid the leader in decision-making and other activities at the second level. The wolf who ranks second in the pack stands the best chance of becoming the pack's leader. The wolves of the third level, wolves, dominate the wolves of the fourth and final level, wolves, who are in charge of ensuring the safety and integrity of the wolf pack.

3.4.1. Hierarchical Structure

The GWO algorithm is based on the social hierarchy of wolves. The best solution α is found at the top of the social hierarchy, i.e. Similarly, β and δ are regarded as the second and third best options. The remaining candidate solutions are assumed to be α wolves, β and δ wolves

3.4.2. Encircling Prey

Wolves usually hunt in groups. This means they work together to catch and eat a prey in an intelligent manner. Grey wolves pursue prey in packs, attempting to encircle it, change its movement direction, and increase the likelihood of a successful hunt. The mathematical equations modeled in GWO for encircling characteristic of the wolves are represented in Eq. (3.6) and Eq. (3.7)

$$\vec{D}_a = \left| \vec{C}_a . \vec{X}_{PQ}(t) - \vec{X}_a(t) \right| \tag{3.6}$$

$$\vec{X}_{a(t+1)} = \vec{X}_{PQ}(t) - \vec{A}.\vec{D}_a \tag{3.7}$$

$\vec{D}_a$ stands for the distance between the prey and the wolf. At iteration t, $\vec{X}_a$ denotes the wolf's position vector, while $\vec{X}_{PQ}(t)$ denotes the prey's position vector. The random vectors and are calculated as shown in Equations (3.8) and (3.9).

$$\begin{aligned} \vec{A}_P &= 2\vec{a}.\vec{r}_a - \vec{a} \\ \vec{C}_P &= 2.\vec{r}_b \end{aligned} \qquad (3.8)\ \ (3.9)$$

Here, $\vec{r}_a$ and $\vec{r}_b$ are the random vectors in the range of [0, 1]

3.4.3 Hunting the Prey

Using the equations above, a wolf can relocate to any location in a hyper-sphere around its prey. This, however, is insufficient to mimic grey wolf social intelligence. To imitate the prey, we assume that the prey's position is the best answer determined thus far (which is the alpha wolf). This is a required assumption because we don't know where the global optimum is in real-world circumstances. We presume the alpha is aware of the prey's whereabouts because he is in charge of the pack. After defining the hierarchy, developing an encircle equation, and identifying the position of prey, the position of each wolf can be updated using the following equations.

$$\vec{X}_{\mathrm{a}(t+1)} = \frac{X_a + X_b + X_c}{3} \tag{3.10}$$

Where,

$$\vec{X}_a = \vec{X}_{a\alpha}(t) - \vec{A}_a.\vec{D}_{a\alpha} \tag{3.11}$$

$$\vec{X}_b = \vec{X}_{a\beta}(t) - \vec{A}_b.\vec{D}_{a\beta} \tag{3.12}$$

$$\vec{X}_c = \vec{X}_{a\delta}(t) - \vec{A}_c.\vec{D}_{a\delta} \tag{3.13}$$

And,

$$\vec{D}_\alpha = \left|\vec{C}_a.\vec{X}_\alpha - \vec{X}_a\right| \tag{3.14}$$

$$\vec{D}_\beta = \left|\vec{C}_b.\vec{X}_\beta - \vec{X}_a\right| \tag{3.15}$$

$$\vec{D}_\delta = \left|\vec{C}_c.\vec{X}_\delta - \vec{X}_a\right| \tag{3.16}$$

3.4.4. Searching & Attacking The Prey

Grey wolves attack only when the prey stops moving. It is mathematically modelled using the **A** vector from Eq (3). **A** is a random vector with a value in the range [a, a], with **A** decreasing from 2 to 0 over the course of iterations using Eq (7).

$$\vec{a} = 2 - (2 \times t / Maximum_{iteration}) \tag{3.17}$$

3.4.5. Pseudo Code of GWO

Step 1. Initialize the number of grey wolves Xai (i = 1, 2,3, . . . , n)

Step 2. Initialize parameters a1, A1 and C1

Step 3. Calculate fitness of Searching Component (Wolfie)

Step 4. Run the algorithm for maximum number of iterations

Step 5. Finding The most suitable Searching Component per wolfie X1

Step 6. Finding The next best Searching Component per wolfie X2

Step 7. Finding The prior best Searching Component per wolfie X3

Step 8. For each Searching Component it has an improvement

Step 9. If the requirement done then update the placement of current Searching Component

Step 10. If the maximum number of iterations is not reached, continue step 4 of the procedure.

Step 11. When the iterations are completed, end the program

3.5. Whale Optimization Algorithm (WOA):

Mirjalili and Lewis initiated the Whale Optimization Algorithm (WOA) in the year 2016 [31]. The idea of this algorithm mimics the humpback whale's distinct foraging behavior. Whales are the world's largest mammals, and they are recognized as extremely intelligent and emotional creatures. They are usually seen in groups, and some species lives in the same family during their lifetime. They are one of the largest baleen whales. The most intriguing part of humpback whales is their distinct hunting method. [32]. This type of foraging activity is known as bubble-net feeding and exclusively found in humpback whales. This required their evolutionary process of figuring out how to catch prey. While encircling prey while hunting, the whales blow bubbles in a circular pattern. Simply described, bubble-net hunting involves humpback whales diving down roughly 12 meters, producing a spiral-shaped bubble around their prey, and then swimming up to the surface to track the bubbles. The mathematical model for this Optimization algorithm is as follows in order to perform optimization:

3.5.1 Mathematical model:

The main mechanisms used in the WOA algorithm are consists of encircling the prey, assaulting with a bubble net, and looking for the prey.

3.5.1.1 Encircling prey:

Humpback whales can track down and surround their prey. The WOA algorithm takes into account the current best search agent position, which is either the target prey or close to it, and. As the best search agent is determined, the other search agents will try updating their position with respect to the best search agent. This phenomenon can be explained using some equations:

$$\boldsymbol{D} = |\boldsymbol{C} \cdot \boldsymbol{X^b}(t_i) - \boldsymbol{X^p}(t_i)| \tag{3.18}$$

$$\boldsymbol{X^p}(t_i + 1) = \boldsymbol{X^b}(t_i) - \boldsymbol{A} \cdot \boldsymbol{D} \tag{3.19}$$

Here | | is the absolute value of the vector. The current iteration is denoted by the t_i. The location vector of the best solution acquired thus for iteration t_i is $X^b(t_i)$. And each search agent's position vector is denoted by $\mathbf{X^p}(t_i)$. '·' is the elements wise Multiplication. The coefficient vectors **A** and **C** that are updated to reflect the prey's most recent position around the search agent following by Eq.

$$\boldsymbol{A} = 2\boldsymbol{a} \cdot rand1 - \boldsymbol{a} \tag{3.20}$$

$$\boldsymbol{C} = 2 \cdot rand2 \tag{3.21}$$

here **a** reduces linearly from 2 to 0 during the iteration process generated for each dimension, and rand1 and rand2 are designated as the random numbers between 0 and 1 for each dimension.

3.5.1.2 Bubble-net attacking method:

The Bubble-net technique combines two approaches that may be modelled mathematically as follows:

Shrinking Encircling Mechanism:

Whale behavior was emulated by lowering the value of a variable in the formula (3.3). Take note of **A**'s variation range is also reduced by **a**. In other words, A is a random number in the range [-a,a], where a decreases from 2 to 0 over iterations. A search agent's new position can be calculated anywhere between its initial location and the current best agent's position by assigning random values for A between -1 and 1.

Spiral Updating Position:

To replicate the helix-shaped behavior of humpback whales, the corresponding spiral equation is created between the positioning of whale and prey:

$$\boldsymbol{X^p}\,(t_i + 1) = \boldsymbol{D^*}.\ \boldsymbol{e^{cr}}.\ cos(2\boldsymbol{\pi}r) + \boldsymbol{X^b}(t_i) \tag{3.22}$$

$$\boldsymbol{D^*} = |\boldsymbol{X^b}(t_i) - \boldsymbol{X^p}\,(t_i)| \tag{3.23}$$

$\mathbf{D^*}$ is the distance between the best solution of ith whale and the prey and c is a constant which denotes the form of the logarithmic shape. And r is a random value between -1 to 1. Humpback whales do, in fact, swim in a spiral pattern while staying within a diminishing circle. Using a 50% chance, repetitions of the program imitate either the diminishing encircling movement or the spiral model movement. So, a probability factor *p* is used to modify the position of whales, between the diminishing encircling mechanism and the spiral model.

3.5.2 Searching the prey:

Random selection is used in almost every meta-heuristic algorithm to find the best solution. Because the location of the ideal design of the bubble-net approach is unknown, Based on their relative positions, humpback whales forage randomly. In contrast to the exploitation phase, which used **A** in the interval of -1 to 1, this phase uses **A** as a vector of random values larger than 1 or less than -1. These two activities can be expressed as follows:

$\mathbf{D} = |\mathbf{C} \cdot \mathbf{X_r}(t_i) - \mathbf{X^p}(t_i)|$

$\mathbf{X^p}\,(t_i + 1) = \mathbf{X_r}(t_i) - \mathbf{A} \cdot \mathbf{D}$

Here, $\mathbf{X_r}(t_i)$ is a position vector generated randomly.

3.5.3 Pseudo Code of WOA Algorithm:

Step 1. Initialize each search agent of whales Xipop (i = 1, 2, 3..., n)
Step 2. Declare the objective function and compute the fitness for each agent
Step 3. Run the algorithm for a specified number of iterations.
Step 4. If $p < 0.5$ & $|A| < 1$ update the search agent position by the Eq 3.1
Step 5. If $p < 0.5$ & $|A| \geq 1$ choose a random agent and modify the search agent by the Eq 3.2
Step 6. Else if $p \geq 0.5$, Modify the position by the Eq 3.3
Step 7: Verify the search space limit.
Step 8: Make changes to the variables.
Step 9: If the entire number of iterations is not reached, go back to step 3 and repeat the process.
Step 10. When all iterations have been finished, stop the program.

3.6. Hybrid Firefly Particle Swarm Optimization Algorithm (HFPSO):

The firefly algorithm lacks a velocity (V) characteristic. As a result, fireflies relocate notwithstanding their previous optimal placements [33]. In some applications, the particle swarm optimization approach can achieve quicker convergence than previous algorithms [34,35]. This quick ability of convergence weakens and slows down in the local search stage of PSO, especially when looking for a solution in the solution space close to a global optimum solution. PSO control parameters can be utilized to properly balance exploration and exploitation. In local search, the amount of particle velocity is important, but determining the right velocity may be difficult, as incorrect numbers might lead particles to bounce about the ideal solution like a pendulum. These oscillations or swings create some delays in the optimization process. Because of this issue, velocity might be overlooked during the exploitation stage.
In 2018, Aydilek proposed an optimization technique that involves search ability of firefly and particle swarm optimization algorithm [36]. The goal of this combination is to achieve a harmony between exploration and exploitation, which enhances both algorithms' capabilities [37]. In comparison to particles, lacks a velocity (V) characteristic or personal best position (p-best) memory. PSO is commonly utilized in the global search in the suggested hybrid combination of two algorithms because it gives quick convergence in exploration. FA is also commonly employed in local search since it allows for fine-tuning of exploitation. Studies using dynamically adjusted inertia weight that take into account gains over prior personal bests have been successful [38].

3.6.1 Mathematical model:

First, input parameters are added that will be used by both algorithms in the next phases. Then, uniform particle vectors are generated at random in the pre-defined search and velocity range. Particles for the global (g-best) and personal best (p-best) are computed and allocated. The subsequent comparison stage compares whether the particle's fitness value has improved in the last iteration. The current location is then recorded in a temp variable (Xi temp), and current position and velocity are determined using Equations (3.24) and (3.25) respectively.

$$\mathrm{Xp_i}(t+1) = \mathrm{X_{pi}}(t) + \mathrm{B}e^{-\gamma r^2}(\mathrm{Xp_i}(t) - \mathrm{gbt}^{t-1}) + \varepsilon i \tag{3.24}$$
$$\mathrm{Vp_i}(t+1) = \mathrm{Xp_i}(t+1) - \mathrm{X_i} \tag{3.25}$$

If a particle has a greater or equal fitness value than the previous global best, local search is presumed to begin and the particle is treated by an imitative FA; otherwise, the particle is handled by PSO, and PSO continues its regular operations with this particle according to Eq. (3.1) and Eq. (3.2). For all particles and firefly, fitness function evaluations and range constraints are examined in the next comparison stage. If the maximum iteration limit is achieved, the hybrid algorithm will be ended, and the result of the proposed hybrid algorithm will be g-best and its fitness value.

The suggested HFPSO optimization technique is given pseudo code in Algorithm 1. MaxFES stands for maximum number of fitness function evaluations. MaxFES is a prominent termination criteria in evolutionary computing that allows for the maximum computation of objective functions [39]. In PSO [40], the inertia weight (w) parameter aids in balancing exploration and exploitation. To restrict the next distance in a direction, a particle's maximum and lowest velocities are used. They are prepared at random in the velocity range at the start of the suggested procedure.

3.6.2 Pseudo Code of HFPSO algorithm

Step 1. Initialize the input parameters of the algorithm

Step 2. Initialize the population matrix in the search range generating particle randomly.

Step 3. Initialize velocity of the particle in the velocity range randomly

Step 4. Specify the associated objective function for the optimization problem.

Step 5. Run the algorithm for maximum number of iterations

Step 6. Compute the local and global fitness values

Step 7. For each particle if it has an improvement

Step 8. If true, then update the position and velocity according to Eq. (3.24) (3.35)

Step 9. Else Update inertia weight (w), position and velocity according to Eq. (3.1) (3.2)

Step 10. Examine the position and velocity range restrictions

Step 11. If particle’s fitness value smaller than local and global best, then assign new best

Step 12. If the maximum number of iterations is not reached, continue step 5 of the procedure.

Step 13. When the iterations are completed, end the program

Chapter 4

IMPLEMENTATION OF NATURE-INSPIRED ALGORITHM BASED PID CONTROLLER FOR DC-DC CONVERTER

By ushering DC-DC converters via the continuous growth of power regulators from linear to switching stage, Power Electronics has ushered in a new sort of industrial revolution in the conversion and management of electric power [41] [42] [43]**.** LED drivers, laptops and computers, electric autos, hydropower plants, solar systems, and a number of other applications employ DC-DC converters. As a result of the switches' operation, these converters have nonlinear temperaments, with higher output voltage ripples and peak overshoot. As a result, multiple control systems concentrating on managing the output voltage are used to improve performance in DC-DC converters. PID control is the most well-known and widely utilized of the several control techniques for power converters. Obtaining the precise value of the PID parameters using the old approach, on the other hand, is difficult and time consuming. As a result, NIA is coupled to the PID controller in order to achieve the optimal PID parameter values for closed-loop stability evaluation of the DC-DC converter. [44] [45] [46] [47] [48] [49] [50].

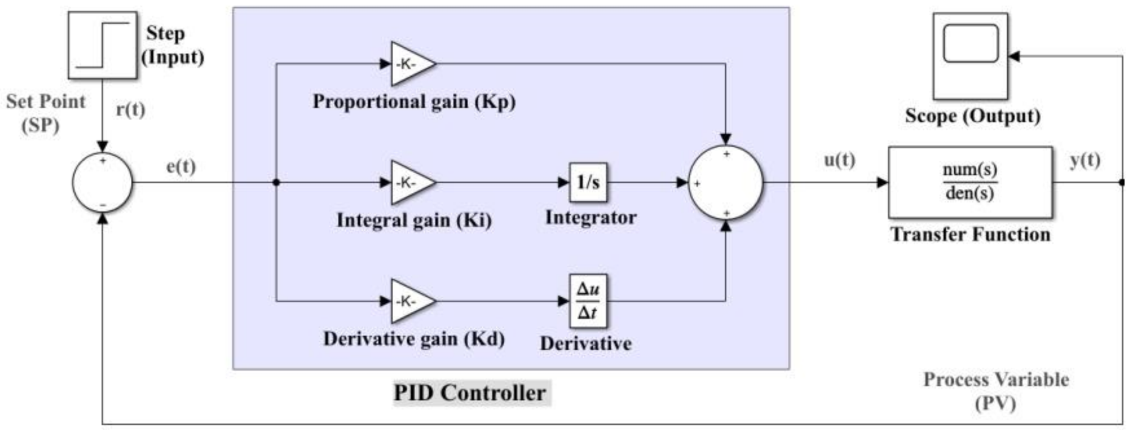
4.1. PID Controller

PID control has had more success and failure than any other controller in the control of dynamic systems. The PID controller is the most extensively utilized of all control design techniques. PID controllers account for almost 85% of all dynamic controllers. The PID controller is offered in a range of models and design techniques. Proportio-Integro-Differential control is abbreviated as PID. In a control algorithm, each of these terms, P, I, and D, plays a different role. Some sentences are removed from the control design since they aren't required. A PI, PD, or even a simple P control might be used. Having an ID control is rare [51].

.

4.2. Interpretation of the Terms

We'll now look at each term separately, assuming that the others are all zero. When Ki = Kd = 0, we simply have u (t) = Kp e (t) (t). As a result, control is proportional to the degree of inaccuracy at any particular point in time. It is a function of the current value of the error. The control



signal increases in direct proportion to the error. This proverb might be interpreted to mean that the more away we are from our objective, the harder we attempt. We don't put in as much effort as we approach closer. If we are on goal, we stop trying. As this illustration indicates, as we get near to the objective, the control practically accomplishes nothing. As a result, if the system deviates from the objective, the control has limited ability to correct it. The crucial word has now appeared.

Assuming now that Kp = Kd = 0, we simply have

$$u(t) = K_i \int_0^t e(\tau) d\tau$$

This integral is used to generate the Type I open-loop forward route. As a result, the steady-state error to a step input is guaranteed to be zero if the system is stable. This is also an application of the internal model idea. The control signal develops in size if e (t) is non-zero for a long time (for example, positive). As a result, if the plant's production starts to slip, it must react. The integral term can be viewed as a collection of the error's past values. The proportional gain and the integral gain are typically connected.

$$K_i = \frac{K_P}{\tau_i}$$

The I word is not commonly used by itself. It is most commonly used with the P word to give a PI control. The I term has a delaying impact on system responses. To speed up system responses, we add the derivative term.

Assuming now that Kp = Ki = 0, we have

$$u(t) = K_d \frac{de(t)}{dt}$$

The sooner the mistake reacts, the larger the control effort. This change in the error shows where it's going. As a result, the derivative term may be thought of as a function of the error's future values [51].

4.3. Objective Function

A goal function is a real-valued function that is maximized or decreased to improve a system's responsiveness. As a consequence, many integral performance functions including

IAE, ITAE, ISE, and ITSE are used in this case to improve system stability while lowering steady-state error [52].

1. **Integral Absolute Error (IAE):**

The IAE approach is based on the integration of absolute error over time, with no weight given to system response faults. The mathematical expression used by IAE is:

$$IAE = \int_0^{\tau} |e(t)|\, dt$$

2. **Integral Time-weighted Absolute Error (ITAE):**

The integration of the absolute error multiplied by time across time with weighted errors is the basis of ITAE. The mathematical expression for ITAE is:

$$ITAE = \int_0^{\tau} t.|e(t)|\, dt$$

3. **Integral Squared Error (ISE):**

In ISE, which is based on the integration of the square of the error over time, large errors are prioritized for eradication. The mathematical expression for ISE is:

$$ISE = \int_0^{\tau} \mathrm{e(t)}^2\, dt$$

4. **Integral Time Squared Error (ITSE):**

The integration of the square of the error multiplied by time across time is the basis of the ITSE approach. The mathematical expression for ITSE is:

$$ITSE = \int_0^{\tau} t.\mathrm{e(t)}^2\, dt$$

Here, for our optimization problem we use ITAE and ITSE objective function.

4.4. Layout of Nature Inspired Algorithm based PID Controller:

Closed-loop stability study for DC-DC converters is performed using the NIA (FA, PSO, GA, and ABC) technique, which involves adjusting a PID controller to maximize KP, KI, and KD values [53] [54] [52].

Figure 4.2 illustrates a configuration. The error is continuously regulated, resulting in a compatible output. The performance of the goal function is evaluated for this purpose utilizing four error formulae (IAE, ITAE, ISE, and ITSE) done by NIA. As a result, the optimal solution achieved through this procedure produces satisfying outcomes.

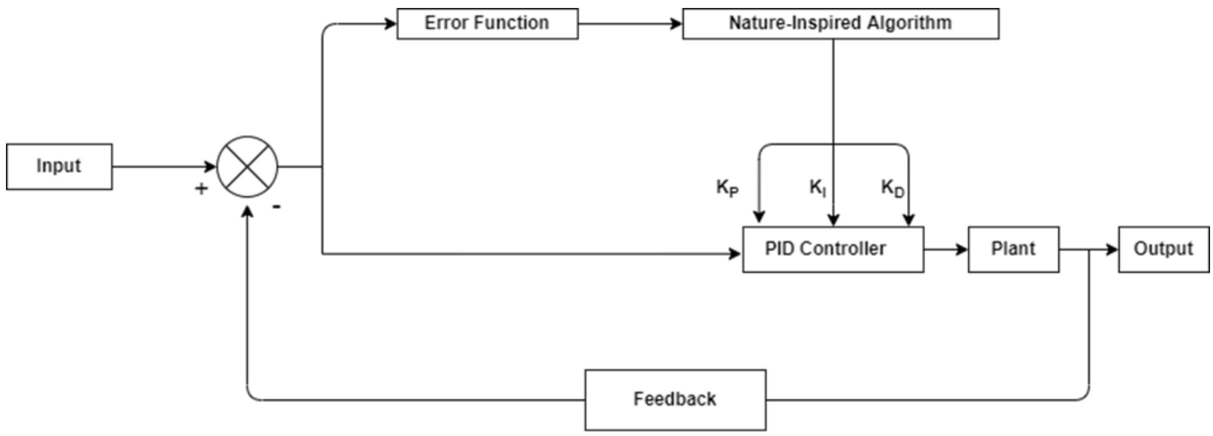


Figure 4.2 Layout of Nature Inspired Algorithm Based Optimized PID Controller

Chapter 5

SIMULATION RESULTS AND PERFORMANCE ANALYSIS

In this thesis project, all simulations based on the above-mentioned mathematical modeling were performed in the MATLAB-Simulink environment to acquire a full understanding of the Interleave-Buck converter's stability and performance analysis. Several comparative studies of different methodologies have been conducted to improve the quality of findings based on a variety of performances. In this part, a comparative study of different metaheuristic algorithms and their hybrid approach is performed to demonstrate that the quality of results obtained by the PID controller is dependent on the types of algorithms utilized. At first, all the simulation parameters for the different algorithms and the Interleaved-Buck converter were discussed. Later, Time Domain Simulation was used to evaluate the performance of each method. The performance of the optimized PID controller is evaluated in time domain simulation using the values of step response characteristics such as percentage of overshoot, rise time, settling time, and peak amplitude.

5.1 Transfer Function:

A transfer function is defined as the mathematical representation of interrelation between the output signal of a control system to the input signal, for all possible input values in terms of frequency. Furthermore, the output and input variables are shaped using the Laplace transform with the initial conditions set to zero [55]. The transfer function of an open-loop and closed-loop system differs. This is due to the introduction of the feedback loop in a closed-loop system.

5.2 MATLAB Simulation and Converter Parameter:

A higher-order transfer function of the Interleaved-Buck converter is produced from the MATLAB Simulink model as a way of simulating the system's stability.
Finally, In Table 5.1, the proposed parameters for the Interleaved-Buck converter are shown. Simulink model of open loop Interleaved-Buck converter is shown in Fig 5.1.

Table 5.1. Parameters of the Interleaved-Buck converter

Parameter	**Symbol**	**Value**
Voltage (input)	V_{in}	10 V
Switching Frequency	fs	100 kHz
Duty Cycle	d	0.67
Inductor	L1	22.5 mH
	L2	22.5 mH
Capacitor	C	470 µF
Load Resistance	R	16 Ω

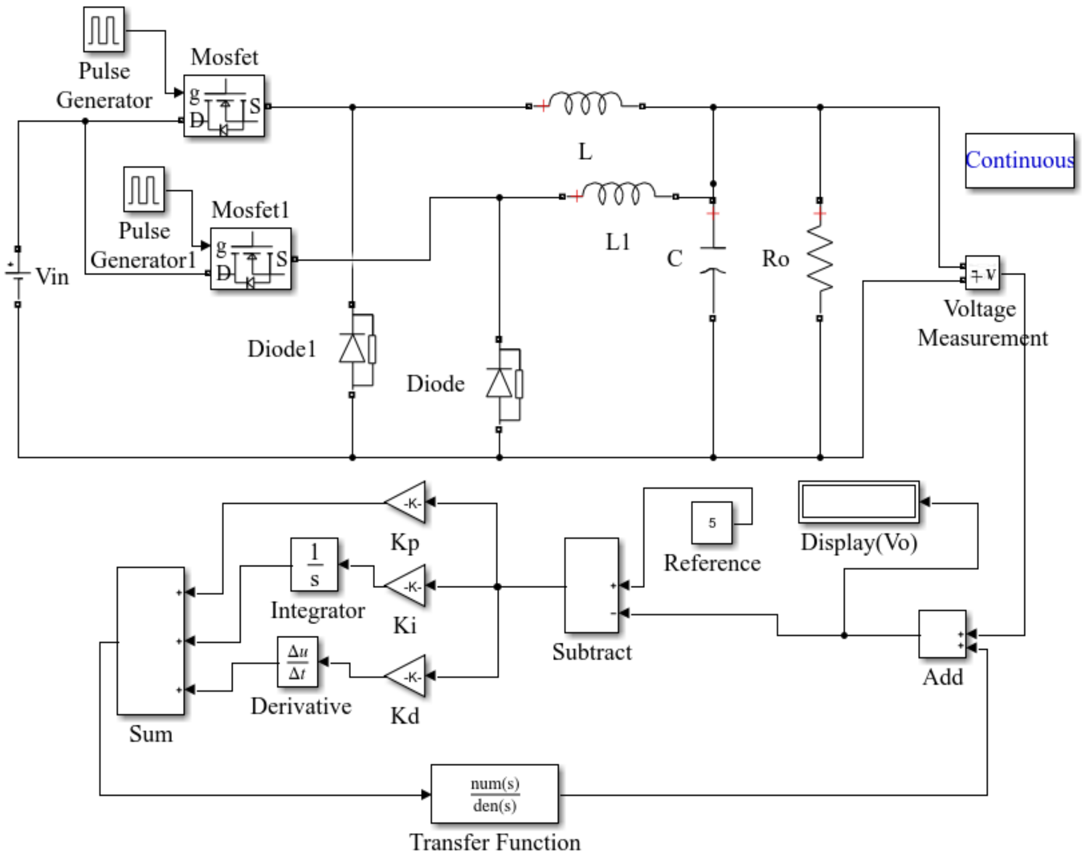


Fig. 5.1 MATLAB Simulink model of Open-loop Interleaved-Buck Converter

5.3 Time-Domain Simulation Analysis:

Different algorithms are compared in this section to evaluate their performance. The Percentage of Overshoot (percent OS), Rise time (Tr), and Setting time (Ts) of the Converter's output voltage are used to compare the findings. These parameters are generated from the system's step response after the PID-controlled metaheuristic algorithms are used. Since a comparison must be done, all of the simulation settings for the optimizing algorithm are maintained the same in order to compare their performance. The initial population size of all the algorithms is taken as 50 and the maximum number of iterations for each case is 30. Search space is kept the same for all the algorithms and within the search space, the initial population was chosen at random. The system's open-loop transfer function is calculated using the state-space model defined before, which is presented below:

$$T(S) = \frac{0.2712\, s\hat{}2 + 1.218e05\, s + 1.746e07}{s\hat{}3 + 304.5\, s\hat{}2 + 2.16e05\, s + 2.708e07}$$

5.3.1 Conventional PID controller:

At first, a traditional PID controller is used with an open-loop system, and the step response is used to analyze the system's stability, as illustrated in Figure 5.2.

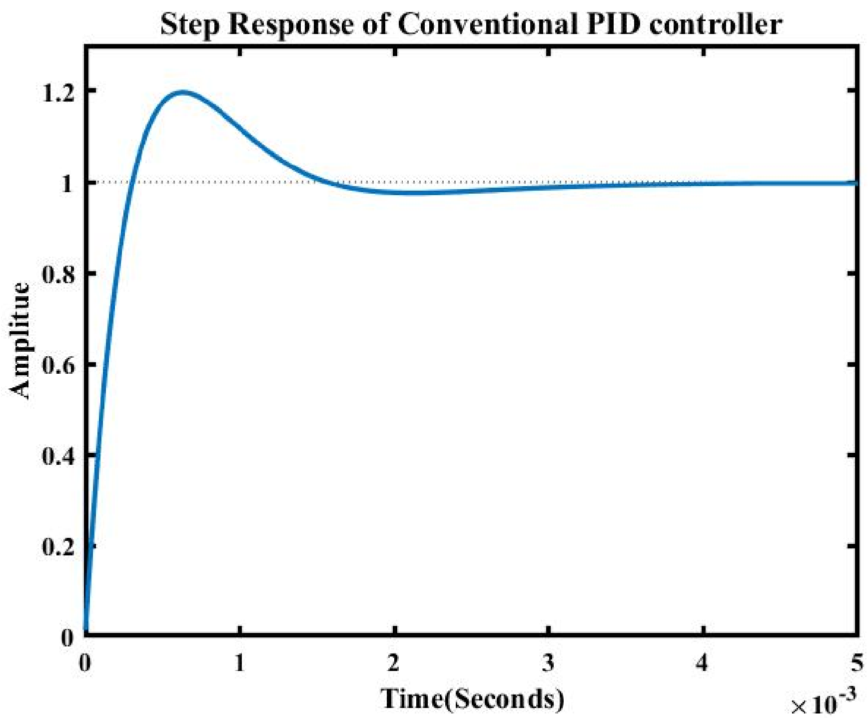


Fig. 5.2 Step Response of Conventional PID Controller

The step response of the closed-loop system with the conventional PID controller displays a significant amount of overshoot with a value of 19.8 percent, as shown in Table 5.2. Afterward, different algorithms and their hybrid approaches are used in the system to find more ideal PID controller settings and decrease overshoot.

Table 5.2 Output Parameters of Conventional PID Controller

Attributes	**Symbols**	**Values**
Gain Values	**Kp**	51.251
	Ki	17591.95
	Kd	0.0288
Performance Parameters	**%OS**	19.8
	Tr(seconds)	0.00035
	Ts(seconds)	0.00371
	Peak Amplitude	1.2

5.3.2 Nature-Inspired Algorithm integrated-PID Controller:

Different Nature-inspired metaheuristic algorithms are applied to optimize the gain parameters of the PID controller after monitoring the performance of the conventional PID controller. As a result, for several Nature-inspired-PID controllers such as PSO-PID, FA-PID, CSA-PID, GWO-PID, WOA-PID, and HFPSO-PID, step responses are observed and gain values, as well as performance indicators, are calculated.

5.3.2.1 PSO-PID Controller:

PSO is firstly used to tune the PID controller, and its parameters are listed in Table 5.3. Step responses are then simulated for the ITAE, ITSE objective functions shown in Figures 5.3 and 5.4. Table 5.4 lists the gain and performance characteristics of the PSO-PID controller for the Interleaved-Buck converter, and it is seen that the ITSE function performs better with an overshoot of 2.06%.

Table 5.3. Parameters of Particle Swarm Optimization

Parameters		**Values**
Acceleration Coefficient	C1	2
	C2	2
Intertia	Wmax	0.9
	Wmin	0.4
Population Size		50

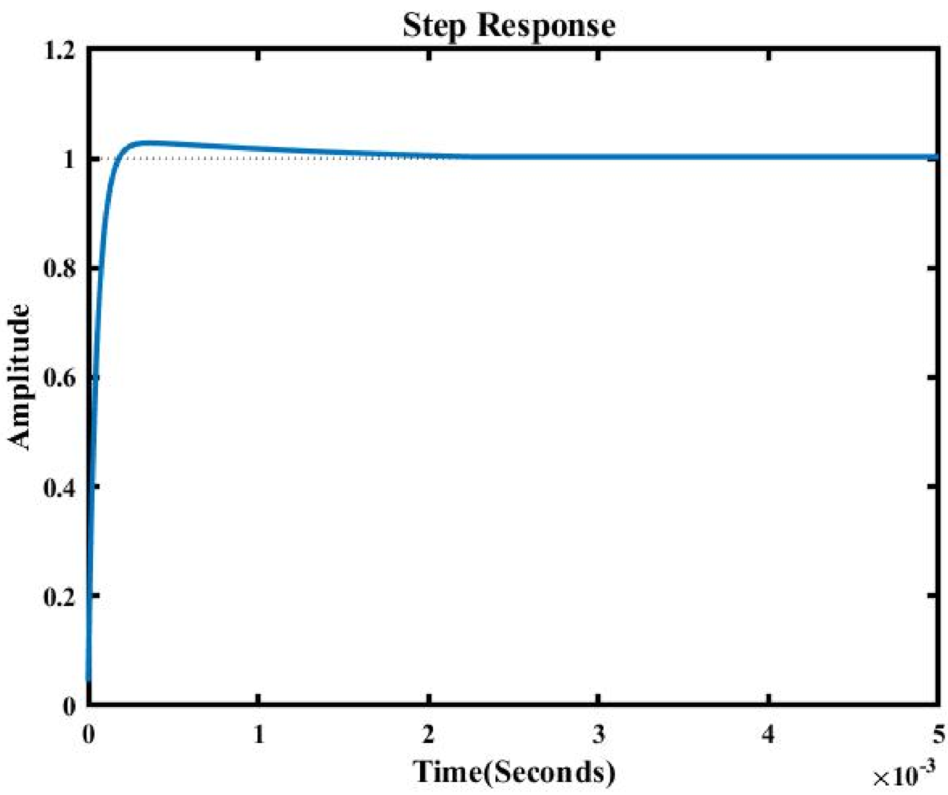


Fig. 5.3 Step Response of PSO-PID Controller for ITAE error function

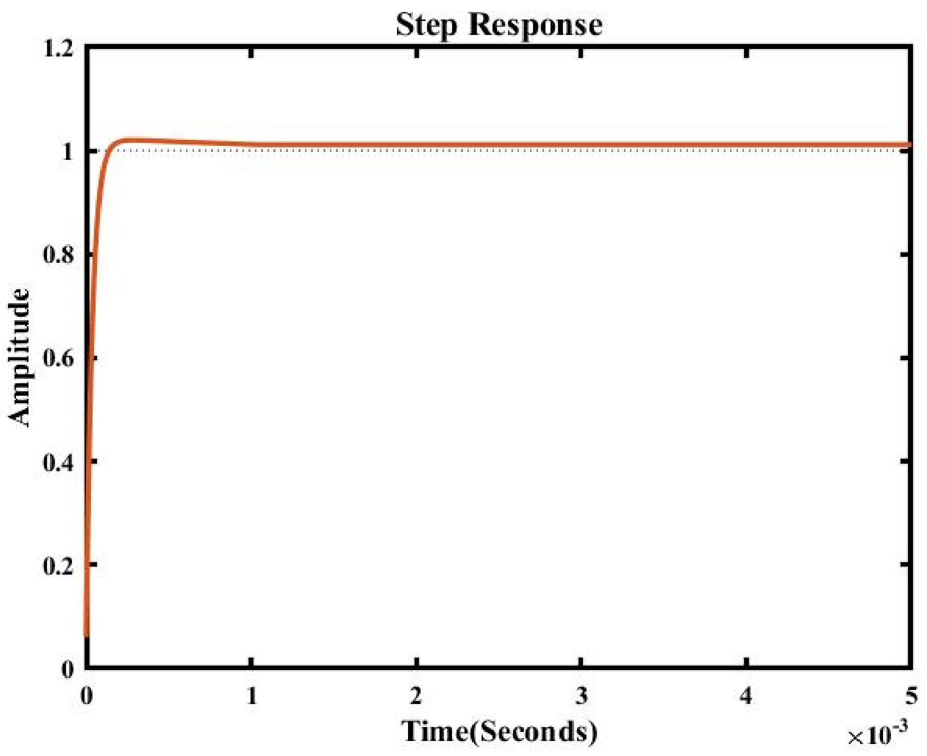


Fig. 5.4 Step Response of PSO-PID Controller for ITSE error function

Table 5.4 Gain and Performance Parameters of PSO-PID controller

Attributes	**Symbols**	**Values**	
		ITAE	**ITSE**
Gain Values	**Kp**	143.159	200
	Ki	60565.91	86918.10
	Kd	0.167	0.237
Performance Parameters	**%OS**	2.87	2.06
	Tr(seconds)	0.000102	0.0000754
	Ts(seconds)	0.000935	0.00046
	Peak Amplitude	1.028	1.02

5.3.2.2 FA-PID Controller:

Following that, FA is implemented for PID controller optimization, and FA parameters are listed in Table 5.5. Step responses are then simulated, as shown in Figs. 5.5 and 5.6. Table 5.6 shows the gain and performance characteristics of the FA-PID controller for the Interleaved-Buck converter.

Table 5.5 Parameters of Firefly Optimization

Parameters	Value
Mutation Coefficient(α)	0.2
Attraction Coefficient(β)	2
Light Absorption Coefficient(γ)	1
Number of fireflies	50

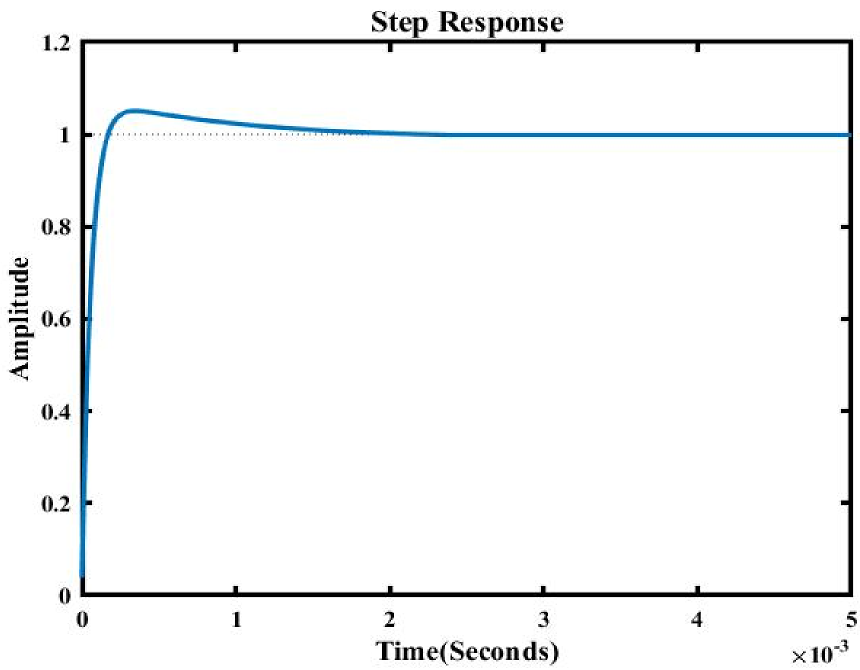


Fig. 5.5 Step Response of FA-PID Controller for ITAE error function

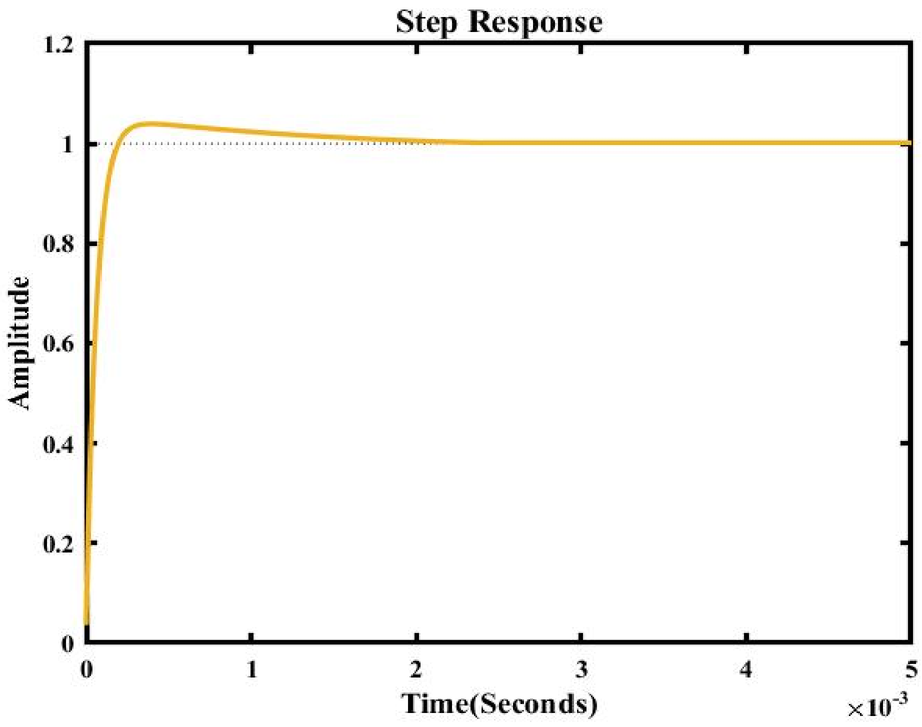


Fig. 5.6 Step Response of FA-PID Controller for ITSE error function

Table 5.6 Gain and Performance Parameters of FA-PID controller

Attributes	**Symbols**	**Values**	
		ITAE	**ITSE**
Gain Values	**Kp**	200	139.2
	Ki	86776.27	58266.15
	Kd	0.148	0.14
Performance Parameters	**%OS**	5.07	3.93
	Tr(seconds)	0.000107	0.000117
	Ts(seconds)	0.00113	0.00119
	Peak Amplitude	1.05	1.03

5.3.2.3 CSA-PID Controller:

Similarly, CSA is used to modify the gain parameters of the PID controller, and the step responses are then simulated, as shown in Figs. 5.7 and 5.8. Table 5.7 lists the gain and performance characteristics of the CSA-PID controller for the Interleaved-Buck converter.

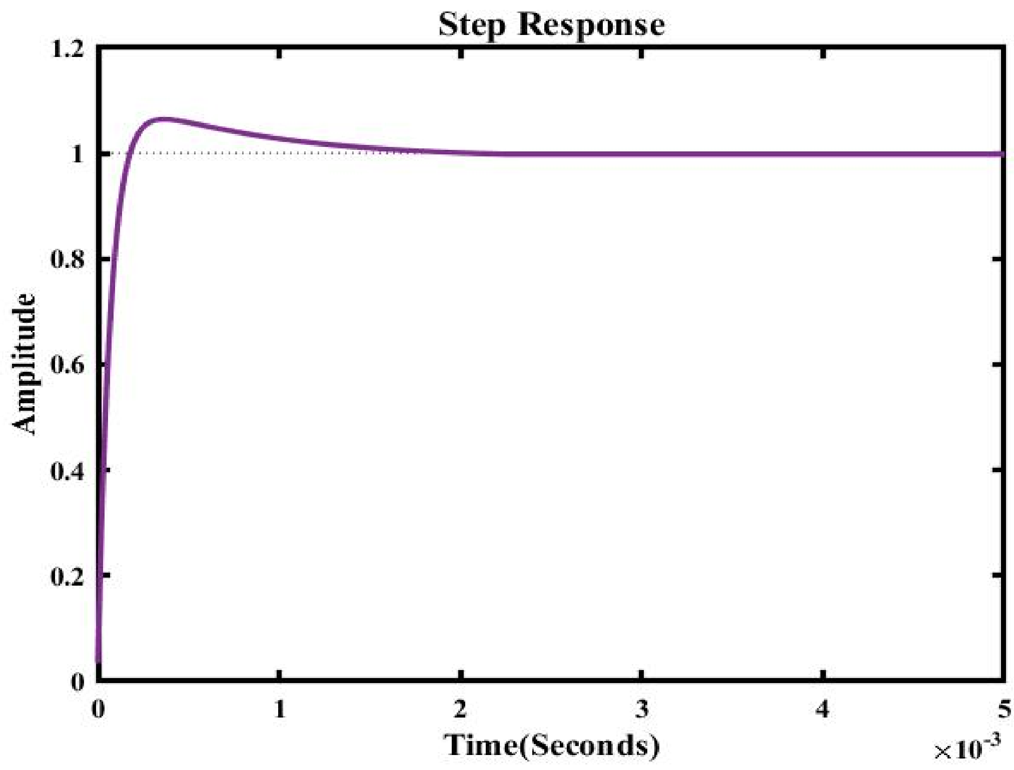


Fig. 5.7 Step Response of CSA-PID Controller for ITAE error function

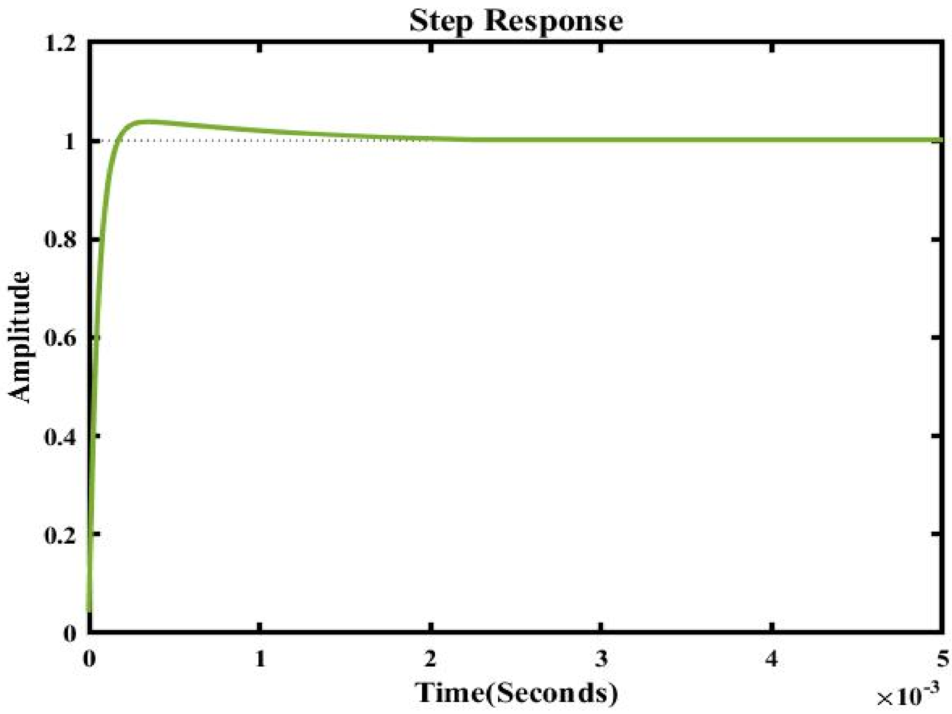


Fig. 5.8 Step Response of CSA-PID Controller for ITSE error function

Table 5.7 Gain and Performance Parameters of CSA-PID controller

Attributes	**Symbols**	**Values**	
		ITAE	**ITSE**
Gain Values	**Kp**	200	175.97
	Ki	95621.257	74232.48
	Kd	0.13	0.161
Performance Parameters	**%OS**	6.44	3.84
	Tr(seconds)	0.000116	0.000103
	Ts(seconds)	0.00119	0.00107
	Peak Amplitude	1.064	1.03

5.3.2.4 GWO-PID Controller:

Later, GWO is implemented for improving PID controller gain settings. Then step responses are simulated, as shown in Figs. 5.9 and 5.10. Following that, Table 5.8 lists the gain and performance characteristics of the GWO-PID controller for the Interleaved-Buck converter. From the Table we can see that GWO gives more overshoot than previously discussed PSO algorithm.

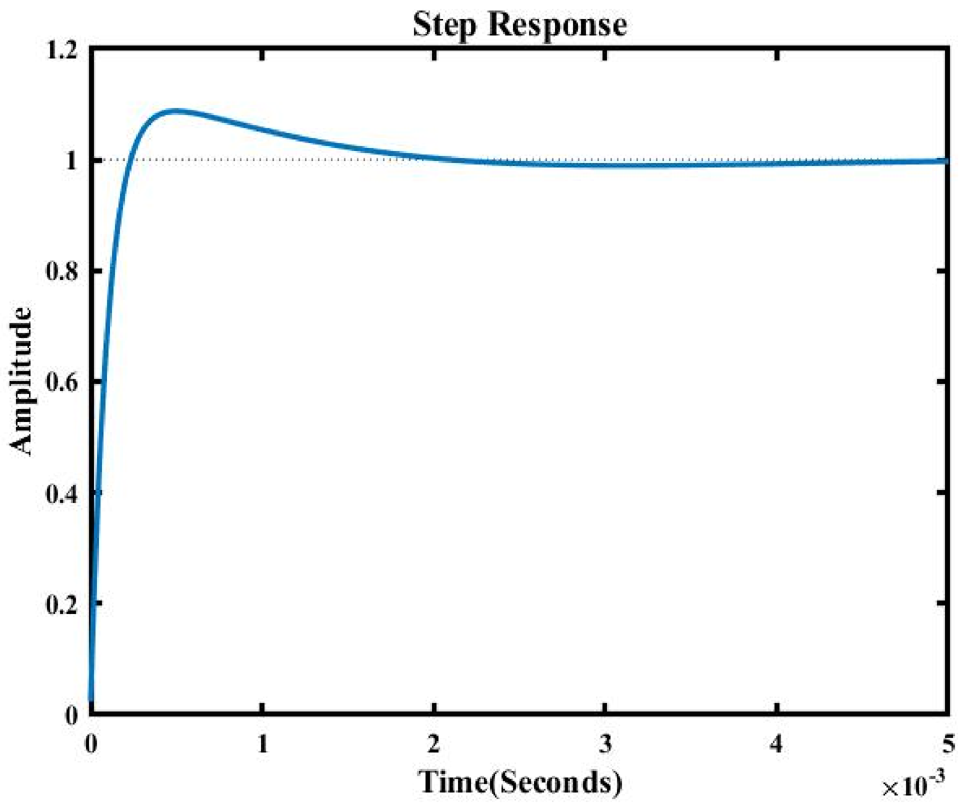


Fig. 5.9 Step Response of GWO-PID Controller for ITAE error function

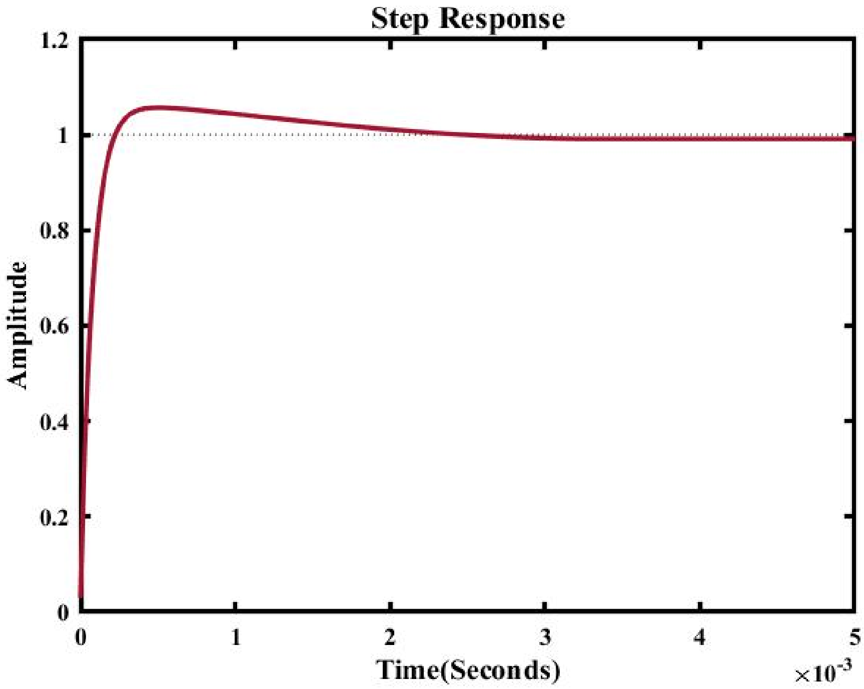


Fig. 5.10 Step Response of GWO-PID Controller for ITSE error function

Table 5.8 Gain and Performance Parameters of GWO-PID controller

Attributes	**Symbols**	**Values**	
		ITAE	**ITSE**
Gain Values	**Kp**	181.737	111.29
	Ki	87338.69	100000
	Kd	0.116	0.111
Performance Parameters	**%OS**	7.228	5.68
	Tr(seconds)	0.000127	0.000141
	Ts(seconds)	0.00125	0.00172
	Peak Amplitude	1.072	1.056

5.3.2.5 WOA-PID Controller:

After that, WOA is used to enhance the PID controller gain values. Then, as illustrated in Figs. 5.11 and 5.12, step responses are simulated. The gain and performance parameters of the WOA-PID controller for the Interleaved-Buck converter are then listed in Table 5.9. The Table shows that the WOA algorithm produces less overshoot with the value of 3.09 percent for the ITSE error function.

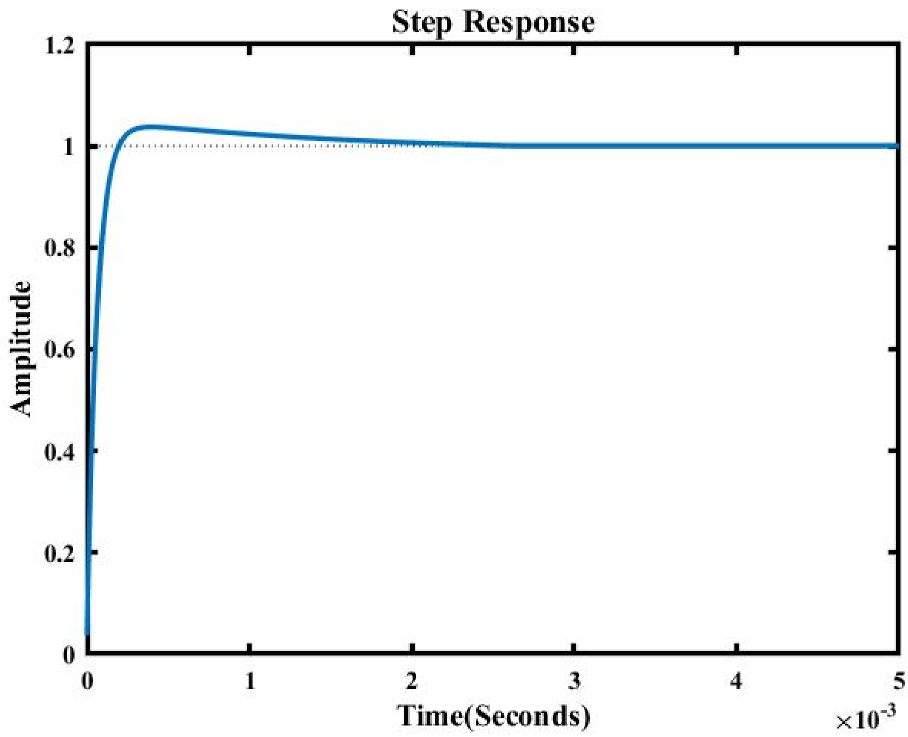


Fig. 5.11 Step Response of WOA-PID Controller for ITAE error function

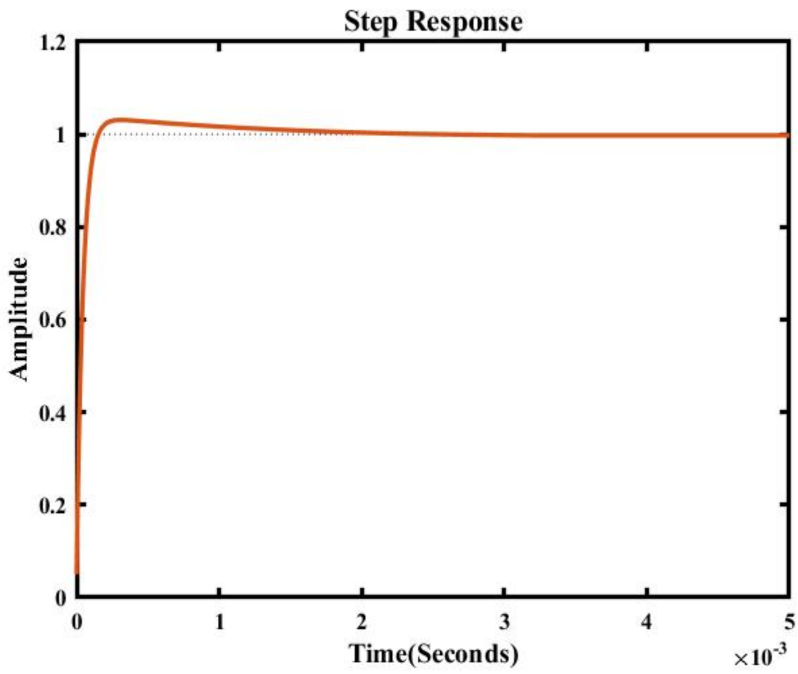


Fig. 5.12 Step Response of WOA-PID Controller for ITSE error function

Table 5.9 Gain and Performance Parameters of WOA-PID controller

Attributes	Symbols	Values	
		ITAE	**ITSE**
Gain Values	**Kp**	133.82	200
	Ki	56720.80	83975.17
	Kd	0.141	0.193
Performance Parameters	**%OS**	3.712	3.095
	Tr(seconds)	0.000117	0.0000884
	Ts(seconds)	0.00117	0.000878
	Peak Amplitude	1.037	1.03

5.3.2.6 HFPSO-PID Controller:

Finally, HFPSO is employed to improve the gain values of the PID controller. Step responses are then generated, as seen in Figs. 5.13 and 5.14. Table 5.10 lists the gain and performance characteristics of the HFPSO-PID controller for the Interleaved-Buck converter. Compared with the all previous algorithm HFPSO gives the best responses in all categories.

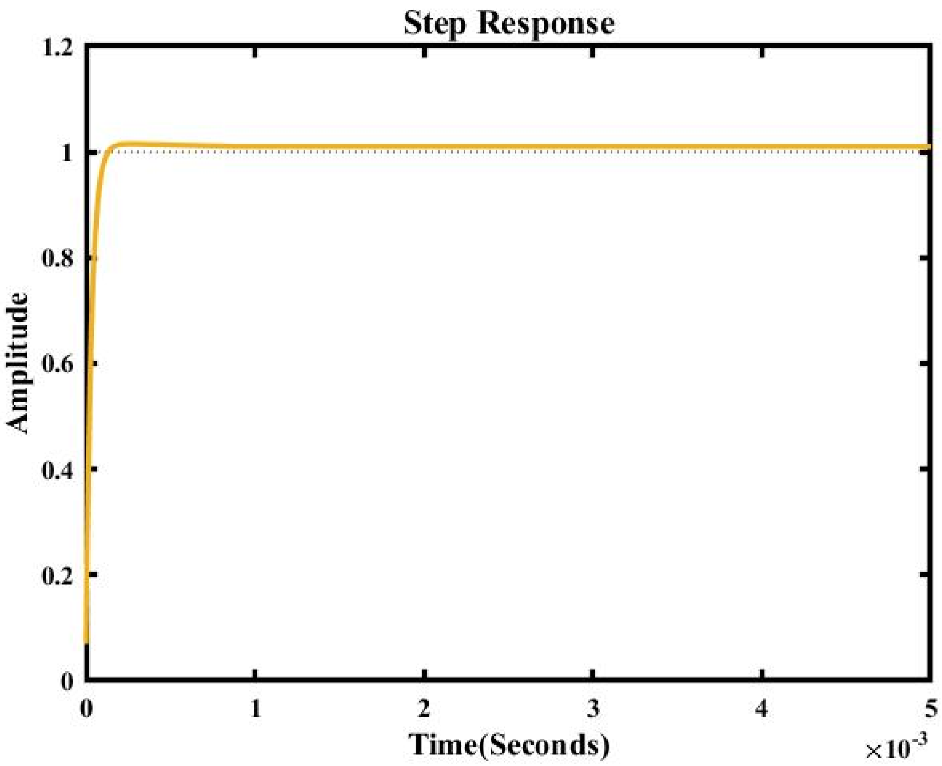


Fig. 5.13 Step Response of HFPSO-PID Controller for ITAE error function

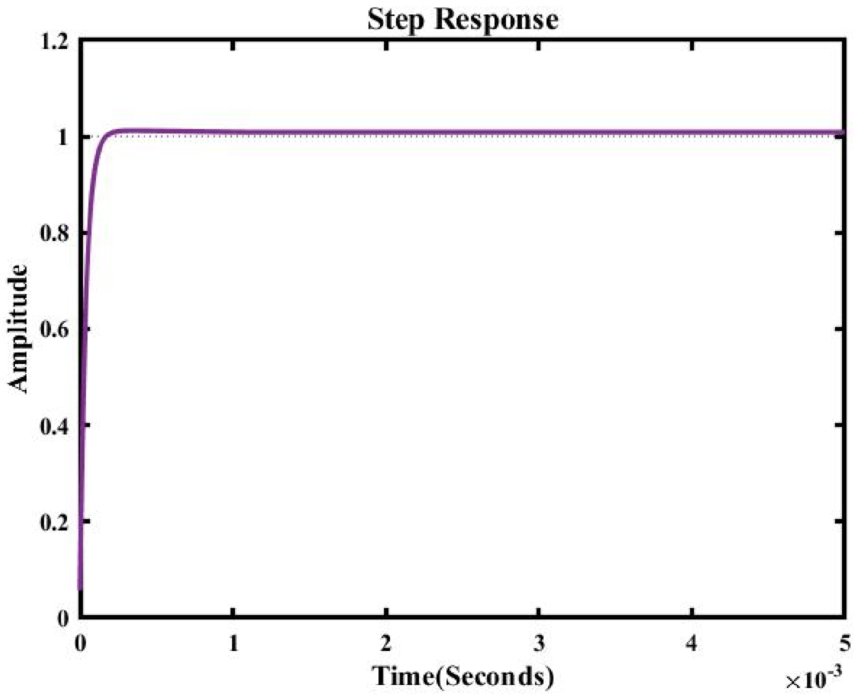


Fig. 5.14 Step Response of HFPSO-PID Controller for ITSE error function

Table 5.10 Gain and Performance Parameters of HFPSO-PID controller

Attributes	**Symbols**	**Values**	
		ITAE	**ITSE**
Gain Values	Kp	200	115.41
	Ki	89553.35	63905.032
	Kd	0.275	0.22
Performance Parameters	%OS	1.513	1.2
	Tr(seconds)	0.0000668	0.0000821
	Ts(seconds)	0.000107	0.000134
	Peak Amplitude	1.015	1.012

5.4 Comparative analysis:

A comparison of performance metrics is shown in Table 5.11. The table shows that ITSE fitness functions have a smaller percentage of overshoot than ITAE fitness functions for all optimization algorithms. Furthermore, ITSE settles faster than ITAE. In terms of optimization algorithms, the best three performers are HFPSO, PSO, and WOA. HFPSO outperforms the others in both the ITAE and ITSE objective functions. ITAE and ITSE had 1.513 percent and 1.2 percent overshoot, respectively. HFPSO also gives the best Settling Time and Rise Time for both the objective function but this time ITAE gives the lowest value. The second best algorithm is PSO with the lowest overshoot of 2.06% for ITSE. The third best algorithm is WOA and the worst performing algorithm is WOA. WOA has a 7.22% and 5.68% overshoot for ITAE and ITSE which is the highest among the algorithms. GWO has also the worst Settling Time and Rise Time. FA and CSA are fourth and fifth best for ITAE with an overshoot of 5.07% and 6.44%. For the ITSE function, FA is fifth and CSA is the fourth-best algorithm. It is clear from the data that HFPSO performs best in all perspectives and also it is shown that hybridizing optimization algorithm gives better performance. Here FA doesn't give a satisfactory result but when it is hybridized with PSO this hybrid algorithm gives the best outcome. Table 5.10 is tabulated as a means to compare the overall performance among all the algorithms. Finally, the overall comparative analysis of the step responses of the best three algorithms is illustrated in Figs. 5.15, 5.16, 5.17 and 5.18 as a means to provide a relative view of the performances.

Table 5.11 Overall comparative analysis of performance parameters

Performance Parameters	**Obj Func**	**Conventional**	**PSO**	**FA**	**CSA**	**GWO**	**WOA**	**HFPSO**
%OS	**ITAE**	19.8	2.87	5.07	6.44	7.228	3.712	1.513
	ITSE		2.06	3.93	3.84	5.68	3.095	1.2
Tr(seconds)	**ITAE**	0.00035	0.000102	0.000107	0.000116	0.000127	0.000117	0.0000668
	ITSE		0.0000754	0.000117	0.000103	0.000141	0.0000884	0.0000821
Ts(seconds)	**ITAE**	0.00371	0.000935	0.00113	0.00119	0.00125	0.00117	0.000107
	ITSE		0.00046	0.00119	0.00107	0.00172	0.000878	0.000134
Peak Amplitude	**ITAE**	1.2	1.028	1.05	1.064	1.072	1.037	1.015
	ITSE		1.02	1.03	1.03	1.056	1.03	1.012

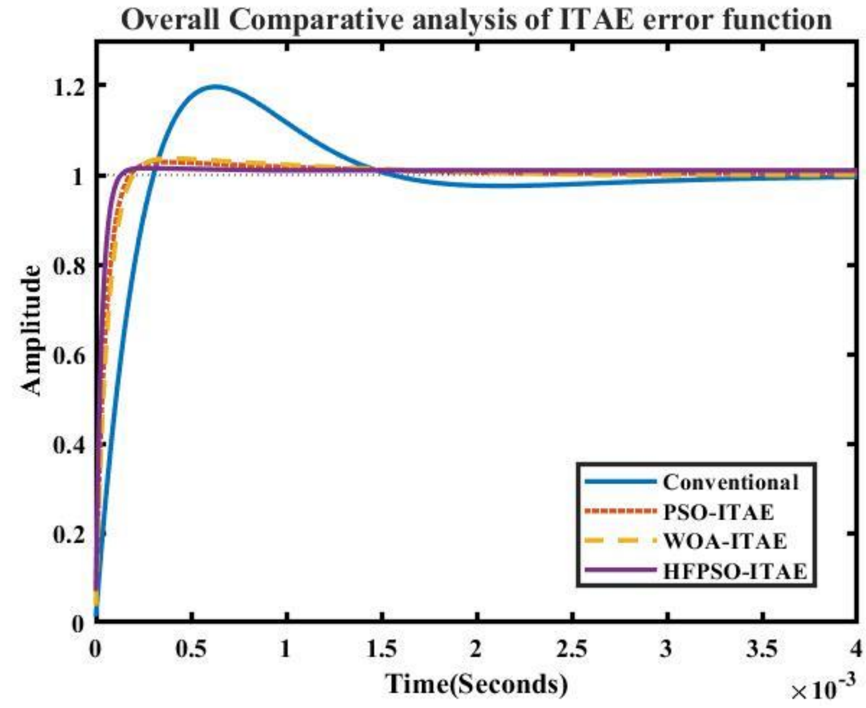


Fig. 5.15 Overall Comparative analysis of ITAE function

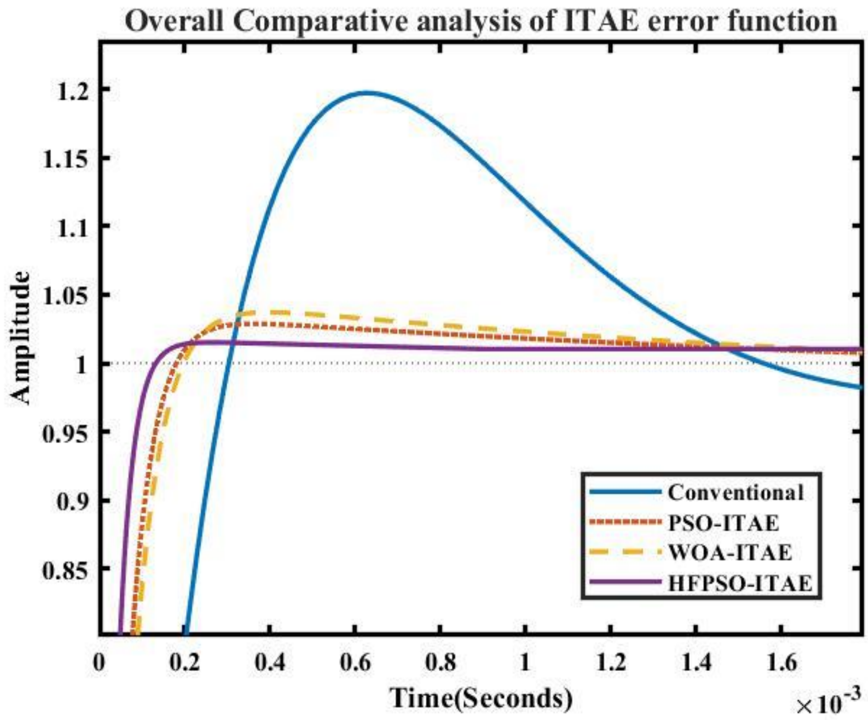


Fig. 5.16 Overall Comparative analysis of ITAE function(Zoomed)

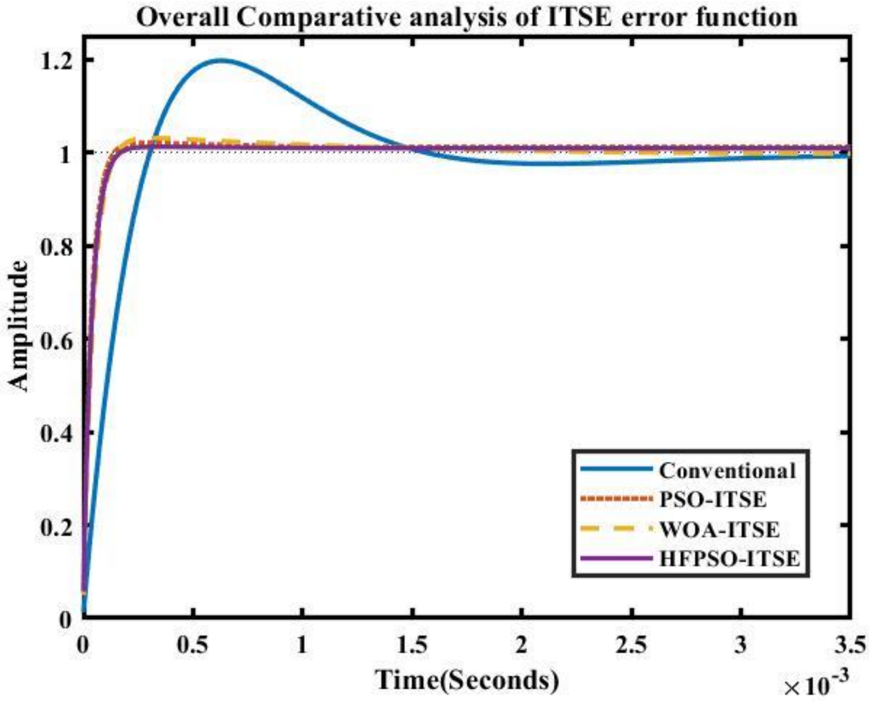


Fig. 5.17 Overall Comparative analysis of ITSE function

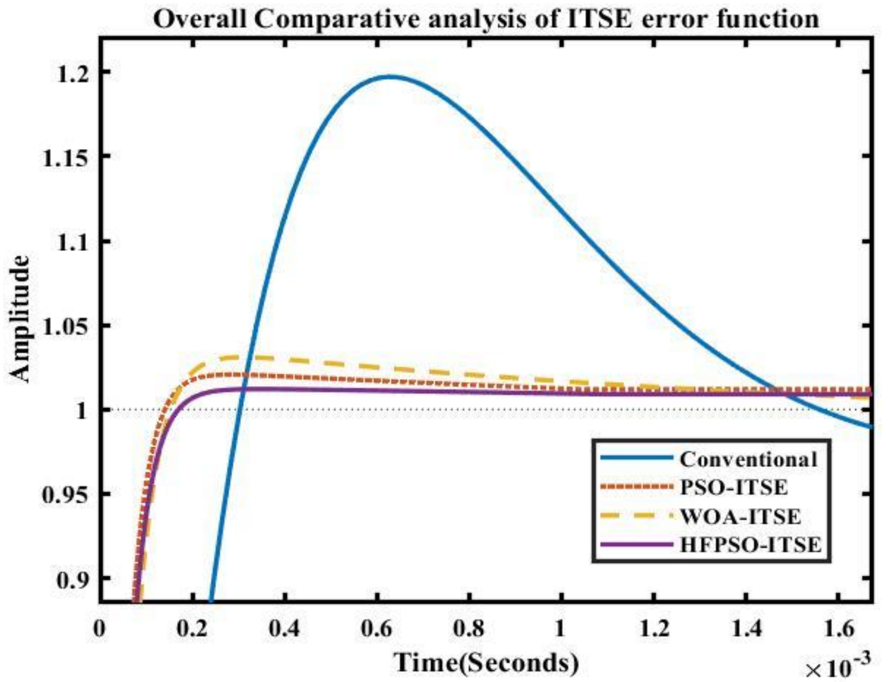


Fig. 5.18 Overall Comparative analysis of ITSE function(Zoomed)

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Synopsis

This thesis discusses how to optimize the PID controller by using a hybrid optimization technique to provide better responses and construct fitness functions for the Interleaved-Buck converter. The objective functions of ITAE and ITSE are utilized to investigate the converter's stability. The operational concept of dc-dc converters is briefly covered in Chapter 2, followed by an overview of the two primary converter topologies and their waveforms. Furthermore, the traditional two-phase Interleaved-Buck converter technique is depicted, as well as the modeling using State Space Averaging Technique. Chapter 3 has presented the overview of different Nature-Inspired Metaheuristic Algorithms and a detailed discussion has been reported about Particle Swarm Optimization (PSO), Firefly Algorithm (FA), Cuckoo Search Algorithm (CSA), Grey Wolf Algorithm (GWO), Whale Optimization Algorithm (WOA) and Hybrid Firefly Particle Swarm Algorithm (HFPSO). The implementation of the techniques for creating an efficient PID controller was discussed in Chapter 4. The tuning of performance parameters has been explored, as well as the required objective functions, in a brief discussion. The simulation results of the algorithm-based tuned PID controller for the Interleaved-Buck converter were discussed in Chapter 5.

With the aid of several types of optimized controllers, a graphical representation of the results was provided, and a comparative analysis was tabulated. The comparative analysis table is created after seeing the graphical representation of all the step responses in order to have a better knowledge of the compatibility of Nature-inspired algorithms in the development of the PID controller. The gain levels and performance characteristics of the standard PID controller have been observed, as well as the parameters of the Interleaved-Buck converter. Following the conventional PID controller, the PID controller is tuned using several optimization approaches in order to reach a better outcome. It is observed that HFPSO-PID is the most optimized controller among all the algorithms in terms of performance parameters. PSO is the second-best performing algorithm after the HFPSO. It is seen from the results that the conventional PID tuning technique gives 19.8 percent of overshoot whereas HFPSO gives 1.2 percent of overshoot. Moreover, rise time and settling time are also within an acceptable limit. Also, FA did not produce good results, however HFPSO beats other algorithms after being hybridized with the Particle Swarm Optimization Algorithm. As a result, it's clear that using hybrid optimization approaches to improve the PID controller yields superior results.

6.2 Future Scope

Different recent algorithms, as well as their hybrid approaches, can be used to develop an optimal PID controller to examine and improve the converters' stability in the future. The performance of the Interleaved-Buck converter can be increased further by developing and getting better parameters. Further hardware implementation tests to validate this technique are also possible. As a result, an overall comparison study may be quite useful in observing stable conditions for various power electronics applications.

References

1. Lilienstein and Miller, THE BIASED TRANSFORMER DCTO-DC CONVERTER, IEEE PESC proceedings, June 1976, pp. 190-199.

2. P. L. Wong, P. Xu, B. Yang, and F. C. Lee, "Performance improvements of interleaving VRMs with coupling inductors," IEEE Trans. Power Electron., vol. 168, no. 4, pp. 499–507, Jul. 2001

3. C. Garcia, P. Zumel, A. D. Castro, and J. A. Cobos, "Automotive DC–DC bidirectional converter made with many interleaved buck stages," IEEE Trans. Power Electron., vol. 21, no. 21, pp. 578–586, May 2006.

4. https://www.emersonautomationexperts.com/2013/control-safety-systems/pid-control

5. Åström, Karl Johan, and Tore Hägglund. "Revisiting the Ziegler–Nichols step response method for PID control." Journal of process control 14.6 (2004): 635-650.

6. Efendi, Moh Zaenal, Farid Dwi Murdianto, and Rangga Eka Setiawan. "Modeling and simulation of MPPT sepie converter using modified PSO to overcome partial shading impact on DC microgrid system." 2017 International Electronics Symposium on Engineering Technology and Applications (IES-ETA). IEEE, 2017.

7. Ragavendra, B., S. Vijayanand, and B. Jayaprakash. "Improved Control Strategy on Cuk Converter fed DC Motor using Artificial Bee Colony Algorithm." Int. J. Recent Dev. Eng. Technol 2 (2014): 181-188.

8. Govindaraj, Thavamani and M. Senthamil. "SIMULATION MODELLING BASED CONTROL OF AN INTERLEAVED BOOST CONVERTER FED INDUCTION MOTOR USING PSO ALGORITHM." (2014).

9. Nishat, M. , Faisal, F. , Evan, A. , Rahaman, M. , Sifat, M. and Rabbi, H. (2020) Development of Genetic Algorithm (GA) Based Optimized PID Controller for Stability Analysis of DC-DC Buck Converter. Journal of Power and Energy Engineering, 8, 8-19.

10. Yaqoob, Muhammad, et al. "Optimization in transient response of DC-DC buck converter using firefly algorithm." 2014 16th International Conference on Harmonics and Quality of Power (ICHQP). IEEE, 2014.

11. P. Zhang, Y. Liu, J. Zhao, H. Zhang and Q. Lv, "An Efficiency Optimization Control Method With Fast Dynamic Response For Muti-Phase Interleaved BUCK Converter," *IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society*, 2021, pp. 1-5, doi: 10.1109/IECON48115.2021.9589403.

12. Arora, Ritika, et al. "PSO optimized PID controller design for performance enhancement of hybrid renewable energy system." 2020 IEEE 9th Power India International Conference (PIICON). IEEE, 2020.

13. P. L. Wong, P. Xu, B. Yang, and F. C. Lee, "Performance improvements of interleaving VRMs with coupling inductors," IEEE Trans. Power Electron., vol. 168, no. 4, pp. 499–507, Jul. 2001.
14. I.Sefa, S.Ozdemir, "Multifunctional interleaved boost converter for PV systems" IEEE International Symposium on Industrial Electronics, ISIE'2010 , Bari, 4-7 July 2010, pp951-956.
15. R. L. Lin, C. C. Hsu, and S. K. Changchien, "Interleaved four-phase buck-based current source with isolated energy-recovery scheme for electrical discharge machine," IEEE Trans. Power Electron., vol. 24, no. 7, pp. 2249–2258, Jul. 2009
16. C. Garcia, P. Zumel, A. D. Castro, and J. A. Cobos, "Automotive DC–DC bidirectional converter made with many interleaved buck stages," IEEE Trans. Power Electron., vol. 21, no. 21, pp. 578–586, May 2006.
17. D. J. S. Newlin, R. Ramalakshmi and S. Rajasekaran, "A performance comparison of interleaved boost converter and conventional boost converter for renewable energy application," *2013 International Conference on Green High Performance Computing (ICGHPC)*, 2013, pp. 1-6
18. N. Mohan,T.M.Undiland & W.P.Robins "Power Electronics Converters Applications & Design", 2nd Edition, John Wiley & Sons, Inc.].
19. Ejury, Jens. "Buck converter design." Infineon Technologies North America (TFNA) Corn Desion Note 1 (2013).
20. Redl, Richard, Brian P. Erisman, and Zoltan Zansky. "Optimizing the load transient response of the buck converter." APEC'98 Thirteenth Annual Applied Power Electronics Conference and Exposition. Vol. 1. IEEE, 1998.
21. Jahanbakhshi, M. H., & Etezadinejad, M. (2019). Modeling and Current Balancing of Interleaved Buck Converter Using Single Current Sensor. ICEE 2019 - 27th Iranian Conference on Electrical Engineering, 1, 662–667. https://doi.org/10.1109/IranianCEE.2019.8786714
22. J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proc. IEEE Int. Conf. Neural Networks, vol. IV, Perth, Australia, 1995, pp. 1942–1948.
23. Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization: An overview. Swarm Intelligence, 1(1), 33–57.

24. Osman, I.H. and Laporte, G. (1996). Metaheuristics: A bibliography. Annals of Operations Research. Vol. 63, pp 513-623

25. Blum, C. and Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. Journal in ACM Computing Surveys. Vol. 35, Issue 3, pp 268-308.

26. Yang, Xin-She, and Mehmet Karamanoglu. "Swarm intelligence and bio-inspired computation: an overview." Swarm intelligence and bio-inspired computation. Elsevier, 2013. 3-23.

27. Johari, Nur Farahlina, et al. "Firefly algorithm for optimization problem." Applied Mechanics and Materials. Vol. 421. Trans Tech Publications Ltd, 2013.

28. Fister, I., Yang, X. S., & Brest, J. (2013). A comprehensive review of firefly algorithms. Swarm and Evolutionary Computation, 13, 34–46. https://doi.org/10.1016/j.swevo.2013.06.001

29. S. Mirjalili, S. M. Mirjalili, and A. Lewis, ‘‘Grey wolf optimizer,’’ Adv. Eng. Softw., vol. 69, pp. 46–61, Mar. 2014.

30. N. Muangkote, K. Sunat, and S. Chiewchanwattana,‘‘An improved grey wolf optimizer for training q-Gaussian radial basis functionallink nets,’’ in Proc. Int. Comput. Sci. Eng. Conf. (ICSEC), Jul.2014, pp. 209–214

31. Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. Advances in Engineering Software, 95, 51–67

32. Watkins WA, Schevill WE. Aerial observation of feeding behavior in four baleen whales: Eubalaena glacialis, Balaenoptera borealis, Megaptera novaeangliae, and Balaenoptera physalus. J Mammal 1979:155–63.

33. A. Sahu, S.K. Panigrahi, S. Pattnaik, Fast Convergence Particle Swarm Optimization for Functions

34. P. Kora, K.S. Rama Krishna, Hybrid firefly and Particle Swarm Optimization algorithm for the detection of Bundle Branch Block, Int. J. Cardiovasc. Acad. 2 (2016) 44–48. doi:10.1016/j.ijcac.2015.12.001.

35. J. Kennedy, R.C. Eberhart, Y. Shi, C. Jacob, J.R. Koza, F.H.B. Iii, D. Andre, M. a Keane, Swarm Intelligence The Morgan Kaufmann Series in Evolutionary Computation, 2001. http://www.citeulike.org/user/pesimov/article/1084312.

36. İbrahim Berkan Aydilek, A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems,Applied Soft Computing,Volume 66, 2018.

37. Y. Shi, R. Eberhart, A modified particle swarm optimizer, 1998 IEEE Int. Conf. Evol. Comput. Proceedings. IEEE World Congr. Comput. Intell. (Cat. No.98TH8360). (1998) 69–73. doi:10.1109/ICEC.1998.699146

38. A. Nickabadi, M.M. Ebadzadeh, R. Safabakhsh, A novel particle swarm optimization algorithm with adaptive inertia weight, Appl. Soft Comput. J. 11 (2011) 3658–3670. doi:10.1016/j.asoc.2011.01.037.

39. M. Taherkhani, R. Safabakhsh, A novel stability-based adaptive inertia weight for particle swarm optimization, Appl. Soft Comput. J. 38 (2016) 281–295. doi:10.1016/j.asoc.2015.10.004

40. J. Xin, G. Chen, Y. Hai, A particle swarm optimizer with multi-stage linearly-decreasing inertia weight, in: Proc. 2009 Int. Jt. Conf. Comput. Sci. Optim. CSO 2009, 2009: pp. 505–508. doi:10.1109/CSO.2009.420

41. Roberts, Steve. "DC/DC book of knowledge." Austria: RECOM Engineering GmbH & Co KG (2015).

42. Miaja, Pablo F., Miguel Rodriguez, Alberto Rodriguez, and Javier Sebastian. "A linear assisted DC/DC converter for envelope tracking and envelope elimination and restoration applications." In 2010 IEEE Energy Conversion Congress and Exposition, pp. 3825-3832. IEEE, 2010

43. Liu, Kwang-Hwa, and Fred C. Lee. "Zero-voltage switching technique in DC/DC converters." In 1986 17th Annual IEEE Power Electronics Specialists Conference, pp. 58- 70. IEEE, 1986.

44. Kumar, J. Sai, and Tikeshwar Gajpal. "A Multi Input DC-DC Converter for Renewable Energy Applications." (2016).

45. Ortiz, G., J. Biela, D. Bortis, and J. W. Kolar. "1 Megawatt, 20 kHz, isolated, bidirectional 12kV to 1.2 kV DC-DC converter for renewable energy applications." InPower Electronics Conference (IPEC), 2010 International, pp. 3212-3219. IEEE, 2010.

46. Li, Wuhua, Xiaodong Lv, Yan Deng, Jun Liu, and Xiangning He. "A review of non-isolated high step-up DC/DC converters in renewable energy applications." InApplied Power Electronics Conference and Exposition, 2009. APEC 2009. Twenty-Fourth Annual IEEE, pp. 364-369. IEEE, 2009.

47. Koutroulis, Eftichios, and Kostas Kalaitzakis. "Design of a maximum power tracking system for wind-energy-conversion applications."IEEE transactions on industrial electronics53, no. 2 (2006): 486-494.

48. Banerjee, Soumitro, and George C. Verghese. Nonlinear phenomena in power electronics. IEEE, 1999.

49. Hamill, David C., Jonathan HB Deane, and David J. Jefferies. "Modeling of chaotic DCDC converters by iterated nonlinear mappings." IEEE transactions on Power Electronics 7, no. 1 (1992): 25-36.

50. A. H. R. Rosa, T. M. de Souza, L. M. F. Morais, and S. I. Seleme. "Adaptive and nonlinear control techniques applied to sepic converter in dc-dc, pfc, ccm and dcm modes using hil simulation." Energies 11, no. 3 (2018): 602

51. Paz, R. A. (2001). The Design of the PID Controller. Computer Engineering, April.

52. Nishat, M. , Faisal, F. , Evan, A. , Rahaman, M. , Sifat, M. and Rabbi, H. (2020) Development of Genetic Algorithm (GA) Based Optimized PID Controller for Stability Analysis of DC-DC Buck Converter. Journal of Power and Energy Engineering, 8, 8-19.

53. Emami, S. A., M. Bayati Poudeh, and S. Eshtehardiha. "Particle Swarm Optimization for improved performance of PID controller on Buck converter." 2008 IEEE International Conference on Mechatronics and Automation. IEEE, 2008.

54. Sonmez, Yusuf, et al. "Improvement of buck converter performance using artificial bee colony optimized-pid controller." Journal of Automation and Control Engineering 3.4 (2015)

55. [Xin-She Yang, Chapter 9 - Cuckoo Search,Editor(s): Xin-She Yang, Nature-Inspired Optimization Algorithms,Elsevier,2014,Pages129-139,ISBN 9780124167438,https://doi.org/10.1016/B978-0-12-416743-8.00009-9]