

## Article

# Optimal Fractional PID Controller for Buck Converter Using Cohort Intelligent Algorithm

Preeti Warriar <sup>1,2</sup>  and Pritesh Shah <sup>1,\*</sup> 

<sup>1</sup> Symbiosis Institute of Technology (SIT), Symbiosis International (Deemed University) (SIU), Pune 411014, India; preetiwarrior06@gmail.com

<sup>2</sup> Army Institute of Technology, Dighi Hills, Pune 411015, India

\* Correspondence: pritesh.shah@sitpune.edu.in; Tel.: +91-20-6193-6499

**Abstract:** The control of power converters is difficult due to their non-linear nature and, hence, the quest for smart and efficient controllers is continuous and ongoing. Fractional-order controllers have demonstrated superior performance in power electronic systems in recent years. However, it is a challenge to attain optimal parameters of the fractional-order controller for such types of systems. This article describes the optimal design of a fractional order PID (FOPID) controller for a buck converter using the cohort intelligence (CI) optimization approach. The CI is an artificial intelligence-based socio-inspired meta-heuristic algorithm, which has been inspired by the behavior of a group of candidates called a cohort. The FOPID controller parameters are designed for the minimization of various performance indices, with more emphasis on the integral squared error (ISE) performance index. The FOPID controller shows faster transient and dynamic response characteristics in comparison to the conventional PID controller. Comparison of the proposed method with different optimization techniques like the GA, PSO, ABC, and SA shows good results in lesser computational time. Hence the CI method can be effectively used for the optimal tuning of FOPID controllers, as it gives comparable results to other optimization algorithms at a much faster rate. Such controllers can be optimized for multiple objectives and used in the control of various power converters giving rise to more efficient systems catering to the Industry 4.0 standards.

**Keywords:** fractional calculus; fractional order PID controllers; power electronics; DC-DC converters; meta-heuristics; cohort intelligence algorithm; Industry 4.0



**Citation:** Warriar, P.; Shah, P. Optimal Fractional PID Controller for Buck Converter Using Cohort Intelligent Algorithm. *Appl. Syst. Innov.* **2021**, *4*, 50. <https://doi.org/10.3390/asi4030050>

Academic Editors: Teen-Hang Meen and Claudio Zunino

Received: 13 June 2021  
Accepted: 2 August 2021  
Published: 4 August 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Power electronic system control is a very crucial and challenging task which has attracted the attention of researchers in the past few decades. Power electronic DC–DC converters are very versatile and can be used for voltage regulation in a wide range of applications. Such converters work in different modes of operation, which introduce non-linearities and are influenced by input and parametric variations [1,2]. The challenge is to design robust and stable controllers which also satisfy the transient response and frequency response specifications with good tracking accuracy [3]. Extensive research has been completed in the quest for improved and more robust controllers for such converters. Such systems are usually controlled using the conventional proportional integral (PI)/proportional integral derivative (PID) control,  $H_\infty$ , sliding mode, predictive control, non-linear methods such as fuzzy and intelligent control, etc. [3–5].

PID controllers are widely used in the industry for the control of power electronic converters which are used in various applications like speed control of DC and AC motor drives, switched mode power supplies, renewable energy applications, etc. [6,7]. These controllers act on the error between the set point value and the controlled parameters, and do not require internal state measurements and, hence, require lesser sensors. They are easy to tune, have an easier control law to implement, and, hence, are easily adaptable to industry [8]. In fact, industrial controllers are still dominated by PID controllers, as

the autotuning process is simple and robust. PID technology has survived years in the industry and is continuously undergoing updation. Even though PID controllers have these advantages, conventional tuning involves trial and error methods and is time consuming. Additionally, many applications, such as power electronic converters, require linearized plant model which limits the range of operation. In actual practice, there will be parametric and load variations, and non-linearities. Under such conditions, the PID controller does not respond well, and advanced controllers are required [9]. Non-linearities can be met by sliding mode controllers, but they are prone to chattering phenomena due to the high frequency of operation of switching converters [1,9].

In the recent years, there has been a significant growth in the field of fractional control, which is one of the major applications of fractional calculus. Fractional order systems use non-integer derivatives and integrals, resulting in various types of fractional order (FO) controllers, including the FO integrator, FO differentiator, FOPI, FOPID controllers, etc. [10,11]. A significant aspect of the fractional order controllers is their robustness to uncertainties and load disturbances. The isodamping property of fractional order systems makes them robust to variations in gain, and hence makes them more attractive compared to the linear PID controllers [12,13]. A typical FO PI/PID controller has more parameters to tune compared to a PID controller, which has only 3 parameters to tune [14]. As the FOPID controllers are extensions of the PID controllers, with additional tuning parameters, the performance of these two types is often compared. The challenge in the tuning of fractional controllers can be considered to be compensated by the increased degrees of freedom, making it possible to satisfy more design requirements [15]. Hence these controllers are more flexible in meeting the design specifications, such as the gain and phase margin, settling time, overshoot, robustness to parameter variations, etc. These characteristics make the fractional order controllers give effective performance in the control of non-linear plants like power converters [16]. It has been observed that the same FOPID controller can be used without retuning on different plants of same ratings. The control effort involved is much lesser in the case of FOPID controllers [17]. Such controllers have been used to control various DC–DC converters like the buck, boost, buck–boost converters for various applications [18–21]. A review of the applications of fractional control in various power electronic systems is given in [22].

One of the main challenges in the use of the FOPID controllers is the tuning of its parameters. Numerical methods for the design of controllers use various optimization techniques, in which a particular algorithm is used to optimize one or more cost functions, which is decided by the design engineer based on the requirements of the application [23]. Some of the common parameters chosen as cost functions are the control performance indices like integral square error (ISE), integral absolute value error (IAE), integral of the absolute error weighted over time (ITAE), integral of the time squared error (ITSE), or time domain specifications like overshoot, rise time, etc. Optimization algorithms like the artificial BEE colony (ABC) algorithm [24], particle swarm optimization (PSO) [25], genetic algorithm (GA) [26], fuzzy/neural network based methods [27], etc., are commonly used for numerical techniques. Such multiple-objective optimization methods are very useful in tuning of fractional order controllers applied to power converters [28–31].

The cohort intelligence (CI) algorithm is a socio-inspired meta-heuristic algorithm, which is based on the behavior of a group of people, referred to as cohort in society [32]. All the members in the cohort are competing with each other to achieve some common goal. Each candidate tries to improve his behavior by observing the others, so that the overall behavior of the cohort evolves, and improves. This method has been used for solving various unconstrained and combinatorial problems and in mechanical systems [33–35], for PID control in DC motors [36], etc.

The objective of this paper is to study the effect of the CI algorithm on the optimal tuning of a fractional order PID controller for a buck converter. The paper is organized into five sections. Section 2 discusses the basics of the buck converter, fractional order systems, and the CI algorithm. Section 3 gives the basic methodology and system description.

Results and comparisons with other methods have been detailed in Section 4. Finally Section 5 concludes the work.

## 2. System Description

### 2.1. Buck Converter

Switched-mode power converters are highly efficient systems prominently used in a wide range of applications like mobile and computer power supplies, electric cars, television, etc. The Buck or step-down converter is a switched mode regulator used to convert a DC voltage of higher magnitude to a lower magnitude DC voltage. A high-power switching device like MOSFET or IGBT is switched on or off at a high frequency to facilitate this conversion. The output voltage can be varied by controlling the on time of the power device by pulse width modulation [37]. The basic buck converter has a DC source, switching device, diode, filter inductor, and capacitor and load, as shown in Figure 1a.

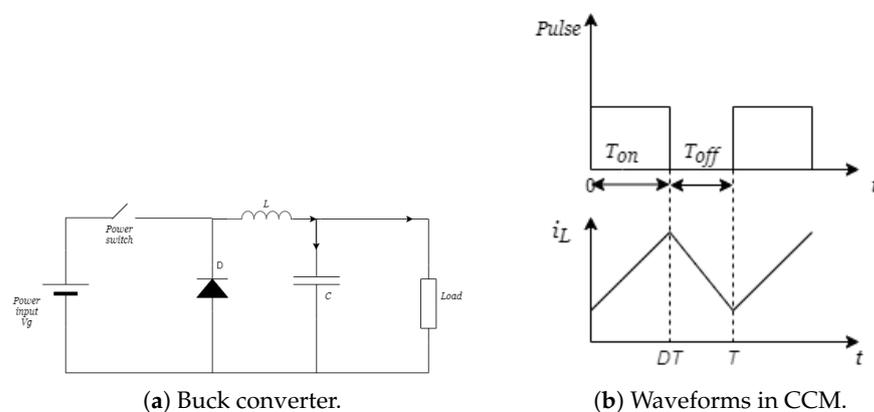


Figure 1. Buck converter and waveforms.

When the control signal is given such that the switch is closed, the diode D is reverse biased and when the switch is turned off, the diode conducts so that the current through the inductor is uninterrupted. From the inductor volt-balance equations, it can be derived that the output voltage  $V_o$  is equal to  $DV_g$ , where the switch duty cycle is given by  $D = T_{on}/T$ ,  $V_g$  is input voltage, and T is the switching period. The waveforms for the continuous conduction mode (CCM), where the inductor current does not become zero, are as shown in Figure 1b. The inductor current increases and decreases during respectively during the on-time ( $T_{on}$ ) and off-time ( $T_{off}$ ) of the pulses. The design parameters of the buck converter can be formulated based on [38,39]. The minimum value of inductor and capacitor for continuous mode conduction can be selected as:

$$L_{min} = \frac{(V_g - V_o)D}{\Delta i_L f} \tag{1}$$

$$C_{min} = \frac{(1 - D)V_o}{8Lf^2\Delta V_o} \tag{2}$$

Here  $f$  is the switching frequency, and  $V_g$  and  $V_o$  are the input and output DC voltages.  $\Delta i_L$  is the inductor current ripple, which is assumed to be 20–30% of output current. Similarly  $\Delta V_o$  is the output voltage ripple, normally assumed to be 1–5% of the output voltage, depending on the design criteria. The value of filter inductance  $L$  should be chosen to be greater than  $L_{min}$  for continuous conduction, and capacitance  $C$  has to be greater than  $C_{min}$ .

### 2.2. Fractional Order Systems

Fractional calculus is a branch of mathematics that deals with the differentiation and integration of arbitrary order, which can be real or even complex. Here the derivative and

integral terms have a fractional order instead of integer order. Here, an integro-differential operator is defined, which combines differentiation and integration. This operator is defined as  ${}_c D_t^\alpha$ , where  $\alpha$  is the non-integer order,  $c$  and  $t$  are the limits of the operation. Many definitions of fractional calculus have been given, but the Riemann–Liouville (RL) and Caputo definitions are most commonly used [40,41]. Basics and definitions of fractional order systems can be found in [42].

Riemann–Liouville Definition

$${}_c D_t^\alpha (f(t)) = D^k J^{k-\alpha} (f(t)) = \frac{1}{\Gamma(k-\alpha)} \left(\frac{d}{dt}\right)^k \int_c^t \frac{f(u)}{(t-u)^{\alpha-k+1}} du \tag{3}$$

Caputo’s Definition

$${}_c D_t^\alpha (f(t)) = \frac{1}{\Gamma(k-\alpha)} \int_c^t \frac{f^k(u)}{(t-u)^{\alpha-k+1}} du \tag{4}$$

In these equations,  $k$  is an integer and  $\alpha$  is a non-integer value between  $k-1$  and  $k$ .

### 2.3. Approximation

The fractional order transfer function has irrational terms and hence integer-order approximations are commonly used for the realization of fractional order systems. The term  $s^\alpha$  can be approximated using analog or digital approximation methods [43]. The Commande Robuste d’Ordre Non-Entier (CRONE), also called the Oustaloup approximation method is a widely used approximation method. The CRONE approximation for the fractional order  $\alpha$  is given as:

$$s^\alpha \approx C \prod_{i=1}^N \frac{1 + (s/\omega_{z,i})}{1 + (s/\omega_{p,i})} \tag{5}$$

This method uses  $N$  poles and  $N$  zeros, alternately placed on the negative real axis, and lying in a limited frequency range  $\{\omega_b, \omega_h\}$  [44].  $N$  is the order of approximation.

$$\omega_{z,i} = \omega_b \left(\frac{\omega_h}{\omega_b}\right)^{(2i-1-\alpha)/2N} \tag{6}$$

$$\omega_{p,i} = \omega_b \left(\frac{\omega_h}{\omega_b}\right)^{(2i-1+\alpha)/2N} \tag{7}$$

### 2.4. Fractional PID Control

Fractional order control is one of the main applications of fractional calculus, and uses fractional orders in the controller parameters. The fractional order PID (FOPID) controller commonly denoted as the  $PI^\alpha D^\beta$  controller, is an extension of the linear PID controller. Here  $\alpha$  and  $\beta$  are the fractional orders of the integrator and differentiator, respectively, and can vary from 0 to 2 [14,45]. The FOPID controller can be expressed as:

$$C(s) = K_p + K_i s^{-\alpha} + K_d s^\beta \tag{8}$$

If the values of  $\alpha$  and  $\beta$  are made equal to one, the linear PID controller is obtained. Similarly, choosing  $(\alpha, \beta) = \{(1,0), (0,1)\}$  gives the PI and PD controller, respectively. It can be easily seen that the FOPID is a generalized version of the conventional PID. By varying the values of  $\alpha$  and  $\beta$  in the range (0,2), fractional PI, PD, PID controllers are obtained. The FOPID controller has five parameters to tune  $[K_p, K_d, K_i, \alpha, \beta]$ , and hence can meet five specifications [15]. The general block diagram for a FOPID controller is shown in Figure 2 [46].

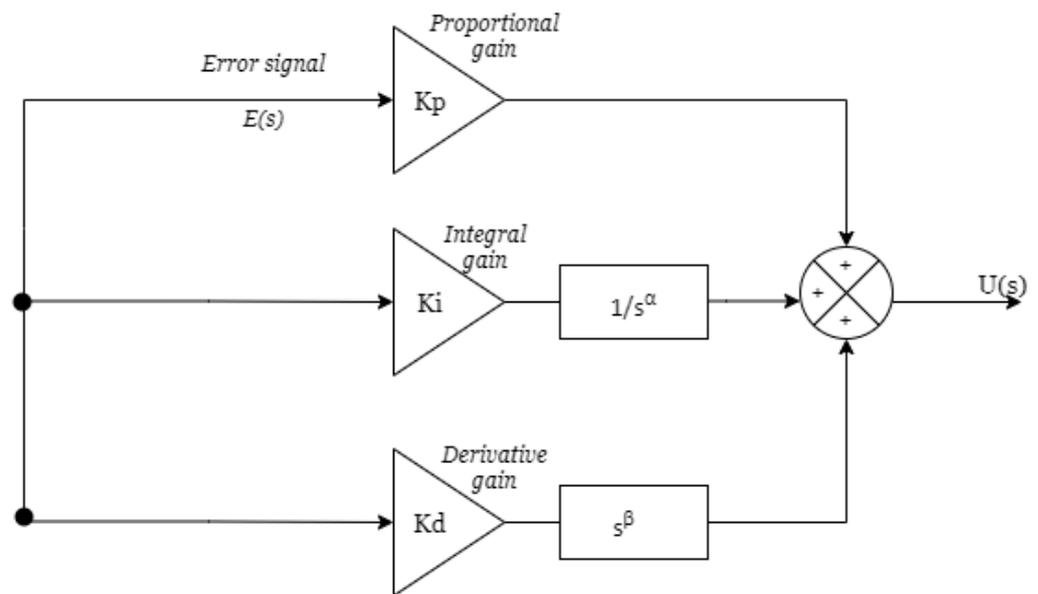


Figure 2. Fractional order PID controller.

### 2.5. CI Algorithm

The cohort intelligence algorithm, proposed by Kulkarni et al., works on the self-supervisory behavior of a group of candidates called a cohort, and has been found to be effective in solving unconstrained, as well as constrained problems [47]. A set of qualities leads to a particular behavioral characteristic. The interaction among a group of candidates in the cohort with a common goal, leads to an improved behavior of the individual candidates. Here, each candidate observes and learns from others in the group and tries to improve its qualities, resulting in an improved overall behavior of the cohort. If there is no improvement in the individual behavior of the candidates after a number of attempts, and remains the same, then the cohort behavior has reached saturation.

The CI algorithm tries to minimize an objective function  $f(x_c)$  which can be considered to be the overall candidate behavior, in which the candidate  $c$  tries to improve its behavior by modifying its set of characteristics given by  $x_c(i) = [x_{c1}, x_{c2}, \dots, x_{ci}]$ . The overall behavior of a particular candidate  $c$ ,  $f(x_c)$ , is improved by modification of its individual characteristics by observing and following the other candidates. There are seven different methods used for improving the behavior, by making variations in the CI algorithm [48]. These variations are for the convergence of the cohort behavior and for reaching global optimization. These are:

1. Best rule;
2. Better rule;
3. Worst rule;
4. Itself rule;
5. Median rule;
6. Roulette wheel selection;
7. Alienation-and-random selection.

In the 'Best rule' method, all the candidates follow the candidate with best behavior, i.e., the candidate with maximum probability of being followed. In the 'Better rule', the candidate follows a random selection of a candidate which is better than itself. The worst candidate is followed in the 'Worst rule', and the candidate follows its own behavior in the 'Itself rule'. The median probability is followed in the 'Median rule'. The 'Roulette wheel method' selects a random number between 0 and 1 and uses the roulette wheel approach to select a candidate behavior. In the 'Alienation-and-random selection method', a candidate is randomly alienated or kept apart at the beginning of the selection, and is not

followed by the other candidates. Any one of these variations can be used for reaching the optimum solution.

The probability of the candidates being followed by other candidates is calculated based on the objective function. A candidate has the best probability of being followed if its cost function is the least. A sampling interval is chosen to set up the lower and upper bounds of the quality vector  $x$ . A sampling reduction factor is also chosen, and each candidate improves its behavior by reducing or increasing the sampling interval till a saturation condition is reached. Once the saturation condition is reached, the current behavior is accepted as the final solution [32].

### 3. Methodology

#### 3.1. Optimal Tuning of the Fopid Controller for a Buck Converter

The objective is to optimally tune the parameters of an FOPID controller,  $PI^\lambda D^\mu$  applied to a buck converter for minimization of a suitable objective function  $J$ . In this paper, the performance indices of ISE, ITSE, ITAE, and IAE have been tested as objective functions for finding the optimal values of  $K_p$ ,  $K_i$ ,  $K_d$ ,  $\lambda$ , and  $\mu$  using the CI algorithm. A performance index is based on the error signal and gives a quantitative measure used for evaluating the performance of a closed loop control system. In optimal control, the system parameters are adjusted so as to minimize this index [49]. The ISE is a performance index which gives more weightage to penalization of larger errors, and ignores smaller errors [50], and has been found to have more practical significance [51]. It is defined as:

$$J_1 = ISE = \int_0^t e^2(t)dt \quad (9)$$

Here the error  $e$  is the difference between the reference voltage  $V_{ref}$  and the output voltage  $V_o$  of the converter, and can be evaluated from the circuit as  $e(t) = V_{ref} - V_o$ . The time  $t$  can be chosen equal or greater to the settling time of the system. Similarly, the other indices are defined as:

$$J_2 = ITSE = \int_0^t te^2(t)dt \quad (10)$$

$$J_3 = ITAE = \int_0^t t |e(t)| dt \quad (11)$$

$$J_4 = IAE = \int_0^t |e(t)| dt \quad (12)$$

#### 3.2. System Description

A buck converter system with a fractional order PID controller has been realized in SIMULINK, for voltage mode control, as shown in Figure 3. Table 1 shows the parameters of the buck converter, which are defined for CCM, according to the equations defined in Section 2.1. The mathematical model of the system using state space averaging is usually used in controller design, but this may lead to approximation errors [31]. Hence the circuit with FOPID controller has been simulated in SIMULINK for a more accurate representation of the system. The fractional order integrator and differentiator of the FOPID controller  $G_c(s)$  are approximated using the Oustaloup approximation of order 5 in a frequency band of [0.01, 1,000,000] rad/s. The FOPID parameters are calculated by the CI algorithm using an m-file in MATLAB, which calls the SIMULINK circuit repeatedly. The FOPID parameters are selected from a sampling interval  $\phi_{K_p} = \phi_{K_i} = \phi_{K_d} = [0, 200]$ , and  $\phi_\lambda = \phi_\mu = [0, 1]$  after some trial runs of the algorithm. The respective objective function (ISE, ITSE, ITAE, or IAE) is calculated in SIMULINK, and the FOPID parameters are continuously updated for each iteration. These parameters and the objective function are returned to the program after the termination condition is achieved.

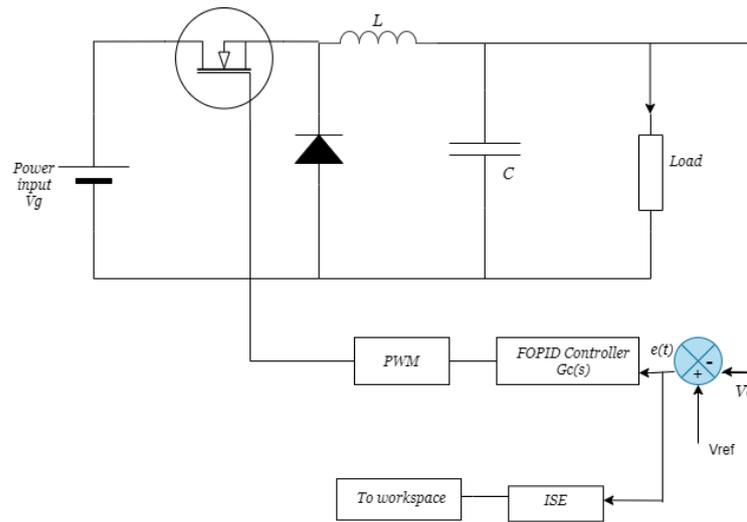


Figure 3. Buck converter with FOPID controller.

Table 1. Buck Converter Circuit Parameters.

| Parameter               | Value        |
|-------------------------|--------------|
| Filter Inductance       | 70 $\mu$ H   |
| Filter Capacitance      | 22 $\mu$ F   |
| Load resistance, R      | 3 $\Omega$   |
| Switching frequency     | 100 KHz      |
| Input voltage           | 24 V         |
| Output voltage          | 15 V         |
| Inductor current ripple | 20% of $i_L$ |

The CI algorithm is used to minimize the respective cost function  $J_i$ , which can be considered as the behavior of a candidate of a cohort which has to be improved. The controller parameters can be considered to be the characteristics of the candidate, randomly generated from a search space. The steps for the CI algorithm can be summarized as follows [52]:

1. Select the no. of candidates 'C' of the cohort, the reduction interval ' $r_i$ ', convergence parameter  $\epsilon$ , and the maximum no.of iterations  $max_{it}$ ;
2. The lower bound and upper bound of the controller parameters  $\phi_{K_p}, \phi_{K_i}, \phi_{K_d}, \phi_\lambda$  and  $\phi_\mu$  are chosen based on some initial runs of the algorithm;
3. For each candidate c in the cohort, the qualities (parameters  $K_p, K_i, \lambda, K_d$  and  $\mu$ ) are generated from the lower and upper bounds as:

$$K_p^c = \min(\phi_{K_p}) + (\max(\phi_{K_p}) - \min(\phi_{K_p})) * \text{rand}(.), \tag{13}$$

where rand is a random number between 0 and 1. Similarly, the values for  $K_i, K_d$ , etc., are generated;

4. The overall behavior (J) of each candidate is calculated from the error, as given by Equation (9) (in this case);
5. The probability function for each candidate is calculated as:

$$P = \frac{1/J}{\sum_{c=1}^C 1/J}; c = 1, 2 \dots C \tag{14}$$

6. A roulette wheel approach is used by each candidate to follow another candidate's behavior. This approach uses one generation as the basis of the next generation. The best solution has the highest probability of being followed;
7. Accordingly, each candidate adjusts its sampling interval as shown below for  $K_p$ :

$$\phi_{K_p} \in [K_p - \|(max(\phi_{K_p}) - min(\phi_{K_p})) * r_i/2\|, K_p + \|(max(\phi_{K_p}) - min(\phi_{K_p})) * r_i/2\|] \quad (15)$$

Similarly, the other four parameters also are adjusted;

8. This process continues till the value of J reaches a saturation condition defined as:

$$max(J)^n - max(J)^{n-1} \leq \epsilon \quad (16)$$

$$min(J)^n - min(J)^{n-1} \leq \epsilon \quad (17)$$

$$max(J)^n - min(J)^{n-1} \leq \epsilon \quad (18)$$

9. If the maximum limit of iterations  $max_{it}$  is exceeded or the saturation condition defined in step 8 is reached, then the algorithm terminates. The final solution for the objective function can be accepted from any one of the C behaviors;
10. If conditions of steps 8 or 9 are not satisfied, jump to step 3.

A flowchart of the methodology is shown in Figure 4.

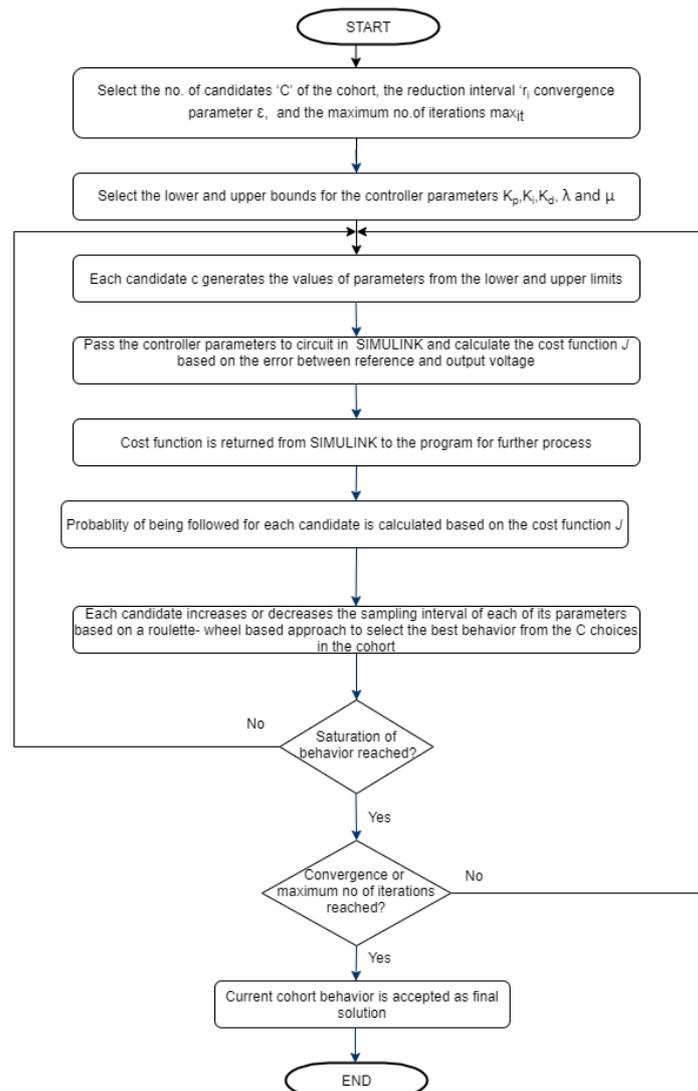


Figure 4. CI Flowchart used for Buck converter control.

#### 4. Results and Discussion

MATLAB R2020b was used for implementation of the cohort intelligence algorithm on Windows 10 with Intel Core i5 processor. The specifications of the algorithm is listed in Table 2. The algorithm was tested with various values of the reduction factor  $r_i$ , and was run for 25 times to test the accuracy of the algorithm. The test runs showed lesser overshoot with a reduction factor of 0.55 and 0.45.

**Table 2.** CI algorithm specifications.

| Parameters               | Values     |
|--------------------------|------------|
| No. of cohort candidates | 4–5        |
| No. of variables         | 5          |
| Reduction factor $r_i$   | 0.45, 0.55 |
| Convergence constant     | 0.001      |
| Maximum iterations       | 25         |

##### 4.1. Start-up Response of the Buck Converter

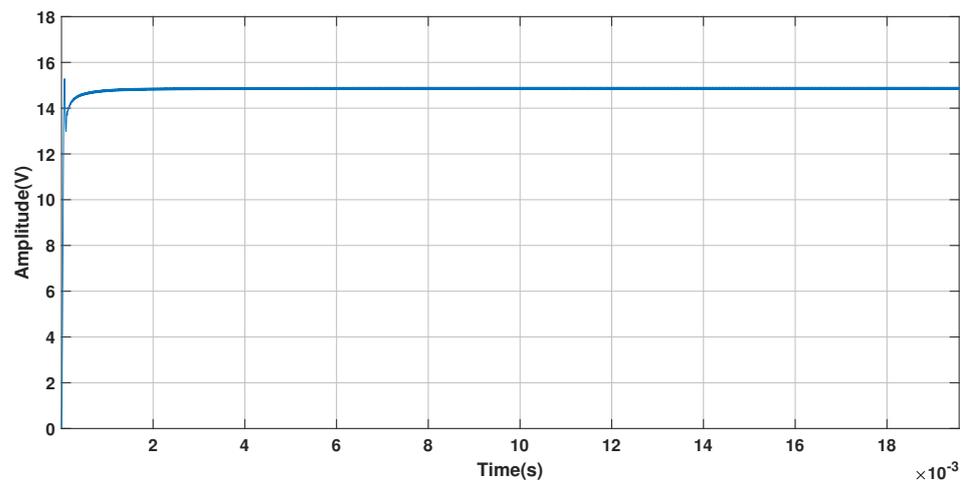
The CI algorithm was used to optimize the voltage response of the buck converter circuit at start-up in SIMULINK with the FOPID parameters as design variables. The initial trial runs showed better results with the sampling interval chosen as  $[0, 200]$  for  $K_p$ ,  $K_i$  and  $K_d$ , and  $[0, 1]$  for  $\lambda$  and  $\mu$ . The start-up response of the buck converter was optimized with different values of the reduction factor  $r_i$ . Figure 5 shows the most optimum time-domain response for the cost function ISE using reduction factors of  $r_i = 0.55$  and  $r_i = 0.45$ .

For the cost function  $J_1 = ISE$ , the value of overshoot was greatly reduced by setting the reduction factor  $r_i$  to 0.45 and 0.55 compared to higher values of  $r_i$ . The inductor current ripple was also limited to less than 20% of the output current (Figure 6). With  $r_i = 0.45$ , the best result gave zero overshoot, and the rise time was  $3.7 \times 10^{-5}$  seconds and settling time around 0.000265 s, which shows a fast transient response. The output voltage ripple was around 75 mV. For  $r_i = 0.55$ , the overshoot and settling time slightly increased to 1.86% and 0.00061 s, respectively, while the rise time remained roughly the same.

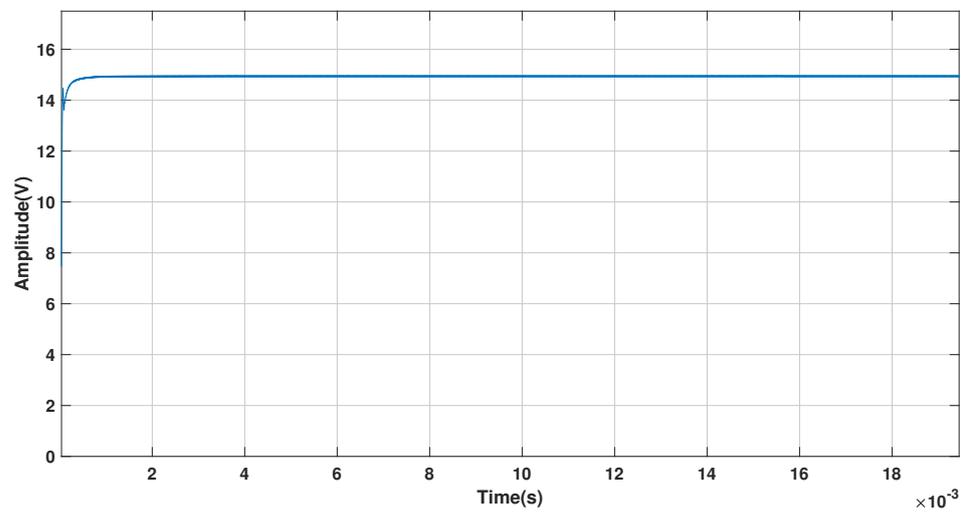
Similarly, the responses were found to be better for all the cost functions with  $r_i = 0.55$  and  $r_i = 0.45$ . The optimal parameters for each cost function has been tabulated in Table 3. It is seen that while the rise time and settling time remains almost the same, the overshoot is more for the cost functions ITAE and IAE. The responses of the CI optimized system with different cost functions is shown in Figure 7.

**Table 3.** Response of CI optimized converter with different cost functions.

| Parameters          | ISE ( $r_i = 0.45$ ) | ITSE ( $r_i = 0.55$ ) | ITAE ( $r_i = 0.45$ ) | IAE ( $r_i = 0.45$ ) |
|---------------------|----------------------|-----------------------|-----------------------|----------------------|
| $K_p$               | 162.08               | 27.9709               | 123.0768              | 197.3838             |
| $K_i$               | 133.84               | 112.7302              | 122.9522              | 115.0357             |
| $K_d$               | 0.5851               | 0.7737                | 54.0215               | 63.0392              |
| $\lambda$           | 0.0673               | 0.1022                | 0.4738                | 0.3730               |
| $\mu$               | 0.6107               | 0.5868                | 0.2537                | 0.1874               |
| Mp (%)              | 0                    | 0                     | 20                    | 21.3                 |
| Tr (ms)             | 0.037                | 0.035                 | 0.0365                | 0.0354               |
| Tss (ms)            | 0.265                | 0.35                  | 0.27                  | 0.3                  |
| Cost function value | 0.006                | 0.0288                | 0.0004                | 0.0119               |



(a) Response with  $r_i = 0.55$ .



(b) Response with  $r_i = 0.45$ .

Figure 5. Startup response of CI-ISE optimized Buck converter with variation of reduction factor.

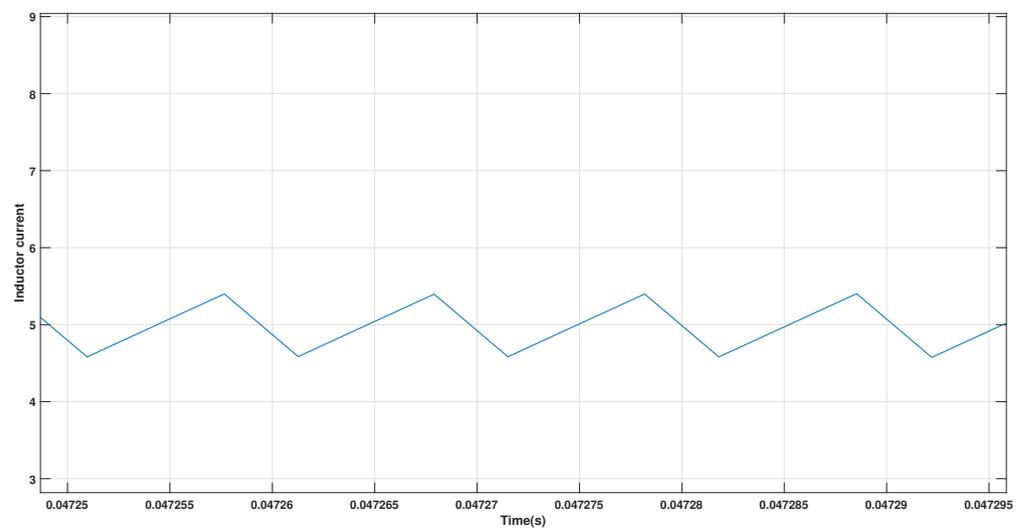


Figure 6. Inductor current ripple of system with CI-ISE and  $r_i = 0.45$ .

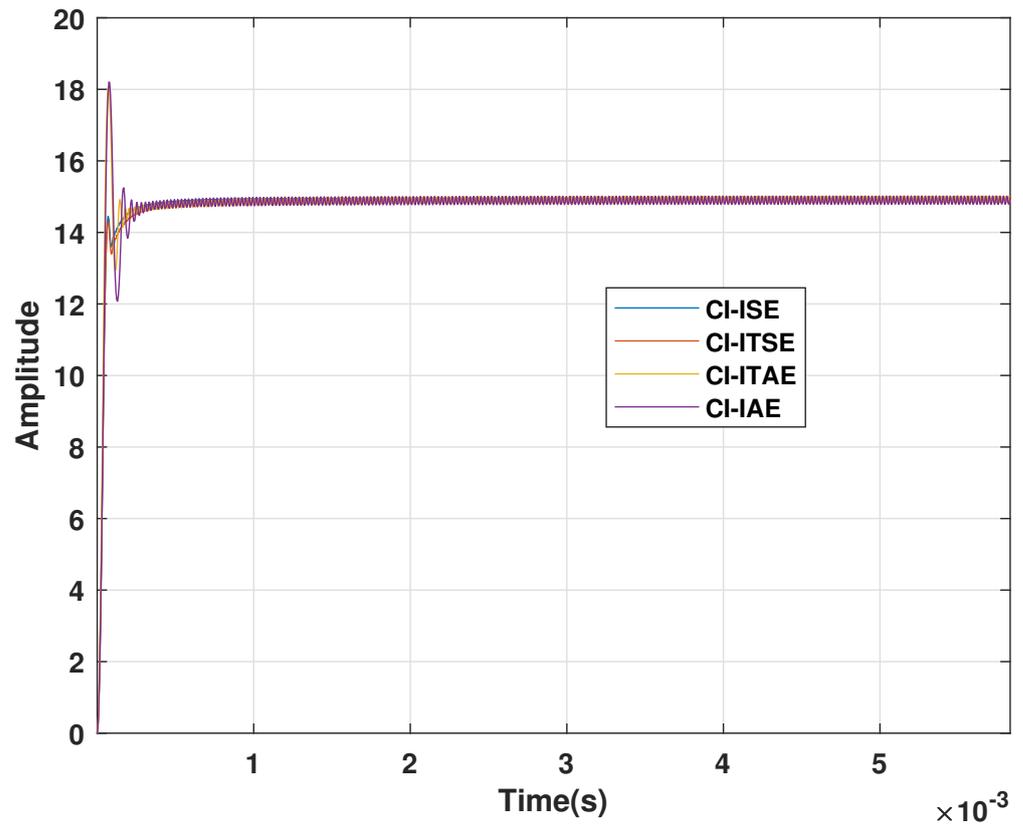


Figure 7. CI-optimized system responses for different cost functions.

4.2. Response of the CI-Optimized System to DC Gain Variations and Dynamic Changes

The response of the circuit to variation of the DC gain upto 45% is shown in Figure 8. The FOPID controller responds well and shows robustness to a range of gain variations. The system settles in approximately 0.25 ms for each case, with a maximum overshoot of 11.9% for a gain increase upto 45%. The rise time is also small, around 0.4 ms for each case. At the same time, the PID controller has a maximum overshoot of 56.7% for gain increase upto 45%. The system settles around 0.45 ms with more oscillations (Figure 9). The FOPID controller has a smoother response compared to PID.

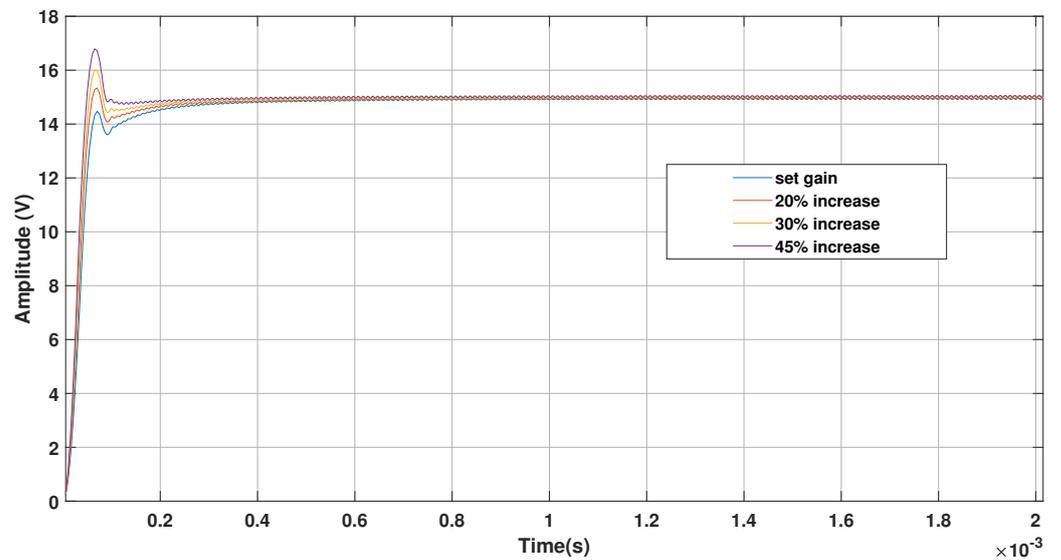
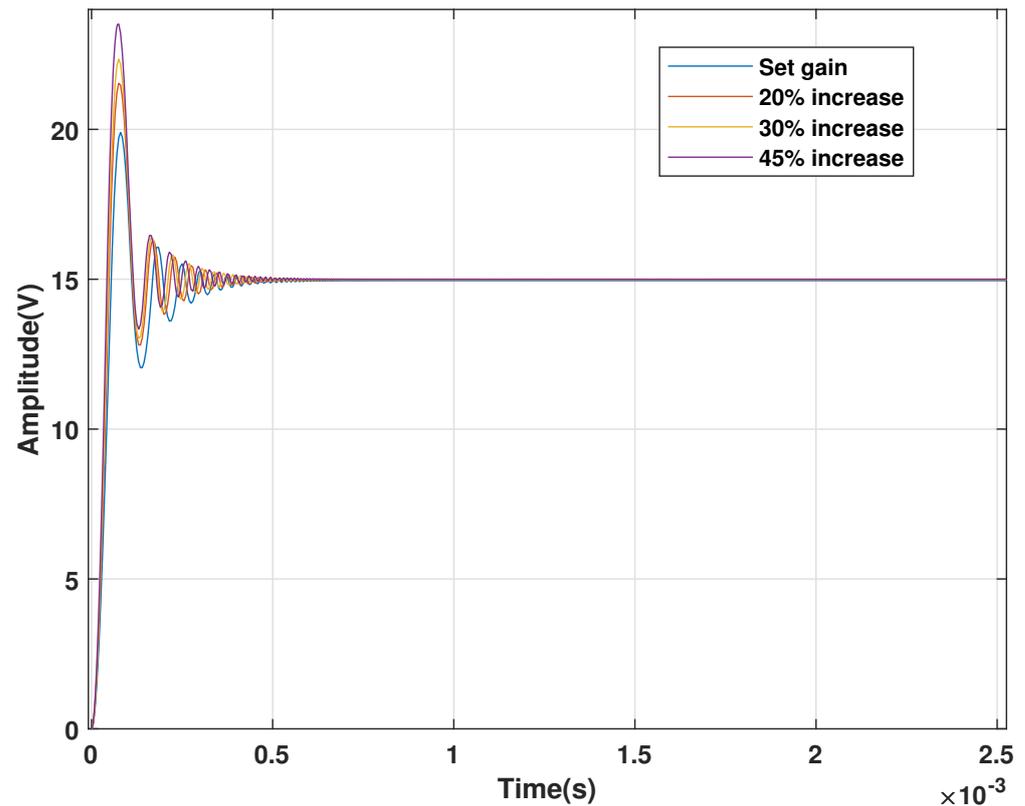


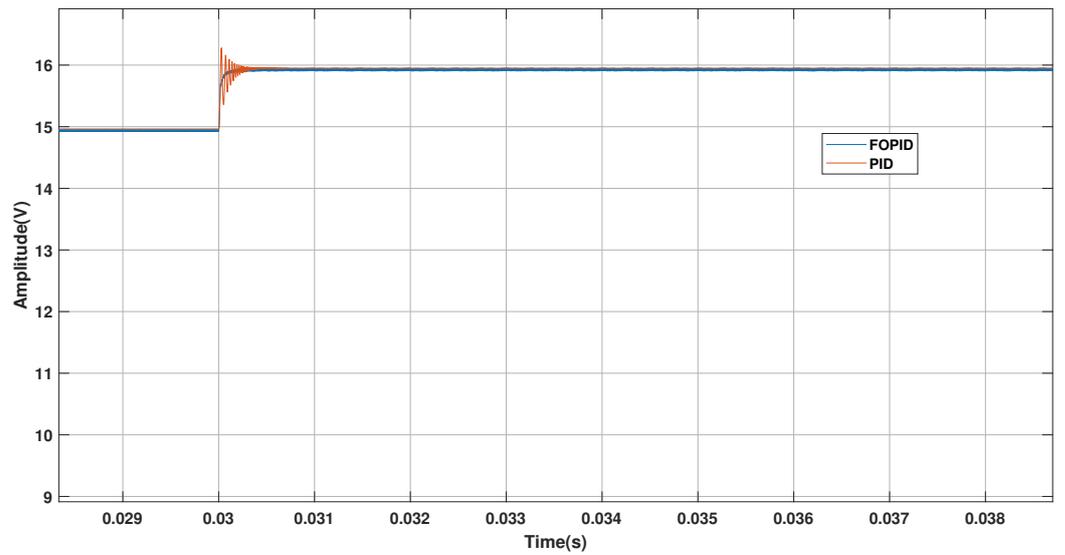
Figure 8. Response of CI optimized FOPID to variation of gain.



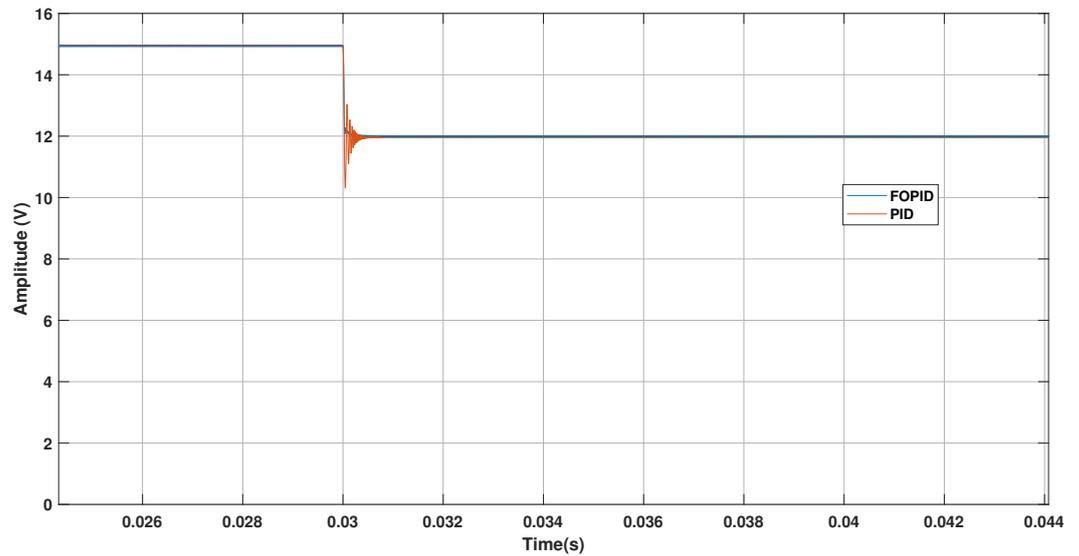
**Figure 9.** Response of CI optimized PID controller to variation of gain.

The response of the FOPID circuit with CI was analyzed for different set-point conditions and compared with that of a PID controller optimized using the CI algorithm. The PID controlled circuit had much higher overshoot than the FOPID based circuit. A change in the reference value from 15 V to 16 V at 0.03 s is shown in Figure 10a. The response of the circuit to a set-point change from 15 V to 12 V at 0.03 s is shown in Figure 10b. The FOPID controller gave a much smoother transition in both cases compared to the PID controller.

The dynamic response of the CI-ISE optimized FOPID system was studied for step changes in the load, and compared with the PID controller under the same conditions. The system was subjected to a step load change from 3  $\Omega$  to 20  $\Omega$  at 0.006 s and from 20  $\Omega$  to 10  $\Omega$  at 0.01 s. The response is shown in Figure 11. The FOPID system had a slightly higher overshoot for the step change at 0.0006 s, but settled to the steady state value in lesser time of 0.4 ms compared to 1.2 ms for the PID controller. At the step load change at 0.1 s, the FOPID controller settled in 0.2 ms, whereas the PID controller took 0.76 ms to settle. The inductor current step variations are more for the PID controller and takes much more time to settle in both the cases.



(a) Set-point varied from 15 V to 16 V.



(b) Set-point reduced from 15 V to 12 V.

**Figure 10.** Response of the FOPID and PID controllers to set-point variation.

#### 4.3. Response of CI-Optimized System to Parametric Variations

The buck converter with CI optimized FOPID controller was subjected to parametric variations. The system response with CI+ISE to variations in the filter parameters is shown in Figure 12. The system response to load variations is shown in Figure 13. It is seen that the same FOPID controller is able to handle parametric variations quite efficiently without retuning.

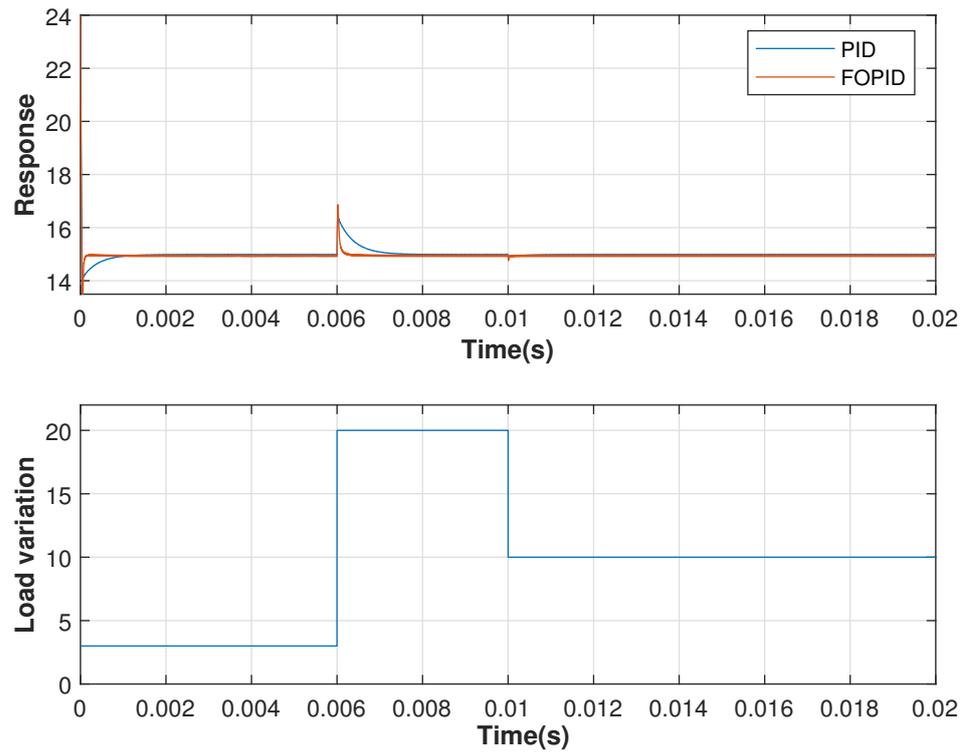


Figure 11. Dynamic Response of CI-ISE optimized system to step-load variation.

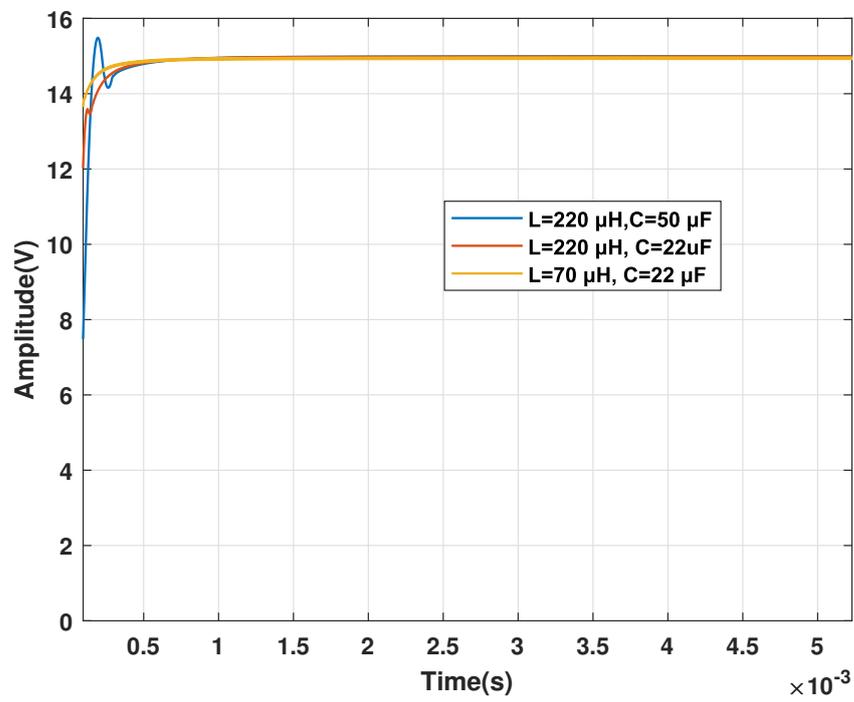


Figure 12. CI optimized system response to filter parameter variations.

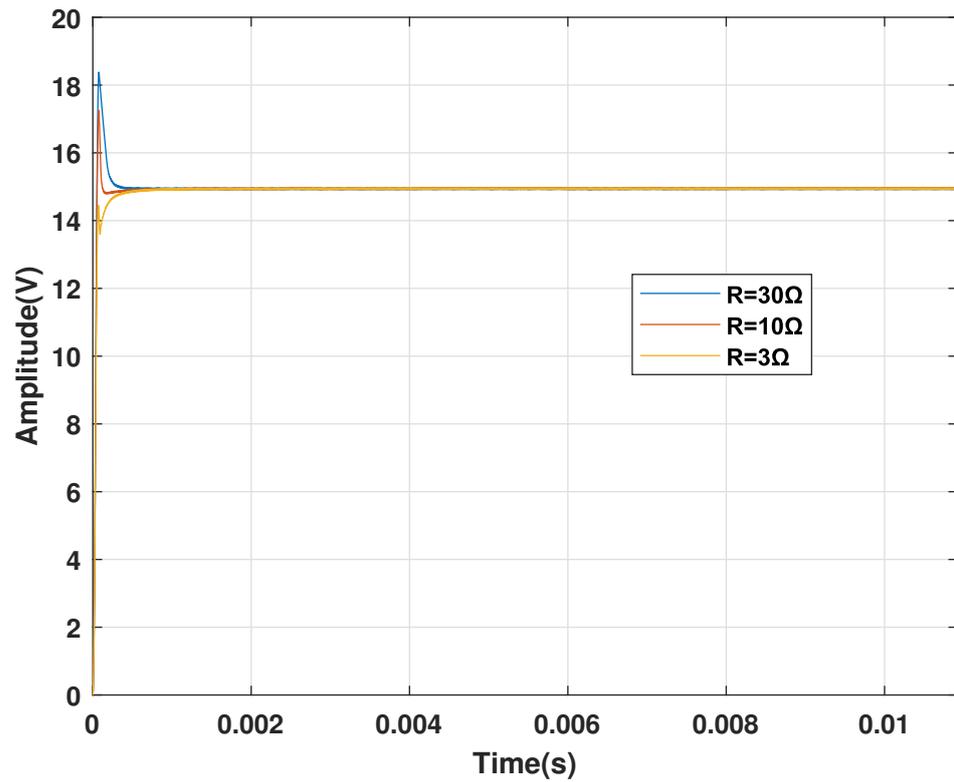
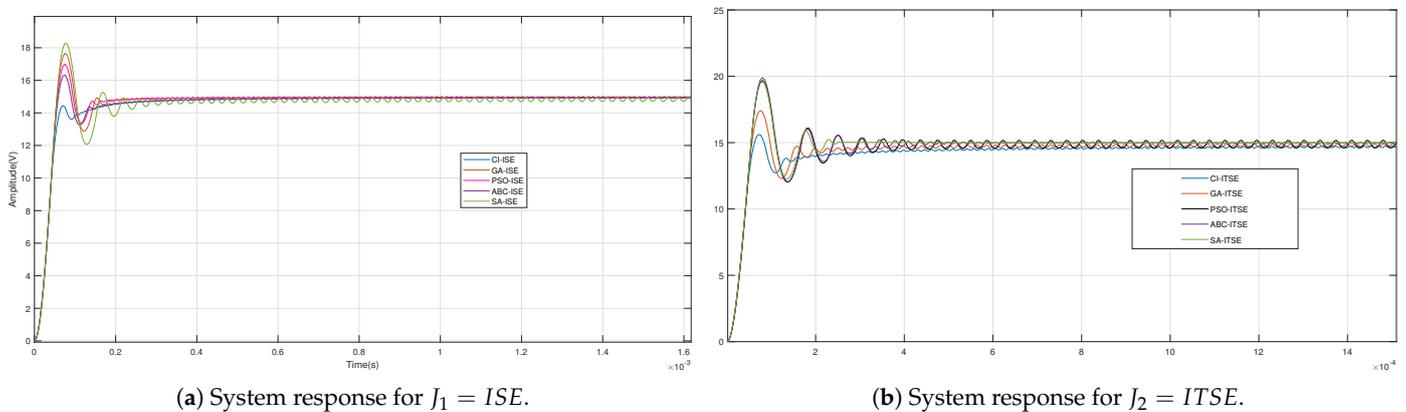


Figure 13. CI optimized system response to load variations.

#### 4.4. Performance Comparison of CI

The efficiency of the CI algorithm for optimization of the FOPID parameters was compared with the genetic algorithm (GA) [26], particle swarm optimization (PSO) [25], artificial bee colony (ABC) [24], and the simulated annealing (SA) [53,54] methods for the same cost function ISE. The PSO and the ABC methods are swarm intelligence methods inspired by the collective behavior of animals or insect colonies, whereas the Genetic algorithm is an evolutionary search algorithm based on natural selection process and genetics. The SA optimization method can be used for multi-objective optimization and works like the annealing process used in metallurgy for heating and cooling of materials. All are meta-heuristic search algorithms like the CI and hence are used for comparison. The response of the FOPID controlled buck converter system was optimized using different algorithms, and is shown in Figures 14 and 15.



(a) System response for  $J_1 = ISE$ .

(b) System response for  $J_2 = ITSE$ .

Figure 14. System response for optimization of ISE and ITSE.

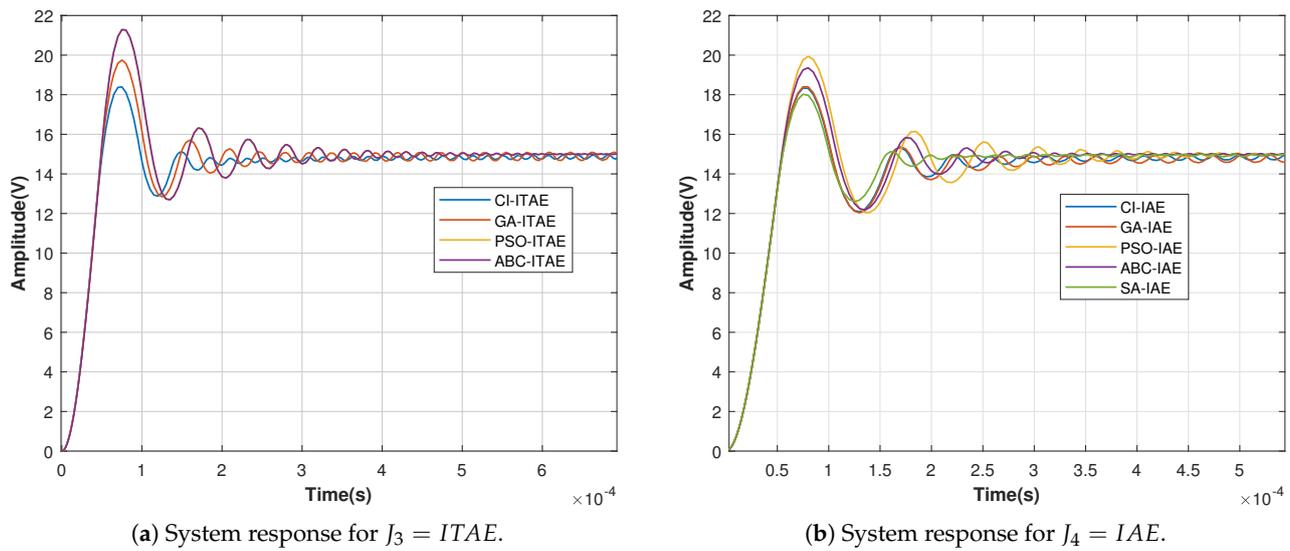


Figure 15. System response for optimization of ITAE and IAE.

A comparison of the time domain specifications and the cost function of the CI method with the other optimization methods is shown in Tables 4–7. The rise time and settling times are comparable in all the methods, but the overshoot and the computational time is least for the CI method for the cost function  $J_1 = ISE$ . The average number of functional evaluations is much lesser in the case of the CI method, which indicates a lesser control effort. The value of the minimized objective function is also least for CI with  $J_1$ .

For the cost function  $J_2$ , the overshoot is small for CI and PSO methods, with the latter having the least average overshoot. The cost function value is small for the PSO and ABC methods, but computational time is very high, increased by 57.4% in PSO and by 96.4% in the ABC method. For the cost function  $J_2$ , the cost function of the PSO is the best and the CI value is worst. However, the computational time is still the least in the case of the CI, being much lesser than all the other methods.

For the cost function  $J_4$ , the cost function values are comparable for all the methods. The simulated annealing method fails to reach the steady state value even after 200 iterations, and hence cannot be suitable for this index. The overshoot is increased in all the methods, with the least overshoot and cost function given by the ABC. Even though the computation time has increased for the CI, it continues to take lesser time for convergence compared to all the other methods.

Table 4. Comparison of different optimization methods for the cost function ISE.

| Parameters                | $J_1 = ISE$ |        |        |        |        |
|---------------------------|-------------|--------|--------|--------|--------|
|                           | CI          | GA     | PSO    | ABC    | SA     |
| Avg. Overshoot %          | 5.6         | 14.6   | 16.2   | 18     | 21     |
| Rise Time (ms)            | 0.037       | 0.038  | 0.036  | 0.035  | 0.035  |
| Settling time (ms)        | 0.265       | 0.19   | 0.167  | 0.24   | 0.32   |
| Cost function             | 0.006       | 0.0057 | 0.0063 | 0.0067 | 0.0064 |
| Avg. computation Time (s) | 542         | 1600   | 720    | 1100   | 1800   |
| Avg. Function evaluations | 84          | 242    | 88     | 100    | 400    |
| Avg. No. of iterations    | 21          | 60     | 22     | 25     | 100    |

**Table 5.** Comparison of different optimization methods for the cost function ITSE.

| Parameters                | $J_2 = ITSE$ |         |                      |                      |         |
|---------------------------|--------------|---------|----------------------|----------------------|---------|
|                           | CI           | GA      | PSO                  | ABC                  | SA      |
| Avg. Overshoot %          | 9.2          | 16.11   | 5                    | 17.3                 | 25      |
| Rise Time (ms)            | 0.042        | 0.038   | 0.043                | 0.039                | 0.039   |
| Settling time (ms)        | 0.39         | 0.42    | 0.37                 | 0.38                 | 0.27    |
| Cost function value       | 0.0001       | 0.00018 | $1.7 \times 10^{-5}$ | $1.7 \times 10^{-5}$ | 0.00039 |
| Avg. computation Time (s) | 762          | 1500    | 1200                 | 1500                 | 1320    |
| Avg. Function evaluations | 84           | 204     | 88                   | 100                  | 1000    |
| Avg. No. of iterations    | 21           | 51      | 21                   | 25                   | 200     |

**Table 6.** Comparison of different optimization methods for the cost function ITAE.

| Parameters                | $J_3 = ITAE$       |                       |                       |                       |                       |
|---------------------------|--------------------|-----------------------|-----------------------|-----------------------|-----------------------|
|                           | CI                 | GA                    | PSO                   | ABC                   | SA                    |
| Avg. Overshoot %          | 20                 | 29                    | 26                    | 18                    | -                     |
| Rise Time (ms)            | 0.036              | 0.037                 | 0.039                 | 0.036                 | -                     |
| Settling time (ms)        | 0.31               | 0.36                  | 0.39                  | 0.40                  | -                     |
| Cost function value       | $4 \times 10^{-4}$ | $3.44 \times 10^{-4}$ | $3.97 \times 10^{-4}$ | $2.65 \times 10^{-5}$ | $3.97 \times 10^{-4}$ |
| Avg. computation Time (s) | 1020               | 1500                  | 1200                  | 1500                  | 1800                  |
| Avg. Function evaluations | 84                 | 204                   | 88                    | 100                   | 1000                  |
| Avg. No. of iterations    | 21                 | 51                    | 21                    | 25                    | 200                   |

**Table 7.** Comparison of different optimization methods for the cost function IAE.

| Parameters                | $J_4 = IAE$ |         |        |        |       |
|---------------------------|-------------|---------|--------|--------|-------|
|                           | CI          | GA      | PSO    | ABC    | SA    |
| Avg. Overshoot %          | 20          | 21      | 25     | 18     | 25    |
| Rise Time (ms)            | 0.036       | 0.038   | 0.039  | 0.036  | 0.037 |
| Settling time (ms)        | 0.28        | 0.42    | 0.39   | 0.38   | 0.29  |
| Cost function value       | 0.0116      | 0.00548 | 0.0017 | 0.0024 | 0.005 |
| Avg. computation Time (s) | 542         | 1500    | 1200   | 1500   | 1320  |
| Avg. Function evaluations | 84          | 204     | 88     | 100    | 1000  |
| Avg. No. of iterations    | 21          | 51      | 21     | 25     | 200   |

4.5. Convergence Plots of the CI Algorithm

The CI algorithm has a self-supervisory learning behavior for the candidates. This can be shown in the convergence plot of Figure 16, which shows the convergence of the candidates towards the required cost function. Similarly, the convergence of the FOPID parameters towards the final value is shown in Figure 17. The fractional controller parameters start converging to the solution after 5 iterations, and reaches optimal value in 21 iterations.

The error and control signals of the CI optimized plant with FOPID controller for a reduction factor of 0.45 is shown in Figure 18.

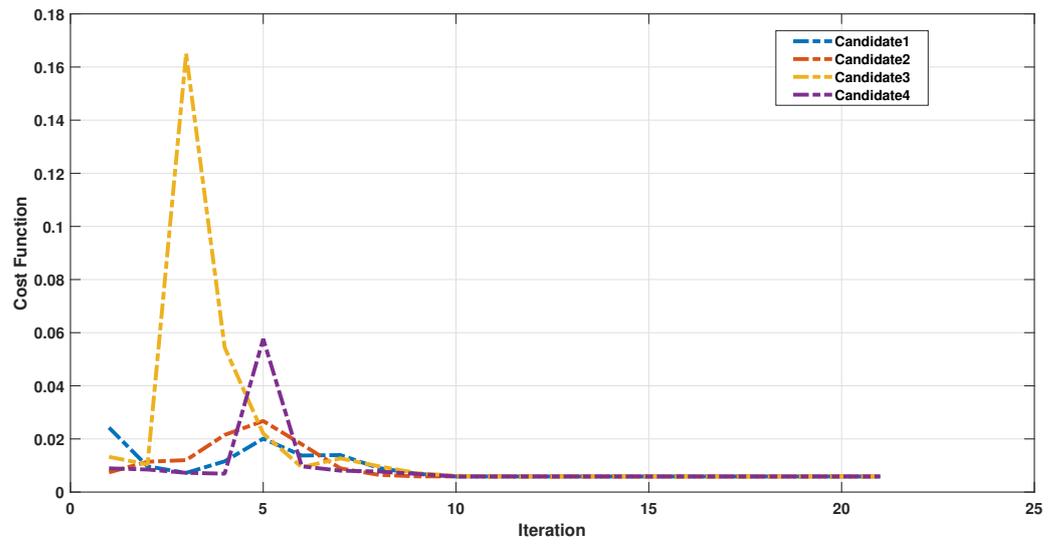


Figure 16. Convergence plot of cost function.

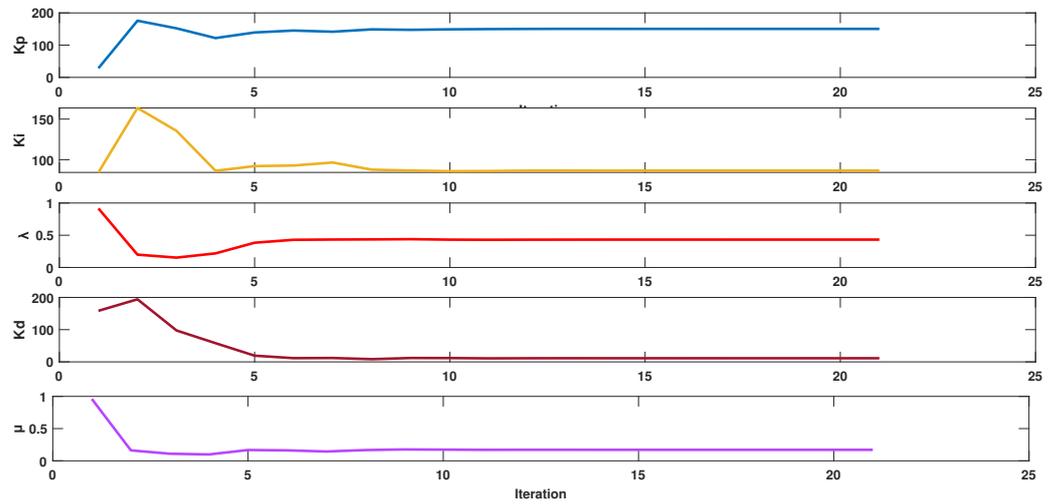
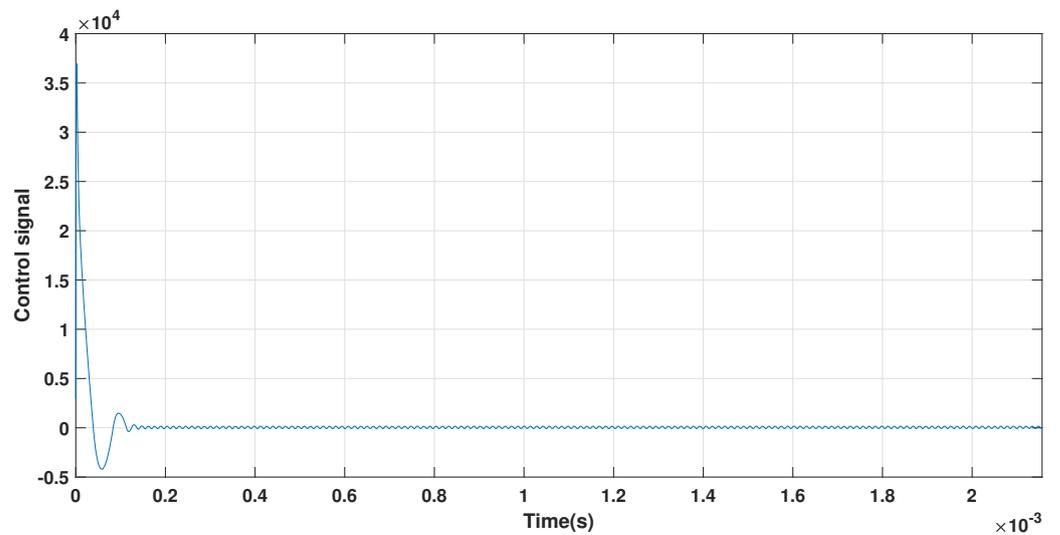
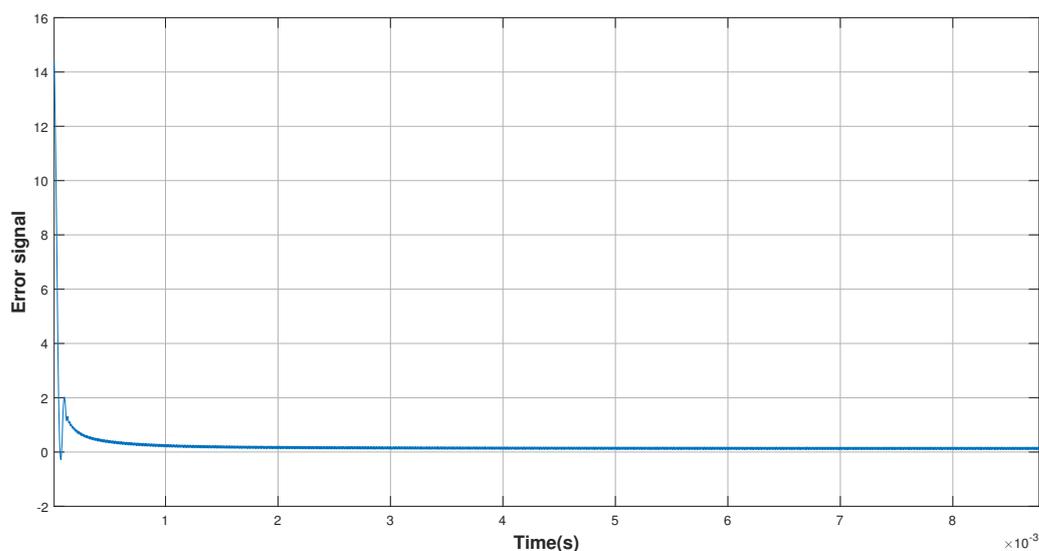


Figure 17. Convergence plot of the FOPID parameters.



(a) Control signal of FOPID.

Figure 18. Cont.



(b) Error signal.

Figure 18. Control and error signals.

## 5. Conclusions

This article discusses a novel method for the design of a fractional order PID controller for a buck converter using the cohort intelligence meta-heuristic algorithm. The performance indices of ISE, ITSE, ITAE, and IAE were optimized using the CI algorithm. The ISE index minimization gave the best results. The proposed system showed a fast start-up response and smooth dynamic response. It was observed that the reduction factor of the CI algorithm had an effect on the time domain response, as well as the cost function value. The performance of the system with CI algorithm was compared with the GA, PSO, ABC, and SA optimization algorithms for different cost functions. The CI algorithm gave faster convergence to the optimal solution in comparison to all the other methods. The much lesser function count and computation time required is a clear advantage of the CI compared to the other methods and also reflects the lesser control effort required. The overshoot and the average value of the objective function was the least for the CI method with ISE. The cost function values for ITSE and IAE were found to be lesser for the PSO and ABC methods, but at the cost of a much higher computational time. These show that the CI algorithm can be an effective method for the optimal tuning of the FOPID parameters. Optimization with CI for minimization of weighted combinations of the performance indices can be tried in future. Future research will be performed on extension of CI algorithm for multi-objective optimization and for the control of converters such as the boost and buck-boost converters.

**Author Contributions:** Conceptualization, P.S.; Methodology, P.W.; Writing—original draft, P.W.; Writing—review & editing, P.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** No data is available in this study.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Raviraj, V.S.C.; Sen, P.C. Comparative study of proportional-integral, sliding mode, and fuzzy logic controllers for power converters. *IEEE Trans. Ind. Appl.* **1997**, *33*, 518–524. [[CrossRef](#)]
2. Seo, S.W.; Choi, H.H. Digital Implementation of Fractional Order PID-Type Controller for Boost DC–DC Converter. *IEEE Access* **2019**, *7*, 142652–142662. [[CrossRef](#)]
3. Habetler, T.G.; Harley, R.G. Power electronic converter and system control. *Proc. IEEE* **2001**, *89*, 913–925. [[CrossRef](#)]
4. Vidal-Idiarte, E.; Martinez-Salamero, L.; Guinjoan, F.; Calvente, J.; Gomariz, S. Sliding and fuzzy control of a boost converter using an 8-bit microcontroller. *IEE Proc. Electr. Power Appl.* **2004**, *151*, 5–11. [[CrossRef](#)]
5. Chan, H.; Chau, K.T.; Chan, C.C. A neural network controller for switching power converters. In Proceedings of the IEEE Power Electronics Specialist Conference—PESC 93, Seattle, WA, USA, 20–24 June 1993; pp. 887–892. [[CrossRef](#)]
6. Zhang, G.; Li, Z.; Zhang, B.; Halang, W.A. Power electronics converters: Past, present and future. *Renew. Sustain. Energy Rev.* **2018**, *81*, 2028–2044. [[CrossRef](#)]
7. Borase, R.; Maghade, D.; Sondkar, S.; Pawar, S. A review of PID control, tuning methods and applications. *Int. J. Dyn. Control* **2021**, *9*, 818–827. [[CrossRef](#)]
8. Aström, K.J.; Hägglund, T. *PID Controllers—Theory, Design, and Tuning*; International Society of Automation (ISA): Pittsburgh, PA, USA, 1995.
9. Ghosh, A.; Prakash, M.; Pradhan, S.; Banerjee, S. A comparison among PID, Sliding Mode and internal model control for a buck converter. In Proceedings of the IECON 2014—40th Annual Conference of the IEEE Industrial Electronics Society, Dallas, TX, USA, 29 October–1 November 2014; pp. 1001–1006. [[CrossRef](#)]
10. Monje, C.A.; Vinagre, B.M.; Feliu, V.; Chen, Y. Tuning and auto-tuning of fractional order controllers for industry applications. *Control Eng. Pract.* **2008**, *16*, 798–812. [[CrossRef](#)]
11. Oustaloup, A.; Levron, F.; Mathieu, B.; Nanot, F.M. Frequency-band complex noninteger differentiator: characterization and synthesis. *IEEE Trans. Circuits Syst. Fundam. Theory Appl.* **2000**, *47*, 25–39. [[CrossRef](#)]
12. Chen, Y.; Petras, I.; Xue, D. Fractional order control—A tutorial. In Proceedings of the 2009 American Control Conference, St. Louis, MI, USA, 10–12 June 2009; pp. 1397–1411. [[CrossRef](#)]
13. Tepljakov, A.; Alagoz, B.B.; Yeroglu, C.; Gonzalez, E.; HosseinNia, S.H.; Petlenkov, E. FOPID Controllers and Their Industrial Applications: A Survey of Recent Results. *IFAC-PapersOnLine* **2018**, *51*, 25–30. [[CrossRef](#)]
14. Shah, P.; Agashe, S. Review of fractional PID controller. *Mechatronics* **2016**, *38*, 29–41. [[CrossRef](#)]
15. Monje, C.; Chen, Y.; Vinagre, B.; Xue, D.; Feliu, V. *Fractional Order Systems and Control—Fundamentals and Applications*; Springer: London, UK, 2010. [[CrossRef](#)]
16. Calderón, A.; Vinagre, B.; Feliu, V. Fractional order control strategies for power electronic buck converters. *Signal Process.* **2006**, *86*, 2803–2819. [[CrossRef](#)]
17. Tepljakov, A.; Alagoz, B.B.; Yeroglu, C.; Gonzalez, E.A.; Hosseinnia, S.H.; Petlenkov, E.; Ates, A.; Cech, M. Towards Industrialization of FOPID Controllers: A Survey on Milestones of Fractional-Order Control and Pathways for Future Developments. *IEEE Access* **2021**, *9*, 21016–21042. [[CrossRef](#)]
18. Prajapati, S.; Garg, M.M.; Prithvi, B. Design of Fractional-Order PI controller for DC-DC Power Converters. In Proceedings of the 2018 8th IEEE India International Conference on Power Electronics (IICPE), Jaipur, India, 13–15 December 2018; pp. 1–6.
19. Djebri, S.; Ladaci, S.; Metatla, A. Fractional-order model reference adaptive control of a multi-source renewable energy system with coupled DC/DC converters power compensation. *Energy Syst.* **2020**, *11*, 315–355. [[CrossRef](#)]
20. Qi, Z.; Tang, J.; Pei, J.; Shan, L. Fractional Controller Design of a DC-DC Converter for PEMFC. *IEEE Access* **2020**, *8*, 120134–120144. [[CrossRef](#)]
21. Sánchez, A.G.S.; Soto-Vega, J.; Tlelo-Cuautle, E.; Rodríguez-Licea, M.A. Fractional-Order Approximation of PID Controller for Buck–Boost Converters. *Micromachines* **2021**, *12*, 591. [[CrossRef](#)] [[PubMed](#)]
22. Warriar, P.; Shah, P. Fractional Order Control of Power Electronic Converters in Industrial Drives and Renewable Energy Systems: A Review. *IEEE Access* **2021**, *9*, 58982–59009. [[CrossRef](#)]
23. Rao, S. *Engineering Optimization: Theory and Practice*, 4th ed.; John Wiley and Sons: Hoboken, NJ, USA, 2009. [[CrossRef](#)]
24. Karaboga, D.; Gorkemli, B.; Ozturk, C.; Karaboga, N. A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artif. Intell. Rev.* **2014**, *42*, 21–57. [[CrossRef](#)]
25. Maiti, D.; Acharya, A.; Chakraborty, M.; Konar, A.; Janarthanan, R. Tuning PID and  $PI^{\lambda}D^{\sigma}$  Controllers using the Integral Time Absolute Error Criterion. In Proceedings of the 2008 4th International Conference on Information and Automation for Sustainability, Colombo, Sri Lanka, 12–14 December 2008. [[CrossRef](#)]
26. Cao, J.-Y.; Liang, J.; Cao, B.-G. Optimization of fractional order PID controllers based on genetic algorithms. In Proceedings of the 2005 International Conference on Machine Learning and Cybernetics, Guangzhou, China, 18–21 August 2005; Volume 9, pp. 5686–5689. [[CrossRef](#)]
27. Das, S.; Pan, I.; Das, S.; Gupta, A. A Novel Fractional Order Fuzzy PID Controller and Its Optimal Time Domain Tuning Based on Integral Performance Indices. *Eng. Appl. AI* **2012**, *25*, 430–442. [[CrossRef](#)]
28. Khubalkar, S.; Junghare, A.S.; Aware, M.V.; Chopade, A.; Das, S. Demonstrative fractional order PID controller based DC motor drive on digital platform. *ISA Trans.* **2017**, *82*, 79–93. [[CrossRef](#)] [[PubMed](#)]

29. Khubalkar, S.; Chopade, A.; Junghare, A.; Aware, M.; Das, S. Design and Realization of Stand-Alone Digital Fractional Order PID Controller for Buck Converter Fed DC Motor. *Circuits Syst. Signal Process.* **2016**, *35*, 2189–2211. [CrossRef]
30. Merrikh-Bayat, F.; Jamshidi, A. Comparing the Performance of Optimal PID and Optimal Fractional-Order PID Controllers Applied to the Nonlinear Boost Converter. *arXiv* **2013**, arXiv:1312.7517.
31. Amirahmadi, A.; Rafiei, M.; Tehrani, K.; Griva, G.; Bartarseh, I. Optimum Design of Integer and Fractional-Order PID Controllers for Boost Converter Using SPEA Look-up Tables. *J. Power Electron.* **2015**, *15*, 160–176. [CrossRef]
32. Kulkarni, A.J.; Durugkar, I.P.; Kumar, M. Cohort Intelligence: A Self Supervised Learning Behavior. In Proceedings of the Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK, 13–16 October 2013; IEEE Computer Society: Washington, DC, USA, 2013; pp. 1396–1400. [CrossRef]
33. Kulkarni, A.J.; Baki, M.; Chaouch, B.A. Application of the cohort-intelligence optimization method to three selected combinatorial optimization problems. *Eur. J. Oper. Res.* **2016**, *250*, 427–447. [CrossRef]
34. Kulkarni, A.J.; Shabir, H. Solving 0–1 Knapsack Problem using Cohort Intelligence Algorithm. *Int. J. Mach. Learn. Cybern.* **2016**, *7*, 427–441. [CrossRef]
35. Kulkarni, O.; Kulkarni, N.; Kulkarni, A.; Kakandikar, G. Constrained Cohort Intelligence using Static and Dynamic Penalty Function Approach for Mechanical Components Design. *Int. J. Parallel Emergent Distrib. Syst.* **2016**, *33*, 570–588. [CrossRef]
36. Bhambhani, A.; Shah, P. PID parameter optimization using Cohort intelligence technique for D.C motor control system. In Proceedings of the 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), Pune, India, 9–10 September 2016; pp. 465–468. [CrossRef]
37. Rashid, M. *Power Electronics: Circuits, Devices, and Applications*; Pearson: London, UK, 2009.
38. AND9135/D LC Selection Guide for the DC-DC Synchronous Buck Converter. ON Semiconductor. 2013. Available online: <https://www.onsemi.com/pub/Collateral/AND9135-D.PDF> (accessed on 4 August 2021).
39. Ejury, J. Buck Converter Design. Infineon Technologies North America. 2013. Available online: <https://www.mouser.de/pdfdocs/BuckConverterDesignNote.pdf> (accessed on 4 August 2021).
40. Caponetto, R.; Dongola, G.; Fortuna, L.; Petráš, I. *Fractional Order Systems*; World Scientific: Singapore, 2010. [CrossRef]
41. Loverro, A. *Fractional Calculus: History, Definitions and Applications for the Engineer*; University of Notre Dame: Notre Dame, IN, USA, 2004.
42. Miller, K.; Ross, B. *An Introduction to the Fractional Calculus and Fractional Differential Equations*; John Wiley and Sons, Inc.: Reading, MA, USA, 1993.
43. Vinagre, B.; Podlubny, I.; Hernández, A.; Feliu, V. Some approximations of fractional order operators used in control theory. *Fract. Calc. Appl. Anal.* **2000**, *3*, 231–248.
44. Valério, D.; da Costa, J. *An Introduction to Fractional Control*; The Institution of Engineering and Technology: London, UK, 2012; pp. 1–358. [CrossRef]
45. Visioli, A. Research trends for PID controllers. *Acta Polytech.* **2012**, *52*, 144–150. [CrossRef]
46. Faieghi, M.; Nemati, A. On Fractional-Order PID Design. In *Applications of MATLAB in Science and Engineering*; IntechOpen: London, UK, 2011. [CrossRef]
47. Kulkarni, A.J.; Krishnasamy, G.; Abraham, A. *Cohort Intelligence: A Socio-Inspired Optimization Method*; Springer: Berlin/Heidelberg, Germany, 2017. [CrossRef]
48. Patankar, N.S.; Kulkarni, A.J. Variations of Cohort Intelligence. *Soft Comput.* **2018**, *22*, 1731–1747. [CrossRef]
49. Tavazoei, M.S. Notes on integral performance indices in fractional-order control systems. *J. Process. Control* **2010**, *20*, 285–291. [CrossRef]
50. Duarte-Mermoud, M.; Prieto, R. Performance index for quality response of dynamical systems. *ISA Trans.* **2004**, *43*, 133–151. [CrossRef]
51. Richard, C.; Bishop, R.H. *Modern Control Systems*; Pearson Prentice Hall, Inc.: Upper Saddle River, NJ, USA, 2008.
52. Shah, P.; Agashe, S.; Kulkarni, A. Design of fractional PID controller using the cohort intelligence method. *Front. Inf. Technol. Electron. Eng.* **2017**, *19*, 437–445. [CrossRef]
53. Fraga-Gonzalez, L.; Fuentes, R.; García-González, A.; Sanchez-Ante, G. Adaptive simulated annealing for tuning PID controllers. *AI Commun.* **2017**, *30*, 347–362. [CrossRef]
54. GirirajKumar, S.M.; Bodla, R.; Narayanan, A. Design of Controller using Simulated Annealing for a Real Time Process. *Int. J. Comput. Appl.* **2010**, *6*, 20–25. [CrossRef]