Chapter 2
An Optimum Setting of PID Controller for Boost Converter Using Bacterial Foraging Optimization Technique

P. Siva Subramanian and R. Kayalvizhi

Abstract In this paper, a maiden attempt is made to examine and highlight the effective application of bacterial foraging (BF) algorithm to optimize the PID controller parameters for boost converter and to compare its performance to establish its superiority over other methods. The proposed BF-PID controller maintains the output voltage constant irrespective of line and load disturbances than particle swarm optimization (PSO)-based PID controller and conventional PID controllers.

Keywords PID controller · Boost converter · State space modeling · Bacterial foraging algorithm

2.1 Introduction

The main target of power electronics is to convert electrical energy from one form to another. To make the electrical energy to reach the load with the highest efficiency is the target to be achieved. Power electronics also targets to reduce the size of the device which aims to reduce cost, size and high availability. In this project, the power electronic device is DC–DC boost converter. Sometimes, it is necessary to increases dc voltage. Boost converter is a DC–DC converter in which the output voltage is always greater than the input voltage which depends on switching frequency [1]. From the energy point of view. From the energy point of view, output

P. Siva Subramanian (✉)
Department of Electronics and Instrumentation Engineering, SSN Institue of Engineering and Technology, Dindigul, Tamil Nadu, India
e-mail: sivasubramanian.v.p@gmail.com

R. Kayalvizhi
Department of Electronics and Instrumentation Engineering, Annamalai University, Chidambaram, Tamil Nadu, India
e-mail: mithuvigknr@gmail.com

V. Vijay et al. (eds.), *Systems Thinking Approach for Social Problems*,
Lecture Notes in Electrical Engineering 327,
DOI 10.1007/978-81-322-2141-8_2

voltage regulation in the DC–DC converter is achieved by constantly adjusting the amount of energy absorbed from the source and that injected into the load. These two basic processes of energy absorption and injection constitute a switching cycle [2]. Some control methods have stated the issue of control through pole placement [3]. Another is the design of boost converter incorporated with PID controller that is used to control the behaviors of the system in linear. This system is a closed-loop system with feedback. A proportional integral derivative controller (PID controller) is a generic control loop feedback mechanism widely used in industrial control systems. A PID controller attempts to correct the error between a measured process variable and a desired set point by calculating and then outputting a corrective action that can adjust the process accordingly. The PID controller tuning involves the calculation of three separate parameters: the proportional, the integral, and the derivative values.

The values of these parameters to control a process depend on the process dynamics and the required response of the process. The adjustment of the controller parameters to achieve satisfactory control is called controller tuning. In many process industries, the process dynamics are poorly known. In such situation, the process model is obtained through the experimental data. The required response of the process is determined based on some performance criteria such as settling time, peak amplitude, peak time, ISE, and IAE. The PID controller is simple but cannot always effectively control systems with changing parameters and may need frequent online retuning. Most of the articles have concentrated on designing PI and PID controllers [4]. In order to obtain better response, a bacterial foraging (BF) algorithm-based PID controller is designed.

BF algorithm proposed by Passino [5] is a newcomer to the family of nature-inspired optimization algorithms. Application of group foraging strategy of a swarm of *Escherichia coli* bacteria to multioptimal function optimization is the key idea of the new algorithm. Bacteria search for nutrients in a manner to maximize energy obtained per unit time. Individual bacterium also communicates with others by sending signals. A bacterium takes foraging decisions after considering two previous factors. The process, in which a bacterium moves by taking small steps while searching for nutrients, is called chemotaxis, and key idea of BF algorithm is mimicking chemotactic movement of virtual bacteria in the problem search space. It is used as optimization method and has shown its effectiveness in various problems. BF algorithm is a powerful search tool that can reduce the time and effort involved in designing systems for which no systematic design procedure exists [6]. They can quickly find close-to-optimal solutions. They are certainly useful tools when trying to solve analytically difficult problems.

In this paper, BF optimization algorithm is developed for tuning the parameters of PID controller. The developed controller is simulated for a DC–DC boost converter and to compare the response of optimized PID controller with the conventional PID controller, particle swarm optimization (PSO)-based PID controller [7]. Simulation results indicate that BF-PID controller guarantees the good performance under various line and load disturbance conditions than others.

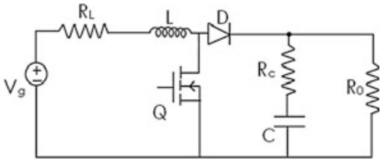

Fig. 2.1 Circuit diagram of boost converter

2.2 Boost Converter

Consider the DC–DC boost converter circuit shown in Fig. 2.1. During the interval, when switch Q is off, diode D conducts the current i_L of inductor L toward the capacitor C_0 and the load R_0. During the interval, when switch Q is on, diode D opens and the capacitor C_0 discharges through the load R. The converter transfers the energy between input and output by using the inductor.

The transfer function in Fig. 2.1 is derived using the standard state space averaging technique. In this approach, the circuits for two modes of operation (ON mode and OFF mode) for the converter are modeled as follows:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{2.1}$$

where

x state variable
u input voltage (V_{in})
y output voltage (V_o)

After modeling, the two modes are averaged over a single switching period T.

2.3 PID Controller

The PID controller shown in Fig. 2.2 is used to improve the dynamic response and to reduce the steady-state error. The derivative controller improves the transient response, and the integral controller will reduce steady-state error of the system.

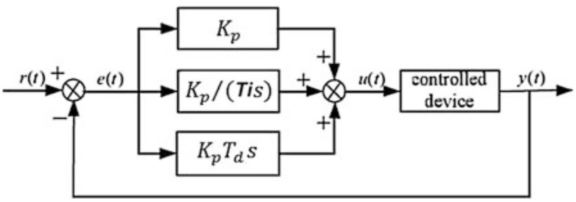

Fig. 2.2 Schematic diagram of PID controller

The transfer function of the PID controller is given as follows:

$$k_p + \frac{k_i}{s} + k_d s = \frac{k_d s^2 + k_p s + k_i}{s} \tag{2.2}$$

The PID controller works in a closed-loop system. The signal $u(t)$ output of the controller is equal to the K_p times of the magnitude of the error plus K_i times the integral of the error plus K_d times the derivative of the error as follows:

$$k_p e + k_i \int e\mathrm{d}t + k_d \frac{\mathrm{d}e}{\mathrm{d}t} \tag{2.3}$$

This control signal will be then sent to the plant, and the new output $y(t)$ will be obtained. This new output will be then sent back to the sensor again to find the new error signal $e(t)$. The controller takes this new error as input signal and computes the gain values (K_p, K_i, K_d).

2.4 Bacterial Foraging Optimization Technique

BF algorithm is a new division of bioinspired algorithm. This technique is developed by inspiring the foraging behavior of *E. coli* bacteria. In the BF optimization process, four motile behaviors of *E. coli* bacteria are mimicked.

2.4.1 Chemotaxis

During the foraging operation, an *E. coli* bacterium moves toward the food location with the aid of swimming and tumbling via flagella. Depending upon the rotation of flagella in each bacterium, it decides whether it should move in a specified direction (swimming) or altogether in modified directions (tumbling), in the entire lifetime. To represent a tumble, a unit length random direction, say $\Phi(j)$, is generated; this will be used to define the direction of movement after a tumble. In particular,

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i) * \Phi(j) \tag{2.4}$$

where $\theta^i(j,k,l)$ represents the ith bacterium, at jth chemotactic, kth reproductive, and lth elimination and dispersal step. $C(i)$ is the size of the step taken in the random direction specified by the tumble.

2.4.2 Swarming

In this process, after finding the direction of the best food position, the bacterium will attempt to communicate with other by using an attraction signal. The signal communication between cells in *E. coli* bacteria is represented by

$$J_{cc}(\theta, D(j,k,l)) = J^{i}_{cc}(\theta, \theta^{i}(j,k,l)) = X + Y \quad (2.5)$$

where

$$X = \sum_{i=1}^{S} [-D_{\text{attract}} * \ \exp(- W_{\text{attract}} * \sum_{m=0}^{P} (\theta m - \theta^{i} m))^{2}]$$

$$Y = \sum_{i=1}^{S} [H_{\text{repellant}} * \ \exp(- W_{\text{repellant}} * \sum_{m=0}^{P} (\theta m - \theta^{i} m))^{2}]$$

where $J_{cc}(\theta, D(j,k,l))$ is the cost function value to be added to the actual cost function to be minimized to present a time-varying cost function, S is the total number of bacteria, P is the number of parameters to be optimized which are present in each bacterium, and D_{attract}, W_{attract}, $H_{\text{repellant}}$, and $W_{\text{repellent}}$ are different coefficients that should be chosen properly.

2.4.3 Reproduction

The least healthy bacteria die, while each of the healthier bacteria (those yielding lower value cost function) asexually splits into two bacteria, which are then placed in the same location. This keeps the swarm size constant.

2.4.4 Elimination and Dispersal

It is possible that in the local environment, the life of bacteria changes either gradually or suddenly due to some other influences. Events can occur such that all the bacteria in a region are killed or a group is dispersed into a new environment. They have the effect of possibly destroying the chemotactic progress, but they also have the effect of assisting in chemotaxis, since dispersal may place bacteria near attractive food sources. From a broader perspective, elimination and dispersal are part of the population-level long-distance motile behavior.

The searching procedure to develop BF-PID controller is as follows:

Step 1 *Initialization*

- D Number of parameters to be optimized. In this project, it is K_P, K_I, and K_D.
- S Number of bacteria to be used for searching the total region.
- N_S Swimming length after which tumbling of bacteria will be done in a chemotactic step.
- N_c Number of iterations to be taken in the chemotactic step.
- N_{re} Maximum number of reproductions to be undertaken.
- N_{ed} Maximum number of elimination and dispersal events to be imposed over bacteria.
- P_{ed} Probability with which the elimination–dispersal will continue.
- θ Location of the each bacterium which is specified by random numbers on [0,1].
- $C(i)$ Chemotactic step size assumed to be constant for our design.

The value of $D_{attract}$, $W_{attract}$, $H_{repellant}$, and $W_{repellent}$ is to denote here that the value of $D_{attract}$ and $H_{repellant}$ should be the same so that the penalty imposed on the cost function through "J_{cc}" will be "0" when all the bacteria will have the same value, i.e., they have converged.

Step 2 *Elimination and dispersal loop*

$l = l + 1$

Step 3 *Reproduction loop*

$k = k + 1$

Step 4 *Chemotactic loop*

$j = j + 1$

(i) For i = 1, 2, 3 … S, take a chemotactic step for each bacterium i as follows:

(ii) Compute the value of $J(i, j, k, l)$.
Let $J(i,j,k,l) = J(i,j,k,l) + J_{cc}(\theta^i(j,k,l), P(j,k,l))$ (i.e., add on the cell to cell attractant effect to the nutrient concentration).

(iii) Let $J_{Last} = J(i, j, k, l)$ to save this value since we may find a better cost via run.
End of for loop.

(iv) For i = 1, 2, 3 … S, take the tumbling/swimming decision
Tumble: Generate a random number vector R^p with each element $m(i)$, (m = 1, 2, 3 … D), a random number on the interval [−1,1].

(v) **Move**: Let

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta T(i) \times \Delta(i)}}$$

This results in a step of size $C(i)$ in the direction of the tumble for bacterium i.

(vi) Compute = $J(i, j, k, l)$
Let $J(i,j,k,l) = J(i,j,k,l) + J_{cc}(\theta^i(j,k,l), P(j,k,l))$

(vii) **Swim**: Note that we use an approximation since we decide swimming behavior of each cell as if the bacteria numbered {1, 2, 3 … i} have moved and {i + 1, i + 2, i + 3 … S} have not; this is simpler to simulate than simultaneous decision about swimming and tumbling by all the bacteria at the same time.
Let m = 0 (counter for swim length)
While m < Ns (if doing better), let
$J_{\text{Last}} = J(i, j+1, k, l)$ and
let $\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta T(i) \times \Delta(i)}}$
And use this $\theta^i(j+1, k, l)$ to compute the new $J(i, j+1, k, l)$ as in step (vi).
Else, let m = Ns, and this is the end of the while statement.

(viii) Go to next bacterium (i + 1) if 'i' is not equal to S (i.e., go to (b)) to process the next bacterium.

Step 5 If $j < N_c$, go to Step 4. In this case, continue chemotaxis since the life of the bacteria is over.

Step 6 *Reproduction*

(i) For the given value of k and l, and for each i = 1, 2, 3 … S. Let $J^i_{\text{Health}} = \sum J(i, j, k, l)$ be the health of bacterium. Sort bacteria on chemotactic parameters $C(i)$ in order of interesting cost J_{Health} (higher cost means lower health).

(ii) The Sr = S/2 bacteria with the highest J_{Health} values die and other Sr bacteria with the best value split and the copies that are made are placed at the same location as their parent.

Step 7 If $k < N_{\text{re}}$, go to Step 2; in this case, we have not reached the number of specified reproduction steps, so we start the next generation in the next chemotactic step.

Step 8 *Elimination and Dispersal*: For i = 1, 2, 3 … S, with probability P_{ed}, eliminate and disperse each bacterium (keeps the bacterium population constant). To do this, if we eliminate a bacterium, simply disperse one into a random location in the optimization domain.

Step 9 If $l < N_{\text{ed}}$, then go to Step 1; otherwise, end.

2.5 Performance Indices for BF Algorithm

The objective function considered is based on the error criterion. The performance of a controller is best evaluated in terms of error criterion. In this work, controller's performance is evaluated in terms of integral square error (ISE).

$$\text{ISE} = \int_0^t e^2 \mathrm{d}t \tag{2.6}$$

Table 2.1 Circuit parameters of the boost converter

Parameter	Symbol	Value
Input voltage	V_{in}	12 V
Output voltage	V_o	20 V
Inductor	L	162 μH
Capacitor	C	220 μF
Internal resistors	R_L, R_C	5 mΩ
Load resistor	R	10 Ω
Duty ratio	D	0.53

The ISE weighs the error with time and hence minimizes the error values nearer to zero.

2.6 Simulation Results

The circuit parameters of the boost converter are shown in Table 2.1.

The responses of boost converter using conventional PID controller and BF-PID controllers are shown in Figs. 2.3, 2.4, 2.5 and 2.6.

The figures show that BF-PID controller will drastically reduce the overshoot and ISE and IAE values as compared to the conventional PID controller. Table 2.2

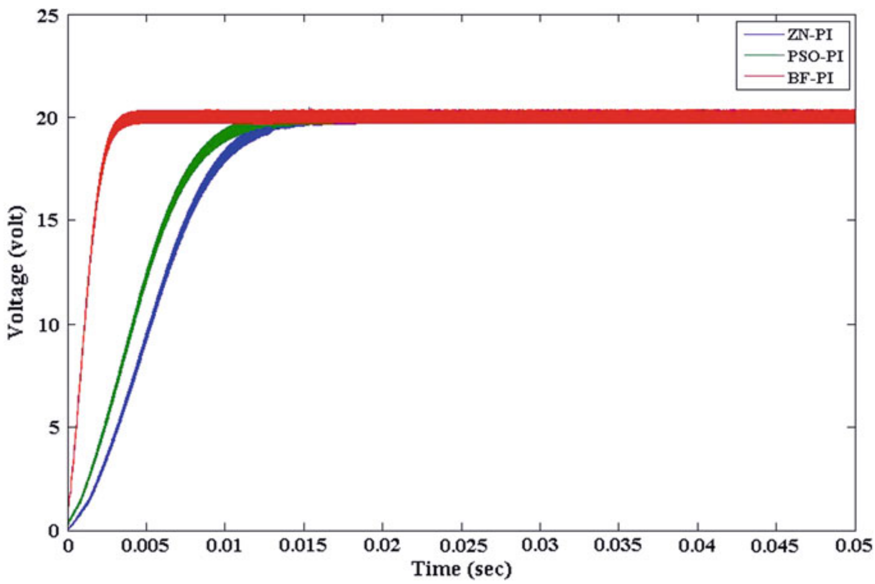

Fig. 2.3 Output voltage of conventional PID controller, PSO-based PID controller, and BF-based PID controller

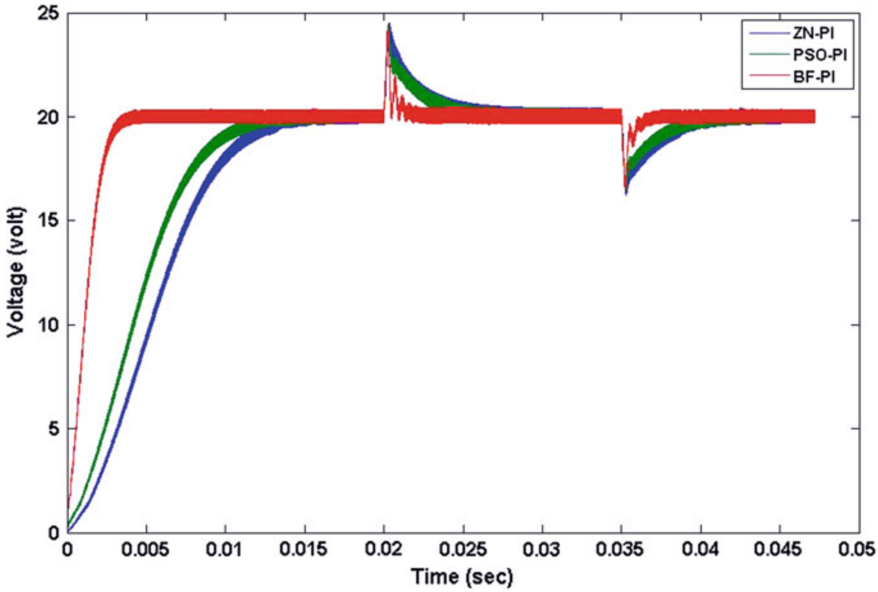

Fig. 2.4 Comparison of closed-loop response of boost converter under sudden change in line voltage of 12–15 V (25 % increase) and 12–10 V (25 % decrease)

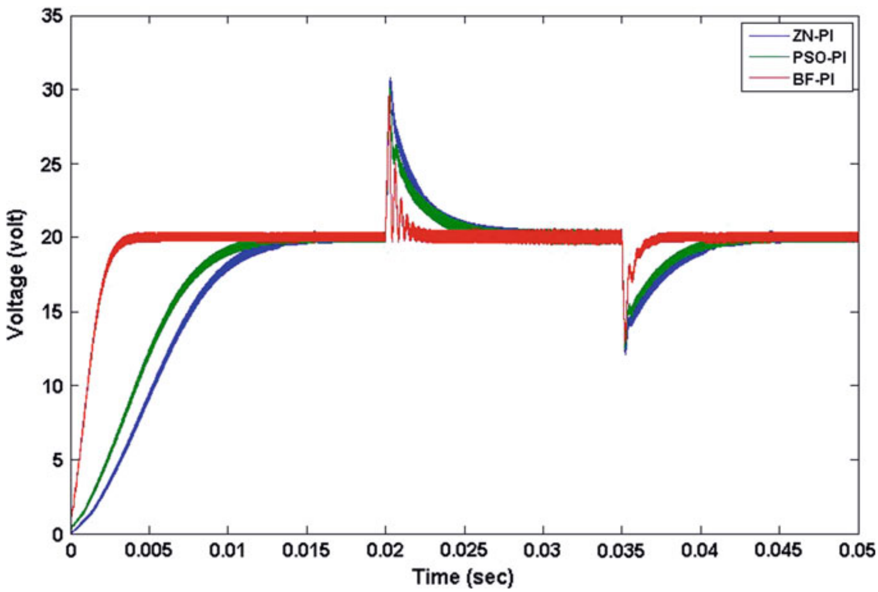

Fig. 2.5 Comparison of closed-loop response of boost converter under sudden change in load disturbance from 10 to 12 Ω (25 % increase) and 10 to 8 Ω (25 % decrease)

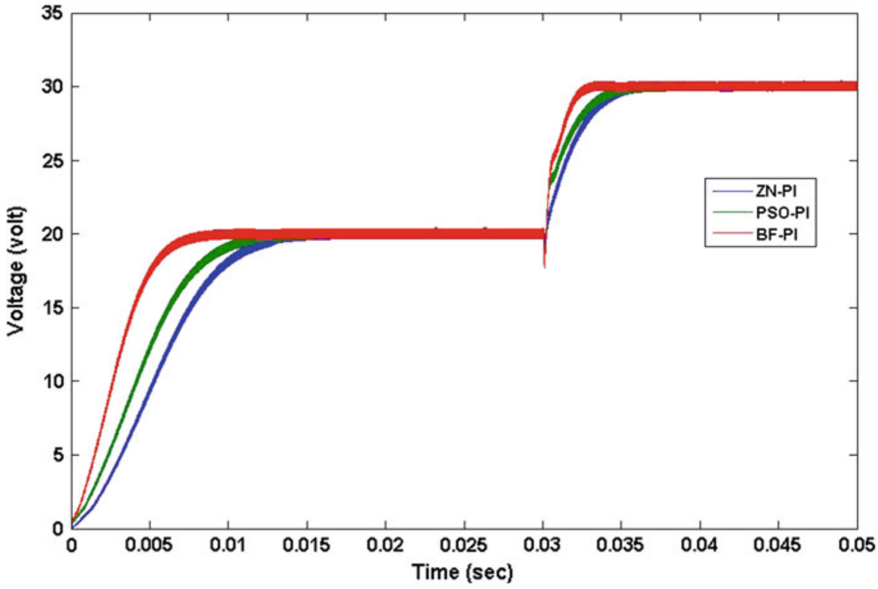

Fig. 2.6 Comparison of servo response of boost converter under sudden change in reference voltage 20–30 V

Table 2.2 Performance analysis of the boost converter

Parameters	Conventional PID controller	PSO-PID controller	BF-PID controller
Peak amplitude (V)	20	20	20
Overshoot (%)	0	0	0
Rise time (ms)	2.3	1.4	0.5
Settling time (ms)	18	13	5
ISE	1.52	1.17	0.35

shows the performance analysis of the boost converter using conventional PID controller, PSO-PID controller, and BF-PID controller.

2.7 Conclusion

In this work, BF algorithm is developed to tune the PID controller parameters that control the performance of DC–DC boost converter. The simulation results confirm that PID controller tuned with BF algorithm rejects satisfactorily both the line and load disturbances. Also, the result proved that BF-based PID controller gives the smooth response for the reference tracking and maintains the output voltage of the boost converter according to the desired voltage.

References

1. Khanchandani KB, Singh MD (2005) Power electronics. Tata McGraw Hill, New Delhi
2. Hasaneen BM, Mohammed AAE (2008) Design and simulation of DC/DC boost converter. 12th international middle-east power system conference, pp 335–340
3. Kelly A, Rinne K (2005) Control of DC–DC converters by direct pole placement and adaptive feed forward gain adjustment. In: Applied power electronics conference and exposition (APEC), vol 3, pp 1970–1975
4. Iruthayarajan MW, Baskar S (2007) Optimization of PID parameters using genetic algorithm and particle swarm optimization. In: International conference on information and communication technology in electrical sciences, pp 81–86
5. Passino KM (2002) Biomimicry of bacterial foraging for distributed optimization and control. IEEE Control syst Mag, pp 52–67
6. Mary Synthia Regis Prabha DM, Pushpa Kumar S, Glan Devdhas G (2011) An optimum setting of controller for a DC–DC converter using bacterial intelligence technique. In: IEEE PES innovative smart grid technologies, New Jersey
7. Suresh Kumar R, Suganthi J (2012) Improving the boost converter PID controller performance using particle swarm optimization. Eur J Sci Res 85(3):327–335. ISSN 1450-216X

http://www.springer.com/978-81-322-2140-1

Systems Thinking Approach for Social Problems
Proceedings of 37th National Systems Conference, December 2013

Vijay, V.; Yadav, S.K.; Adhikari, B.; Seshadri, H.; Fulwani, D.K. (Eds.)

2015, XVI, 430 p. 229 illus., Hardcover
ISBN: 978-81-322-2140-1