International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 3 (2015) pp. 6883-6900

http://www.ripublication.com

Design Of Fractional Order PI Controller For DC-DC Boost Converter

Rajamani SENTHILKUMAR[1] and Venugopal MANIKANDAN[2]

1. Research Scholar,
Faculty of Electrical Engineering, Anna University, Chennai, INDIA,
rsenthilkumarpe@gmail.com
2. Professor, Department of Electrical and Electronics Engineering,
Coimbatore Institute of Technology, Coimbatore, Tamil Nadu, INDIA,
manikandan@cit.edu.in

Abstract

Fractional order controller is used in most areas of science and engineering, being acknowledged its ability to give up a superior control in many dynamical systems. This work proposes the applications of a Fractional Order Proportional and Integral (FOPI) controller in the area of Power Electronics for a DC-DC power converter with soft computing techniques. Genetic Algorithm and Particle Swarm Optimization techniques are used for designing a Fractional order PI controller to analysis the performances of dc to dc boost converter, and it is simulated in MATLAB environment. The objective function is written to find the optimum value for the FOPI controller parameters by minimizing a given integral square error (ISE) as an objective function which can obtain most of the systems requirements such as reducing the overshoot, keeping a fast system response, attain a good load disturbances rejection and maintaining robustness. The results show that this PSO-FOPI design method may effectively tune the parameter of the Fractional order controller as compared to GA.

Keywords: Boost converter, genetic algorithm, PSO, Fractional Order PI controller.

1. INTRODUCTION:

The boost converter is capable of producing a regulated output voltage having a magnitude greater than the source voltage. Power for the boost converter can be obtained from any suitable DC sources such as battery sources, fuel cells and photo voltaic solar systems. When the boost converter is operates in open loop mode, it shows poor voltage regulation and dynamic response. Hence, the closed loop control provides for better voltage regulation. The control strategy lies in the operation of the duty cycle of the switch that causes the voltage change. The boost converter operated in two modes, where the mode of operation is changed from ON state to the OFF state of the power switch. Small signal linearization techniques have mostly been employed for controller design. However, boost converter is a non-minimum phase system due to the presence of a right half plane zero which causes poor dynamic response [1] and confine the available bandwidth for the stable operation of the boost converter [2]. There are many methods for controller design based on the linearized small signal model of a boost converter, and its implementation is [3] - [6]. In order to deal with this problem and improve the dynamic response of DC-DC converters, a Fractional Order PI controller has been proposed based on soft computing techniques.

A Fractional Order PI controller has been applied to regulate the constant output DC voltage of a Boost converter within a specified tolerance limit. A very significant aspect of designing FOPI controllers is to choose the values of K_P, K_i and λ. The Genetic Algorithm and Particle Swarm Optimization Algorithms are used for tuning the FOPI controller. Recently, the importance of fractional order dynamic systems and controllers has been increasing in many areas of science and engineering. Controllers consisting of fractional order derivatives and integrals have been used in industrial applications such as power electronics [7], system identification [8], robotic manipulators [9], etc., It should be noted that there are a growing number of physical systems whose behavior can be compactly determined using the fractional order system theory and can be controlled with Fractional Order Proportional-Integral (FOPI) controller [10].

The present work proposes a design of the fractional order PI controller for a boost converter using optimization technique. The contribution of this work consists mainly in the design of feedback fractional order PI controller in order to get optimum time domain specifications in which integral square error (ISE) has been minimized and hence the controller parameters K_P, K_i, and λ are identified. It can be achieved through GA and PSO algorithm. The Simulink model of a boost converter works along with the evolutionary algorithm to evaluate the objective function which results with robust feedback fractional order PI controller. The development and implementation of the proposed system and controller are done using MATLAB/Simulink.

2. MATHEMATICAL MODEL OF BOOST CONVERTER

The structure of the conventional boost converter shown in Fig.3, where SW is a switch, D output diode and R, L, C are Load resistor, input inductor, and output capacitor respectively. By state space averaging method, the state averaging equations (1) (2) (3) describes the boost converter behavior is

$$\left.\begin{aligned} \frac{dX}{dt} &= AX + Bv_{in} \\ v_o &= cX \end{aligned}\right\} \tag{1}$$

Where

$$X = \begin{bmatrix} i_L \\ v_C \end{bmatrix} \tag{2}$$

The derived state equation is

$$\left.\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} -\dfrac{1}{L}\left(r_L + \dfrac{(1-D) R r_C}{R_L + r_C}\right) & -\dfrac{(1-D) R_L}{L(R_L + r_C)} \\ \dfrac{(1-D) R_L}{C(R_L + r_C)} & \dfrac{-1}{C(R_L + r_C)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \dfrac{1}{L} \\ 0 \end{bmatrix} v_{in} \\ v_O &= \begin{bmatrix} \dfrac{(1-D) R_L r_C}{R_L + r_C} & \dfrac{R_L}{R_L + r_C} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned}\right\} \tag{3}$$

Where, the state variables x_1 and x_2 are inductor current i_L and capacitor voltage v_c respectively.

The transfer function of the boost converter is given by

$$G_p(s) = \frac{(R_L + r_C)\left((sCr_C + 1)\left\{-(sL + r_L)(R_L + r_C) + (1-D)^2 R^2\right\}\right)}{P(S)\left[(1-D) R_L\left((1-D) R_L + r_C\right) + r_L (R_L + r_C)\right]} R_L v_{in} \tag{4}$$

Where

$$\begin{aligned} P(s) &= s^2 LC (R_L + r_C)^2 + s\left\{L(R_L + r_C) + r_L C(R_L + r_C)^2 + (1-D) R_L r_C C(R_L + r_C)\right\} \\ &+ r_L (R_L + r_C) + (1-D) R_L \left((1-D) R_L + r_C\right) \end{aligned} \tag{5}$$

The above state equation is modeled in MATLAB tool and gets the dynamic characteristic of boost converter.

3. FRACTIONAL ORDER PI$^{\lambda}$D$^{\delta}$ CONTROLLER DESIGN:

The most commonly used controllers in the industries are PID controllers for many years due to its simplicity of design and having better dynamic characteristics. The widespread use of these PID controllers has motivated many researchers to look for better alternative controllers. Hence, fractional order controller for the dynamic system has been introduced, and the performance of CRONE controller is demonstrated by Oustaloup.

In order to improve the performance of PID controller podlubny has proposed a generalization of PID controller as $PI^{\lambda}D^{\delta}$ controller which is known as fractional order PID controller, where, I and D are in general non-integer orders. The order of s in integral and derivative action is λ and δ are real numbers respectively. This new technique is verified to provide more agility and flexibility to enhance control of dynamics [11]. The controller has a less sensitive. Hence, the external disturbances do not change the system's response. Compare to conventional PID controller here we get two more adjustable parameters. These two adjustable parameters give an extra degree of freedom to the dynamic properties of fractional order systems.

The Grunewald-Letnikov, Riemann-Liouville and Caputo statements [12, 13] are the well-established definitions of fractional derivatives and integrals in fractional calculus. These definitions are given below

Grunewald-Letnikov definition:
This definition is defined as

$$ {}_aD_t^{\alpha} f(t) = \lim_{h\to 0} \frac{1}{h^{\alpha}} \sum_{j=0}^{\left[\frac{t-a}{h}\right]} (-1)^j \binom{a}{j} f(t-jh) \tag{6}$$

Where $w_j^{\alpha} = (-1)^j \binom{a}{j}$ represents the coefficients of a polynomial $(1-z)^{\alpha}$
The coefficients can also be obtained from

$$ w_0^{\alpha} = 1 \quad w_j^{\alpha} = \left(1 - \frac{\alpha+1}{j}\right) w_{j-1}^{\alpha} \quad j = 1, 2, \ldots.. \tag{7}$$

Riemann-Liouville definition:
The fractional order integration is defined by

$$ {}_aD_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_a^t (t-\tau)^{\alpha-1} f(\tau)\, d\tau \tag{8}$$

Where $0 < \alpha < 1$ and a is the initial time instance, assumed to be zero, i.e., a =0

4. PROPOSED FOPI CONTROLLER:

The FOPI controller generalizes the conventional integer order PID controller and expands from the point to a plane. This expansion gives more flexibility in PID controller design. The main advantage of FOPID is, by selecting a suitable value of λ and δ in between 0 and 1, we get the four possible configurations of the controller as shown in Fig.1.

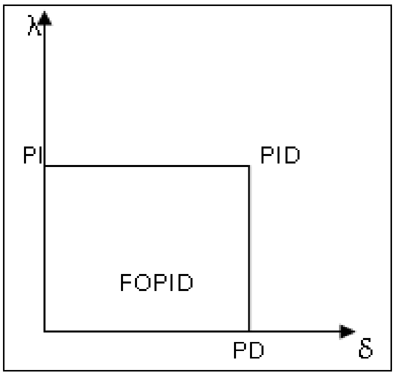

Fig.1 FOPID

This proposed fractional controller has three unknown parameters. The objective is to find optimal values of Kp, K_i and λ to get the better dynamic response of the systems. These values are found by standard Genetic Algorithm and Particle Swarm Optimization technique for obtaining the finest value.

The most common form of a fractional order PI controller is the PI^{λ} controller [14], involving an integrator of order λ where λ can be any real numbers. The differential equation of fractional order PI controller is described as

$$u(t) = k_p\, e(t) + k_i\, D_t^{-\lambda}\, e(t) \tag{9}$$

The transfer function of FOPI is obtained through Laplace transform and is given by

$$G_c(S) = k_p + k_i\, s^{-\lambda} \tag{10}$$

Where k_p, k_i are the controller gain, and λ is an integral power in a real number. The design of fractional order PI controller involves finding the optimal values of Kp, K_i, and λ.

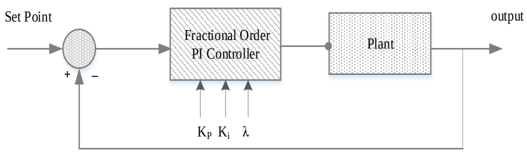

Fig.2 Block diagram of FOPI controller

It can be expected that the PI^{λ} controller shown in Fig.2 may enhance the system control performance. The most significant advantages of the PI^{λ} controller are the superior control of dynamical systems, which are described by fractional order mathematical models. The added advantage lies in the fact that the PI^{λ} controllers are less amount of sensitive to modify the parameters of a control system.

5. PROBLEM FORMULATION OF BOOST CONVERTER WITH GA AND PSO:

The output voltage of the boost converter is changed with respect to duty cycle. For this reason, the controller circuit is designed in such a way to adjust the duty cycle to follow the reference voltage.

The main aim of this work is to improve the dynamic response of DC-DC converter by selecting the appropriate values of the FOPI controller. The measurement of the boost converter considered in this paper is [15], the applying input voltage is 36V and switching frequency 2KHz.The boost converter under consideration converts 36 V to 80V.Table 1 shows the parameter of the boost converter.

Table.1 Boost converter parameters

Parameter	Value	Unit
Load Resistance, R	100	Ω
Filter inductance, L	33	mH
r_L	3	Ω
Filter Capacitance, C	1000	μF
r_c	0.5	Ω

The optimization technique is used to improve the dynamic performance of the boost converter. As a result, the following control quality criterion is used in this paper:

Minimize

$$J = \int_0^{\infty} e(t)^2 dt \tag{11}$$

Where $e(t) = \left[V_{ref} - V_0\right]$ is the error signal

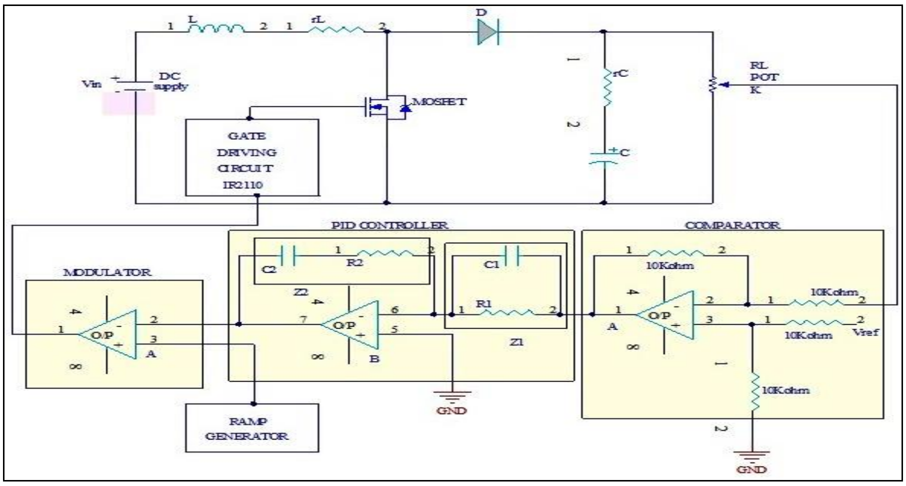

Fig.3 Boost converter topology

6. GENETIC ALGORITHM BASED TUNING OF FRACTIONAL ORDER PI CONTROLLER

A genetic algorithm is a search Metaheuristic that mimics the natural selection process. It was introduced by Holland in 1975 and belonged to the larger class of an evolutionary algorithm. GA generates solutions to optimization problems using techniques inspired by natural selection such as inheritance, mutation, selection, and crossover.

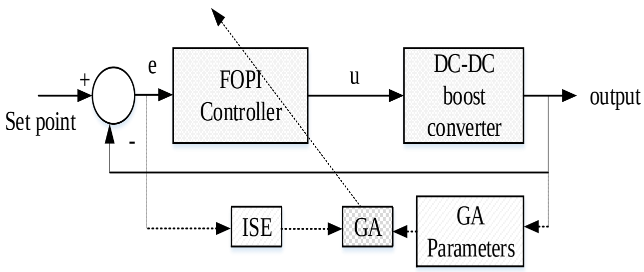

Fig.4 Structure and design of GA-FOPI controller

The primary attitude behind GA is survival of the fittest. Consequently, GA is used primarily for maximization problems in optimization. GA does not experience from the basic setback of conventional optimization technique such as getting stuck in local minima. It is because GA works on the principle of natural genetics, which include a large number of randomness.

6.1. Optimal values of Kp, K_i and λ using a genetic algorithm:

The objective is to find out the optimal set of values of Kp, Ki and λ, in fractional order PI controller. Fig.4 represents the GA based finding of optimal values of k_p, k_i and λ. A good set of values of controller parameters are measured as parents and to get the new set of values of K_p, K_i and λ, a crossover operation is performed between two good values. As a result, the first generation of offspring's is produced. The total population increases and selection process is carried out on the population. The right chromosomes are kept whereas the worst chromosomes are left out.

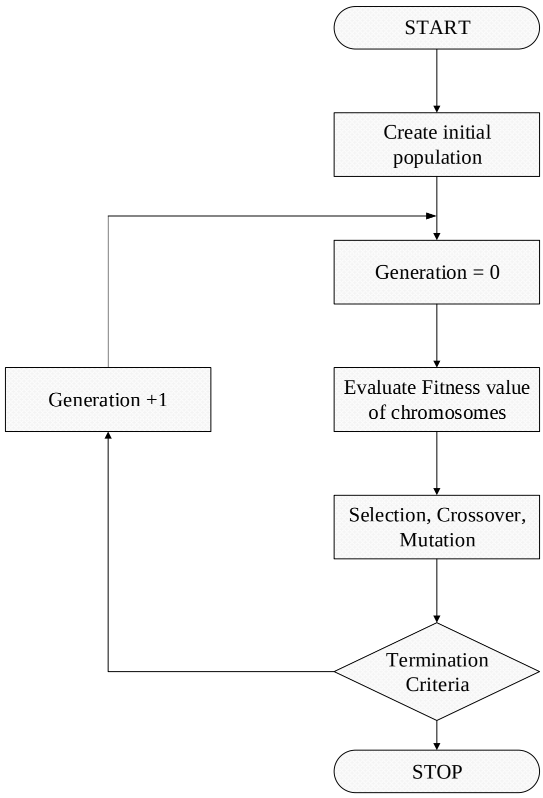

Fig.5 Flow chart of GA algorithm

The objective function ISE value of the right chromosomes is then verified to find out whether these are the best chromosomes or optimal chromosomes. If the objective function value is not satisfied, then again crossover between the best chromosomes is performed, till the objective function value, ISE comes within the limit. To obtain more optimal results, the mutation operation is performed.

The optimal values of k_p, k_i and λ are achieved by using genetic algorithms first initialize the values as follows:

Population size	: 40
Dimension of a problem	: 3
Bit length of chromosomes	: 8
Maximum number of generations	: 250
Selection scheme	: Roulette wheel

Mutation rate/probability : 0.2
Crossover rate/probability : 0.8

The following algorithm simplifies the main steps in GA.
Step-1: Initialization:
Initially, many chromosomes are randomly generated to form an initial population. This generated population allows the entire range of a feasible solution in the search space. These chromosomes represent the values of Kp, Ki and λ.
Step-2: Fitness function.
It is an objective function of problem to be optimized. Calculate the fitness value of each in the population using the objective function given by (11).
Step-3: Selection:
Individuals are selected from a population randomly. For selecting individuals, Roulette wheel method is used for improving the population itself.
Step-4: crossover and mutation:
1. Perform crossover between parents to create new offspring.
2. Apply mutation with a definite probability.
3. Calculate the fitness of each offspring.

Step-5: Termination criteria
If the current iteration is greater than or equal to the maximum iteration, the process is stopped otherwise the above steps will be repeated. The Fig.5 shows the operational flow chart of a genetic algorithm [16, 17]. This operation is carried out till, the genetic algorithm reaches the maximum number of iterations.

7. PARTICLE SWARM OPTIMIZATION BASED TUNING OF FRACTIONAL ORDER PI CONTROLLER

Particle Swarm Optimization (PSO) is a new population-based stochastic optimization technique. The PSO algorithm attempt to mimic the natural process of group communication of individual knowledge, which can be occurs when such swarms flock, migrate, and forage, etc., in order to achieve some optimum property such as configuration or location.

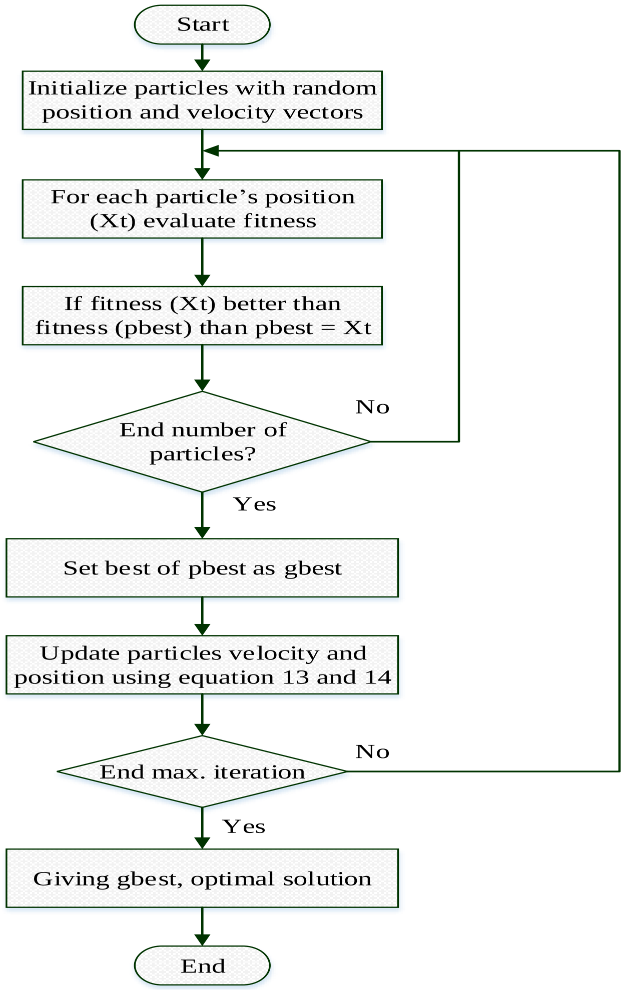

Fig.6. Flow chart for PSO algorithm

In PSO, the swarm is initialized with a population of random solutions. Each particle in the swarm is a different possible set of the unknown parameters to be optimized. Each particle is considered as a point in an N-dimensional solution space, which adjusts its flying toward a possible area according to its own flying experience, as well as other companions flying experience. Each particle keep path of its coordinates in the solution space by swarming the particles toward the best fitting (fitness) solution called *pbest*, to meet previous iterations with the objective to find better solutions [18-21] and finally converging on a single minimum error solution. The searching method of implemented PSO- FOPI controller is given in the flowchart.

Using present velocity and the distances from *pbest* and *gbest* the updated velocity of each particle can be calculated. The updated velocity, position, and inertia weight are shown.

$$W^{t} = W_{max} - Iter \times \left[\frac{W_{max} - W_{min}}{Iter_{max}} \right] \quad (12)$$

$$Vi^{t+1} = W^{t} \cdot Vi^{t} + C1 \cdot R1 \cdot \left(Pi^{t} - Si^{t} \right) + C2 \cdot R2 \cdot \left(Gi^{t} - Si^{t} \right) \quad (13)$$

$$Si^{t+1} = Si^{t} + Vi^{t+1} \quad (14)$$

Where $c1$ and $c2$ are two positive constants, called cognitive learning rate and social learning rate respectively.1, 2…t_{max} indicates the iterations, Vi^{t+1} stands for velocity of I^{th} particle, R_1, R_2 is a two random functions in the range [0, 1]. Equation (10) shows how the inertia weight is updated. The velocities of the particles are limited in [*Wmin, Wmax*].

Steps in PSO Based FOPI Controller Optimization

Step 1: Initialize the swarm from the solution space.
Initialize a swarm of particles with random velocity and position in the solution space.
Step 2: Evaluate the objective function of every particle.

Evaluate the fitness value of each particle in the population. The advantage of each particle in the swarm is found by using an objective function called evaluation function. This function is defined to minimize the cost function given by (11).

Step 3: Compare the fitness of particle with *pbest* and if the current value is better than *pbest* replace the local best value.
Step 4: updating: Modify the current velocity and position of the particle as per (13) and (14).
Step 6: The above steps are repeated until the number of iterations reached. Record the optimized k_p , *ki and* λ.
Step 7: perform closed loop test with optimized values of controller parameters and calculate time domain specification for the system.

7.1 OPTIMIZATION OF FOPI CONTROLLERS:

Optimizations of FOPI controllers initially decided the optimization goal, and then encode the parameters to be searched. The solution space is a three-dimensional, being k_p , *ki and* λ. So each particle has three-dimensional positions and velocity vector.PSO algorithm is running until the stop condition is satisfied. The particles of the last generation are the optimized parameters of the FOPI controller. At the end of the iterations, the best position of the swarm will be the solution of the problem. It cannot be always possible to get the optimum result of the problem, but the obtained solution will be an optimal one [18-21]. The flowchart for implemented PSO-FOPI controller is shown in Fig.6.

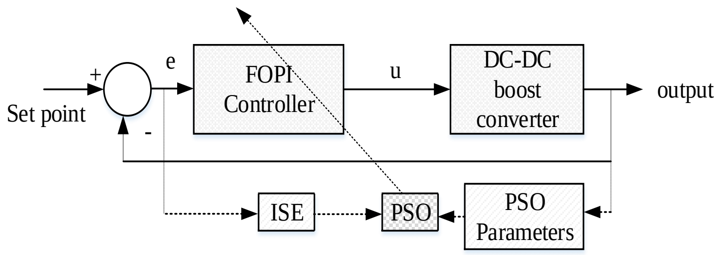

Fig.7. Structure and design of PSO-FOPI controller

The PSO algorithm method has been implemented as M-file that interconnected to Simulink model, where the FOPI controller parameters are computed and feed to the GUI of the controller. The optimization performed with this initial parameter, number of particles 20, a maximum iteration 50, number of dimensions 3, with the objective function ISE. The PSO method will generate initial values of three parameters K_p, K_i and λ of the fractional order PI controller. The objective function is then computed for each value of controller parameters. At the end of iteration, the three parameters of the fractional order PI controller K_p, K_i and λ has been obtained directly according to the minimum value of the objective function ISE. The structure of PSO-FOPI controller as shown in Fig.7

8. SIMULATION RESULTS AND DISCUSSION:

The value of k_p, k_i and λ is applied to the Fractional Order PI Controller of the DC-DC boost converter, is to control the output voltage. The stability of the converter system is verified by means of Genetic Algorithm and Particle Swarm Optimization algorithms. Fig.8 shows the nominal response of the proposed system with a step change of 80 V. Figs.9 & 10 show the voltage and current response of DC Boost converter for load and supply disturbances for the reference voltage of 80V. Fig 11 shows the voltage and current response of DC Boost converter for decrement in set point from 80V to 40V at time period t= 1 sec and increment in set point at from 40V to 80V at time period t=2 sec. Fig.12 Shows the voltage and current response of DC Boost converter for disturbance in feedback at time t=1 sec 10 % decrement and time t=2 sec 10% increase. Table 2 shows the obtained fractional order PI controller parameter using GA and PSO. Table 3 shows the performance of GA and PSO tuned DC Boost converter. From Table 3, it is obvious that the rise time, settling time, and steady state error of the proposed PSO tuned FOPI system is superior to that of GA tuned FOPI controller for Boost converter application.

From the dynamic response of (Figs.9-12) proposed PSO tuned FOPI controller has quicker disturbance rejection and good dynamic stability.

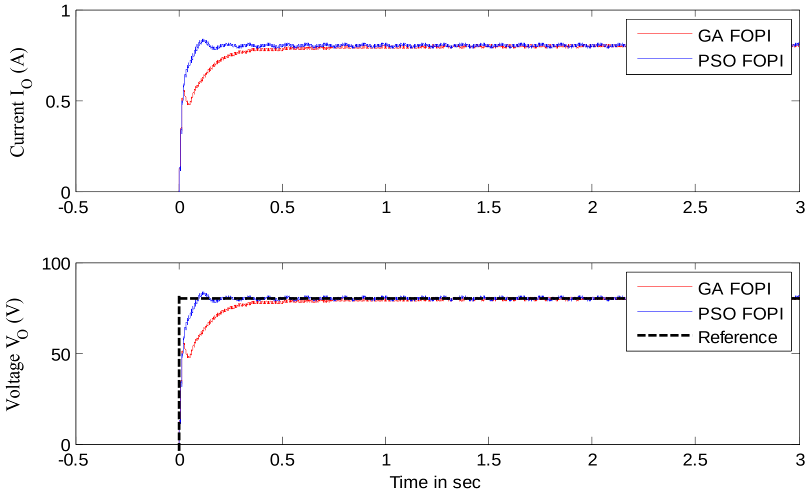

Fig.8. Simulated voltage and current response of GA & PSO tuned FOPI controlled DC-Boost converter for a set value of 80 V

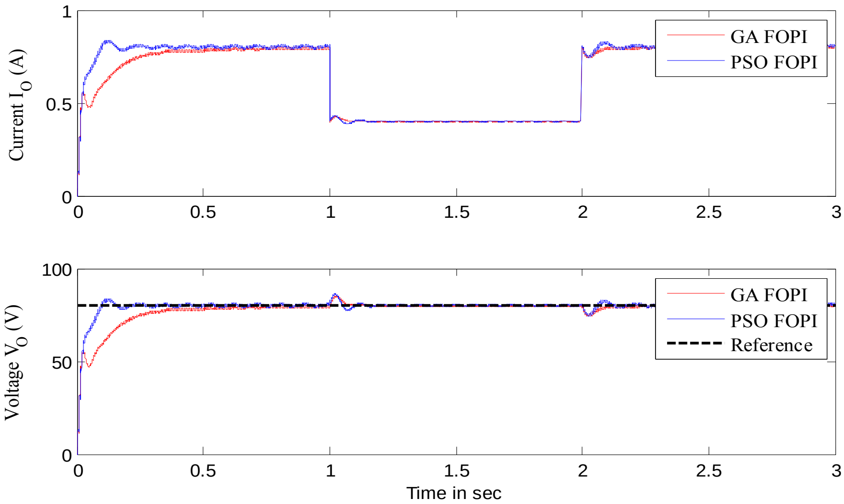

Fig.9. Simulated voltage and current response of GA & PSO tuned FOPI controlled DC-Boost converter for a change in load from 100 Ω to 200 Ω at time t=1 Sec and 200 Ω to 100 Ω at time t=2 Sec.

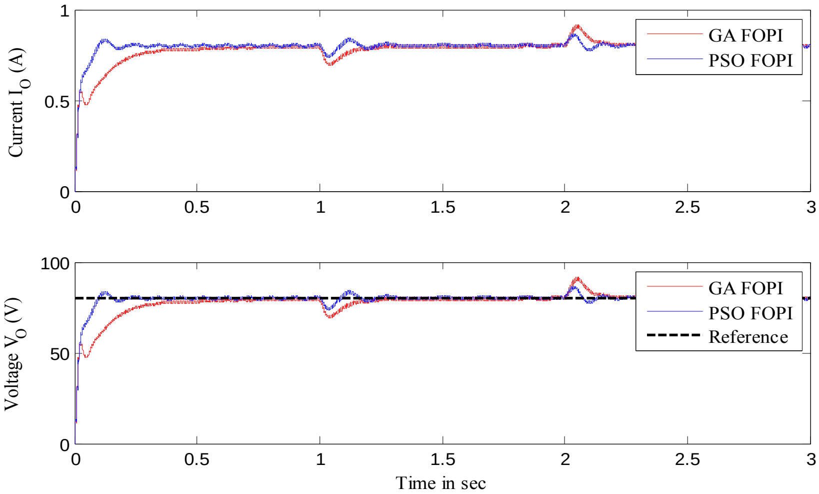

Fig.10. Simulated voltage and current response of GA & PSO tuned FOPI controlled DC-Boost converter for 10% decrement and increment in supply/input voltage at time t=1 sec and t=2 sec respectively.

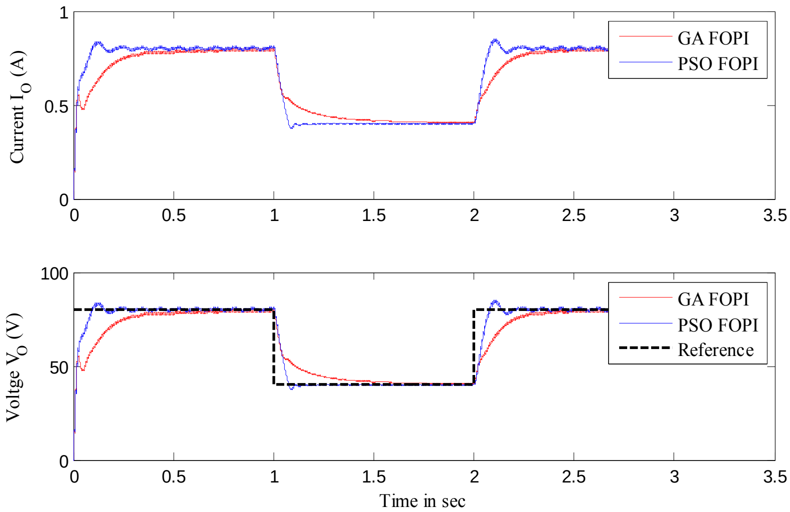

Fig.11. Simulated servo response of voltage and current for GA & PSO tuned FOPI controlled DC-Boost converter at time t=1 sec 50% decrement and time t=2 sec 50% increase.

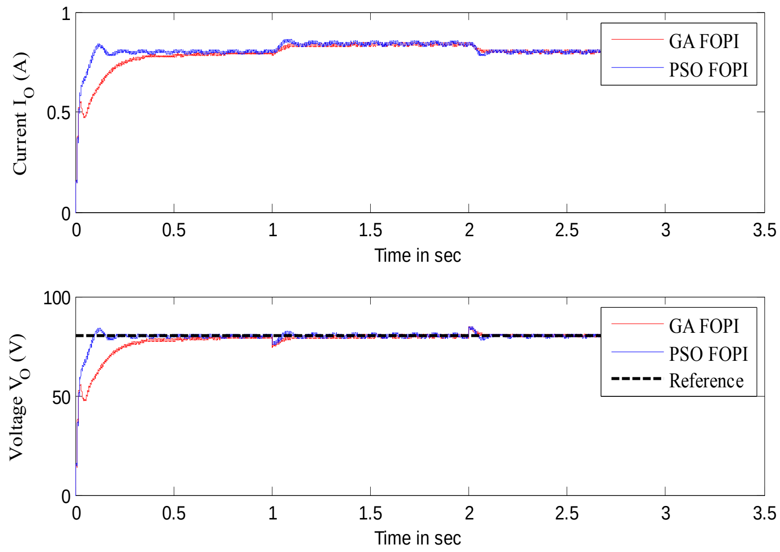

Fig.12. Simulated regulatory response of voltage and current for GA & PSO tuned FOPI controlled DC-Boost converter at time t=1 sec 10% decrement and time t=2 sec 10% increment.

TABLE – 2: FOPI Controller Parameters

Controller	K_p	K_i	Lambda	ISE
GA FOPI	0.2824	10.2315	0.9172	1.6649e+07
PSO FOPI	0.1336	32.5579	0.9839	1.5663e+07

TABLE 3: Simulated performance analysis of DC-DC Boost converter

Controller	Rise time	Peak Time	Settling time	Steady state error
GA FOPI	0.2188	0.3755	0.3755	-0.43
PSO FOPI	0.073	0.1285	0.1587	0.22

10. CONCLUSION:

In this paper, the PSO and GA method of tuning are used to find the optimal parameters of fractional order controller which minimizing ISE. This obtained value is applied to the Fractional Order PI Controller of the DC-DC boost converter to confirm the response of the controller. The stability of the system is verified with servo and regulatory responses using Genetic Algorithm and Particle swarm optimization algorithms. Using fractional order controller the dynamic performance of DC to DC converter is significantly improved. The dynamic performance of boost converter is carried out using MATLAB environment. In view of all the results from the simulation, the PSO based FOPI controller can achieve good time domain performance, disturbance rejection, and robustness, as compared to GA. Moreover, PSO search can achieve faster search and better solutions compared with GA.

11. REFERENCES:

[1]. D.M.Sable, H.Cho, and R.B.Ridley," Use of leading-edge modulation to transform boost and fly back converters into minimum phase zero systems", *IEEE Transactions on Power Electronics*. vol.6.no.4, oct.1991.

[2] S. Kapat, A. Patra, and S. Banerjee, "A current-controlled tristate boost converter with improved performance through RHP zero elimination," *IEEE Trans. Power Electronics*, vol. 24, no. 3, pp. 776–786, Mar. 2009.

[3] A. G. Perry, G.Feng, Y. F. Liu, and P.C. sen, "A design method for PI-like fuzzy logic controller for DC-DC converter", *IEEE Trans. Ind. Electron*, vol. 54, no. 5. pp. 2688-2695, Oct. 2007.

[4] B. Aldo, D. Corsanini, A. Landi, and L. sani, "Circle based Criteria for performance evaluation of controlled DCDC switching Converters", *IEEE Trans. Ind. Electron*, Vol. 53, no. 6, pp. 1862-1869, Dec. 2006.

[5] Jun Zhang, Henry Shu-Hung Chung, Wai-LUN Lo, S.Y.Ron Hui, "Implementation technique For Design of switching Regulators Using Genetic Algorithm", *IEEE Transactions on power electronics*, vol. 16, no. 6, November 2001.

[6] J. Y. Hung, W. Gao, and J. C Hung, "variable structure control: A survey," *IEEE Transactions. ind. electron*, vol. 40. No. 1. pp. 2-22, Feb. 1993.

[7] A.J. Calder´on, B.M. Vinagre, V. Feliu, Fractional order control strategies for power electronic buck converters, Signal Processing, 86, 2006, 2803-2819.

[8] M. Schlegel, M.ech, Fractal system identification for robust control - the moment approach, Proceedings of the 5th International Carpathian Control Conference, 2004, 2004.

[9] N.M. Fonseca Ferreira, J.A. Tenreiro Machado, Fractional-order hybrid control of robotic manipulators, 11th International Conference on Advanced Robotics, 2003, 393-398.

[10] I. Podlubny, Fractional-order systems and $PI^{\lambda}D^{\mu}$ controllers, IEEE Transactions on Automatic Control, 44, 1999, 208-214.

[11]. Y. Chen, H. Dou, B. M. Vinagre, and C. A.Monje, "A Robust Tuning Method For Fractional Order PI Controllers," in Proceedings of the 2nd IFAC Workshop on Fractional Differentiation and its Applications, 2006.

[12] K.S. Miller, B. Ross "An Introduction to the Fractional Calculus and Fractional Differential Equations", New York: Wiley, 1993.

[13] K.B. Oldham, J. Spanier, "The Fractional Calculus", New York: Academic Press, 1974.

[14] Y.Q.Chen, K.L.Moore, "Discretization schemes for fractional differentiators and integrators", IEEE Transactions on Circuits System 1: Fundamental Theory Applications, Vol.49, No.3, pp.363-367, 2002.

[15]. Kinattingal and V.T.Sreedevi, "Boost converter controller design using Queen-Bee-Assisted GA", *IEEE Transaction on industrial Electronics*.Vol.56.NO.3 March 2009.

[16] Jun-Yi Cao, Jin Liang and Bing-Gang Cao, "Optimization of fractional order PID controllers based on genetic algorithm", Proceedings of International

Conference on Machine Learning and Cybernetics, 2005, Volume 9, pp. 5686-5689, Guangzhou, China, 18-21 Aug. 2005.

[17] Li Meng and Dingyu Xue, "Design of an optimal fractional-order PID controller using multi-objective GA optimization", Proceedings of Chinese Control and Decision Conference, pp. 3849-3853, Guilin, 17-19 June 2009.

[18] M. Clerc, J. Kennedy, The particle swarm: explosion stability and convergence in a multi-dimensional complex space, IEEE Trans. Evolution. Computing. 6 (1) (2002) 58–73.

[19] R.C. Eberhart, Y. Shi, Comparing inertia weights and constriction factors in particle swarm optimization, in: Proc. CEC, San Diego, CA, 2000, pp. 84–88.

[20] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Proc. IEEE Conf. on Neural Networks, IV, Piscataway, NJ, 1995, pp. 1942–1948.

[21] J. Kennedy, R.C. Eberhart, Y. Shi, Swarm Intelligence, Morgan Kaufmann Publishers, San Francisco, CA, 2001.

Authors

Rajamani Senthilkumar was born on March 11, 1977. He received his B.E. in Electrical and Electronics Engineering and M.E. in Power Electronics and Drives in 2007 from Anna University, Chennai, Tamil Nadu, India. He is currently pursuing his Ph.D. in Power Electronics and Drives at Anna University, Chennai, Tamil Nadu, India. His research interests include power electronics and drives. He is currently an assistant professor (Sr.Grade) at the Department of Electrical and Electronics Engineering at PPG Institute of Technology, Coimbatore, Tamil Nadu, India.

Dr.Venugopal Manikandan is currently working as Professor in the department of Electrical and Electronics Engineering at Coimbatore Institute of Technology, Coimbatore - 641 014, India. He is also guiding researchers in the area of soft computing. He obtained his doctoral degree from Anna University, Chennai. He did his masters in Applied Electronics at PSG College of Technology, Coimbatore, and Bachelor's degree in Electrical and Electronics Engineering at Government College of Technology, Coimbatore.