ANALYSIS OF BOOST CONVERTER USING SOFT COMPUTING TECHNIQUES WITH RLE LOAD FOR PHOTO-VOLTAIC APPLICATION

Reeba Rex S[1] ,Pravin Rose T[2], Manjusha M[3],Amudaria S[4] ,Aravind J.M[5]
[1]Associate Professor,Department of ECE, Mar Ephraem College of Engineering and Technology, Elavuvilai, India. Email:-reebarex@gmail.com
[2]Associate Professor,Department of EEE, Vidya Academy of Science and Technology, Trivandrum, India.Email:-pravinrose25@gmail.com
[3]Assistant Professor,Department of EEE, Mar Ephraem College of Engineering and Technology, Elavuvilai, India. Email;-*manjupaul02@gmail.com*
[4]Assistant Professor,Department of CSE, Arunachala College of Engineering for Women, Manavilai, India. Email:-daria.amu@gmail.com
[5]Assistant Professor,Department of ECE, Mar Ephraem College of Engineering and Technology, Elavuvilai, India. Email;-*jamesaravind@gmail.com*

Abstract:-
This paper aims the designof a DC-DC boost converter for providing stabilized output to the variation in RLE-load.A suitable controller which can be practically realized for DC-DC boost converter is done with various optimizationtechniques that possess simple implementation, better convergence quality and enhanced computational ability.The PID controller parameters optimally using various optimization algorithms for enhancing the dynamic response of the converter is done in the presence of RLE-load.To tune the PID controller parameters using various optimization algorithms to enhance the efficiency of the RLE-load.The effectiveness in terms of time domain specifications, such as settling time, rise time, peak time, and overshoot, and the time integral domain specifications, namely Integral Square Error (ISE), Integral Absolute Error (IAE), and Integral Time Absolute Value Error (ITAE) is analysed.The research modeling of thistechnique is done in MATLAB/SIMULINK 2014 platform in terms of various performancemetrics.This research can be applied to Photo-Voltaic Application for generation of energy.

Keywords- PID, Boost-Converter, Integral Square Error (ISE), Integral Absolute Error (IAE), and Integral Time Absolute Value Error (ITAE).

I.Introduction:-

Power electronics deals with different types of converters that are generally used at power level rather than signal level. A power electronic system comprises of one or more number of power electronic converters made up of some power semiconductor devices that are controlled using the integrated circuit by [1], [2]. The switching features of the power semiconductor devices allow a power electronic converter to change the input power of one form into the output power of some other form in [3]. Power Electronics is also ushering in a new type of industrial revolution with its versatility in applications, such as conservation of energy, energy storage for bulk utility, renewable energy system, and industrial automation. In case of power conversion, a DC-DC converter acts an important role with the widespread applications of laptops, Light Emitting Diode (LED) drivers, cellular phones, electric vehicles, hydro power plants, maximizing the harvest of energy for photovoltaic systems and wind turbines, and so on in [4], [5]. This application needs the converter to achieve increased efficiency, and Power Factor (PF) at the load side, while reducing the size and price of the device with the increase in availability by [6], [7]. A DC-

DC converter either increases or decreases the input voltage, based on the need of the connected load, with the adjustment of the duty cycle applied to the switching device, such as Metal-Oxide-Semiconductor Field Effect Transistors(MOSFET) and Insulated Gate Bipolar Transistor (IGBT)in [8], [9]. One of the most prominent research interests in the field of power converters is the application of DC-DC converters with high step-up voltage gain by [10], [11]. Various control methods, such as Fuzzy Logic Controller(FLC), Artificial Neural Network (ANN), and PID controller have been designed to assure the stability and the fast transient response. The optimization techniques, like Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Bacterial Foraging Optimization (BFO) have also been developed by [12]. Among all the converters, boost converter is the most commonly used DC-DC converter. It is a step up converter that provides a higher voltage at the load side as compared to the source voltage. Open loop mode of operation of the boost converter possesses substandard regulation of voltage and undesirable dynamic response. Thus, closed loop mode of operation is chosen for proper voltage regulation and performance enhancement by [13].

The paper is organized in 8 sections.Section II explains the basic function of Boost Converter.Section III explains the fitness function for optimizing the parameters of Boost Converter.Section IV explains the QBGA Algorithm to optimize the parameters of the Boost Converter. Section V explains the SFLA Algorithm to optimize the parameters of the Boost Converter. Section VI explains the CSA Algorithm to optimize the parameters of the Boost Converter. Section VII explains the FFA Algorithm to optimize the parameters of the Boost Converter. Section VIII explains the simulation done in MATLAB and results are discussed.

II.Basic function of Boost Converter

The converter whose voltage at its output is more than that of the input voltage is called as boost converter in [14]. The increase in output voltage gives its name as boost or step-up converter. The DC-DC boost converter requires only four external components, such as diode, electronic switch, inductor, and an output capacitor. The converter can thus be operated in the two different modes based on its capacity to store energy and its relative duration of the switching period by [15]. The role of the capacitor at the resultant side is to make the resultant voltage free from ripple and an inductor in the input side is to make the ripple free current corresponding to the voltage. The boost converter is generally analyzed in Continuous Conduction Mode (CCM).

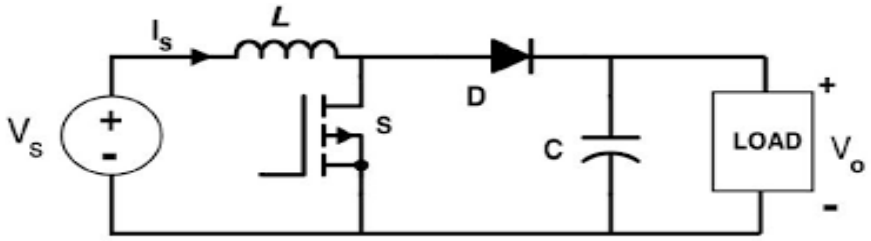

Figure 1: SchematicBoost Converter

a) Switch-on state

When switch is in ON condition, the diode gets reversed-biased with the isolation of the output stage, and thus the inductor is supplied with energy from the input.

b) Switch-off state

When the switch is in OFF condition, the output draws energy from the input side and from the inductor. The output capacitance is considered to be very large to make sure of a constant voltage at the output. The output of the power converters are needed to be regulated within a specified tolerance limit to get the desired output by varying the voltage set. The resultant is compared with the reference value and the error value is used to regulate the duty cycle of the switch used in the converter by [11].

III) Fitness evaluation:

The term fitness is termed as the minimal value of error signal that is estimated as the difference among the acquired value of the boost converter and the reference value. The error for fitness is estimated in Equation (1) for RLE load as,

$N_{error} = N_{ref} - N_{out}$ For RLE load (1)

where, N_{out} represents the output from the boost converter and N_{ref} is the reference value. The fitness or the objective functions taken are rise time, settling time, ISE, IAE, and ITAE and the error is the output, which is expressed in Equation (1). The optimization is done using four optimization algorithm and is explained in below sections.

IV)QBGA -PID OPTIMIZATION

The QBGA-PID optimization is an iterative process that is performed till obtaining an optimum solution. The objective is to minimize the fitness function of rise time, settling time, ISE, IAE, and ITAE. The sequential steps of the QBGA are outlined as below, and by applying QBGA, the PID parameters are optimized.

Step 1: Initialization of algorithm parameters and the optimization problem:

The optimization is performed to obtain the optimal values of k_p, k_i, and k_d. In this step, the optimization problem (Minimize (rise time, settling time, ISE, IAE, and ITAE)), and the decision variables (n) are defined. In addition, the population size (R), the recombination probability (p_r), and the maximum number of iterations $(I_{\max})$ are also defined. The objective function is represented in Equation (2) as,

Minimize
$F(x) = (1+t_r)(1+t_s)(1+E_{ISE})(1+E_{IAE})(1+E_{ITAE})$ (2)

where, $F(x)$ is the objective, in terms of rise time t_r, settling time t_s, ISE, IAE, and ITAE, and D is the count of decision variables. The parameters of QBGA algorithm are also specified in this step.

Step 2:Generation of bees

Let the number of bees that are involved in the solution space be represented as $B_1, B_2, ..., B_i, ..., B_n$, where n is the population size of the bees.

Step 3: Queen Bee Identification

Among the bees that are generated in random, the best queen bee B_q possessing the best binary structure in minimizing the function $F(x)$ in terms of rise time, settling time,ISE, IAE, and ITAEis chosen. The closed loop feedback control system is simulated digitally with all the bees, and $F(x)$ is evaluated for each bee. Equation (3) provides the queen bee from the n number of bees,denoted as B_q.

$$B_q = Max\left(\frac{1}{1+F(x_i)}\right) \quad (3)$$

Thus, the best queen bee is obtained from Equation (3) and is separated from the remaining $(n-1)$ bees, which are then identified as the drones.

Step 4:Reproduction or mating flight

The queen bee develops a mating flight for the next generation of the bees to be reproduced. All the drones do not have the capability to fly quickly to reach a queen that offers a provision for incorporating a recombination probability associated with all the drones. For this, the probability of recombination, represented as p_r, is fixed between 0 and 1. The probability of each drone is represented as p_i. For the condition, $p_i \geq p_r$, the queen bee recombines with the drone resulting in two virgin queen bees. The recombination process of queen bee with the drone is exactly similar to the process of crossover in the standard GA. However, in QBGA, the crossover for only one parameter is described for clarity, and the multipoint crossover is advised for enhanced efficiency. Recombination generates two offspring among which, only the fittest one survives and the other one is discarded, which is similar to killing the offspring by the virgin queen bee.

Step 5:Recombination

For *p*i <*p*r, no recombination occurs, and hence, no offspring is generated at this condition. After the completion of steps 3 and 4, that is at the completion of reproduction, the generation of virgin queen bees is noticed to be less than the population of the bees *n*.

Step 6:Piping

In a bee hive, there occur a fight between all the virgin queens, and only the fittest virgin bee survives. The population of all the virgin queen bees in addition with their mother queen bees are evaluated again using the Equation (3), and the new queen bee is identified. Finally, all the other bees except the selected one are discarded from the hive.

Step 7:Termination
Terminate the program when the termination criterion is reached and choose the newly generated queen bee as the optimum one. Otherwise go to the next step.
Step 8: Development of a new population of drone for next mating flight
All the drones dies once mates with the queen bee, and hence, population size of (n –1) drones are generated again for the next reproduction in a random manner. The drones that are generated in random and the new queen bee develop the population for the next generation, and goes to step 4.

Table 1: Parameter setting of QBGA –PID

Parameters	Values
Decision variables, n	10
Population size, R	10
Mutation probability	0.2
Cross over probability	0.8
Recombination probability p_r	0.8
No of iteration, $I_{\max}$	500

V)SFLA-PID Optimization

The SFLA-PID optimization is an iterative process that is used in tuning the PID controller parameters of the boost converter. The objective of the design of SFLA is to minimize the fitness function of rise time, settling time,ISE, IAE, and ITAE.

i) Algorithmic steps of Shuffled Frog Leaping Algorithm

The steps in the procedure of SFLA are,

Step 1: Initialization of problem and the parameters involved in the algorithm:

The optimization problem is represented as in Equation (5.4), and the optimization is performed to obtain the optimal values of k_p, k_i, and k_d. In this step, the optimization problem (Minimize (rise time, settling time, ISE, IAE, and ITAE)) is defined. In addition, the population size (R), the number of memeplexes (m), number of frogs in each memeplexes, maximum step size, and the maximum number of iterations $(I_{\max})$ are also defined.The objective function is represented in Equation (4) as,

Minimize

$$F(x) = (1 + f_1)(1 + f_2)(1 + f_3)(1 + f_4)(1 + f_5) \quad (4)$$

where, $F(x)$ is the objective, in terms of rise time (f_1), settling time (f_2), ISE (f_3), IAE (f_4), and ITAE (f_5).The position of each frog is represented using Equation (5) as,

$$X_g = lb_j + rand\ (0,1)(ub_j - lb_j) \quad (5)$$

where,the number of frogs, i = 1, 2, ..., X, j = 1, 2, ..., D (D dimensional vector) and, lb_j and ub_j represents the bound exhibited by lower and upper limits in the dimension j.

Step 2:Fitness evaluation

Evaluate fitness for all the frogs, and sort them on the basis of their fitness measures in descending order in terms of rise time, settling time,ISE, IAE, and ITAE and divide into m subsets, which are termed as memeplexes. The population of n number of frogs are placed in every subnet, and the distribution is achieved based on the fact that, the most fitness value of the frogs is passed to the first subnet and the next oneis

placed in the second subnet. In each subnet, the best frogs x_b and the worst frogs x_w are verified. This type of process is also indicated as memeplex evolution. The positions of worse individualsareexpressed using the Equations (6) and (7),

$$D_i = rand\left(x_b - x_w\right) \quad (6)$$

$$x_w^{new} = x_w + D_i; \qquad \left(-D_{\max} \le D_i \le D_{\max}\right) \quad (7)$$

where, $rand(\ )$ indicates randomly generated number within the range 0 to 1, D represents the movement of frog, i is the movement of i^{th} frog, $D_{\min}$ and $D_{\max}$ are the minimum and maximum allowable variation in position of the frogs, respectively. The maximum evolution generation in each set is denoted as N_{gen}. When the new position x_w^{new} of the worst frog is better compared to the old position, then the position of the worst frog is replaced by the new position. If not, the evaluation in Equation (5.6) and Equation (7) are repeated with respect to the entire population of the best frog x_g. In other words, x_b is replaced by x_g as predicted and is given in Equation (5.8) as,

$$D_i = rand\left(x_g - x_w\right) \quad (8)$$

Thus, the Equation (5.7) also changes correspondingly to find the position of the worst frog in the memeplexes.

Step 3:Shuffling

If the attempt taken in step 2 is still unable to provide anenhanced performance, the position for the worst frog is generated in random manner. The iteration is continued for a pre-defined count of memetic evolutionary times within all the memeplexes, and again the whole population is mixed in the shuffling process.

Step 4:Termination Condition

The local evolution process and the global shuffling process are carried outtill reaching the maximum criteria of convergence.

Table 2**:**Parameter setting of SFLA-PID

Parameters	Values
Population size, P	60
Number of memeplexes, m	6
Number of frogs in each memeplexes	10
Maximum step size	2
Number of generations for each memeplex	10
Maximum iteration	100

•

VI.CSA-PID Optimization:-

The CSA-PIDoptimization is a process which is performed to tune the PID controller of the boost converter. The need of the design is to minimize the fitness function of rise time, settling time,ISE, IAE, and ITAE.

i)Algorithmic steps of Cuckoo Search Algorithm

The steps that involve in the procedure of CSA are,

Step 1: Initialization of parameters and the problem of optimization

The optimization is performed to obtain the optimal values of k_p, k_i, and k_d. In this step, the optimization problem (Minimize (rise time, settling time, ISE, IAE, and ITAE)) is defined. In addition, the population size (P), step constant α_0, standard normal random variables, u and v, control factor of levy flight, β, probability p_a and the maximum number of iterations $\left(I_{\max}\right)$ are also defined.The objective function is represented as,

Minimize

$$F(x) = (1+\lambda_1)(1+\lambda_2)(1+\lambda_3)(1+\lambda_4)(1+\lambda_5) \quad (9)$$

where, $F(x)$ is the objective, in terms of rise time (λ_1), settling time (λ_2), ISE (λ_3), IAE (λ_4), and ITAE (λ_5).

Step 2: Calculate fitness for new solution

Develop new positions in the levy flight method, $x = (x_1, x_2, ..., x_D)^T$, and calculate fitness for each solution. Levy flight normallymake in use of the random walk strategy as expressed in Equation (10) and (11) as,

$$x_i^{t+1} = x_i^t + \alpha_0 \frac{\phi \times u}{v^{\frac{1}{\beta}}} \left(x_i^t - x_{best}^t\right) \quad (10)$$

where,

$$\phi = \left[\frac{\Gamma(1+\beta) \times \sin(\Pi \times \beta / 2)}{\Gamma\left(\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{(\beta-1)/2}\right)} \right]^{1/\beta} \quad (11)$$

where, x_i^t is the i^{th} solution in t^{th} iteration, x_{best}^t is the current best solution, α_0 represents the step constant, u and v indicates the standard normal random variables, and β is the control factor.

Step 3: Selection of best solution

Choose a solution among n in random manner and evaluate the fitness measure. When the fitness of the new solution is better on comparison with the old one, then replace the old one with the new solution.

Step 4: Elimination of worst solution:

The fraction p_a of the worst solutions is abandoned, and the new ones are developed to be involved in the searching mechanism.

Step 5: Ending judgement and termination:

Keep the high quality solutions, and then rank them to find the current best solution. When the iterative condition of termination $(t < I_{\max})$ is reached, stop the algorithm, or return to step 2.

Table 3**:** Parameter setting of CSA-PID

Parameters	**Values**
Initial population P	25
Step constant, α_0	$\left(\frac{u}{\lvert v \rvert}\right)^{\alpha} \bullet \left(\frac{1}{\beta}\right)$
Standard normal random variables, u and v	$u = \sigma *$ rand v=rand
Control factor of Levy flight, β	1.1
Probability p_a	0.25
No of iteration, $I_{\max}$	100

VII.FFA-PID Optimization:-

The FFA-PIDoptimization is a process which is performed to tune the PID controller of the boost converter. The need of the design is to minimize the fitness function of rise time, settling time,ISE, IAE, and ITAE.

i)Algorithmic steps of Firefly Algorithm

The steps in the procedure of FFA are as follows,

Step 1: Problem initialization and algorithm parameters:

The optimization is performed to obtain the optimal values of k_p, k_i , and k_d . In this step, the optimization problem (Minimize (rise time, settling time, ISE, IAE, and ITAE)) is defined. In addition, the population size (R) , Light absorption coefficient γ , Light intensity I ,

Attractiveness β, and the maximum number of iterations $(I_{\max})$ are also defined. The objective function is represented in Equation (12) as,

Minimize

$$F(x) = (1+\delta_r)(1+\delta_s)(1+\delta_{ISE})(1+\delta_{IAE})(1+\delta_{ITAE}) \quad (12)$$

where, $F(x)$ is the objective, in terms of rise time δ_r, settling time δ_s, ISE (δ_{ISE}), IAE (δ_{IAE}), and ITAE (δ_{ITAE}).

Step 2: Define light absorption coefficient

The light absorption coefficient γ or the fitness is evaluated in terms of IAE, IAE, and ITAE), as it is considered as the typical initial measure of Γ in the proposed optimization technique.

Step 3: Check for maximum generations

When the iterations are less than the maximum number of iterations, evaluate the light intensity for i^{th} firefly. If the intensity of the i^{th} firefly is greater than j^{th} firefly (selected in random among the population), then move the i^{th} firefly towards j. Otherwise, move the firefly j in random manner.

Step 4: Evaluation of attractiveness

The attractiveness β that is directly proportional to the light intensity of each firefly is evaluated as it changes with respect to distance.

Step 5: Determination of new solution:

Determine all the newly generated solutions and re-estimate the light intensity for all the fireflies. Rank all the fireflies depending on their light intensity and choose the current best solution.

Step 6: Termination:

When the condition of termination is met, then stop the algorithm. Otherwise return to step 2.

Table 4:Parameter Setting of FFA-PID

Parameters	**Values**
Population size	20
Light absorption coefficient, γ	0.1
Light intensity, I	1
Mutation Co-efficient	0.8
Attractiveness, β	0.2
Maximum iterations $I_{\max}$	100

Table 4 shows the parameters initialized and assumed during the optimization of FFA-PID algorithm in the proposed boost converter fed R-load or RLE-load.

VIII.Simulation / Results

The simulation of the entire system is carried out in MATLAB/SIMULINK environment. The best control parameters for QBGA algorithm was selected, and the *m.files*were performed to partition the data of fitness and the gain values. The algorithms were executed with 10 simulation-runs. Each simulation-run has 500 numbers of iterations. The count of simulation runs acts as stopping criteria in QBGA algorithm. From the overall simulation-runs, the runtime that returned the least fitness and best optimal solution was selected.The solution is enhanced in all successive iterations and the process is continued until finding the optimal solution.The simulation is done with R-load and RLE-load for QBGAtuned PID controller. Thus,QBGA tuned PID were used to find the better parameters of fitness k_p, k_i, and k_d .

Table 5: Gain Parameters of QBGA Tuned PID Controller

Sl. No.	Methods	k_p	k_i	k_d
1.	QBGA Tuned PID Controller	0.8101	0.6863	0.0220

Table 6: Gain Parameters of SFLA Tuned PID Controller

Sl. No.	**Methods**	k_p	k_i	k_d
1.	SFLA Tuned PID Controller	1.4392	0.6032	0.01223

Table 7: Gain Parameters of CSA Tuned PID Controller

Sl. No.	Methods	k_p	k_i	k_d
1.	CSA Tuned PID Controller	1.0734	0.6336	0.0234

Table 8: Gain Parameters of FFA Tuned PID Controller

Sl. No.	Methods	k_p	k_i	k_d
1	FFA Tuned PID Controller	0.9001	0.8844	0.0781

Table 9: Comparative Analysis of Performance Indices(RLE-Load)

Sl. No.	Type of Error	QBGA TunedPIDController	SFLA TunedPIDController	CSA Tuned PID Controller	FFA TunedPIDController
1.	ISE	0.3038	0.0805	0.0707	0.0606
2.	IAE	0.6221	0.1616	0.1255	0.1023
3.	ITAE	0.1666	0.0801	0.0625	0.0356

Comparative analysis of performance indices (RLE-Load) is depicted in Table 6.8. From the Table 6.8 it is noticed that FFA tuned PID controller has reduced ISE, IAE and ITAE when compared toother optimization algorithms. The value of ISE using QBGA tuned PID controller, SFLA tuned PID controller, CSA tuned PID controller, and FFA tuned PID controller is 0.3038, 0.0805, 0.0707, and 0.0606, respectively. Similarly, the value of IAE using QBGA tuned PID controller, SFLA tuned PID controller, CSA tuned PID controller, and FFA tuned PID controller is 0.6221, 0.1616, 0.1255, and 0.1023, respectively. The value of ITAE using QBGA tuned PID controller, SFLA tuned PID controller, CSA tuned PID controller, and FFA tuned PID controller is 0.1666, 0.0801, 0.0625, and 0.0356, respectively. Thus, from the Table 6.8 it is evident that FFA tuned PID controller has reduced ISE, IAE and ITAE as compared to other optimization algorithms .

Sl.No:	Parameters	QBGA	SFLA	CSA	FFA
1.	Rise Time (sec)	0.9868	0.9247	0.7351	0.6170
2.	Settling Time (sec)	4.1883	3.7703	2.8963	2.2
4.	Settling (Max) (V)	48.0271	47.7965	47.9528	48.0192
5.	Peak Over Shoot	0.1426	0.1291	0.1209	0.0497
6.	Peak Value (V)	48.0271	47.7965	47.9528	48.0192

Table 10: Comparative Analysis for Time Domain Specifications (RLE-Load)

Comparative analysis for time domain specifications (RLE -Load) is given in Table 10. The settling time and rise time is minimized to 2.2 sec and 0.6170sec with FFA. Results show that the performance of the FFA gives better improvement in settling time, rise time, peak overshoot and maximum settling time.

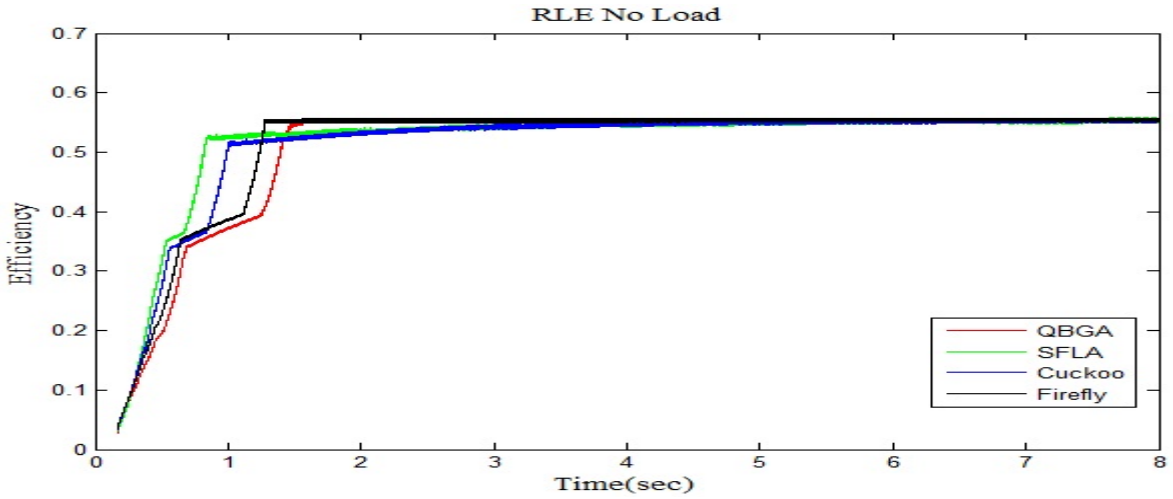

Figure 2:Efficiency of QBGA,SFLA,CSA and FFA Based PID Controller with RLE-No Load

Figure 2 efficiency of QBGA,SFLA,CSA and FFA based PID controller with RLE-no load.The efficiency noted in FFA based PID controller with resistive load is 57% which outperforms QBGA,SFLA and CSA.

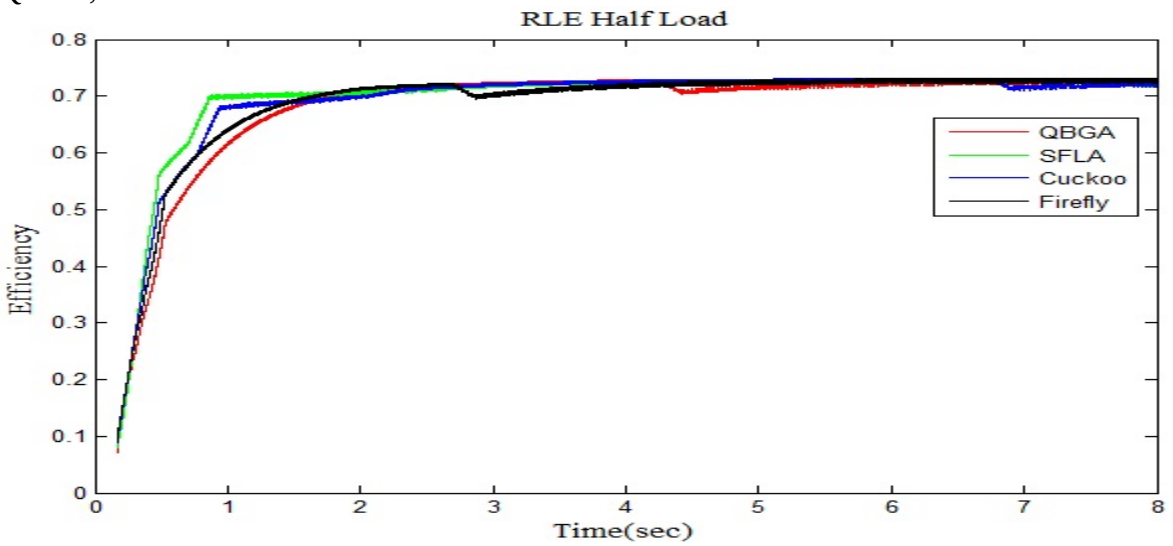

Figure 3:Efficiency of QBGA,SFLA,CSA and FFA Based PID Controller with RLE-Half Load

Figure 3 efficiency of QBGA, SFLA, CSA and FFA based PID controller with RLE-half load.The efficiency noted in FFA based PID Controller with RLE-half load is 73% which outperforms QBGA,SFLA and CSA.

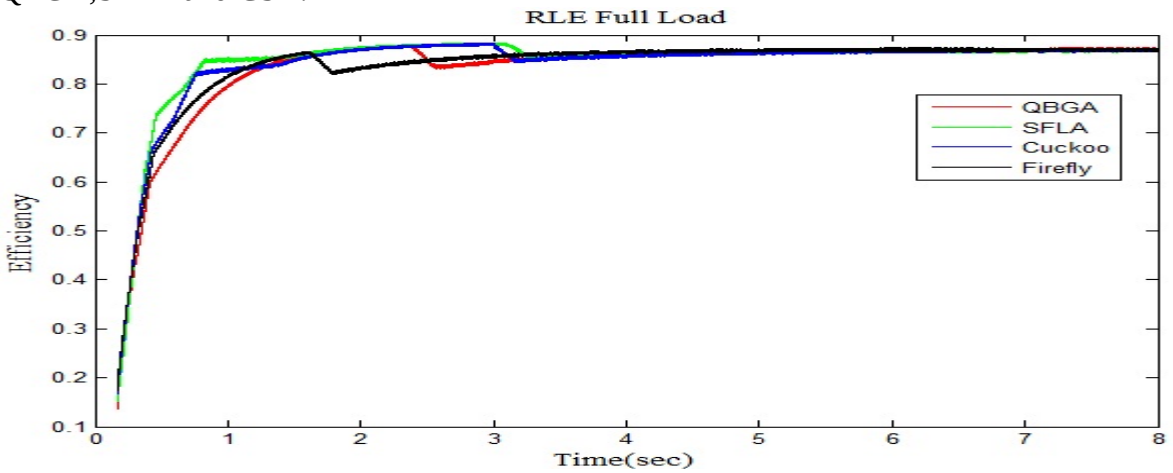

Figure 4: Efficiency of QBGA,SFLA,CSA and FFA Based PID Controller with RLE Full Load

Figure 4efficiency of QBGA,SFLA,CSA and FFA based PID controller with RLE full load.The efficiency noted in FFA based PID controller with RLE-fullload is 88% which outperforms QBGA,SFLA and CSA.

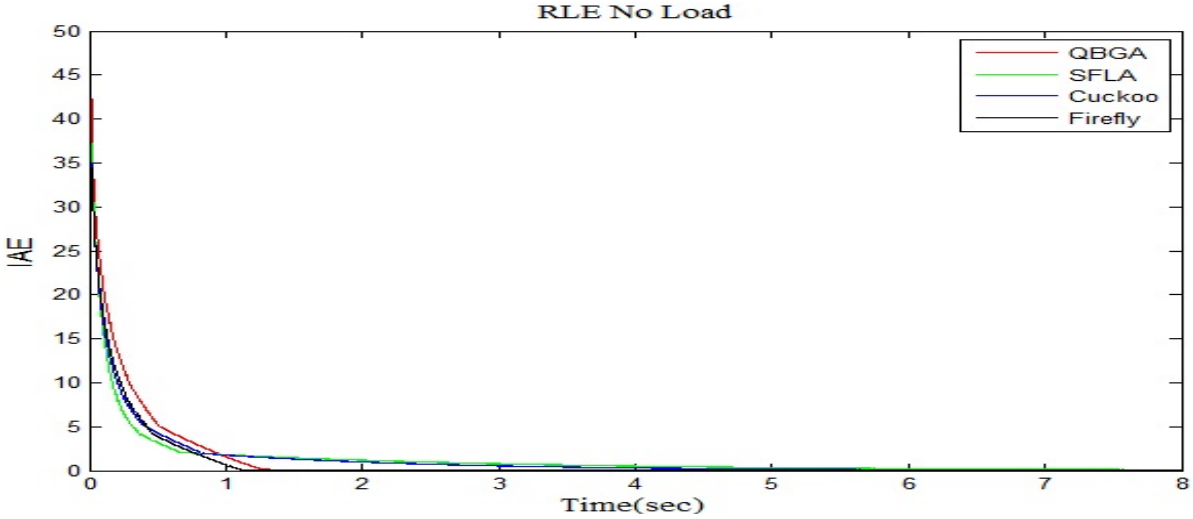

Figure 5: Comparative Response Curve with Respect to QBGA,SFLA,CSA and FFA Based PID Controller IAE with RLE-No Load

Figure 5 depict the comparative response curve with respect to QBGA,SFLA,CSA and FFA based PID controller IAE with RLE-no load.From the response curve it is evident that FFA converges earlier when compared to QBGA,SFLA and CSA.

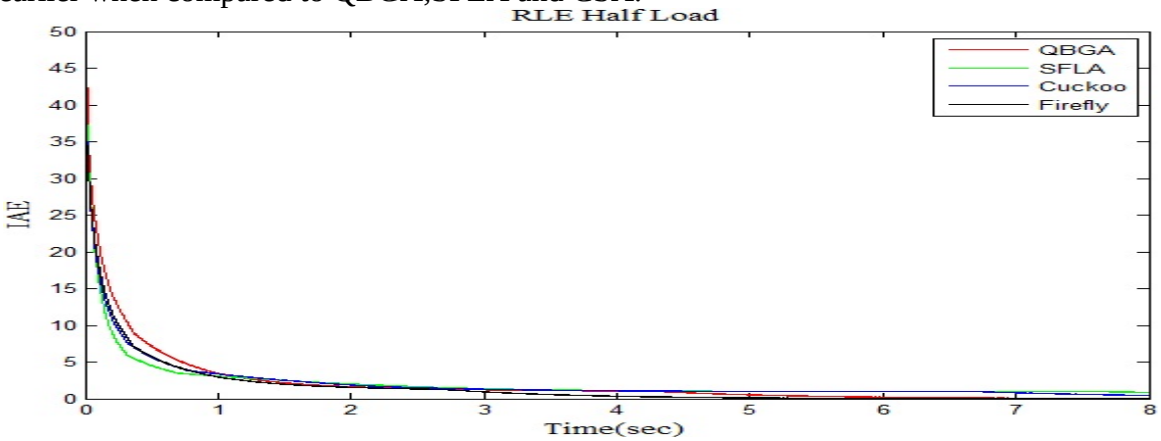

Figure 6: Comparative Response Curve with Respect to QBGA,SFLA, CSA and FFA Based PID Controller IAE with RLE-Half Load

Figure6 depict the comparative response curve with respect to QBGA,SFLA, CSA and FFA based PID controller IAE with RLE-half load.From the response curve it is evident that FFA converges faster when compared to QBGA,SFLA and CSA.

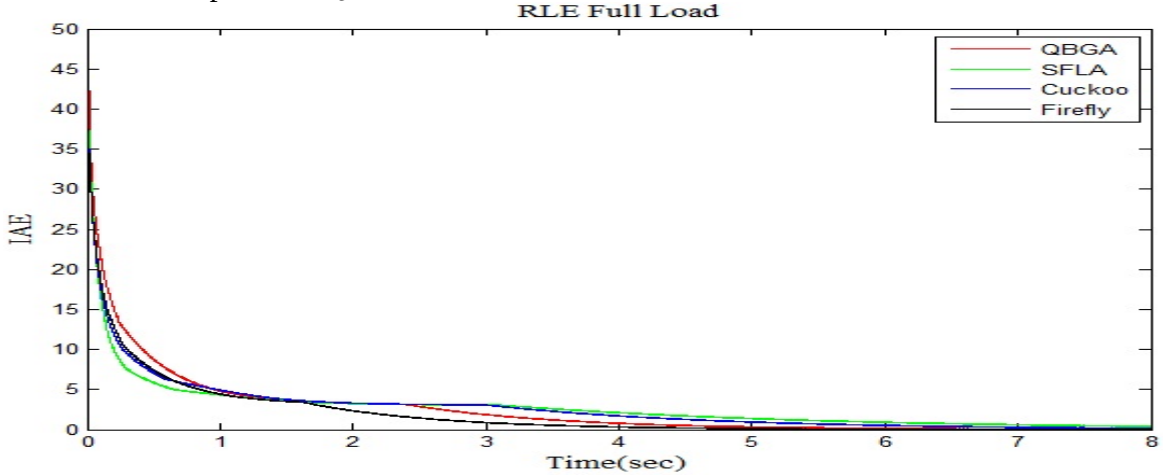

Figure 7: Comparative Response Curve with Respect to QBGA,SFLA, CSA and FFA Based PID Controller IAE with RLE-Full Load

Figure 7 depict the comparative response curve with respect to QBGA,SFLA, CSA and FFA based PID controller IAE with RLE-full load.From the response curve it is evident that FFA converges faster when compared to QBGA,SFLA and CSA.

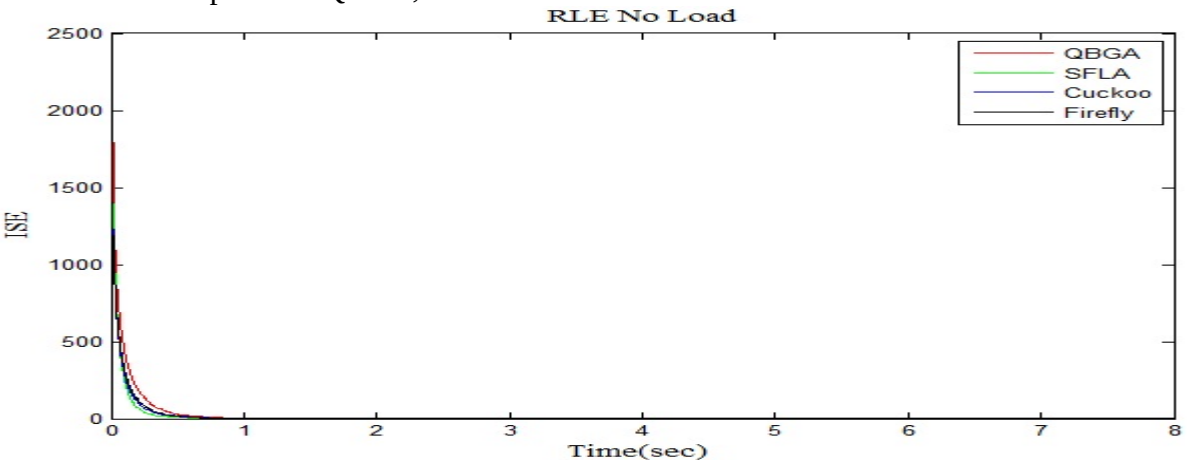

Figure 8: Comparative Response Curve with Respect to QBGA,SFLA, CSA and FFA Based PID Controller ISE with RLE-No Load

Figure 8 depict the comparative response curve with respect to QBGA,SFLA,CSA and FFA based PID controller ISE with RLE-no load.From the response curve it is evident that FFA converges faster when compared to QBGA,SFLA and CSA.

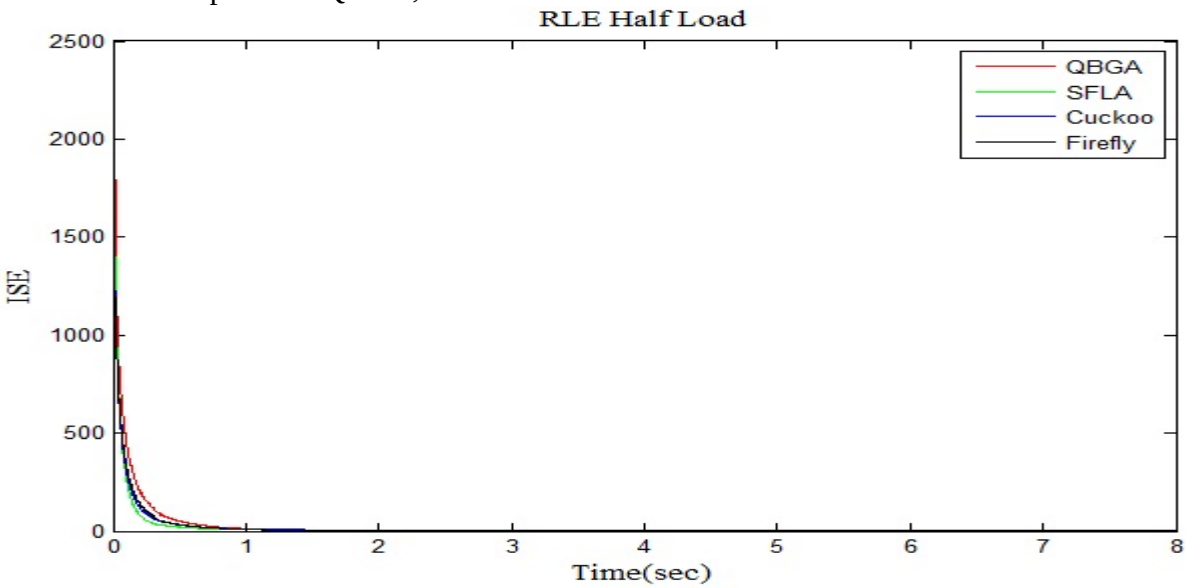

Figure 9: Comparative Response Curve with Respect to QBGA, SFLA, CSA and FFA Based PID Controller ISE with RLE-Half Load

Figure 9 depict the comparative response curve with respect to QBGA,SFLA, CSA and FFA based PID controller ISE with RLE-half load.From the response curve it is evident that FFA converges faster when compared to QBGA,SFLA and CSA.

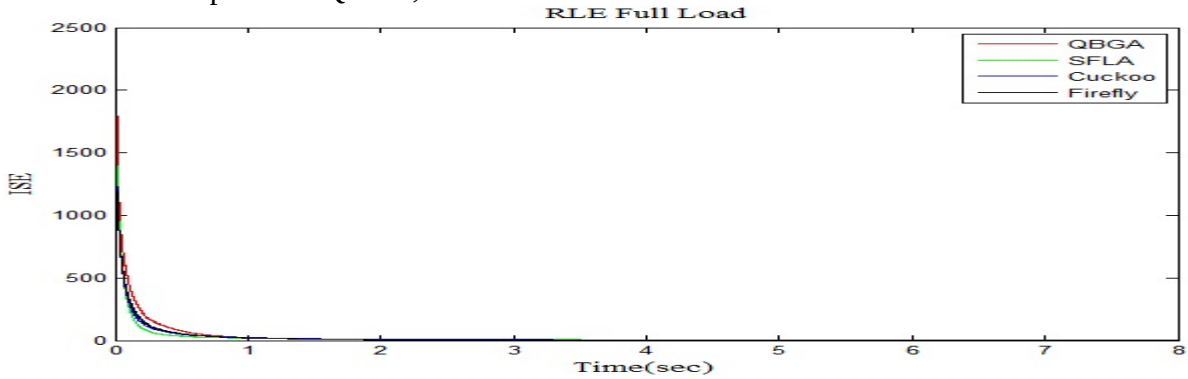

Figure 10: Comparative Response Curve with Respect to QBGA,SFLA, CSA and FFA Based PID Controller ISE with RLE-Full Load

Figure 10 depict the comparative response curve with respect to QBGA,SFLA,CSA and FFA based PID controller ISE with RLE-full load.From the response curve it is evident that FFA converges faster when compared to QBGA, SFLA and CSA.

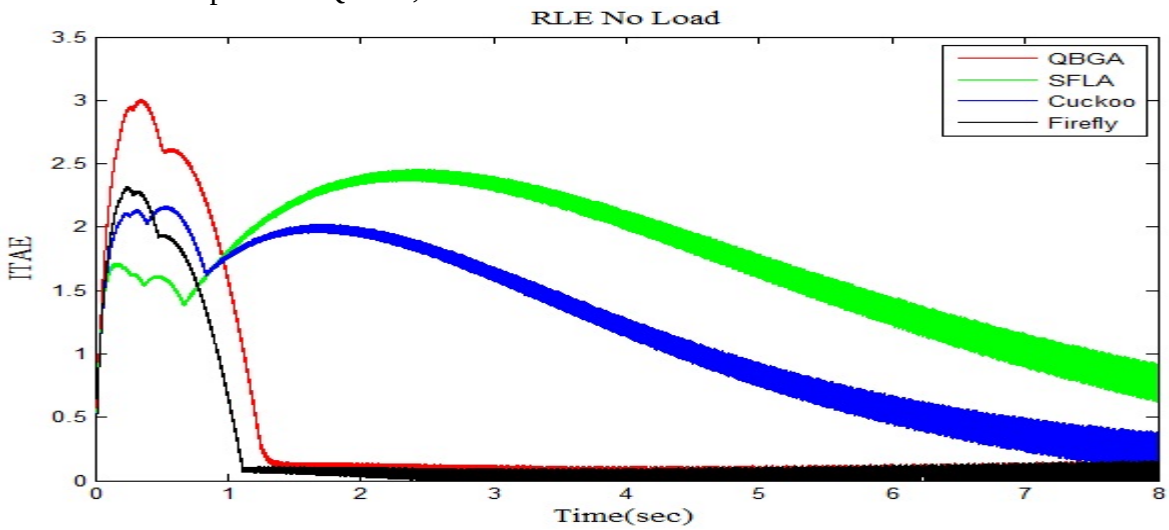

Figure 11: Comparative Response Curve with Respect to QBGA,SFLA, CSA and FFA Based PID Controller ITAE with RLE-No Load

Figure 11 depict the comparative response curve with respect to QBGA,SFLA, CSA and FFA based PID controller ITAE with RLE-no load.From the response curve it is evident that FFA converges faster when compared to QBGA,SFLA and CSA.

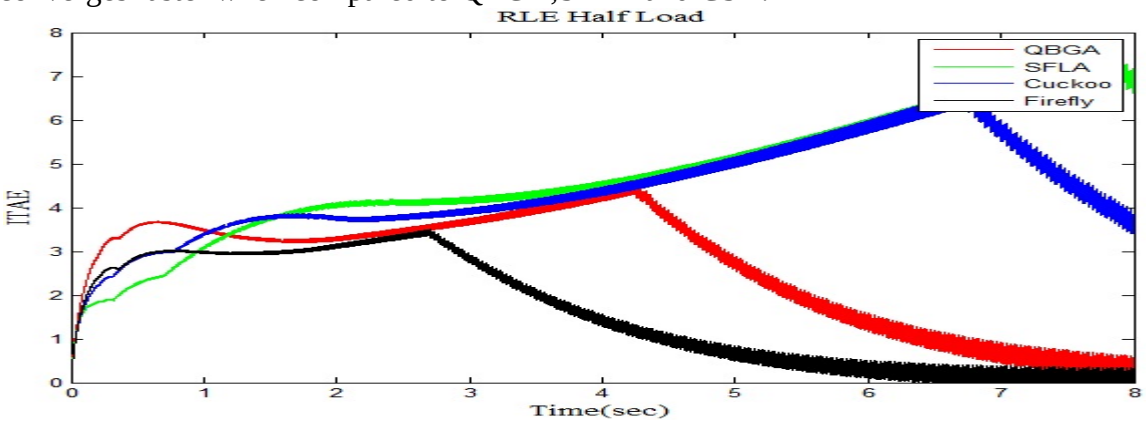

Figure 12: Comparative Response Curve with Respect to QBGA,SFLA, CSA and FFA Based PID Controller ITAE with RLE-Half Load

Figure 12 depict the comparative response curve with respect to QBGA,SFLA, CSA and FFA based PID controller ITAE with RLE-half load.From the response curve it is evident that FFA converges faster when compared to QBGA, SFLA and CSA.

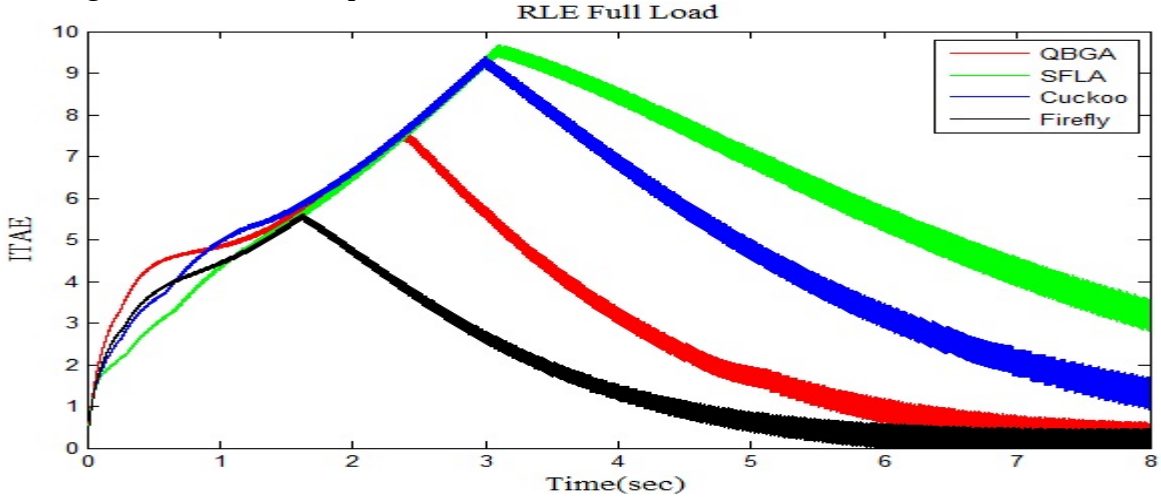

Figure 13: Comparative Response Curve with Respect to QBGA,SFLA, CSA and FFA Based PID Controller ITAE with RLE-Full Load

Figure 13 depict the comparative response curve with respect to QBGA,SFLA,CSA and FFA based PID controller ITAE with RLE-full load.From the response curve it is evident that FFA converges faster when compared to QBGA,SFLA and CSA. Results show that the performance of the FFA gives better improvement in settling time, rise time, peak overshoot and maximum settling time.Results show that the performance of the FFA gives better improvement in settling time, rise time, peak overshoot and maximum settling time.

Conclusion

This paper discusses the need for the proposed FFA algorithm in tuning the PID parameters in such a way to maintain constant output at the output of the boost converter. The duty cycle of the boost converter is changed based on the optimally tuned PID controller, and the effectiveness of the proposed approach is compared with the conventional methods in terms of settling time, rise time, and peak overshoot, and the time integral performance indices, namely ISE, IAE, and ITAE. The results indicate that the proposed FFA is effective in controlling the parameters of the PID controller. The dynamic response of the system using the proposed FFA is also analyzed and found to be effective when compared to other optimization algorithms. Thus, the performance of the proposed FFA is noted to be superior in terms of the expected objectives. This research work can be applied to photovoltaic application for generation of energy.

References:

1.Kahraman H. T., **Aras S.**, **Guvenc U.**, and **Sonmez Y.** (2017), Exploring the effect of distribution methods on meta-heuristic searching process. *In 2017 International Conference on Computer Science and Engineering (UBMK)*, 371–376. IEEE.

2. **Karthikeyan M.**, **Elavarasu R.**, **Ramesh P.**, **Bharatiraja C.**, **Sanjeevikumar P.**, **Mihet-Popa L.**, and **Mitolo M.**, (2020), A hybridization of Cuk and boost converter using single switch with higher voltage gain compatibility, *Energies*, **13**(9), 2312.

3.**He S.** and **Li X.** (2008), Application of a group search optimization based artificial neural network to machine condition monitoring. *In 2008 IEEE International Conference on Emerging Technologies and Factory Automation*, 1260–1266. IEEE.

4.**Jung S. H.**, (2003), Queen-bee evolution for genetic algorithms, *Electronics letters*, **39**(6), 575–576.

5.**Khabou H.**, **Souissi M.**, and **Aitouche A.**, (2020), MPPT implementation on boost converter by using T–S fuzzy method, *Mathematics and Computers in Simulation*, **167**, 119–134.

6.**Khedr S. F. M.**, **Ammar M.**, and **Hassan M. A. M.** (2013), Multi objective genetic algorithm controller's tuning for non-linear automatic voltage regulator. *In 2013 International Conference on Control, Decision and Information Technologies (CoDIT)*, 857–863. IEEE.

7.**Krohling R. A.** and**Rey J. P.**, (2001), Design of optimal disturbance rejection PID controllers using genetic algorithms, *IEEE Transactions on evolutionary computation*, **5**(1), 78–82.

8.**Guo L.**, **Hung J. Y.**, and **Nelms R. M.**, (2009), Evaluation of DSP-based PID and fuzzy controllers for DC–DC converters, *IEEE transactions on industrial electronics*, **56**(6), 2237–2248.

9.**Kumar K. R.** and **Jeevananthan S.** (2011), Design of a Hybrid Posicast Control for a DC-DC Boost Converter Operated in Continuous Conduction Mode. *In 2011 International Conference on Emerging Trends in Electrical and Computer Technology*, 240–248. IEEE.

10.**Kim T.-H.**, **Maruta I.**, and **Sugie T.**, (2008), Robust PID controller tuning based on the constrained particle swarm optimization, *Automatica*, **44**(4), 1104–1110.

11.**Koutroulis E.** and **Kalaitzakis K.**, (2006), Design of a maximum power tracking system for wind-energy-conversion applications, *IEEE transactions on industrial electronics*, **53**(2), 486–494.

12.**Lei Z.**, (2004), Simplex method based optimal design of PID controller, *Information and Control*, **33**(3), 376–379.

13.**Gohar Ali H.**, **VilanovaArbos R.**, **Herrera J.**, **Tobón A.**, and **Peláez-Restrepo J.**, (2020), Non-linear sliding mode controller for photovoltaic panels with maximum power point tracking, *Processes*, **8**(1), 108.

14.**Manjula K.** and **Bhuvaneswari S.** (2016), Analysis of harmonic reduction in closed loop

controlled double inductor boost converter fed DC motor with EMI input filter. *In 2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, 832–837. IEEE.

15.**Guo L.**, **Hung J. Y.**, and **Nelms R. M.**, (2009), Evaluation of DSP-based PID and fuzzy controllers for DC–DC converters, *IEEE transactions on industrial electronics*, **56**(6),2237–2248.