



# Implementation of Soft Computing Techniques for Luo Converter

Dr. B. Achiammal<sup>1</sup>, Dr. M. Dhinakaran<sup>2</sup>, P. Deepthi<sup>3</sup>

<sup>1</sup>Assistant Professor, Department of Electronics and Instrumentation Engineering,  
Government College of Technology, Coimbatore, Tamilnadu

<sup>2,3</sup>Associate Professor, Department of Electronics and Communication Engineering,  
Government College of Engineering, Salem, Tamilnadu

**Abstract:** The Luo converters are used for converting positive source voltage to better load voltage in the main industrial equipment. Dynamic behaviour of these converters is highly nonlinear because of their changing time and design. By changing PI parameters using optimised techniques, the efficiency of the Luo converters is improved. Genetic algorithm (GA) is developed for PI optimization in this work and also the Modified Bacterial Foraging (MBF) algorithm. The results show that MBF-PI and GA-PI controller performance is higher to the classic ZN-PI controller performance.

**Keywords:** DC-DC converter, positive elementary Luo converter, Genetic algorithm, Modified bacterial algorithm.

## 1. INTRODUCTION

DC-DC converters are configured with big industrial equipment. DC voltage input to DC voltage output vector is converted by these DC-DC converters. The sub-set of DC-DC converters is a positive output Luo conversion, with these output positive elementary Luo conversion devices (Fig. 1). DC-DC conversions are performed from DC voltage positive input to DC voltage positive output.

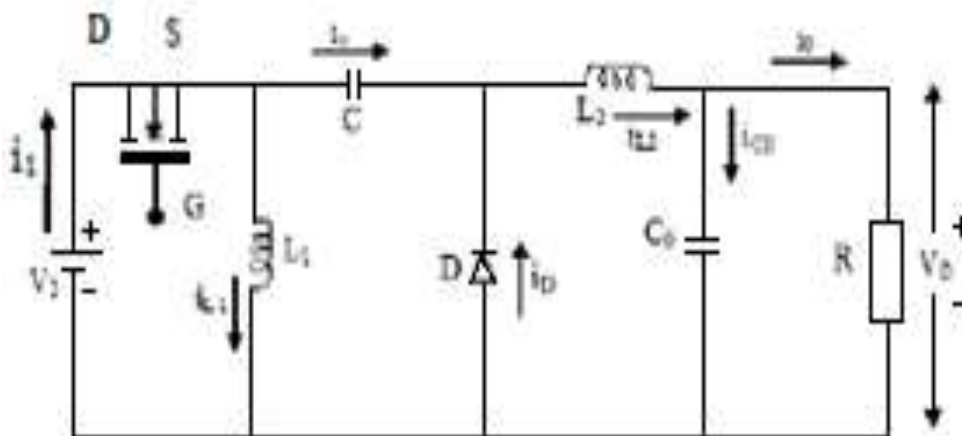


Fig.1 Luo converter Positive output elementary

These Luo Converter Positive outputs have complicated non-linear modes with constraint variance difficult. Due to their time varying existence, PI controllers do not respond satisfactorily to these converters. Via optimized techniques, these positive Luo converters are controlled. The work is carried out on the basis of the integral squared error (ISE) and the integral absolute error (IAE) by the PI controller, GA PI controller and the MBF PI controller for Luo positive conversion devices and the output comparison of the controllers.



2. LUO CONVERTER DESIGN

Luo converter voltage transfer ratio of the positive output portion is  $(k/(1-k))$ . The turn on and off modes in the positive output feature Luo converter are improved by a State Space approach (Fig.2 and Fig.3) Luo converters are also developed.

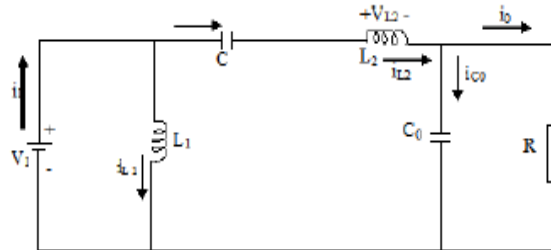


Fig. 2 Luo converter Positive output elementary – mode 1

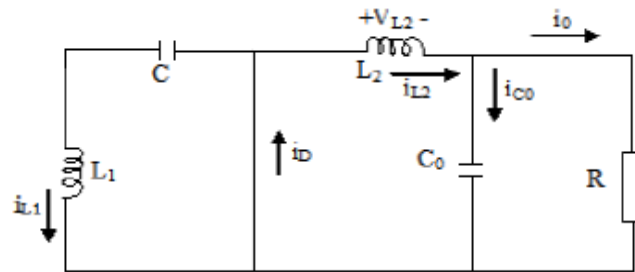


Fig. 3 Luo converter Positive output elementary – mode 2

During a single switching cycle T these two models are averaged by the average state-space technology. The state variables are:

$$X_1=I_{L1}, X_2=I_{L2}, X_3=V_O, X_4=V_{C0}..... (1)$$

A1 and A2 device matrices are obtained using the above state variables, B1, B2 are input matrices as well as C1, C2 are output matrices.

3. TUNING METHOD OF PI CONTROLLER

3.1. PI Controller

The PI controller is widely known for enhancing dynamic response and reducing or removing steady state errors. The integral controller adds an initial pole, which increases the device form by one and lowers the steady state error.

The PI controller transfer function is given by:

$$K_p + \frac{K_i}{s} = \frac{k_p s + k_i}{s} \quad (2)$$

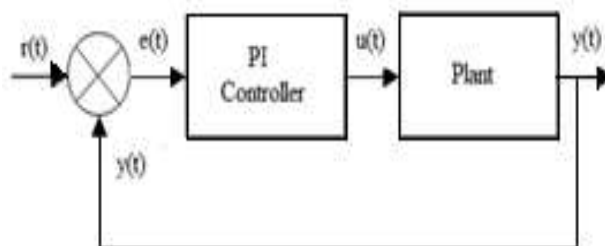


Fig. 4 Schematic diagram of PI controller

The PI control works with the schematic shown in Fig.4 in a closed loop method. The variable (e), the variation among the target input value (r) and the definite output (y), represents the fault in monitoring. The PI controller is given



this error signal (e) with controller is computing the integral signal of that error. The proportional gain (Kp) of the error with integral gain (Ki) of the error plus equals that of the signal (u) that just passes the operator.

$$u = k_p e + k_i \int e dt \quad (3)$$

By sending a signal (u), the new output(y) to the plant. The new signal error(s) was observed when a new output (y) was returned to the sensor. This latest signal error is taken by the controller and measured as a complete gain.

### 3.2. Genetic algorithm Based PI Controller

The Holland first suggested the fundamental principles of GA. This system was motivated by Natural Selection method, a biological method that is likely to gain a stronger individual in a competitive environment. To solve extremely complex issues, GA takes a simple comparison of such normal growth to achieve overall optimisation. It assumes that a subject can resolve a problem and can be described by a position of parameters. It is understood that these parameters are the genes of a chromosome with a combined set of values that can be ordered. The representation of variables is described by the training system. Depending on the application details, the variables may be identified by real and binary numbers or previous types. Her range is normally specified by the issue, the search space.

A variety of problems have been successfully solved, such as: salesmen travelling graph partition trouble, harmonic analysis, power electronics, etc. Machine learning, dynamic control systems with system also adaptive manage has also been implemented. It has been used. Initially, an initial population of chromosomes is generated randomly. Present solution of the dilemma is the chromosomes. Then, all chromosomes have their fitness values measured by decoding the objective function. Thus, a collection of the top chromosomes is chosen during the selection method based on each individual's condition. Crossover and mutation, genetic values are used to boost the future generation solution in this 'surviving' population. In order to produce offspring, Crossover is a recombination operator that mixes two chromosomes into subparts. In direct to attain even improved solutions; the operators extract similar properties from various chromosomes. Mutation is a chromosome operator that creates changes. This procedure often takes place with a little possibility. In the event of binary coding, it changes a bit's arbitrarily value. It variable the whole value of a chromosome in real coding. The search space is explored through the mutation operator in search of better points. Until individuals collide or another stop condition is met, the process proceeds.

#### 3.2.1. Genetic operator

Genetic operators are used in every generation to establish a new population for chosen individuals from the existing population. Three major reproductive, crossover and mutation genetic operators are typically working. By using various probabilities for using these operators, the speed of convergence may be controlled. Crossover and mutation operators must be carefully prepared, as their preference greatly contributes to the performance of the whole genetic algorithm.

#### 3.2.2. Reproduction

A selection of individuals from the present inhabitants can only be made by changing them without modification. There is also the opportunity for new people to choose solutions already created. A variety of other selection methods are available and each procedure must be selected by the user. A method of selection is depending upon the equal concept, i.e. a greater likelihood for selection of a fitter chromosome. Four general techniques for identify are:

1. Roulette Wheel technique
2. Stochastic common sampling
3. Normalized geometric technique
4. Tournament technique

Roulette Wheel technique is used in this work.

#### 3.2.3. Crossover

As offspring of two parents (i.e. a binary crossover operator), new individuals are normally formed. One or more so-called crossover points (usually random) are particular in each parent's chromosome, at the same location. The crossover sections are then splitting among the parents. The crossover points. In this mode the inhabitants are the descendants. There are many crossover styles beyond one point and several crossover points. In later stages of growth, it is more important to keep people in touch, and it is therefore a better thought to employ an adaptively evolving crossover speed: advanced rates in early stages and a minor rate at the end of GA. An offspring, a wise linear combination of parents, is generated by arithmetic crossover. Often, various forms of crossover may also be used at various stages of development.

#### 3.2.4. Mutation

By changing one selected individual, a new group is created. The changes may include modifying one or more values in the frame or adding / deleting sections of the present. Mutation is a source of changeability in GA, and a mutation rate which is too high results in lower quality, except for the specific problems.



This mutation must consequently be implemented carefully since it is an arbitrary search operator; or else the algorithm is not anything more than a random search at high rates of mutation. In addition, various mutation operators may be used in different levels. At the outset, mutation operators may be favoured leading to larger jumps in the search space. Later, a mutation operator that leads to minor changes in the search room could be promoted when the solution is near.

### 3.3. Advanced Technique (MBFOA)

MBFOA emulates the bacteria E filtration process. Each bacterium conducts a chemotactic  $N_c$  times within a period called generation (P). Later than all bacteria during their chemotactic cycles, they replicate the greatest bacteria, removing the worst and creating new bacteria at casual. It makes a process in which the best bacteria are transferred to the following generations amongst all chemotactic steps.

The four processes of MBFOA merge the chemotaxis with swarming into one loop to make it simpler for reproductive to disposable dispersal processes. For a solution called a bacterium and its location, MBFOA uses real-life encoding as,

$$\theta^i(j, P) = \bar{x}_i \quad (4)$$

Where,

i, j and p is the number of bacterium, chemotactic loop and cycle numbers.

#### 3.3.1. Chemotaxis

The chemotactic cycle present of a tumble (random search direction) and bacterial motions in the search region to detect nutrients. The movement of the attractor is applied twice in a chemical loop while the rest of the steps are carried out with tumble-swim movement. The regulations are used to detect the worst bacterium in a swarm in the chemotactic loop in the reproduction stage and in the disposal. The chemotactic process consists of the movement of tumble-swim in the present swarm by bacteria.

The tumble movement is represented by

$$\phi(i) = \frac{\Delta(i)}{\sqrt{\Delta(i)^T \Delta(i)}} \quad (5)$$

Where,

In interval [-1, 1],  $\Delta_i$  is a randomly generated vector of m size with elements. Each bacterium i change its location after that, by swimming and is shown as

$$\theta^i(j+1, P) = \theta^i(j, P) + B(i)\phi(i) \quad (6)$$

Where,

Bacterium i (new solution)'s new position at chemotactic step j + 1,  $\theta^i(j, P)$  is actually the position in chemotactic step j of bacteria i. In MBFOA vector b(i) phase values are determined with equation 5, taking into account the relevant boundaries of each design variable.

$$B(i)_k = S * \left( \frac{\Delta x_k}{\sqrt{m}} \right), k = 1, \dots, m \quad (7)$$

Where,

$\Delta x_k$  is the variation among upper with lower limits for design parameter  $x_k$ :  $U_k - L_k$ , m is an indication of the amount of propose variables with S is an indicated percentage of the value by the bacteria as step size. In order to allow each bacterium in the swarm to behind the bacterium in the most safe search area MBFOA implements an attractor movement, which is described as an

$$\theta^i(j+1, P) = \theta^i(j, P) + \beta \left( \theta^B(P) - \theta^i(j, P) \right) \quad (8)$$

Where,

$\theta^i(j+1, P)$  is the recent location of bacterium i,  $\theta^i(j, P)$  is the new location of bacteria I and  $\beta$  defines the nearness of bacterium (j) to the location of best bacterium  $\theta^B(P)$  in the existing position of bacterium i,  $\beta$  define the new location of the recent bacterium i to the location of suitable bacterium in the swarm at generation P.

The movement of the attractor occurs twice in a chemotactic loop, while the movement of the tumble swim is performed in the remaining steps. The goal is to foster stability among research and exploitation.

#### 3.3.2. Reproduction

The method of reproduction presents of the swarm sorting according to the restriction technique. For the next generation, the first half of the population is being repeated to stay the same. It involves removing the second worst bacterium from the current population with rearrange it with a another of the best bacterium while removing and replacing the worst bacteria with one created randomly.



### 3.3.3.. Elimination and Dispersal

Removal-dispersal extracts only the worst bacterium and replaces it by adding a new randomly generated bacterium. On the end of the generation cycle, single reproduction steps with a single disposal move are completed. The removal – dispersal stage is simpler since only the worst of the population bacteria is removed.

## 4. PERFORMANCE INDICES

The objective function is based upon the error criterion. The controller's output is better measured by error criterion. In this work the output of the controller is measured in terms of ISE and IAE.

$$ISE = \int_0^t e^2 dt \quad (10)$$

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

The error is weighted with time by the ISE and IAE, thus minimising error values closer to zero.

## 5. SIMULATION RESULTS

Table 1 displays the circuit parameters of the Luo elementary output positive. The parameter device values of the traditional controller ZN-PI, GA-PI and MBF-PI are acquired. A matlab m file with the specifications provided in Tables 2 and 3 is created based on both the GA and MBF algorithms. Figures 5, 6 and 7 demonstrate the responses of the Luo elemental positive output converter using the traditional control ZN-PI, GA-PI and MBF-PI.

The statistics show that in contrast to the traditional PI controller and GA-PI controllers, the overflow, ISE and IAE values are significantly reduced. Table 4 demonstrates performance analysis with traditional ZN-PI, GA-PI, and MBF-PI controllers of positive elementary Luo performance.

**TABLE 1: LUO CONVERTER DESIGN PARAMETERS OF POSITIVE OUTPUT ELEMENTARY**

Parameter	Symbol	Value
Input Voltage	$V_{in}$	10 V
Output Voltage	$V_o$	30V
Inductor	L	100 $\mu$ H
Capacitor	C	5 $\mu$ F
Load resistor	R	10 $\Omega$
Duty ratio	D	0.1-0.9

**Table: 2 GA design values**

Parameter	Value
Population size	50
Iteration	100
Crossover probability	>0.8
Mutation probability	<0.05

**TABLE 3: MBF Parameter values**

Parameter	Value
Number of bacteria ( $S_b$ )	50
Chemotactic loop limit ( $N_e$ )	10
Number of generations ( $G_{MAX}$ )	100
Number of bacteria for reproduction ( $S_r$ )	25
Step size R	2.1e <sup>-3</sup>
Scaling factor $\beta$	0.44

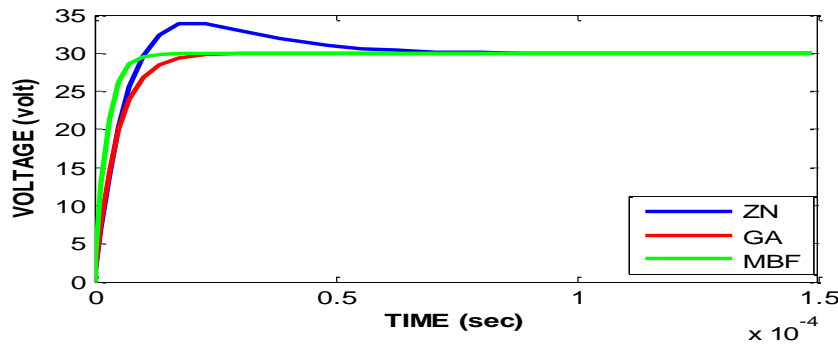


FIG.5 CLOSED LOOP RESPONSES OF CONVENTIONAL ZN-PI, GA-PI AND MBF-PI CONTROLLERS

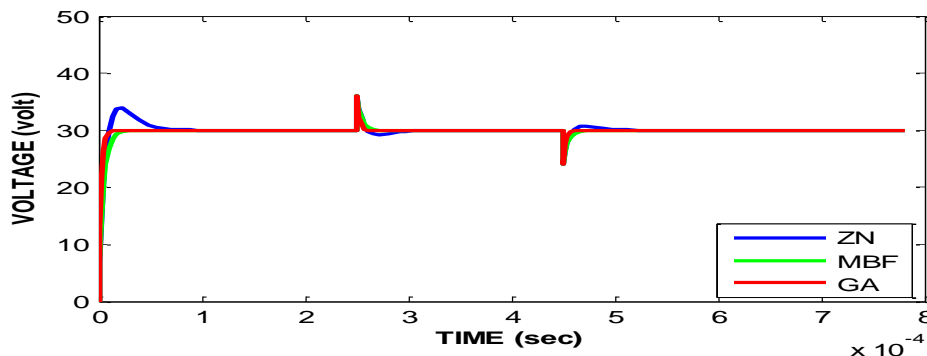


FIG.6 Comparison of closed loop responses of Positive output Elementary Luo converter under sudden load disturbance from 10V– 12V(20%) at 0.25e-3 sec and 10V– 8V(20%) at 0.45e-3 sec

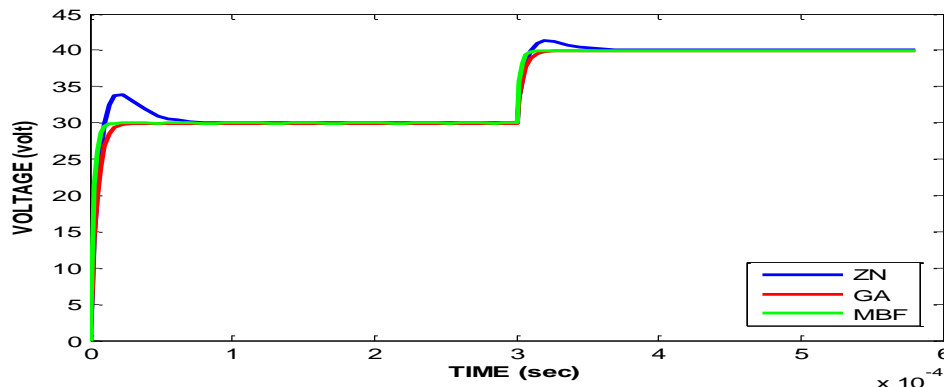


FIG.7 Comparison of servo responses of Positive output Elementary Luo converter with sudden increase in reference voltage from 30V – 40V at 0.3e-3 sec

Table 4 Performance Analysis

Parameters	PI	GA- PI	MBF-PI
Rise Time	7.48e-6	9.91e-6	5.23e-6
Peak	1.13	0.998	0.999
Settling Time ( t <sub>s</sub> )	5.52e-5	1.79e-5	9.38e-6
Peak Overshoot	13.4	0	0
ISE	0.0022	0.0020	0.00106
IAE	0.0006	0.00018	9.831e-5
K <sub>P</sub>	6.32	1.326e+4	52.59
K <sub>P</sub>	20.15	12.98	27.75



### CONCLUSION

Genetic algorithm (GA-PI) and Modified Bacterial Forging Algorithm (MBF-PI) have been built in this work to change the parameters for the PI controller that governs the efficiency of the positive Luo elementary output converter. The simulations show that both line and load disturbances are satisfactorily dismissed by the PI control with GA algorithm and MBF algorithm. The results also show that the MBF-PI controller provides a smooth response to reference tracking and retains a positive output voltage elementary to the target voltage from the Luo converter.

### REFERENCES

1. Luo, F.L., "Positive output Luo converter voltage lift Technique", IEEE processdings, 146(4), July 1999, pp.415-432.
2. Yıldız, Y.E., Altun, O. "Hybrid achievement oriented computational chemotaxis in bacterial foraging optimization: a comparative study on numerical benchmark". Soft. Comput. Vol.19 (12), 2015, pp. 3647-3663.
3. Niu, B., Wang, C., Liu, J., Gan, J., Yuan, L. "Improved bacterial foraging optimization algorithm with information communication mechanism for nurse scheduling". LNCS, vol.9226, Springer, Heidelberg 2015, pp.701-707.
4. E.Mezura-Montes and B.Hernandez-Ocana, "Modified bacterial foraging optimization for engineering design", in proceedings of the Artificial Neural Networks in Engineering Conference (ANNIE 2009), Vol.19, 2009, pp.357-364.
5. Janardan Nanda, S. Mishra and Lalit Chandra Saikia, "Maiden Application of Bacterial Foraging-Based Optimization Technique in Multiarea Automatic Generation Control", IEEE Transactions on Power systems, Vol.24, No 2, 2009.
6. J.F.Huang, F.B.Dong, "Modelling and Control on Isolated DC-DC Converter", Power Electronics, vol.44, 2015, pp.87-89
7. BetaniaHernandez-Ocana, Ma. Del Pilar Pozos - Parra and Efren Mezura-Montes Improved Modified Bacterial Foraging Optimization Algorithm to Solve Constrained Numerical Optimization Problems, Applied mathematics and Information Science, Vol. 10, No. 2, 2016, pp. 607-622
8. T O.Mahony, C J Downing and K Fatla, "Genetic Algorithm for PID Parameter Optimization: Minimizing Error Criteria" method manage and Instrumentation 2000 26-28 July 2000, University of Stracthclyde, pp.148~153
9. Houck, C.R., Joines, J.A., and Kay, M.G.: 'A genetic algorithm for function optimization: a Matlab implementation'. Technical Report NCSU-IE TR 95-09, 1995, North Carolina State University
10. J.You, S.B.Kang, "Generalized State Space Averaging based PWM Rectifier Modeling", Electrical Measurement & Instrumentation Vo1.46, 2009, pp.67-70.
11. Dr.N.Dhanasekar ,Dr.R.Kayalvizhi "Design and simulation of PI control for positive output triple lift Luo converter " International journal of modern engineering research . Vo1.2, issue 6. 2012, pp.4186-4188.