Moth Flame Optimisation Algorithm for Control of LUO Converter

N. Nachammai, R. Kayalvizhi

Abstract: Because of the effects of the parasitic elements, the output voltage and power transfer efficiency of all DC-DC converters are restricted. In order to eliminate the limitations caused by parasitic elements, the voltage lift technique is successfully applied to DC-DC converters resulting in a new series called Luo converters. Linear control methods ensure stability and good control only in small vicinity around the operating point. These classical controllers are designed using mathematical models by linearising non-linearities around the nominal operating point. Since these controllers are also sensitive to the operating points and parameters variations, a high degree of accuracy cannot be guaranteed from them. To ensure that the controllers work well in large signal conditions and to enhance their dynamic responses, intelligent methods using fuzzy technique is suggested. The performance of a fuzzy logic controller depends on its control rules and membership functions. Hence, it is very important to adjust these parameters to the process to be controlled. A method is presented for tuning fuzzy control rules by Moth Flame Optimization (MFO) algorithm to make the fuzzy logic control systems behave as closely as possible to the operator or expert behavior in a control process. The tuning method fits the membership functions of the fuzzy rules given by the experts with the inference system and the defuzzification strategy selected, obtaining high-performance membership functions by minimizing an error function. Moth-Flame Optimization (MFO) algorithm is one of the newest bio-inspired optimization techniques in which the main inspiration of this optimizer is the navigation method of moths in nature called transverse orientation. MFO has a fast convergence rate due to use of roulette wheel selection method. Moth-Flame Optimizer (MFO) is used to control the LUO converter. MFO-Fuzzy is used to search the fuzzy rules and membership values to achieve minimum ISE, ITAE, settling time and peak overshoot. The proposed method is compared with fuzzy controller. Simulation results prove that the MFO algorithm is very competitive and achieves a high accuracy.

Keywords: Moth Flame Optimisation Algorithm, Fuzzy Logic Controller, Positive Output Elementary LUO Converter.

I. INTRODUCTION

All the modern electronic systems require high quality, small, light weight, cheap, reliable and efficient power supplies. DC-DC converters are widely used in computer peripheral power supplies, car auxiliary power supplies, servomotor drives and medical equipments. Because of the effects of the parasitic elements, the output voltage and power transfer efficiency of such converters are restricted.

In order to eliminate these limitations, the voltage lift technique has been successfully applied to DC–DC converters resulting in a new series named as Luo converters [1]. Recently fuzzy control technique has been applied to many industrial processes. Fuzzy logic controller (FLC) are rule-based systems which are useful complex ill-defined processes, especially those which can be controlled by a skilled human operator without knowledge of their underlying dynamics. The essential part of the FLC system is a set of fuzzy control rules related by means of a fuzzy implication and compositional rule of inference. The correct choice of the membership functions of the linguistic label set plays an essential role in the performance of an FLC, it being difficult to represent the experts knowledge perfectly by linguistic control rules. An FLC contains a number of sets of parameter that can be altered to modify the controller performance. A tuning method for obtaining high performance fuzzy control rules by means of MFO algorithm has been proposed. Moth-flame optimization MFO algorithm has been developed by seyedalimirjalili in 2015 is a nature inspired method [2] from navigating mechanism of moths in nature called transverse orientation, which has not received yet much attention in the control of power electronic converters. MFO algorithm has two main advantages [3] such as MFO avoids the local optima problem, while many other optimization algorithms such as Genetic Algorithm (GA) still face this problem and MFO has high exploration and exploitation which may assist to outperform other algorithms.

II. POSITIVE OUTPUT ELEMENTARY LUO CONVERTER

Positive output Luo converters perform the conversion from positive DC input voltage to positive DC output voltage. The elementary Luo converters perform step-down or step-up DC-DC conversion.

Fig. 1 Positive Output Elementary Luo Converter
The elementary circuit is shown in Fig.1 Switch S is a N-channel power MOSFET (NMOS) device. It is driven by a PWM switching signal with repeating frequency \( f_s \) and duty ratio \( d \). The switching period is \( T = 1/f_s \) so that the switch-on period is \( dT \) and the switch-off period is \( (1-d)T \). The load \( R \) is resistive. \( R = V_o / I_o \) where \( V_o \) and \( I_o \) are the average output voltage and current. Table I gives the circuit parameters of positive output elementary Luo converter.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Voltage</td>
<td>( V_{in} )</td>
<td>10 V</td>
</tr>
<tr>
<td>Output Voltage</td>
<td>( V_o )</td>
<td>20 V</td>
</tr>
<tr>
<td>Inductor</td>
<td>( L )</td>
<td>100( \mu )H</td>
</tr>
<tr>
<td>Capacitor</td>
<td>( C )</td>
<td>5( \mu )F</td>
</tr>
<tr>
<td>Load resistor</td>
<td>( R )</td>
<td>10( \Omega )</td>
</tr>
<tr>
<td>Duty ratio</td>
<td>( D )</td>
<td>0.67</td>
</tr>
</tbody>
</table>

The elementary circuit is shown in Fig.1. Switch S is a N-channel power MOSFET (NMOS) device. It is driven by a PWM switching signal with repeating frequency \( f_s \) and duty ratio \( d \). The switching period is \( T = 1/f_s \) so that the switch-on period is \( dT \) and the switch-off period is \( (1-d)T \). The load \( R \) is resistive. \( R = V_o / I_o \) where \( V_o \) and \( I_o \) are the average output voltage and current. Table I gives the circuit parameters of positive output elementary Luo converter.

### III. DESIGN OF FUZZY LOGIC CONTROLLER

The control action is determined in a fuzzy logic controller from the evaluation of a set of simple linguistic rules. The development of the rules requires a thorough understanding of the process to be controlled [11] but it does not require a mathematical model of the system.

**A. Identification of Inputs and Output**

The inputs to the fuzzy controller are the error in output voltage \( e \) and the change of error \( ce \) which are defined as

\[
\begin{align*}
    e &= V_{\text{ref}} - V_o \\
    ce &= e_k - e_{k-1}
\end{align*}
\]

where \( V_o \) is the present output voltage, \( V_{\text{ref}} \) is the reference or desired output voltage and subscript \( k \) denotes values at the sampling instants.

**B. Fuzzification of Inputs and Output**

The inputs and output of the controller are not quantized in the classical sense that each input or output is assigned a “membership grade” \( \mu \) to each fuzzy set. Mamdani type input and output membership functions are used for control of Luo converters. In the present work, seven triangular fuzzy sets are chosen as shown in Fig. 3 and Fig. 4 and are defined by the library of fuzzy set values such as NB - Negative Big, NM - Negative Medium, NS - Negative Small, ZE - Zero, PS - Positive small, PM - Positive Medium and PB - Positive Big for the error \( e \), change in error \( ce \) and for the change in duty cycle \( \delta d_k \).

### Table II: Rule base for FLC

<table>
<thead>
<tr>
<th>( ce )</th>
<th>NB</th>
<th>NM</th>
<th>NS</th>
<th>ZE</th>
<th>PS</th>
<th>PM</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>( e )</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
</tr>
<tr>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
</tr>
<tr>
<td>NM</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
</tr>
<tr>
<td>NS</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
</tr>
<tr>
<td>ZE</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
</tr>
<tr>
<td>PS</td>
<td>NM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
</tr>
<tr>
<td>PM</td>
<td>NS</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
</tr>
<tr>
<td>PB</td>
<td>ZE</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
</tr>
</tbody>
</table>

**C. Defuzzification**

In the defuzzification operation, a logical sum of the inference result from each of the four rules is performed. This logical sum is the fuzzy representation of change in duty cycle inferred by the \( i^{th} \) rule.
cycle is calculated in this work using the center of gravity method

$$\delta d_k = \frac{\sum^4_{i=1} w_i m_i}{\sum^4_{i=1} w_i}$$  \hspace{1cm} (3)$$

IV. OVERVIEW OF MFO ALGORITHM

MFO algorithm based on the navigation method of moths in nature. Moths fly in night by maintaining a fixed angle with respect to the moon, a very effective mechanism for travelling in a straight line for long distances[4]. But moths fly spirally around the light, because the light source is very near. The moth eventually converges towards the light. Moths and flames are both solutions. The moths are actual search agents, whereas flames are the best position for moths that obtains so far. Therefore, each moth searches around a flame and updates it in case of finding a better solution. Different steps involved in the proposed algorithm are as follows[5]:

The set of moths represented in a matrix form is ,

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & \cdots & m_{1,d} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ m_{n,1} & m_{n,2} & \cdots & \cdots & m_{n,d} \end{bmatrix}$$ \hspace{1cm} (4)$$

Where, \( n \) is the number of moths and \( d \) is the number of variables (dimension). For all the moths it is assumed that there is array for storing the corresponding fitness values as

$$OM = \begin{bmatrix} OM_1 \\ OM_2 \\ \vdots \\ OM_n \end{bmatrix}$$ \hspace{1cm} (5)$$

A matrix similar to the moth matrix is considered as

$$F = \begin{bmatrix} F_{1,1} & F_{1,2} & \cdots & \cdots & F_{1,d} \\ F_{2,1} & F_{2,2} & \cdots & \cdots & F_{2,d} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ F_{n,1} & F_{n,2} & \cdots & \cdots & F_{n,d} \end{bmatrix}$$ \hspace{1cm} (6)$$

The dimension of \( M \) and \( F \) arrays are equal. For the flames, it is also assumed that there is an array for storing the corresponding fitness value as

$$OF = \begin{bmatrix} OF_1 \\ OF_2 \\ \vdots \\ OF_n \end{bmatrix}$$ \hspace{1cm} (7)$$

The MFO algorithm is a three-tuple that approximates the global optimal of the optimization problems and defined as

$$MFO= (I, P, T)$$

\( l \) is a function that generates a random population of moths and corresponding fitness value. The methodical model of this function is

$$l : \phi \rightarrow \{ M, OM \}$$ \hspace{1cm} (8)$$

The \( P \) function, which is the main function, moves the moths around the search space. This function received the matrix of \( M \) and returns is updated one eventually.

$$P: M \rightarrow M$$ \hspace{1cm} (9)$$

The \( F \) function returns true if the termination criterion is satisfied and false if the termination criterion is not satisfied.

$$T: M \rightarrow \{ true, false \}$$ \hspace{1cm} (10)$$

The matrixes which define the upper and lower bounds of the variables are as

$$ub = [ub_1, ub_2, ub_3, \ldots, ub_{n-1}, ub_n]$$ \hspace{1cm} (11)$$

Where \( ub_i \) indicates the upper bound of the \( i \)th variable.

$$lb = [lb_1, lb_2, lb_3, \ldots, lb_{n-1}, lb_n]$$ \hspace{1cm} (12)$$

Where \( lb_i \) indicates the lower bound of the \( i \)th variable.

The position of each moth is updated with respect to a flame using the below equation,

$$M = S(M, F_i) \rightarrow M$$ \hspace{1cm} (13)$$

Where \( M_i \) indicate the \( i \)th moth, \( F_j \) indicates the \( j \)th flame, and \( S \) is the spiral function.

A logarithmic spiral is defined for the MFO algorithm [6] as

$$S(M, F_j) = D_i e^{bt} \cos(2\pi t) + F_j$$ \hspace{1cm} (14)$$

Where \( D_i \) indicates the distance of the \( i \)th moth for the \( j \)th flame, \( b \) is a constant for defining the shape of the logarithmic spiral, and \( t \) is a random number in [-1, 1].

\( D_i \) is calculated as

$$D_i = \left| F_j - M_i \right|$$ \hspace{1cm} (15)$$
The spiral equation allows a moth to fly “around” a flame and not necessarily in the space between them.[7] The best solutions obtained so far are considered as the flames and stored in $F$ matrix. The number of flames is decreased adaptively over the course of iterations are

$$\text{flames no} = \text{round}\left(N - I \times \frac{N - I}{T}\right)$$

Where, $I$ is the current number of iteration, $N$ is the maximum number of flames and $T$ indicates the maximum number of iterations[8]. In the initial steps of iterations there is $N$ number of flames. The moths update their positions only with respect to the best flame in the final steps of iterations. The gradual decrement in number of flames balances exploration of the search space [9].

### PSEUDO-CODE OF MFO [10]

1. Initialize the position of moths
2. While (Iteration < Max Iteration)
3. Update flame no using (i)
4. OM = FitnessFunction($M_i$)
5. $I$ = iteration + 1
6. $F$ = sort($M_i$)
7. $OF$ = $\text{sort (OM)}$
8. else
9. $F$ = sort($M_i$)
10. $OF$ = $\text{sort (OM)}$
11. end
12. for $i = 1 : n$
13. for $j = 1 : d$
14. Update $r$ and $t$
15. Calculate $D$ using (7) with respect to the corresponding moth
16. Update $M(i,j)$ using (5) and (6) with respect to the corresponding moth
17. end
18. end

### VII. SIMULATION RESULTS AND DISCUSSION

Different objectives considered in this work are minimization of ISE, ITAE, settling time and peak overshoot. This method is simulated and tested on the LUO Converter. Figs. 5-7 show the simulated output voltage of the Luo converters with MFO-fuzzy controller under line and load disturbances. The above responses are compared with corresponding responses with fuzzy controller. The MFO-fuzzy controller implemented regulates the output voltage within a maximum of 21 ms and 20 ms after line and load disturbances respectively. Fig.5 display the start up transient in the output voltage of positive output elementary Luo converter. The start up transient exists for a maximum of 17.5 ms. From the Table IV, it is also observed that the peak overshoot for MFO-fuzzy (3.7%) is less than the fuzzy controller (25.9%) with sudden increase in the line disturbance.

The ISE and IAE weight the error with time and hence minimize the error values near to zero.

**Fig.5 Closed loop responses of Fuzzy and MFO-Fuzzy Controllers**

**Fig.6 Closed loop responses of Fuzzy and MFO-Fuzzy controllers with sudden line disturbance (25%) at 0.04 sec and at 0.07 sec.**
MFO–FUZZY vs FUZZY with load disturbances

Fig. 7 Closed loop response of Fuzzy and MFO-Fuzzy controllers with sudden load disturbance (25%) at 0.04 sec and at 0.07 sec.

Table III: Parameter Settings

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of moths and flames(Agents no)</td>
<td>50</td>
</tr>
<tr>
<td>Number of variables(dim)</td>
<td>7</td>
</tr>
<tr>
<td>Maximum iteration number(Max. iteration)</td>
<td>500</td>
</tr>
<tr>
<td>Lower bound (lb)</td>
<td>Min value of control variables</td>
</tr>
<tr>
<td>Upper bound (ub)</td>
<td>Max value of control variables</td>
</tr>
</tbody>
</table>

Table IV. Performance Evaluation of Positive Output Elementary LUO Converter

VIII. CONCLUSION

A new nature inspired algorithm named Moth-Flame Optimization (MFO) algorithm has been used for tuning the fuzzy controller for LUO converter. The performance of MFO – Fuzzy and Fuzzy controllers have been investigated. The results show that MFO tuned fuzzy controller provides better dynamic response in terms of less settling time and less peak overshoot. The performance of MFO algorithm misevaluated by comparing with fuzzy controller for LUO converter. The simulation results infer that the line and load disturbance reject in faster rate and settles very quickly, which shows superiority of MFO tuned fuzzy controller over fuzzy controller.

REFERENCES


N. Nachammar received her B.E (Electronics and Instrumentation) from Annamalai University in 1998 and M.Tech (CEDT) from IISC, Bangalore in 1996. She is presently working as an Associate Professor in Department of Electronics and Instrumentation Engineering, Annamalai University, where she has put in a total service of 24 years. She is presently pursing PhD in the Department of Electronics and Instrumentation Engineering, Annamalai University. Her areas of interest are modelling, simulation and implementation of optimization techniques for control of power electronic converters.

Published By: Blue Eyes Intelligence Engineering & Sciences Publication Pvt. Ltd.
Dr. R. KAYALVIZHI has obtained B.E (Electronics and Instrumentation), M.E (Power Systems) and PhD in Instrumentation Engineering from Annamalai university in 1984,1988 and 2007 respectively. She is currently working as a Professor in the Department of Electronics and Instrumentation Engineering at Annamalai University where she has put in 32 years of service. She produced 5 PhDs and presently guiding 5 PhD scholars. Her research papers 25 has been presented in the International and National conferences. She has 35 publications in National Journals and 30 in International Journals. Her areas of research interest include Power Electronics, Power Systems and Digital Image Processing. She is a life member of Indian society for Technical Education.