PARAMETER INVESTIGATION AND ANALYSIS FOR ELITE OPPOSITION BACTERIAL FORAGING OPTIMIZATION ALGORITHM

*Maliki, D[1], Muazu, M. B[2], Kolo, J.G[3], & Olaniyi, O. M[4]
[1,4]Department of Computer Engineering, Federal University of Technology, PMB 65 Minna Niger State, Nigeria
[2]Department of Computer Engineering, Ahmadu Bello University Zaria Kaduna State, Nigeria
[3]Department of Electrical and Electronics, Federal University of Technology, PMB 65 Minna Niger State, Nigeria

*Corresponding author email: danlami.maliki@futminna.edu.ng, +2348062502102

ABSTRACT
The investigation and analysis of algorithm parameters is an important task in most of the global optimization techniques. However, finding the best set of parameter value for the optimum performance of an algorithm still remain a challenging task in a modified Bacteria Foraging Optimization Algorithm (BFOA) since most of the existing research focuses on the application of the algorithm and likewise it benchmarking with the global test function. The Elite Opposition Bacterial Foraging Optimization Algorithm (EOBFOA) is a modified nature inspired optimization algorithm from BFOA which focuses on the generation of an elite solution from the opposition solution for an optimization process. This research is focused on the investigation of such parameters population size, probability of elimination dispersal, step size and number of chemotaxis so as to determine the extent to which they affect the optimal solution from the EOBFOA with respect to global minimum or least minimum standard deviation. From the results obtained, it was observed that the global minimum in EOBFOA depend on the exploitation ability of the bacteria in the search space.

Keywords: BFOA, EOBFOA, elite solution, opposition solution, parameters.

1 INTRODUCTION

Every organism in life is faced with the problem of survival needs. The needs for survival ranges from the availability of water, food, minerals, sunlight and even oxygen for sufficient movement and growth. These resources are been acquired using different strategies, the behavioural process of using different strategies for survival is refer to as foraging (Hai, 2014). In optimization process foraging activities are relevant in providing inspiration for the design of different algorithms. The search for nutrient, the quality and quantity of the nutrient with the characteristics of the forager must also be taken into consideration during the algorithm development (Murugan, 2015). In the foraging search process, the ability of exploring the entire search space also depends on the elite individual participating in the foraging process (Naresh, 2011).

Some algorithms developed were obtained from biological processes which mimic the behaviour of some particular organisms (plants and/or animals). For example, Particle Swarm Optimization (PSO) proposed by Eberhart and Kennedy from the Social behaviour of bird and fish schooling, Genetic Algorithm (GA) proposed by Holand from genetics and Darwinian evolution theory, and the Ant Colony Optimization (ACO) developed by Dorigo based on the foraging behaviour of ants (Hanning, 2011). Nature inspired algorithms can be grouped into either population or trajectory-based. The population-based algorithms involve the use of agents that perform the search process so as to arrive or converge to a better solution. Whereas, the trajectory-based uses a single agent that moves through the design space (Mouayad, 2018).

The performance of nature inspired algorithms revolves around the exploration and exploitation abilities of the search agent during the search process in finding a good solution. The exploration process searches the entire search space so as to determine the possible region(s) of the best solution(s) while exploitation is a local search process that uses ideas from problem of interest to obtain a best possible solution from the outcome of exploration (Yang, 2014). In nature inspired algorithms, population size is one of the important parameters in the optimization process. In addition, it favours more of the exploration during the search process. The relationship between population size and dimension of the optimization problem is of great important. Varying the population size or selecting the right population can enhance the performance of the algorithm (Li, 2010).

The BFOA inspired by the *E.Coli* foraging strategy can be simulated based on different processes of chemotaxis, swarming, reproduction, elimination and dispersal (xiaohual, 2016). The chemotaxis is a search behaviour of the bacteria that implements the method of optimization in which the bacteria try to move up to a nutrient concentration level by avoiding harmful substances by finding its way into acidic or alkaline free environment. The bacteria when in a favourable medium swim for a long time by releasing a chemical substance during swarming to attract its fellow bacteria and when in an acidic region turns to move away to find an environment that favours its survival (Nachammai, 2017).

Furthermore, after swarming process, the bacteria extend to another stage of elimination and dispersal. In this stage, the bacteria with the lowest value of cost function will survive and also split into two making the population of the bacteria remains constant (Vipul, 2012). The elimination dispersal event greatly influences the performance of chemotaxis. This process can also result in the death of some bacteria due to increase in temperature as well as other factors that can disperse the bacteria such as water (Heng, 2018). The value of step size is one of the most sensitive parameters in BFOA when solving costraint optimization problems. The right choice of step size can make the algorithm to converge to a better result (Betania,2014).

This research is focused on the variation of population size, probability of elimination-dispersal, step size and number of chemotaxis as an optimization parameter value in EOBFOA. The population size provides how small or large the search space will be during the exploration activities of the bacteria. In foraging process of the bacteria, there are possibilities that certain bacteria will be eliminated by the process of probability of elimination-dispersal so that the algorithm will not be trapped in a local optima search. The step size and number of chemotaxis step are important to the performance of the algorithm since the bacteria will try to avoid noxious substance and tend to concentrate in a region of better nutrient for best performance.

2 ELITE OPPOSITION BACTERIA FORAGING PROCESS

The process is shown using the following relevant equations

$$V_i^t = [v_{i\,1}^t, v_{i\,2}^t \ldots\ldots v_{i\,D}^t] \quad (1)$$

represent the ith solution in the current population at generation t. its elite opposition-based solution is represented in equation (2).

$$EP_i^t = [ep_{i\,1}^t, ep_{i\,2}^t, \ldots, ep_{i\,D}^t] \quad (2)$$

is given by

$$ep_{i\,1}^t = k.\left(F_j^t + G_j^t\right) - x_{i\,j}^t \quad (3)$$

$EF_j^t = \min(ex_{mj}^t)$,

$EG_j^t = \max(ex_{mj}^t)$,

$ep_{i\,1}^t = rand(EF_j^t, EG_j^t)$,

if $ep_{i\,1}^t < LG_i \| ep_{i\,1}^t > UG_i$

$i = 1,2, \ldots..SN;\ \ j = 1,2, \ldots, D;$

$m = 1,2, \ldots\ldots ER;\ \ k = rand(0,1)$

Where

$$EV_m^t = [\,ex_{m1}^t,\ ex_{m2}^t, \ldots\ ex_{m,D}^t] \quad (4)$$

$m = 1,2 \ldots.ER$ are the chosen elite solutions

Where $ep_{i\,j}^t$ is the elite opposition-based value of $x_{i\,j}^t$, ER is the size of the selected elite solution.

Where EF_j^t and EG_j^t are the minimum and maximum value of the jth dimension of the selected elite solutions (Guo, 2015).

[1] Initialize parameters p, S, N_c, N_s, N_{re}, N_{ed}, P_{ed}, $C(i)$, $(i = 1,2, \ldots.S)$, θ^i.

[2] Elimination-dispersal loop: $l = l + 1$

[3] Reproduction loop: $k = k + 1$

[4] Chemotaxis loop: $j = j + 1$

(a) for $i = 1,2 \ldots S$

Chose the ER elite solutions from the current population

Calculate the lower and upper boundary of the chosen elite solution

for $i = 1\ to\ SN$

$k = rand(0,1)$

Create the elite opposition solution and evaluate EP_i^t

Chose the top best SN solution from population and the elite opposition population

else

compute the normal procedure of the traditional BFOA

Perform a chemotactic step for bacterium i as follows

(b) Compute fitness function, $J(i,j,k,l)$.

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

(c) Let $J_{last} = J(i,j,k,l)$

(d) Tumble by generating a random vector $\Delta(i) \in \Re^p$ with each element Δ_m (i), , $m = 1,2, \dots p$ a random number on , $[-1, 1]$

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

There will be a step size $C(i)$ in the direction of tumble for bacterium i

(f) Compute $J(i,j+1,k,l)$ and let

$J(i,j+1,k,l) = J(i,j,k,l) + J_{cc}(\theta^i(j+1,k,l), P(i,j+1,k,l)$

(g) Swim

Let $m = 0$ (counter for swim length)

While $(m < N_s)$. Do

Let $m = m + 1$.

If $J(i,j+1,k,l) < J_{last}$ (if doing better) then

Let $J_{last} = J(i,j+1,k,l)$ and let

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

And use this $\theta^i(j+1,k,l)$ to compute the new $J(i,j+1,k,l)$ as in (f)

Else

let $m = N_s$

end if

end while

(h) Move to the next bacterium $(i+1)$ if $i \neq S$ (i.e move to [b] and process the next bacterium)

[5] IF $j < N_c$, the life of the bacteria is not over move to step 4.

[6] Reproduction:

(a) for each $i = 1,2, \dots S$, and for given k and l,

Let $J_{health}^i = \sum_{j=1}^{N_c+1} J(i,j,k,l)$

and sort the bacteria in ascending order of J_{health}

(b) The bacteria with the highest J_{health} die and the one with lowest J_{health} survive and split into two.

[7] If $k < N_{re}$, move to step 3. Since the maximum number of Nre is not attain.

[8] Elimination-dispersal loop.

for $i = 1,2 \dots S$ with P_{ed} do

eliminate and disperse each bacterium.

if $k < N_{ed}$ then

Go to step 2.

else

end

end if

end for

3 RESULTS AND DISCUSSION

The parameter investigation and analysis of EOBFOA is perform using zakrov function for the unimodal test with the range of [-5 10] as shown in figure 1 and Goldstein and price function for multimodal test with the range of [-2 2] as shown in figure 2. The objection function for zakrove function with Goldstein and price function are given in equation (5) and equation (6).

$$f_1(x) = \sum_{i=1}^{n} x_i^2 + \left(\frac{1}{2}\sum_{i=1}^{n} ix_i\right)^2 + \left(\frac{1}{2}\sum_{i=1}^{n} ix_i\right)^4 \quad (5)$$

$$f_2 = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2] \times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 - 48x_2 + 36x_1x_2 + 27x_2^2] \quad (6)$$

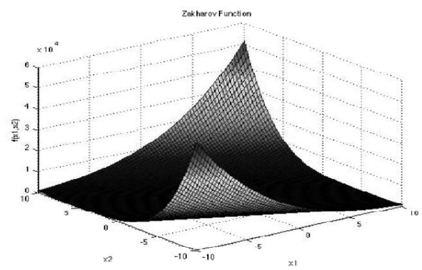

Figure 1: Zakrov function

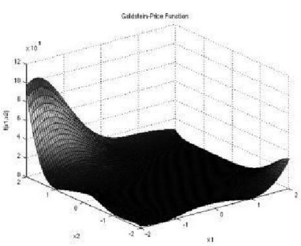

Figure 2: Goldstein and price function

The EOBFOA is run ten times for twenty iteration, for each chosen value of S, P_{ed}, $C(i)$, and N_c. The best minimum convergence value, worst convergence value, mean value and standard deviation value were calculated. Where M1 = Mean Value for unimodal test function, B1= Best convergence value for unimodal test function, W1 = Worst Value for unimodal test function Std1 = Standard deviation value for unimodal test function, M2 = Mean Value for multimodal test function, B2= Best convergence value for multimodal test function, W2 = Worst Value for multimodal test function, Std2 = Standard deviation value for multimodal test function.

3.1 EFFECT OF POPULATION SIZE (S) on EOBFOA

The chosen population size (S) for EOBFOA is shown in table 1.1. The test for each population size is perform using unimodal and multimodal benchmark test function. The best global minimum (B1= 0.000 and B2= 3.1284) as shown in figure 3 and figure 4, and the least standard deviation (std1 = 0.0012 and std2 =0.5206) are obtain at population size of 20. The population size of 30 and 40 also obtain exact the global minimum but with standard deviation of std1 of 0.0047 and 0.0014, and std2 of 1.5724 and 0.5732 higher than that of population size of 20. Increasing the population size of the EOBOFA will favour more exploration of the bacteria in search space which will directly increase the complexity of the algorithm.

TABLE 1.1: POPULATION SIZE (S) ON EOBFOA

S	UNIMODAL TEST FUNCTION				MULTIMODAL TEST FUNCTION			
	M1	B1	W1	Std1	M2	B2	W2	Std2
20	0.0038	0.0000	0.0072	0.0012	3.4233	3.1284	4.3477	0.5206
30	0.0011	0.0004	0.0018	0.0047	4.3319	3.2032	6.9831	1.5724
40	0.0016	0.0065	0.0029	0.0014	3.8229	3.2436	4.4873	0.5732
50	0.0024	0.0083	0.0059	0.0022	3.8896	3.2689	5.6636	1.0502

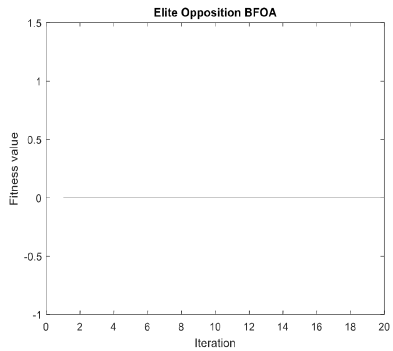

Figure 3: Unimodal test for best population size

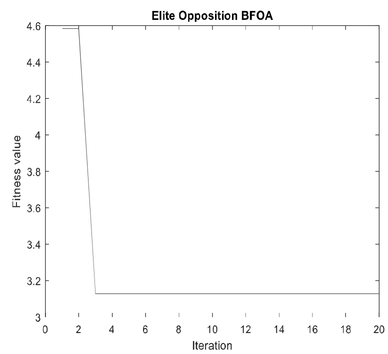

Figure 4: Multimodal test for best population size

3.2 EFFECT OF PROBABILITY OF ELIMINATION-DISPERSAL (P_{ed}) on EOBFOA

The results in table 1.2 shows the different range of P_{ed} values and it effect on EOBFOA. The global minimum (B1= 0.004 and B2 = 3.0147) is shown in figure 5 and figure 6. The least standard deviation (std1 = 0.0006 and std2 = 0.5303) is obtain at the P_{ed} value of 0.5. increasing the value of P_{ed} above 0.5 affect the chance of getting to global minimum. From the result obtain, if the P_{ed} value is large, the EOBFOA can degrade to a random exhaustive search. If a smaller value of P_{ed} is chosen appropriately, it will help the algorithm to escape from local minima and move to global minimum.

TABLE 1.2: PROBABILITY OF ELIMINATION-DISPERSAL (P_{ed}) ON EOBFOA

P_{ed}	UNIMODAL TEST FUNCTION				MULTIMODAL TEST FUNCTION			
	M1	B1	W1	Std1	M2	B2	W2	Std2
0.1	0.0011	0.0004	0.0021	0.0007	3.9249	3.2671	5.0101	0.6485
0.5	0.0007	0.0000	0.0015	0.0006	3.5103	3.0417	4.4100	0.5303
0.9	0.0018	0.0005	0.0041	0.0017	3.9942	3.0537	4.0647	1.5358
1.3	0.0034	0.0006	0.0106	0.0041	5.0580	3.2206	9.9413	2.7841

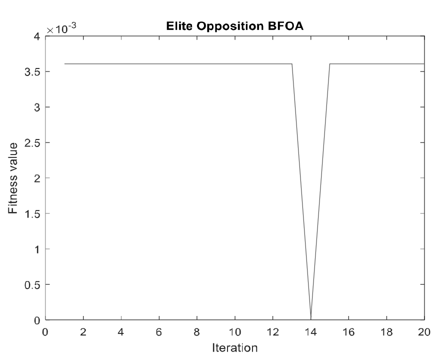

Figure 5: Unimodal test for best Ped Value

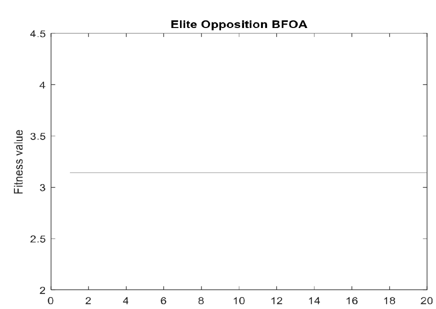

Figure 6: Multimodal test for best Ped Value

3.3 EFFECT OF STEP SIZE $C(i)$ on EOBFOA

The step size value ($C(i)$) is an important parameter with greater influence on the convergence behaviour of the EOBFAO. The best performance of EOBFOA is at $C(i)$ value of 0.01 of B1= 0.0002 and B2=3.0778 as shown in figure 7 and figure 8. The least standard deviation value of std1 = 0.0012 and std2 = 1.0535 on both unimodal and multimodal benchmark test function is also shown in table 1.3. A higher value of $C(i)$ above 0.01 will promote exploration of the search space by the bacteria and reducing the step size value will promote the exploitation by the bacteria in the search space.

TABLE 1.3: STEP SIZE $C(i)$ ON EOBFOA

$C(i)$	UNIMODAL TEST FUNCTION				MULTIMODAL TEST FUNCTION			
	M1	B1	W1	Std1	M2	B2	W2	Std2
0.01	0.0030	0.0002	0.0043	0.0012	4.1957	3.0778	4.7561	1.0535
0.05	0.0143	0.0017	0.0163	0.0104	31.5270	5.6320	74.8100	30.0859
0.09	0.0538	0.0114	0.1823	0.0723	26.4181	3.1323	57.2976	21.4121
0.13	0.0117	0.0024	0.0276	0.0101	38.1482	12.9209	114.5691	43.1189

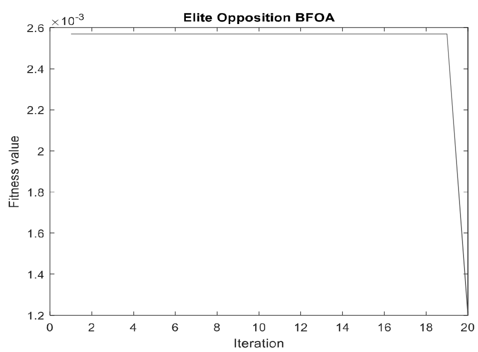

Figure 7: Unimodal test for best Cvalue

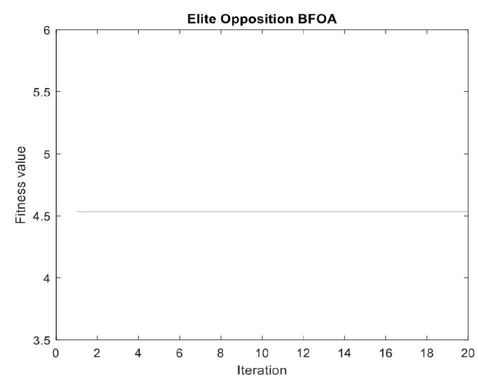

Figure 8: Multimodal test for best Cvalue

3.4 EFFECT OF NUMBER CHEMOTAXIS (Nc) on EOBFOA

The number of chemotaxis indicate the how meaningful movement the bacteria make from one location to another. From table 1.4, at N_c = 20, the EOBFOA performs better by arriving at the global minimum of B1 = 0.000 and B2 = 3.0811 as shown in figure 9 and figure 10. The least standard deviation is a obtain at std1 =0.0003 and std2 = 0.3715. increasing the value N_c above 20, will enable EOBFOA to move away from bad region during the optimization search process while decreasing the value of N_c below 20 will reduce the active participation of the bacteria in the search space.

TABLE 1.4: NUMBER OF CHEMOTAXIS (Nc) ON EOBFOA

N_c	UNIMODAL TEST FUNCTION				MULTIMODAL TEST FUNCTION			
	M1	B1	W1	Std1	M2	B2	W2	Std2
10	0.0006	0.0004	0.0005	0.0004	3.6178	3.1396	4.9483	0.7558
20	0.0007	0.0000	0.0018	0.0003	3.6794	3.0811	4.1247	0.3715
30	0.0032	0.0006	0.0085	0.0033	4.6853	3.3650	5.3604	1.3017
40	0.0035	0.0022	0.0041	0.0009	4.5466	3.5275	5.5310	0.8866

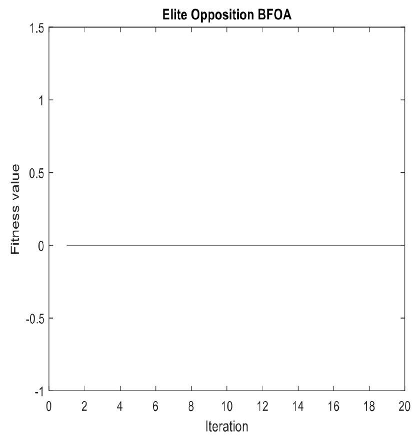

Figure 9: Unimodal test for best Nc value

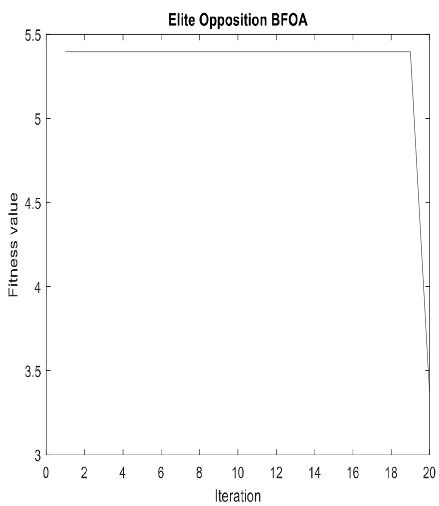

Figure 10: Multimodal test for best Nc value

4 CONCLUSION

In this research paper, an investigation and analysis of some parameters value has been performed on EOBFOA base on convergence to a better solution or the least standard deviation. From the performance analysis, it has been observed that the EOBFOA had its best performance at the population size of 20, probability of elimination of dispersal of 0.5, step size value of 0.01 and number of chemotaxis of 20. Furthermore, increasing the size of some parameter values like population will also increase the complexity or running time of the algorithm since the agent will take more time during the exploration process. Further research will focus on performance evaluation of BFOA and EOBFOA on statistical test function and another optimization application in the field of image processing.

REFERENCES

Betania, H., Ma D.P.P, & Efren, M. M. (2014). Step Size Control on the Modified Bacterial Foraging Algorithm for Constrained Numerical Optimization. GECCO '14 *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation.* Pp 25-32.

Guo, Z., Wang, S., Yue, D., & Jian, K. L. (2015). Elite Opposition-based Artificial Bee Colony Algorithm for Global Optimization. *IJE Transactions C: Aspect* Vol.28, No. 9, Pp 1268-1275.

Hai, S., & Yunlong, Z. (2014). Adaptive Bacterial Foraging Optimization Algorithm Based on Social Foraging Strategy. *Journal of Networks*. Pp 799-806.

Heng, L., Wayan, I. M., Noor, A. S. (2018). Discrete Bacterial Foraging Optimization for Resource Allocation in Macrocell Femtocell Networks. *Wiley ETRI Journal*. Pp 726-735.

Li, M.S., Ji, T.Y., Tang, W.J., Wu, Q.H., & Saunders. (2010). Bacterial Foraging Algorithm with Varying Population. *Elsevier Journal of Bio System*. Pp 185-197.

Murugan, P., Karthikeyan, R., & Pandiaraj, K. (2015). Implementation of Bacterial Foraging Optimization Algorithm in Leaf Spring Cutting Stock Problem. *International Journal of Advanced Technology in Engineering and Science*. Pp 381-391.

Nachammai, N., & kayalvizhi, R. (2017). Performance Analysis of Fuzzy Logic Controller Using Bacterial Foraging Optimization Algorithm. *Journal of Engineering Technologies and Innovative Research (JETIR).* Vol. 4, Issue 12. Pp 177-182.

Naresh, G., Ramalinga, M., & Nasasimham, S.V.L. (2011). Bacterial Foraging Algorithm for the Robust Design of Multi Machine Power System Stabilizer. *International Conference on Signal, Image Processing and Applications*. Pp 181-187.

Sankalap, A., & Satvir, S. (2013). The Firefly Optimization Algorithm: Convergence Analysis and Parameter

Selection. *International Journal of Computer Applications*. Vol. 69, No. 3. Pp 48-52.

Vipul, S., Pattnaik, S.S, & Tanuj, G. (2012). A Review of Bacterial Foraging Optimization and Its Applications. National Conference of Future Aspect of Artificial Intelligence in Industrial Automation. *International Journal of Computer Application.* Pp 9-12.

Xiaohul, F., Yuyao, H., Honghi, Y., & Yu, J. (2016). Self-Adaptive Bacterial Foraging Optimization Algorithm Based on Evolution Strategies. *Rev. Téc. Ing. Univ. Zulia*. Vol. 39, No. 8. Pp 350-358.

Yang, X.S. (2014). Nature Inspired Optimization Algorithms. *Elsevier Journal*. Pp 1-20.