An improved chaotic-based African buffalo optimisation algorithm

Chinwe Peace Igiri*, Yudhveer Singh and Ramesh Chandra Poonia

Amity Institute of Information Technology,
Amity University Rajasthan,
Jaipur, India
Email: chynkemdirim@gmail.com
Email: yudhvir.chahal@gmail.com
Email: rameshpoonia@gmail.com
*Corresponding author

Abstract: Optimisation remains inevitable in any organisation as the need to maximise the limited resources persists. It justifies the seemingly endless research in this area. This study explores the effectiveness of chaos to mitigate false or premature convergence problem in African buffalo optimisation (ABO) algorithm. Chaos employs the ergodic and stochastic properties to handle this limitation. Three resourceful chaotic functions in the literature are evaluated to find the best strategy for ABO improvement. The same strategy is applied across the algorithms under study to provide an unbiased judgment. The study validates the proposed system's performance with a range of nonlinear test functions. The proposed system's result is compared with standard ABO, Particle swarm optimisation (PSO), and chaotic particle swarm optimisation (CPSO) algorithms. Although chaotic ABO (CABO) gave 92% performance in comparison with standard ABO, chaotic PSO, and standard PSO; it requires further investigation. To be more explicit, the reason for no significant difference between chaotic-ABO and standard ABO in some functions calls for further research attention. The present study also highlights the research future scope. In all, the study gives insight to researchers on the appropriate algorithm for a real-world problem.

Keywords: chaotic-optimisation; premature convergence; African buffalo optimisation; ABO; bio-inspired algorithm; nonlinear benchmark optimisation problems; chaotic African buffalo optimisation algorithm; chaotic particle swarm optimisation algorithm; logistic map; iterative chaotic map with infinite collapses; tent map; swarm intelligence.

Reference to this paper should be made as follows: Igiri, C.P., Singh, Y. and Poonia, R.C. (2019) 'An improved chaotic-based African buffalo optimisation algorithm', *Int. J. Innovative Computing and Applications*, Vol. 10, Nos. 3/4, pp.147–153.

Biographical notes: Chinwe Peace Igiri is an Assistant Professor at the Department of Computer Engineering of Port Harcourt Polytechnics, Nigeria. She is also pursuing her doctoral studies in Software Engineering at Amity University, Rajasthan, India. Her research interests focus on artificial intelligence applications, bio-inspired computation, industrial optimisation, and machine learning. She is a certified member of Council for Regulation of Engineers in Nigeria (COREN), Nigeria Society of Engineers (NSE), among other professional bodies.

Yudhveer Singh has served as an Assistant Professor at the Department of Amity Institute of Information Technology, Amity University Rajasthan. He has more than ten years rich and illustrious teaching experience at undergraduate and postgraduate levels. He has published more than 14 papers in good indexed international journal and serves as reviewer and editor in renowned international journals of repute. Besides he has presented more than ten research papers in national and international conferences. He has also supervised four research scholars for PhD degree. His specialisation area is special function, fractional calculus, integral equation and integral transform and its applications.

Ramesh Chandra Poonia has rich experience of 15+ years as an academician. At present, he is an Associate Professor of Computer Science at Amity Institute of Information Technology, Amity University Rajasthan, India. He has received his PhD in Computer Science from the Banasthali University, Rajasthan, India. His research interests include internet of things for sustainable agriculture, ITS and WSN protocol evaluation, internet congestion control and simulation and modelling.

1 Introduction

Optimisation is an active subject in literature especially in operations research, computer science, mathematics, and many more. Irrespective of the field, the focus is the same. To be more precise, the operation researcher's focal point is seeking a balance between cost and profit. While computer scientist and mathematician search for the best out of many feasible options. In other words, optimisation is the converging point for these three disciplines. It draws research attention in exponential progression due to the demand for high computation effort to solve industrial applications especially as the problem size increases (Binitha and Sathya, 2012; Kumar and Kumari, 2018).

Studies show that bio-inspired optimisation methods could effectively mitigate high computation effort challenge. It has many real-world applications such as digital marketing (Kumar et al., 2017, in press), engineering design (Arana-Daniel et al., 2018; Kaur and Arora, 2018), networking, energy optimisation, and many more. Ideally, the model of natural phenomenon determines its efficiency (Fister et al., 2013). The global and local search spaces proper exploration would improve the solution quality (Fister et al., 2013). However, there is no perfect bio-inspired algorithm, especially in standard phase. Several variants of even the most robust bio-inspired algorithm exist in an attempt to improve performance (Igiri et al., 2019). For example, particle swarm optimisation (PSO), firefly algorithm (FA), cuckoo search (CS), bat algorithm (BA), and many more have multiple modifications. FA, for instance, have the discrete, chaotic-based, hybridised versions, among other variants (Yang and He, 2013). Besides, ABO has been hybridised with levy flight to enhance its performance in nonlinear optimisation problem (Igiri et al., 2018). The bio-inspired approach is better alternatives to traditional methods (Binitha and Sathya, 2012).

One might question the need for further research despite that much achievement. Strictly speaking, the model representation of these natural phenomena determines the solution. Altering the characteristics of an organism is impossible; however, improving their models could enhance their performance. Thus, the enormousness and continuous research in modifying bio-inspired techniques (Igiri et al., 2019). It is, therefore, the motivation for this study. Thus, the research question is: what is the effect of chaos on ABO? What is its relative performance regarding, standard ABO, PSO, and chaotic PSO? What are the possible limitations to chaotic ABO?

Chaos, on the other hand, is an unpredictability induced in a simple-deterministic system. Metaphorically, chaos is a 'butterfly effect' (Yang and He, 2013). Chaotic optimisation induces a system to improve its performance. There are various features such as ergodic, stochastic or randomisation, regularity, and sensitivity that control chaos (Yang and He, 2013; Feng et al., 2017). The concept of chaotic-optimisation is invoking these features to generate chaos variable (Hamaizia et al., 2012). However, the ergodic and stochastic properties are key factors that enable algorithms to skip the problem of local optima entrapment (Yang and He, 2013). Algorithms such as PSO, ABO, BA, FA, among others suffer this type of limitation. Studies prove the effectiveness of chaos in handling such challenges especially in PSO, BA, FA, and many more (Binitha and Sathya, 2012; Kaur and Arora, 2018; Fister et al., 2013; Yang and He, 2013; Zilong et al., 2006). Chaos has many advantages including the ability to escape local minimum, slow global convergence rate, and insensitivity to an initial value. However, it does not yield a satisfactory result in large problem size (Hamaizia et al., 2012).

Chaotic optimisation has demonstrated an unbeaten record in many nature-inspired algorithms. For example, studies show that chaotically improved PSO yielded a better result than standard PSO (Alatas et al., 2009; Meng et al., 2004; Liu et al., 2005). Chen and Yu (2008) obtained a high prediction soft sensor model of real-time ethylene measurement using a chaotic hybrid PSO. Chaotic BA outperformed standard bat and other none chaotic-based algorithms (Gandomi and Yang, 2014; Jordehi, 2015). This study structure is as follows: Section 1 is the introduction and Section 2 is the theoretical framework. The methodology, the function evaluation/results, and conclusion are presented in Sections 3, 4 and 5, respectively.

1.1 Research highlight

- The ABO algorithm is induced by a chaotic map.
- Standard nonlinear optimisation benchmark functions are used to evaluate the performance of the new chaotic ABO (CABO) variant.
- CABO has higher convergence speed than CPSO, standard ABO and PSO in the test functions under study.
- The CABO gave 92% performance in comparison with the other swarm intelligence under study.
- The logistic map is the best chaotic strategy for ABO algorithm.

2 Theoritical framework

2.1 African buffalo optimisation algorithm

ABO is a metaheuristic algorithm in the class of swarm intelligence. It is developed by Odili and Kahar in 2016, and inspired by the foraging and defensive behaviour of African buffaloes that belong to cow family (Odili and Kahar, 2016). African buffaloes live in herds with peculiar defensive, democratic, and exceptionally intelligent features that make them distinctive (Odili and Kahar, 2016). Typically ABO is a sound-based algorithm similar to echolocation of BA. Two distinct sounds, 'maaa' and 'waaa' control the herds (Odili and Kahar, 2016). The 'waaa' alert is the exploration signal while the 'maaa' call is the exploitation signal (Odili and Kahar, 2016). These two signals constitute the significant components of ABO algorithm. The same also form the two equations that control the kernel of the algorithm. The exploration equation (maaa, ma_j) also referred to as the democratic equation controls the movement of the herds using two learning parameters, lb_1 and lb_2. While the exploitation equation is controlled by (waaa, wa_j). The ma_j and wa_j represent the diversification and intensification movements of j^{th} buffalo ($j = 1, 2, \ldots, n$) respectively as illustrated in (1) and (2).

As swarm intelligence, the algorithm employs a coordination mechanism of the herds to find the best solution to an optimisation problem. It involves seven iterative steps stated in Algorithm 1. Equations (1) and (2) control the iterative steps. While searching for food, the herds vote to determine the best grazing location by comparing each buffalo's best position $bp_{\max j}$ and the global best $bg_{\max j}$ position. The fitness value ma_{j+1} determines the next processing step. Two conditions decide the criteria: first, no further better solution; second, when the maximum iteration value is reached.

$$ma_{j+1} = ma_j + lb_1\left(bg_{\max j} - wa_j\right) + lb_2\left(bp_{\max j}\right) \tag{1}$$

$$wa_{j+1} = \frac{wa_j + ma_j}{\pm 0.5} \tag{2}$$

Algorithm 1 Standard ABO

Step 1:	Objective function $f(u) = u = (u_1, u_2, \ldots, u_n)^{\Omega}$
Step 2:	Randomly initialise the buffaloes within the search space;
Step 3:	Update the fitness values with (1)
Step 4:	Update the buffalo's location $j^{th} = (bg_{\max}$ and $bp_{\max})$ using (2)
Step 5:	If $bp_{\max}$ is updating, proceed to Step 6, else return to Step 2
Step 6:	If stopping criteria is reached, proceed to Step 7, otherwise, return to Step 3
Step 7:	Output best result

Source: Odili and Kahar (2016)

2.2 *ABO weakness*

The ABO algorithm uses few parameters which have successfully mitigated the delay problem of many algorithms such as PSO, FA and BA. However, ABO exhibits other weaknesses including premature convergence and efficiency in search space exploration. Consequently, ABO gives only feasible results against the expected optimum solution. The weak kernel could be the reason for poor performance. Thus, enhancing the kernel could improve its performance. The kernel of an algorithm is the equation(s) or mathematical representation that controls its operation. Implicitly, modifying the mathematical model of a biological sequence could enhance its performance. Thus, these weaknesses are the motivation for this study.

2.3 *Chaotic functions*

The ergodic and stochastic features of chaos have successfully handled premature convergence. Typically, the PSO (Chen and Yu, 2008; Alatas et al., 2009; Meng et al., 2004; Liu et al., 2005), BA (Gandomi and Yang, 2014; Jordehi, 2015), among others. This study focuses on three chaotic functions, namely, logistic, the iterative chaotic map with infinite collapses (ICMIC) and tent map (Afrabandpey et al., 2014). These three are chosen because of their remarkable performance as referenced in the literature. However, the best strategy would be applied to algorithm kernels under study.

2.3.1 *The logistic map function*

Logistic map is the simplex polynomial map (Chen and Yu, 2008). The function is defined by (3)

$$v_{j+1} = \beta v_j\left(1 - v_j\right) \tag{3}$$

where $v_j \in [0, 1]$ and $v_0 \in [0, 1]$, j is the number of iterations, β is the control parameter.

2.3.2 *The ICMIC map function*

The ICMIC is a one-dimension chaotic map with infinite collapses at the iterative regions in contrast to logistic and tent map with finite collapses (Jordehi, 2015). The map generates chaotic numbers in ([0, 1]) and characterised by (4)

$$v_{j+1} = abs\left(\sin\left(\frac{\beta}{v_j}\right)\right) \tag{4}$$

2.3.3 *Tent map function*

The tent map exhibits a similarity with the logistic map. It can replicate itself to the required population size for better exploration (Afrabandpey et al., 2014). It is effective for the evolutionary algorithm and BA due to this exploration feature. Studies have also proven that 1/0.7 is an effective multiplier value (Afrabandpey et al., 2014). The function is defined by (5).

$$v_{j+1} = p\left(v_j\right) = \begin{cases} \dfrac{v}{0.7}, & v < 0.7 \\ \dfrac{1}{0.3}v(1-v), & \text{otherwise} \end{cases} \tag{5}$$

3 Methodology

The standard ABO uses a simple random number generator for computation during the iterative processes. The random number generator could be the reason for local optima entrapment in standard ABO, especially in a multi-modal optimisation. Ideally, the chaos generates a well-distributed search within the solution space; thereby enabling the algorithm to escape premature convergence.

This study attempts to mitigate this problem by using chaotic variables. It considers three popular chaotic functions for enhancing bio-inspired optimisation in the literature. The study validates the three to obtain the best variant for ABO. The best is used to intensify the ABO democratic and exploitative search capabilities. The linearly decreasing function is multiplied by the chaotic variable to facilitate the search mechanism. The democratic equation is updated at each iteration using (6). Algorithm 2 is the proposed chaotic ABO. In comparison with the standard ABO (see Algorithm 1), there are eight steps in CABO against seven steps in the former. The additional step is the chaotic function generator (step 2), the same is the novelty of this study highlighted in bold.

The simulation is carried out in MATLAB. The system specification includes Intel(R) Pentium(R), CPU N3510 @ 1.99 GHz, 4.00 GB RAM, and 64-bit operating system X64-based processor. The parameter of any bio-inspired algorithm is essential in determining its performance. ABO uses only two parameters: $lb_1 = 0.8$ and $lb_2 = 0.8$.

The performance of the chaotic ABO would be evaluated with standard ABO, chaotic PSO, and PSO.

$$ma_{j+1} = ma_j + lb_1 \boldsymbol{v} * \left(b_o g_{\max j} - wa_j\right) + lb_2 \left(b_o p_{\max j}\right) \tag{6}$$

$$wa_{j+1} = \frac{wa_j + ma_j}{\pm 0.5} + \boldsymbol{v} \tag{7}$$

where $\boldsymbol{v}$ is the suitable map and other variables definition remain as in Section 2.1.

Algorithm 2 Chaotic ABO

Step 1:	Objective function $f(u) = \mathrm{u} = (u_1, u_2, \ldots, u_n)^{\Omega}$
Step 2:	**Generate chaotic map**
Step 3:	Randomly initialise the buffaloes within the search space;
Step 4:	Update the fitness values with (6)
Step 5:	Update the buffalo's location $j^{\text{th}} = (bg_{\max}$ and $bp_{\max})$ using (7)
Step 6:	If $bp_{\max}$ is updating, proceed to Step 7, else return to Step 3
Step 7:	If stopping criteria is reached, proceed to Step 8, otherwise, return to Step 4
Step 8:	Output best result

4 Functions evaluation and results

1 Sphere

$$g(x) = \sum\nolimits_{i=0}^{n} x_i^2, \quad x \in [-10, 10] \tag{8}$$

2 Schaffer

$$g(x) = 0.5 + \frac{\sin\left(x_1 - x_2\right)^2 - 0.5}{\left[1 + 0.001\left(x_1^2 + x_2^2\right)\right]^2}, \quad x \in [-100, 100], 0 \tag{9}$$

3 Beale

$$\begin{aligned} g(x) &= \left(1.5 - x_1 + x_1 x_2\right)^2 + \left(2.25 - x_1 x_2^2\right) \\ &\quad + \left(2.625 - x_1 x_2^3\right)^2, \quad x \in [-4.5, 4.5], 0 \end{aligned} \tag{10}$$

4 Bochachvesky

$$\begin{aligned} g(x) &= x_1^2 + 2x_2^2 - 0.3\cos\left(3\pi x_1\right) \\ &\quad -0.4\cos\left(4\pi x_{12}\right) + 0.7, \quad x \in [-100, 100], 0 \end{aligned} \tag{11}$$

5 Easom

$$\begin{aligned} g(x) &= -\cos\left(x_1\right)\cos\left(x_2\right)\exp\left(-\left(x_1 - \pi\right)^2\right. \\ &\quad \left. -\left(x_2 - \pi\right)^2\right), \quad x \in [-100, 100], -1 \end{aligned} \tag{12}$$

The three chaotic maps (logistics, ICMIC, and tent) are evaluated to find the best strategy across the algorithms under study. These three are selected as a result of the proven performance in literature. Sphere function is used for performance validation. Logistic map emerged the best out of the three. The same is utilised to generate chaos sequence to enhance the algorithms' kernels. In order to secure a fair comparable computation condition; first, the chaotic map is applied across the algorithms under consideration. Second, all the algorithms have the same starting point. Last, a variety of numerical benchmark functions are used to evaluate the performance of the algorithms based on convergence rate, robustness, and reliability.

4.1 Convergence rate assessment

Since the kernels control the bio-inspired algorithms, the iteration number is used to evaluate their efficiencies. To illustrate, at 100th iteration chaotic ABO outperforms standard ABO and Chaotic PSO, and standard PSO on sphere and Schaffer functions as shown in Table 1 and Table 3. The convergence plot for sphere and Schaffer function are illustrated in Figures 1 and 2 for proper visualisation. Also, chaotic ABO converged at the global minimum at the 38th and 44th iteration on Beale and Bochachvesky functions respectively, as shown in Table 2 and Table 4. It reveals the strength of chaos in chaotic ABO. In other words, chaotic ABO outperforms standard ABO and chaotic PSO, and standard PSO as shown in the convergence plot in Figures 3 and 4, respectively. Although chaotic PSO also converged at global minimum on Bochachvesky function, but at the 87th iteration. ABO on the other hand only gave a feasible solution across all the test functions under study. It further reveals the efficacy of chaos to enhance bio-inspired-based optimisation algorithms.

Contrarily, CABO has no significant impact on Easom function unlike the standard ABO. However, the CPSO converged at global minimum in Easom function as shown in Table 5 and Figure 5. The result variation could be justified with no-free-lunch (NFL) theorem which states that 'there is no perfect algorithm' (Wolpert and Macready, 1997).

Table 1 Comparison of chaotic variants on sphere function (F1)

Chaotic map	*Best solution at 100 iterations*				*Global minimum value*
	ABO	*CABO*	*CPSO*	*PSO*	
Logistic	2.8428e-13	**1.0873e-54**	1.1433e-23	0.0045441	0.0000
ICMIC	2.8428e-13	3.1221e-56	1.2894e-06	0.0045441	0.0000
Tent	2.8428e-13	9.9091e-55	2.0767e-22	0.0045441	0.0000

Table 2 Schaffer function (F2) table

Evaluation	*ABO*	*CABO*	*CPSO*	*PSO*	*Global min. value*
Best solution	5.2398e-12	**0.000**	1.5085e-09	0.03348	0.000
Number of iterations	100	**43**	100	100	-

Table 3 Beale function (F3) table

Evaluation	*ABO*	*CABO*	*CPSO*	*PSO*	*Global min. value*
Best solution	8.7258e-13	**0.0000**	0.0000	0.021354	0.000
Number of iterations	100	**38**	80	100	-

Table 4 Bochachvesky function (F4) table

Evaluation	*ABO*	*CABO*	*CPSO*	*PSO*	*Global min. value*
Best solution	1.9661e-08	**0.000**	0.0000	0.052831	0.000
Number of iterations	100	**44**	87	100	-

Table 5 Easom function (F5) table

Evaluation	*ABO*	*CABO*	*CPSO*	*PSO*	*Global min. value*
Best solution	–1.4383e-05	–0.99795	**–1.0000**	–0.061084	–1.0000
Number of Iterations	100	100	**55**	100	-

Figure 1 Sphere function (F1) convergence plot (see online version for colours)

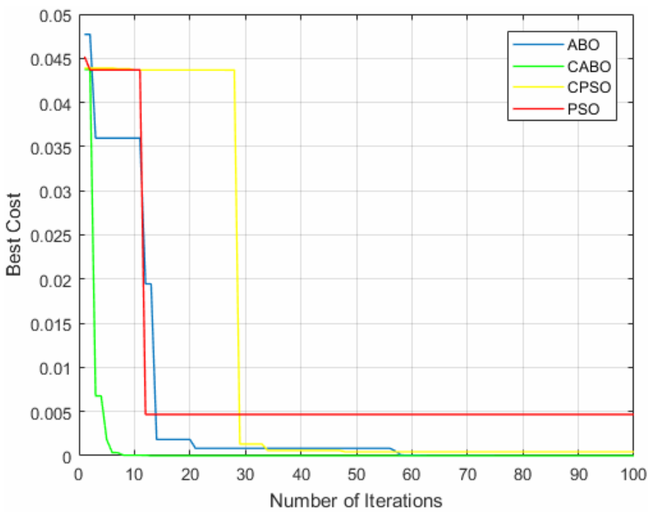


Figure 2 Schaffer function (F2) convergence plot (see online version for colours)

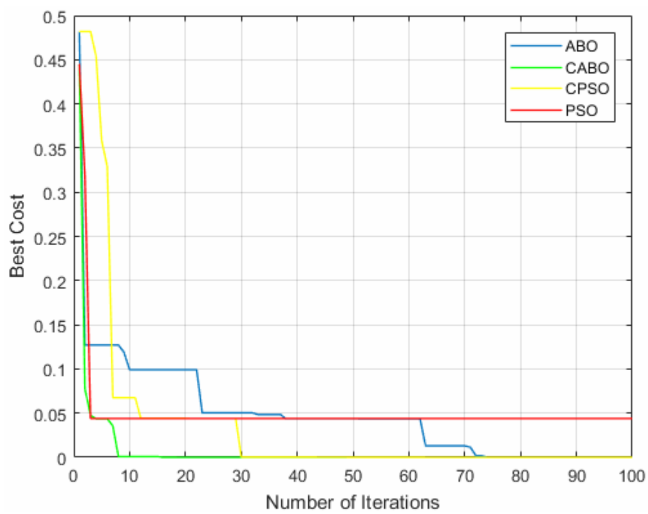


Figure 3 Beale function (F3) convergence plot (see online version for colours)

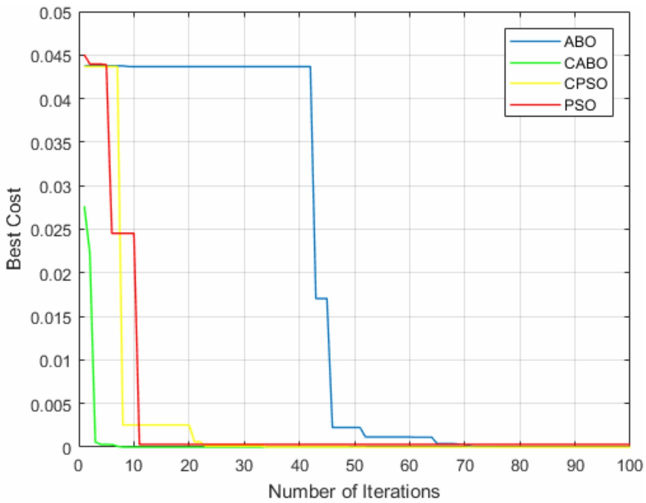


Figure 4 Bochachvesky 3 function (F4) convergence plot (see online version for colours)

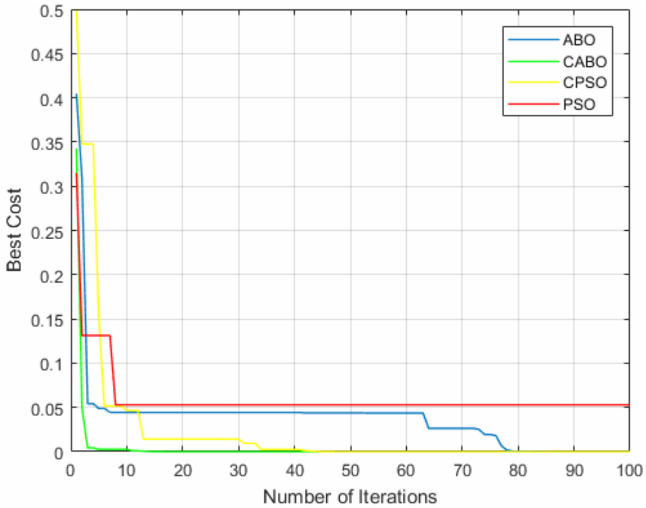
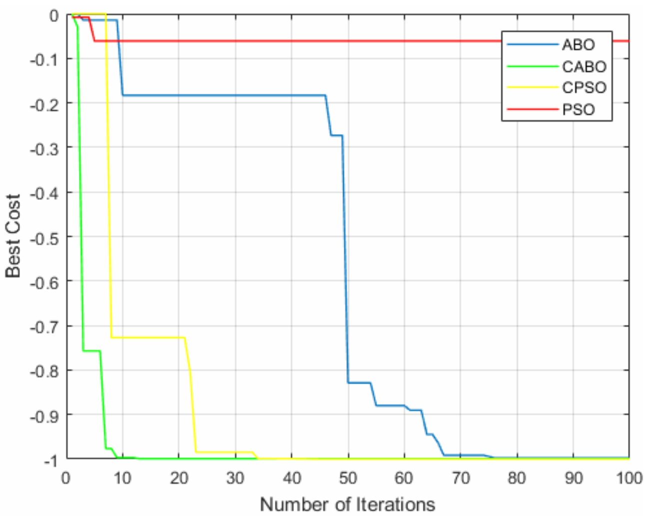


Figure 5 Easom function (F5) convergence plot (see online version for colours)



4.2 Robustness and reliability assessment

The test for robustness and reliability of an algorithm is validated on its efficiency over a wide range of problems (Beiranvand et al., 2017). In reference to the numerical benchmark functions under this study, chaotic ABO shows 92% performance better than CPSO, standard ABO and PSO. The success rate is validated based upon best solution (objective function values) and the number of iteration of the algorithms to achieve the results herein. Aside, standard ABO is used to comparative study to reveal the impact of chaos on the proposed system. Chaotic PSO is used to justify the result in a fair algorithm environment, while the standard ABO and PSO is used to portray the effectiveness of chaotic-based optimisation.

5 Conclusions and future scope

This study investigates the best chaotic variant to improve the bio-inspired algorithms under study. Logistic map function emerged the best and the same is utilised to enhance the search efficiency of chaotic ABO and chaotic PSO. The overall performance of improved algorithm in terms of convergence rate, robustness and reliability is validated over a variation of standard benchmark test functions in the literature. The result gave 92% performance in comparison with CPSO, ABO, and PSO as shown in Table 6. The convergence of chaotic ABO at global minimum as discussed in Section 4.1 suggests the efficiency of chaos in skipping local optima. It accounts for the outperformance of the improved proposed method over standard ABO, PSO and chaotic PSO.

Table 6 Performance ranking table

Algorithm/ function	*F1*	*F2*	*F3*	*F4*	*F5*	*Sum/n*10 (%)*	*Ranking*
CABO	10	10	10	10	6	92	1
CPSO	8	6	8	8	10	80	2
ABO	6	8	6	6	8	68	3
PSO	4	4	4	4	4	40	4

However, despite the success rate of the computational effort record of chaotic ABO, there is need for further improvement as revealed on its poor performance on Easom function. Also, the theoretical insight for the lack of chaos impact on Easom function is beyond the scope of this study and stands to be investigated in the future study. Also ABO could be further be hybridised with Bottle-nose Dolphin algorithm (Pazhaniraja et al., 2017) in the future work. Besides the improved CABO could be applied to real-world optimisation problems such as networking, supply chain, and engineering design in further studies.

References

Afrabandpey, H., Ghaffari, M., Mirzaei, A. and Safayani, M. (2014) 'A novel bat algorithm based on chaos for optimization tasks', *2014 Iranian Conference on Intelligent Systems (ICIS)*, February, pp.1–6, IEEE.

Alatas, B., Akin, E. and Ozer, A.B. (2009) 'Chaos embedded particle swarm optimization algorithms', *Chaos, Solitons & Fractals*, Vol. 40, No. 4, pp.1715–1734.

Arana-Daniel, N., Lopez-Franco, C. and Alanis, A.Y. (2018) *Bio-inspired Algorithms for Engineering*, Butterworth-Heinemann, The Boulevard, Langford Lane, Kidlington, Oxford OX5 1GB, UK.

Beiranvand, V., Hare, W. and Lucet, Y. (2017) 'Best practices for comparing optimization algorithms', *Optimization and Engineering*, Vol. 18, No. 4, pp.815–848.

Binitha, S. and Sathya, S.S. (2012) 'A survey of bio inspired optimization algorithms', *International Journal of Soft Computing and Engineering*, Vol. 2, No. 2, pp.137–151.

Chen, R.Q. and Yu, J.S. (2008) 'Study and application of chaos-particle swarm optimization-based hybrid optimization algorithm', *Journal of System Simulation*, Vol. 20, No. 3, pp.685–688.

Feng, J., Zhang, J., Zhu, X. and Lian, W. (2017) 'A novel chaos optimization algorithm', *Multimedia Tools and Applications*, Vol. 76, No. 16, pp.17405–17436.

Fister Jr., I., Yang, X.S., Fister, I., Brest, J. and Fister, D. (2013) *A Brief Review of Nature-Inspired Algorithms for Optimization*, arXiv preprint arXiv:1307.4186.

Gandomi, A.H. and Yang, X.S. (2014) 'Chaotic bat algorithm', *Journal of Computational Science*, Vol. 5, No. 2, pp.224–232.

Hamaizia, T., Lozi, R. and Hamri, N.E. (2012) 'Fast chaotic optimization algorithm based on locally averaged strategy and multifold chaotic attractor', *Applied Mathematics and Computation*, Vol. 219, No. 1, pp.188–196.

Igiri, C.P., Singh, Y. and Bhargava, D. (2018) 'An improved African buffalo optimization algorithm using chaotic map and chaotic-levy flight', *International Journal of Engineering & Technology*, Vol. 7, No. 4, pp.4570–4576.

Igiri, C.P., Singh, Y. and Poonia, R. (2019) 'A review study of modified swarm intelligence: particle swarm optimization, firefly, bat and gray wolf optimizer algorithms', *Recent Patents on Computer Science*, Vol. 12, doi: 10.2174/2213275912666190101120202.

Jordehi, A.R. (2015) 'Chaotic bat swarm optimisation (CBSO)', *Applied Soft Computing*, Vol. 26, No. C, pp.523–530.

Kaur, G. and Arora, S. (2018) 'Chaotic whale optimization algorithm', *Journal of Computational Design and Engineering*, Vol. 5, No. 3, pp.275–284.

Kumar, B.S., Bharghava, D., Kar, A.K. and Igiri, C.P. (2017) 'Identification of suitable websites for digital marketing – an approach using bio-inspired computing', *International Journal of Engineering & Technology*, Vol. 7, Nos. 1–2, p.239, doi: 10.14419/ijet.v7i1.2.9313.

Kumar, B.S., Bharghava, D., Kar, A.K. and Igiri, C.P. (in press) 'Introducing the rock hyrax intelligent optimization algorithm: an exploration for Web 3.0 domain selection', *International Journal of Advanced Intelligence Paradigms*, Inderscience.

Kumar, S. and Kumari, R. (2018) 'Artificial bee colony, firefly swarm optimization, and bat algorithms', *Advances in Swarm Intelligence for Optimizing Problems in Computer Science*, pp.145–182, Chapman and Hall/CRC, 5 Howick Place, London, SW1P 1WG.

Liu, B., Wang, L., Jin, Y.H., Tang, F. and Huang, D.X. (2005) 'Improved particle swarm optimization combined with chaos', *Chaos, Solitons & Fractals*, Vol. 25, No. 5, pp.1261–1271.

Meng, H.J., Zheng, P., Wu, R.Y., Hao, X.J. and Xie, Z. (2004) 'A hybrid particle swarm algorithm with embedded chaotic search', 2004 IEEE Conference on Cybernetics and Intelligent Systems, Vol. 1, pp.367–371, IEEE.

Odili, J. and Kahar, M.M. (2016) 'African buffalo optimization', *International Journal of Software Engineering and Computer Systems*, Vol. 2, pp.28–50, doi: 10.15282/ijsecs.2.2016.3.0014

Pazhaniraja, N., Paul, P.V., Roja, G., Shanmugapriya, K. and Sonali, B. (2017) 'A study on recent bio-inspired optimization algorithms', *2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN)*, March, pp.1–6, IEEE.

Wolpert, D.H. and Macready, W.G. (1997) 'No free lunch theorems for optimization', *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, pp.67–82.

Yang, X.S. and He, X. (2013) *Firefly Algorithm: Recent Advances and Applications*, arXiv preprint arXiv:1308.3898.

Zilong, G., Sun'an, W. and Jian, Z. (2006) 'A novel immune evolutionary algorithm incorporating chaos optimization. *Pattern Recognition Letters*, Vol. 27, No. 1, pp.2–8.