Elephant Herding Optimization

Gai-Ge Wang*
School of Computer Science and Technology
Jiangsu Normal University
Xuzhou, China
E-mail: gaigewang@163.com

Suash Deb
Dept. of Computer Science & Engineering
Cambridge Institute of Technology
Ranchi, INDIA
Email: suashdeb@gmail.com

Leandro dos S. Coelho
Industrial and Systems Engineering Graduate Program
Pontifical Catholic University of Parana
Curitiba, Brazil
Email: lscoelho2009@gmail.com

***Abstract*—In this paper, a new kind of swarm-based metaheuristic search method, called Elephant Herding Optimization (EHO), is proposed for solving optimization tasks. The EHO method is inspired by the herding behavior of elephant group. In nature, the elephants belonging to different clans live together under the leadership of a matriarch, and the male elephants will leave their family group when they grow up. These two behaviors can be modelled into two following operators: clan updating operator and separating operator. In EHO, the elephants in each clan are updated by its current position and matriarch through clan updating operator. It is followed by the implementation of the separating operator which can enhance the population diversity at the later search phase. To demonstrate its effectiveness, EHO is benchmarked by fifteen test cases comparing with BBO, DE and GA. The results show that EHO can find the better values on most benchmark problems than those three metaheuristic algorithms.**

Keywords- Elephant herding optimization; Benchmark functions; Global optimization; Swarm intelligence

I. INTRODUCTION

The ever increasing complexity of the real-world problems is making it extremely difficult for the traditional methods to address those. On the other hand, though modern metaheuristic methods cannot provide exact answers, they can generate satisfactory solutions within a reasonable time span. Over the past few years, various kinds of metaheuristic algorithms have been proposed and successfully applied to solve myriads of real-world optimization problems. Among all metaheuristic methods, swarm-based algorithms [1] are one of the most representative paradigms & widely used ones.

Swarm intelligence (SI) methods are one of the most well-known paradigms in metaheuristic methods which has been widely used in various applications. Its inspiration originates from the collective behavior of animals. Two of the widely used SI algorithms are particle swarm optimization (PSO) [2-9] and ant colony optimization (ACO) [10]. The idea of PSO [2] originated from the social behavior of bird flocking.

The ants, in nature, are well capable of keeping the past paths in mind by pheromone. Inspired by this phenomenon, ACO [10, 11] was proposed by Dorigo et al. Recently, more effective swarm intelligence algorithms have been proposed, such as artificial bee colony (ABC) [12, 13], cuckoo search (CS) [14-20], bat algorithm (BA) [21-24], firefly algorithm (FA) [25-28], animal migration optimization (AMO) [29], ant lion optimizer (ALO) [30], big bang-big crunch algorithm (BB-BC) [31-34], charged system search (CSS) [35-37], chaotic swarming of particles (CSP) [38], monarch butterfly optimization (MBO) [39], krill herd (KH) [40-47], multi-verse optimizer (MVO) [48], dragonfly algorithm (DA) [49], grey wolf optimizer (GWO) [50, 51], wolf search algorithm (WSA) [52], among others.

In general, wide elephants are social in nature and the elephant group is composed of several clans. The elephants belonging to different clans live together under the leadership of a matriarch, and male elephants remain solitary and will leave their family group while growing up. Inspired by the herding behavior of elephant group, a new kind of swarm-based heuristic search method, called EHO, is proposed for solving global optimization tasks. These habitation of elephants can be used to solve optimization problems. The behavior of elephant herding in nature are idealized into clan updating operator and separating operator. In EHO, each elephant implements clan updating operator to update its position based on its current position and matriarch in the responding clan. Subsequently, the worst elephant is replaced by separating operator. By comparing with BBO [53-58], DE [59-64] and GA [65], the performance of EHO is investigated by several experiments implemented on fifteen test cases. The results show that EHO can find much fitter solutions on most benchmark problems than the three other methods.

Section 2 reviews the herding behavior of elephants in nature. Section 3 discusses how the herding behavior of elephants is to formulate a general-purpose heuristic search. Several simulation results comparing EHO with other methods on fifteen functions, are presented in Section 4. Section 5 concludes this paper.

II. HERDING BEHAVIOR OF ELEPHANTS

Elephants are one of the largest mammals on land. The African elephant and the Asian elephant are two of traditionally recognized species. A long trunk is the most representative feature that are multipurpose, such as breathing, lifting water and grasping objects.

In nature, elephants are social animals, and they have complex social structures of females and calves. An elephant group is composed of several clans under the leadership of a matriarch, often the oldest cow [66]. A clan consists of one female with her calves or certain related females. Females prefer to live in family groups, while male elephants tend to live in isolation, and they will leave their family group when growing up. Though male elephants live away from their family group, they can stay in contact with elephants in their clan through low-frequency vibrations [66].

In this paper, the herding behavior of the elephants is considered as two operators, which are subsequently idealized to form a general-purpose global optimization method.

978-1-4673-8501-5/15 $31.00 © 2015 IEEE
DOI 10.1109/ISCBI.2015.8

III. ELEPHANT HERDING OPTIMIZATION

In order to make the herding behaviour of elephants solve all kinds of global optimization problems, we preferred to simplify it into the following idealized rules.

1) The elephant population is composed of some clans, and each clan has fixed number of elephants.

2) A fixed number of male elephants will leave their family group and live solitarily far away from the main elephant group at each generation.

3) The elephants in each clan live together under the leadership of a matriarch.

A. Clan updating operator

As mentioned before, all the elephants live together under the leadership of a matriarch in each clan. Therefore, for each elephant in clan *ci*, its next position is influenced by matriarch *ci*. For the elephant *j* in clan *ci*, it can be updated as

$$x_{new,ci,j} = x_{ci,j} + \alpha \times \left(x_{best,ci} - x_{ci,j}\right) \times r \tag{1}$$

where $x_{new,ci,j}$ and $x_{ci,j}$ are newly updated and old position for elephant *j* in clan *ci*, respectively. $\alpha \in [0,1]$ is a scale factor that determines the influence of matriarch *ci* on $x_{ci,j}$. $x_{best,ci}$ represents matriarch *ci*, which is the fittest elephant individual in clan *ci*. $r \in [0, 1]$. Here, uniform distribution is used.

The fittest elephant in each clan cannot be updated by Eq. (1), i.e., $x_{ci,j}=x_{best,ci}$. For the fittest one, it can be updated as

$$x_{new,ci,j} = \beta \times x_{center,ci} \tag{2}$$

where $\beta \in [0,1]$ is a factor that determines the influence of the $x_{center,ci}$ on $x_{new,ci,j}$. We can see, the new individual $x_{new,ci,j}$ in Eq. (2) is generated by the information obtained by all the elephants in clan *ci*. $x_{center,ci}$ is the centre of clan *ci*, and for the *d*-th dimension it can be calculated as

$$x_{center,ci,d} = \frac{1}{n_{ci}} \times \sum_{j=1}^{n_{ci}} x_{ci,j,d} \tag{3}$$

where $1 \le d \le D$ indicates the *d*-th dimension, and *D* is its total dimension. n_{ci} is the number of elephants in clan *ci*. $x_{ci,j,d}$ is the *d*-th of the elephant individual $x_{ci,j}$. The centre of clan *ci*, $x_{center,ci}$, can be calculated through *D* calculations by Eq. (3).

Based on the description above, the clan updating operator can be formulated as shown in Figure 1.

B. Separating operator

In elephant group, male elephants will leave their family group and live alone when they reach puberty. This separating process can be modelled into separating operator when solving optimization problems. In order to further improve the search ability of EHO method, let us assume that the elephant individuals with the worst fitness will implement the separating operator at each generation as shown in Eq. (4).

$$x_{worst,ci} = x_{\min} + \left(x_{\max} - x_{\min} + 1\right) \times rand \tag{4}$$

where $x_{\max}$ and $x_{\min}$ are respectively upper and lower bound of the position of elephant individual. $x_{worst,ci}$ is the worst elephant individual in clan *ci*. $rand \in [0, 1]$ is a kind of stochastic distribution and uniform distribution in the range [0, 1] is used in our current work.

Accordingly, the separating operator can be formed as shown in Figure 2.

for *ci*=1 to *nClan* (for all clans in elephant population) **do**
  for *j*=1 to n_{ci} (for all elephants in clan *ci*) **do**
    Update $x_{ci,j}$ and generate $x_{new,ci,j}$ by Eq. (1).
    if $x_{ci,j}=x_{best,ci}$ **then**
      Update $x_{ci,j}$ and generate $x_{new,ci,j}$ by Eq. (2).
    end if
  end for *j*
end for *ci*

Figure 1. Pseudo code of clan updating operator

for *ci*=1 to *nClan* (all the clans in elephant population) **do**
  Replace the worst elephant in clan *ci* by Eq. (4).
end for *ci*

Figure 2. Pseudo code of separating operator

Based on the description of clan updating operator and separating operator, the EHO method is developed and its mainframe can be summarized as shown in Figure 3.

Step 1: Initialization. Set generation counter *t*=1; initialize the population; the maximum generation *MaxGen*.
Step 2: While *t*<*MaxGen* **do**
  Sort all the elephants according to their fitness.
  Implement clan updating operator by Figure 1.
  Implement separating operator as shown in Figure 2.
  Evaluate population by the newly updated positions.
  t=*t*+1.
Step 3: end while

Figure 3. Pseudo code of EHO algorithm

IV. SIMULATION RESULTS

In this section, EHO is verified by benchmark evaluation in comparison with three methods (BBO [53], DE [59] and GA [65]) on fifteen test problems (see Table 1). F01-F15 are basic benchmarks while F16-F20 are rotated, shifted, and composition functions selected from IEEE CEC 2005. More information about these test problems can be found in [53, 67, 68]. The dimension of F01-F20 is set to fifteen in this work.

In order to obtain fair results, all the implementations are conducted under the same conditions as shown in [69].

For four methods, both the population size and maximum generations are set to fifty. The parameters in EHO are set as: the scale factor α=0.5, β=0.1, and the number of clan *nClan*=5. In our current work, all the clans have the same number elephants, i.e., n_{ci}=10. For BBO, DE and GA, their parameter settings can be found in [69, 70].

TABLE I. BENCHMARK FUNCTIONS

No.	*Name*	*No.*	*Name*	*No.*	*Name*
F01	Ackley	F06	Griewank	F11	Penalty #2
F02	Alpine	F07	Holzman 2	F12	Perm
F03	Brown	F08	Levy	F13	Powell
F04	Dixon & Price	F09	Pathological	F14	Quartic
F05	Fletcher-Powell	F10	Penalty #1	F15	Rastrigin

In general, all the metaheuristic methods are depended on certain stochastic distribution. Therefore, different runs will generate different results. In this work, 100 independent runs are implemented in order to get the most representative statistical results (see Tables II-III). In the following tables, the fittest solution is highlighted in bold font.

In Table II, the "6.57±0.95" indicates the mean and standard deviation (Std) of function value are 6.57 and 0.95, respectively. For the mean and Std values as shown in Table II, EHO method has proven its best performance on F01-F04, F06-F08, F10, F11, F13, and F15. BBO has shown its best performance on F05, F09 and F14. For F12, DE has the fittest solutions, while GA has the smallest Std on this cases.

TABLE II. MEAN FUNCTION VALUES OBTAINED BY FOUR METHODS

	BBO	DE	EHO	GA
F01	6.57±0.95	18.30±0.47	**1.7E-3±1.9E-4**	16.69±0.80
F02	1.58±0.78	14.74±1.70	**2.5E-4±3.6E-5**	24.02±4.27
F03	0.50±1.00	2.90±0.65	**2.9E-6±2.2E-7**	9.89±4.59
F04	8.7E5±7.8E5	3.5E6±1.5E6	**0.89±0.05**	1.3E7±9.7E6
F05	**8.9E4±2.6E4**	2.4E5±5.1E4	6.4E5±1.9E5	2.1E5±8.6E4
F06	8.94±2.75	22.33±4.17	**1.00±9.9E-8**	44.82±19.61
F07	327.50±317.19	1.9E3±960.30	**2.8E-13±1.4E-13**	1.3E4±7.7E3
F08	2.65±1.16	15.27±3.84	**1.73±0.15**	34.71±11.65
F09	5.28±**0.46**	**3.67**±0.64	4.90±0.52	5.74±0.58
F10	2.5E4±7.1E4	3.2E5±4.1E5	**0.45±0.11**	3.1E6±5.3E6
F11	3.3E5±9.7E5	2.7E6±1.7E6	**1.89±0.17**	1.4E7±1.7E7
F12	6.7E51±2.2E51	**5.5E45**±1.4E46	7.5E49±2.0E50	6.0E51±**1.3E37**
F13	153.28±85.38	1.1E3±349.80	**5.3E-6±1.9E-6**	820.66±472.24
F14	**2.2E-16±0.00**	0.77±0.31	2.3E-15±7.5E-16	26.88±17.04
F15	2.18±1.45	115.82±14.78	**3.6E-5±9.1E-6**	47.40±11.29

TABLE III. BEST FUNCTION VALUES OBTAINED BY FOUR METHODS.

	BBO	DE	EHO	GA
F01	4.08	15.82	**1.3E-3**	13.76
F02	0.37	9.97	**1.8E-4**	12.69
F03	**2.2E-16**	1.63	2.3E-6	**2.2E-16**
F04	4.8E4	8.7E5	**0.74**	9.3E5
F05	**2.6E4**	1.3E5	2.5E5	8.5E4
F06	3.52	13.09	**1.00**	10.27
F07	60.00	475.88	**7.6E-14**	1.6E3
F08	**0.40**	7.17	1.27	12.41
F09	4.34	**1.46**	3.20	4.07
F10	3.26	32.71	**0.23**	13.90
F11	118.50	2.6E5	**1.33**	2.2E5
F12	6.0E51	**5.8E37**	4.0E45	6.0E51
F13	23.00	334.18	**8.8E-7**	191.00
F14	**2.2E-16**	0.18	8.4E-16	**2.2E-16**
F15	**2.2E-16**	79.00	1.8E-5	9.00

For the best solutions as shown in Table III, EHO has the strong search ability and can find the fittest solution on F01, F02, F04, F06, F07, F10, F11, F13, and F16-F18. BBO and DE have the best solutions on F05, F08, F15, F20 and F09, F12, F19, respectively. It should be noted that, for F03 and F14 cases, both BBO and GA can find the same final function values coincidentally.

Moreover, the convergent process of four methods on the most representative benchmarks can be given as follows (see Figures 4-5). Figure 4 shows the convergent history of F02 Alpine function. For this case, though BBO is able to find the final solution that is little worse than EHO, EHO can converge to the optimal sharply within five generations. Figure 5 shows the convergent history of F03 Brown function. It is clear that, EHO has the better performance than BBO, DE and GA during the whole optimization process. BBO, DE and GA have the similar performance.

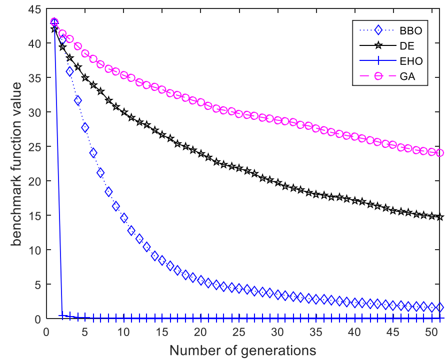

FIGURE 4. CONVERGENT CURVES OF THE F02 ALPINE FUNCTION

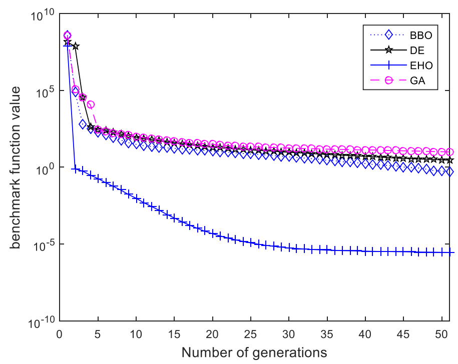

FIGURE 5. CONVERGENT CURVES OF THE F03 BROWN FUNCTION

V. CONCLUSION

In this paper, the behavior of elephant herding are idealized into clan updating operator and separating operator. Through modelling the behavior of elephant herding in nature, a new kind of swarm-based heuristic search method, called EHO, is proposed for solving global optimization tasks. At the early phase of EHO, each elephant in clan is updated by using clan information through clan updating operator. And then, the worst elephant is replaced by randomly generated elephant individual through separating operator. By comparing with BBO, DE and GA, EHO is benchmarked by fifteen test cases. EHO can find much better solutions on most benchmark problems than three other algorithms.

ACKNOWLEDGMENT

This work was supported by Jiangsu Province Science Foundation for Youths (No. BK20150239) and National Natural Science Foundation of China (No. 61503165).

REFERENCES

[1] Z. Cui, and X. Gao, "Theory and applications of swarm intelligence," *Neural Computing & Applications,* vol. 21, no. 2, pp. 205-206, 2012.
[2] J. Kennedy, and R. Eberhart, "Particle swarm optimization," in Proceeding of the IEEE International Conference on Neural Networks, Perth, Australia, 1995, pp. 1942-1948.
[3] G. Ram, D. Mandal, R. Kar, and S. P. Ghoshal, "Optimal design of non-uniform circular antenna arrays using PSO with wavelet mutation," *International Journal of Bio-Inspired Computation,* vol. 6, no. 6, pp. 424-433, 2014.
[4] S. Mirjalili, G.-G. Wang, and L. d. S. Coelho, "Binary optimization using hybrid particle swarm optimization and gravitational search algorithm," *Neural Computing and Applications,* vol. 25, no. 6, pp. 1423-1435, 2014.
[5] G.-G. Wang, A. H. Gandomi, A. H. Alavi, and S. Deb, "A hybrid method based on krill herd and quantum-behaved particle swarm optimization," *Neural Computing and Applications*, 2015.
[6] G.-G. Wang, A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "A novel improved accelerated particle swarm optimization algorithm for global numerical optimization," *Engineering Computations,* vol. 31, no. 7, pp. 1198-1220, 2014.
[7] X. Zhao, B. Song, P. Huang, Z. Wen, J. Weng, and Y. Fan, "An improved discrete immune optimization algorithm based on PSO for QoS-driven web service composition," *Applied Soft Computing,* vol. 12, no. 8, pp. 2208-2216, 2012.
[8] X. Zhao, "A perturbed particle swarm algorithm for numerical optimization," *Applied Soft Computing,* vol. 10, no. 1, pp. 119-124, 2010.
[9] S. Mirjalili, and A. Lewis, "S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization," *Swarm and Evolutionary Computation,* vol. 9, pp. 1-14, 2013.
[10] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics,* vol. 26, no. 1, pp. 29-41, 1996.
[11] K. Krynicki, J. Jaen, and J. A. Mocholí, "Ant colony optimisation for resource searching in dynamic peer-to-peer grids," *International Journal of Bio-Inspired Computation,* vol. 6, no. 3, pp. 153-165, 2014.
[12] D. Karaboga, and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization,* vol. 39, no. 3, pp. 459-471, 2007.
[13] X. Li, and M. Yin, "Self-adaptive constrained artificial bee colony for constrained numerical optimization," *Neural Computing and Applications,* vol. 24, no. 3-4, pp. 723-734, 2012.
[14] X. S. Yang, and S. Deb, "Cuckoo search via Lévy flights." pp. 210-214.
[15] X.-S. Yang, and S. Deb, "Cuckoo search: recent advances and applications," *Neural Computing and Applications,* vol. 24, no. 1, pp. 169-174, 2013.
[16] X. Li, J. Wang, and M. Yin, "Enhancing the performance of cuckoo search algorithm using orthogonal learning method," *Neural Computing and Applications,* vol. 24, no. 6, pp. 1233-1247, 2013.
[17] G.-G. Wang, A. H. Gandomi, X. Zhao, and H. C. E. Chu, "Hybridizing harmony search algorithm with cuckoo search for global numerical optimization," *Soft Computing*, 2014.
[18] G.-G. Wang, S. Deb, A. H. Gandomi, Z. Zhang, and A. H. Alavi, "Chaotic cuckoo search," *Soft Computing*, 2015.
[19] G.-G. Wang, A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "A new hybrid method based on krill herd and cuckoo search for global optimization tasks," *International Journal of Bio-Inspired Computation*, 2012.
[20] X. Li, and M. Yin, "Modified cuckoo search algorithm with self adaptive parameter method," *Information Sciences,* vol. 298, pp. 80-97, 2015.
[21] X. S. Yang, *Nature-inspired metaheuristic algorithms*, 2nd ed., Luniver Press, Frome, 2010.
[22] S. Mirjalili, S. M. Mirjalili, and X.-S. Yang, "Binary bat algorithm," *Neural Computing and Applications,* vol. 25, no. 3-4, pp. 663-681, 2013.
[23] J.-W. Zhang, and G.-G. Wang, "Image matching using a bat algorithm with mutation," *Applied Mechanics and Materials,* vol. 203, no. 1, pp. 88-93, 2012.
[24] G.-G. Wang, B. Chang, and Z. Zhang, "A multi-swarm bat algorithm for global optimization." pp. 480-485.
[25] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Mixed variable structural optimization using firefly algorithm," *Computers & Structures,* vol. 89, no. 23-24, pp. 2325-2336, 2011.
[26] X. S. Yang, "Firefly algorithm, stochastic test functions and design optimisation," *International Journal of Bio-Inspired Computation,* vol. 2, no. 2, pp. 78-84, 2010.

[27] G.-G. Wang, L. Guo, H. Duan, and H. Wang, "A new improved firefly algorithm for global numerical optimization," *Journal of Computational and Theoretical Nanoscience,* vol. 11, no. 2, pp. 477-485, 2014.

[28] L. Guo, G.-G. Wang, H. Wang, and D. Wang, "An effective hybrid firefly algorithm with harmony search for global numerical optimization," *The Scientific World Journal,* vol. 2013, pp. 1-10, 2013.

[29] X. Li, J. Zhang, and M. Yin, "Animal migration optimization: an optimization algorithm inspired by animal migration behavior," *Neural Computing and Applications,* vol. 24, no. 7-8, pp. 1867-1877, 2014.

[30] S. Mirjalili, "The ant lion optimizer," *Advances in Engineering Software,* vol. 83, pp. 80-98, 2015.

[31] O. K. Erol, and I. Eksin, "A new optimization method: Big Bang-Big Crunch," *Advances in Engineering Software,* vol. 37, no. 2, pp. 106-111, 2006.

[32] A. Kaveh, and S. Talatahari, "Optimal design of Schwedler and ribbed domes via hybrid Big Bang-Big Crunch algorithm," *Journal of Constructional Steel Research,* vol. 66, no. 3, pp. 412-419, 2010.

[33] A. Kaveh, and S. Talatahari, "A discrete big bang-big crunch algorithm for optimal design of skeletal structures," *Asian Journal of Civil Engineering,* vol. 11, no. 1, pp. 103-122, 2010.

[34] A. Kaveh, and S. Talatahari, "Size optimization of space trusses using Big Bang–Big Crunch algorithm," *Computers & Structures,* vol. 87, no. 17-18, pp. 1129-1140, 2009.

[35] A. Kaveh, and S. Talatahari, "A novel heuristic optimization method: charged system search," *Acta Mechanica,* vol. 213, no. 3-4, pp. 267-289, 2010.

[36] S. Talatahari, and R. Sheikholeslami, "Optimum design of gravity and reinforced retaining walls using enhanced charged system search algorithm," *KSCE Journal of Civil Engineering,* vol. 18, no. 5, pp. 1464-1469, 2014.

[37] A. Kaveh, and S. Talatahari, "Charged system search for optimal design of frame structures," *Applied Soft Computing,* vol. 12, no. 1, pp. 382-393, 2012.

[38] A. Kaveh, R. Sheikholeslami, S. Talatahari, and M. Keshvari-Ilkhichi, "Chaotic swarming of particles: a new method for size optimization of truss structures," *Advances in Engineering Software,* vol. 67, pp. 136-147, 2014.

[39] G.-G. Wang, S. Deb, and Z. Cui, "Monarch butterfly optimization," *Neural Computing and Applications*, 2015.

[40] A. H. Gandomi, and A. H. Alavi, "Krill herd: a new bio-inspired optimization algorithm," *Communications in Nonlinear Science and Numerical Simulation,* vol. 17, no. 12, pp. 4831-4845, 2012.

[41] G.-G. Wang, A. H. Gandomi, and A. H. Alavi, "A chaotic particle-swarm krill herd algorithm for global numerical optimization," *Kybernetes,* vol. 42, no. 6, pp. 962-978, 2013.

[42] A. H. Gandomi, S. Talatahari, F. Tadbiri, and A. H. Alavi, "Krill herd algorithm for optimum design of truss structures," *International Journal of Bio-Inspired Computation,* vol. 5, no. 5, pp. 281-288, 2013.

[43] G.-G. Wang, A. H. Gandomi, A. H. Alavi, and G.-S. Hao, "Hybrid krill herd algorithm with differential evolution for global numerical optimization," *Neural Computing and Applications,* vol. 25, no. 2, pp. 297-308, 2014.

[44] G.-G. Wang, A. H. Gandomi, and A. H. Alavi, "An effective krill herd algorithm with migration operator in biogeography-based optimization," *Applied Mathematical Modelling,* vol. 38, no. 9-10, pp. 2454-2462, 2014.

[45] G.-G. Wang, A. H. Gandomi, and A. H. Alavi, "Stud krill herd algorithm," *Neurocomputing,* vol. 128, pp. 363-370, 2014.

[46] G.-G. Wang, A. H. Gandomi, A. H. Alavi, and S. Deb, "A Multi-Stage Krill Herd Algorithm for Global Numerical Optimization," *International Journal on Artificial Intelligence Tools*, 2015.

[47] L. Guo, G.-G. Wang, A. H. Gandomi, A. H. Alavi, and H. Duan, "A new improved krill herd algorithm for global numerical optimization," *Neurocomputing,* vol. 138, pp. 392-402, 2014.

[48] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: a nature-inspired algorithm for global optimization," *Neural Computing and Applications*, 2015.

[49] S. Mirjalili, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Computing and Applications*, 2015.

[50] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software,* vol. 69, pp. 46-61, 2014.

[51] S. Saremi, S. Z. Mirjalili, and S. M. Mirjalili, "Evolutionary population dynamics and grey wolf optimizer," *Neural Computing and Applications,* vol. 26, no. 5, pp. 1257-1263, 2014.

[52] S. Fong, S. Deb, and X.-S. Yang, "A heuristic optimization method inspired by wolf preying behavior," *Neural Computing and Applications*, 2015.

[53] D. Simon, "Biogeography-based optimization," *IEEE Transactions on Evolutionary Computation,* vol. 12, no. 6, pp. 702-713, 2008.

[54] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Let a biogeography-based optimizer train your Multi-Layer Perceptron," *Information Sciences,* vol. 269, pp. 188-209, 2014.

[55] X. Li, and M. Yin, "Multi-operator based biogeography based optimization with mutation for global numerical optimization," *Computers & Mathematics with Applications,* vol. 64, no. 9, pp. 2833-2844, 2012.

[56] S. Saremi, S. Mirjalili, and A. Lewis, "Biogeography-based optimisation with chaos," *Neural Computing and Applications,* vol. 25, no. 5, pp. 1077-1097, 2014.

[57] X.-T. Li, and M.-H. Yin, "Parameter estimation for chaotic systems using the cuckoo search algorithm with an orthogonal learning method," *Chinese Physics B,* vol. 21, no. 5, pp. 050507, 2012.

[58] X. Li, and M. Yin, "Multiobjective binary biogeography based optimization for feature selection using gene expression data," *IEEE Transactions on NanoBioscience,* vol. 12, no. 4, pp. 343-353, 2013.

[59] R. Storn, and K. Price, "Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization,* vol. 11, no. 4, pp. 341-359, 1997.

[60] D. Zou, H. Liu, L. Gao, and S. Li, "An improved differential evolution algorithm for the task assignment problem," *Engineering Applications of Artificial Intelligence,* vol. 24, no. 4, pp. 616-624, 2011.

[61] D. Zou, J. Wu, L. Gao, and S. Li, "A modified differential evolution algorithm for unconstrained optimization problems," *Neurocomputing,* vol. 120, pp. 469-481, 2013.

[62] D.-x. Zou, L.-q. Gao, and S. Li, "Volterra filter modeling of a nonlinear discrete-time system based on a ranked differential evolution algorithm," *Journal of Zhejiang University SCIENCE C,* vol. 15, no. 8, pp. 687-696, 2014.

[63] X. Li, and M. Yin, "An opposition-based differential evolution algorithm for permutation flow shop scheduling based on diversity measure," *Advances in Engineering Software,* vol. 55, pp. 10-31, 2013.

[64] X. Li, and M. Yin, "Modified differential evolution with self-adaptive parameters method," *Journal of Combinatorial Optimization*, 2014.

[65] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine learning*, Addison-Wesley, New York, 1998.

[66] R. Sukumar, *The Asian elephant: ecology and management*, Cambridge University Press, New York, 1993.

[67] X.-S. Yang, Z. Cui, R. Xiao, A. H. Gandomi, and M. Karamanoglu, *Swarm Intelligence and Bio-Inspired Computation*, Elsevier, Waltham, MA, 2013.

[68] P. Suganthan, N. Hansen, J. Liang, K. Deb, Y. Chen, A. Auger, and S. Tiwari, *Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization*, Nanyang Technological University, Singapore, 2005.

[69] G. Wang, L. Guo, H. Wang, H. Duan, L. Liu, and J. Li, "Incorporating mutation scheme into krill herd algorithm for global numerical optimization," *Neural Computing and Applications,* vol. 24, no. 3-4, pp. 853-871, 2014.

[70] G.-G. Wang, L. Guo, A. H. Gandomi, G.-S. Hao, and H. Wang, "Chaotic krill herd algorithm," *Information Sciences,* vol. 274, pp. 17-34, 2014.