Performance of Elephant Herding Optimization Algorithm on CEC 2013 real parameter single objective optimization

VIKTOR TUBA
John Naisbitt University
Graduate School of Computer Science
Bulevar umetnosti 29, Belgrade
SERBIA
viktortuba@gmail.com

MARKO BEKO
Univ. Lusófona de Humanidades e Tecn.
Computer Engineering Department
Campo Grande 376, Lisabon
PORTUGAL
marko@isr.ist.utl.pt

MILAN TUBA
John Naisbitt University
Graduate School of Computer Science
Bulevar umetnosti 29, Belgrade
SERBIA
tuba@ieee.org

Abstract: Numerous real life problems represents hard optimization problems that cannot be solved by deterministic algorithm. In the past decades various different methods were proposed for these kind of problems and one of the methods are nature inspired algorithms especially swarm intelligence algorithms. Elephant herding optimization algorithm (EHO) is one of the recent swarm intelligence algorithm that has not been thoroughly researched. In this paper we tested EHO algorithm on 28 standard benchmark functions and compared results with particle swarm optimization algorithm. Comparison show that EHO has good characteristics and it outperformed other approach from literature.

Key–Words: hard optimization problems, optimization algorithms, swarm intelligence, elephant herding optimization, EHO

1 Introduction

Optimization and solving different optimization problems represent an active research fields for decades. Numerous real life optimization problems are hard optimization problems that cannot be solved in reasonable time by deterministic algorithm and they belong to NP difficult problems. One of the well known NP difficult problem is traveling salesman where salesman needs to visit N cities once in such order that cost the least. This can be represented by graph where nodes are cities and edges of the graph have weight that represent cost of traveling cites that are connected with it. Deterministic approach has complexity of N! thus in case of more than 20 cities, calculation time will be unreasonably long.

For solving hard optimization problems different stochastic algorithms that use random factors and set of the search rules were proposed in the past. Stochastic algorithms does not guarantee optimal solution or the same solution each time, but if good algorithm runs long enough, obtained solution will be *good enough* which means that it will be in tolerance margin from optimal solution. Since different solution can be obtained for the same problem solved by the same algorithm, as final solution average result of numerous runs is usually used as final solution.

The most stochastic algorithms are natural based, i.e. they imitate some natural phenomena. It has been shown that those kind of algorithms provides good solutions even though it is not completely understood why or how exactly. All nature based stochastic algorithms can be divided into three groups: evolutionary, artificial immune systems and swarm intelligence algorithms.

As the name says, evolution algorithms use the idea of survival evolutionary. After initial population of solutions that can be randomly created the next generation combines the best solutions from previous generation. Concept of mutation is usually used as random factor. In evolution algorithms population goes trough numerous iterations of breeding and in each iteration we are closer to the solution. Evolution algorithms searched for good solution and combine them while artificial immune systems use negative selection where bad solutions are searched so they can be eliminated from the population.

Swarm intelligence algorithms are recent stochastic algorithms. Idea is to mimic collective behavior of spices from nature. Each individual represents one possible solution and by collective intelligence best solution is searched. Movement of each individual is based on its own memory, global data from swarm and random factor. Swarm intelligence algorithms represent active research area and in the past decades numerous of them were proposed. Some of the well known swarm intelligence algorithms are particle swarm optimization (PSO), ant colony optimization (ACO), artificial bee colony (ABC), bat algorithm (BA), fireworks algorithm (FWA) and others.

In this paper performance of the recent swarm intelligence algorithm, elephant herding optimization algorithm (EHO) was tested. We tested (EHO) on standard CEC 2013 benchmark functions. Obtained results were compared with PSO.

The rest of the paper is organized as follows. In Section 2 some of swarm intelligence algorithm and applications that use them were presented. In Section 3 elephant herding optimization algorithm was explained in details. Experimental results along with comparison with PSO were presented in Section 4. At the end conclusion is given in Section 5.

2 Swarm intelligence algorithms

Various applications need to solve some kind of unconstrained or constrained optimization problem. For solving it numerous techniques and methods were proposed. Two major groups of metaheuristics are commonly used, inspired by nature and not inspired by nature. In this paper nature inspired metaheuristic called swarm intelligence was tested.

Swarm intelligence algorithms are based on the collective behavior of the social groups from nature and it is an important research topic. The main idea of this algorithms is to use simple set of rules that control individuals which exhibit collective intelligence. Swarms of worms, ants, bees, birds and fish were the main source inspiration for these methods. Brief analysis of swarm intelligence algorithm is given in [30].

Particle swarm optimization (PSO) is one of the earliest swarm intelligence algorithms [11] inspired by social behavior of fish or birds. Original form but also upgraded versions of PSO were widely used for solving various global optimization problems [15].

Ant colony optimization (ACO) imitates social behavior of ants. ACO models ants property of disposing pheromone on their way from nest to the food source. This metaheuristic have numerous variants that can be found in the literature. ACO was successfully used on minimum weight vertex cover problem [8], [26], minimum connected dominating set problem [9], and many others.

Artificial bee colony (ABC) was inspired by social behavior of honey bee swarm [10]. In ABC algorithm three types of bees are included: employed, onlookers and scouts. This algorithm was widely used and it was shown that is effective and efficient for different problems and numerous upgraded and enhanced versions of ABC were proposed [4], [6], [13]. ABC showed robustnesses when tackling engineering optimization [25].

Seeker optimization algorithm (SOA) performs search process by modeling human reasoning, memory, interactions, and past experience. Global optimization problems were successfully solved with this technique [7]. In [23] hybridization of SOA was proposed.

Bat algorithm is based on the echolocation behaviour of bats with varying pulse rates of emission and loudness [29]. It was successfully applied to numerous problems such as handwritten digit recognition [19], parameter tuning for support vector machine [18], multilevel image thresholding [1], etc. Beside original BA numerous hybridizations and improvements were proposed [22], [2], [20].

Fireworks algorithm was proposed in 2010 and as inspiration explosion of the fireworks was used [14]. During the last years it was intensively used for many different problems [16], [17].

Firefly algorithm was inspired by the social and flashing behavior of fireflies [28]. This algorithm was implemented for many different applications such as image processing [5], [21], for cardinality constrained mean-variance portfolio optimization problem [3], etc. In [24] firefly algorithm was used to improve seeker optimization algorithm.

3 Elephant herding optimization algorithm

Elephant herding optimization algorithm is one of the recent swarm intelligence algorithm that was proposed in 2016 by Wang et. al. [27]. Algorithm was inspired by herding behavior of elephants. Complex behavior was simplified for purpose of swarm intelligence algorithm and described as follows. Elephant population is made of several clans and the elephants in the clans live under the leadership of a matriarch. In each generation constant number of elephants leave clan and go to live far away. It can be easily concluded that behavior in clans represents exploitation while leaving members are used fo exploration.

EHO algorithm is defined as follows. Complete elephant population is firstly divided into k clans.

Each member j of clan i moves according matriarch where matriarch is the elephant c_i with the best fitness value in generation [27]:

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

where $x_{new,ci,j}$ represents new position of elephant j in clan i and $x_{ci,j}$ is its old position, $x_{best,ci}$ is the best solution of clan c_i, $\alpha \in [0, 1]$ is algorithm's parameter which determines the influence of the matriarch and $r \in [0, 1]$ is random number used to improve diversity of the population in the later stages of the algorithm.

Position of the best elephant in clan $x_{best,ci}$ is updated by the following equation [27]:

$$x_{new,ci} = \beta \times x_{center,ci} \quad (2)$$

where $\beta \in [0, 1]$ is the second parameter of the algorithm which controls influence of the $x_{center,ci}$ defined as:

$$x_{center,ci,d} = \frac{1}{n_{ci}} \times \sum_{l=1}^{n_{ci}} x_{ci,l,d} \quad (3)$$

where $1 \leq d \leq D$ is d^{th} dimension and D is total dimension of the space while n_{ci} is the number of elephants in clan i.

Elephants that move away from the clan are used to model exploration. In each clan i some number of the elephants with the worst values of the objective function are moved to the new positions according to the following equation:

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

where x_{min} and x_{max} are lower and upper bound of the search space, respectively. Parameter $rand \in [0, 1]$ is random number chosen from uniform distribution.

Complete elephant herding optimization algorithm is presented in Algorithm 1.

Algorithm 1 Pseudo-code of the EHO algorithm

Initialization
Set generation counter $t = 1$, set maximum generation $MaxGen$
Initialize the population and
repeat
    Sort all the elephants according to their fitness
    for all clans c_i in the population **do**
        for all elephants j in the clan c_i **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
    end for
    for all clans c_i in the population **do**
        Replace the worst elephant in clan c_i by Eq. 4
    end for
    Evaluate population by the newly updated positions
until stop criteria=FALSE
return the best solution among all population

4 Experimental Results

To test our proposed method we used Matlab R2016a and experiments were done on the platform with Intel ® Core™ i7-3770K CPU at 4GHz, 8GB RAM, Windows 10 Professional OS.

We tested elephant herding optimization algorithm on 28 standard benchmark functions proposed for CEC 2013 competition [12].

Parameters of EHO algorithms were set as follows. Parameter α was set to 0.8 and β was 0.001. Population size was 100 and it was divided into 5 clans. Maximal evaluation number of objective function was 500,000. Search range was [-100,100] and 10-dimensional problems were considered. For each function median, standard deviation, the best and the worst value of the function in 51 runs were calculated.

EHO was compared with other approach from literature. We compared it with [31] where PSO was implemented and tested on the same benchmark functions. The obtained results were presented in Table 1.

As it can be seen, both algorithms found the optimal function value for f_1 (sphere). Standard deviation is 0 for both functions which means that EHO as well as PSO successfully determined optimal function value every time. EHO algorithm found exact optimal value for f_5 (different powers function) with standard deviation 0 while PSO found the optimal values but with some deviation. EHO as well as PSO were not able to find nearly good solutions for functions f_2, f_3 and f_4. For this function obviously some specific parameter settings are needed and probably more iterations.

Table 1: Comparison of PSO and EHO

Fun.	Alg.	Optimal	Best	Median	Worst	St.Dev.
f_1	PSO	-1.400E+03	-1.400E+03	-1.400E+03	-1.400E+03	0.000E+00
	EHO	-1.400E+03	-1.400E+03	-1.400E+03	-1.400E+03	0.000E+00
f_2	PSO	-1.300E+03	7.597E+02	3.504E+04	4.755E+05	**7.356E+04**
	EHO	-1.300E+03	**1.853E+02**	**2.934E+04**	**4.129E+05**	8.328E+04
f_3	PSO	-1.200E+03	**-1.200E+03**	2.670E+05	**8.251E+07**	1.656E+07
	EHO	-1.200E+03	-1.158E+03	**1.284E+05**	1.795E+08	**6.834E+06**
f_4	PSO	-1.100E+03	2.454E+02	7.769E+03	1.856E+04	4.556E+03
	EHO	-1.100E+02	**1.195E+02**	**2.359E+03**	**5.270E+03**	**1.631E+03**
f_5	PSO	-1.000E+03	-1.000E+03	-1.000E+03	-1.000E+03	3.142E-05
	EHO	-1.00E+03	-1.000E+03	-1.000E+03	-1.000E+03	**0.00E+00**
f_6	PSO	-9.000E+02	**-**9.000E+02	-8.902E+02	-8.898E+02	4.974E+00
	EHO	-9.000E+02	-9.000E+02	-8.902E+02	-8.898E+02	**4.140E+00**
f_7	PSO	-8.000E+02	-7.974E+02	-7.789E+02	7.434E+02	1.327E+01
	EHO	-8.000E+02	**-7.974E+02**	**-7.870E+02**	**-7.793E+02**	1.013E+01
f_8	PSO	-7.000E+02	-6.789E+02	-6.797E+02	-6.796E+02	6.722E-02
	EHO	-7.000E+02	**-6.797E+02**	-6.797E+02	-6.797E+02	**4.338E-03**
f_9	PSO	-6.000E+02	-5.987E+02	-5.952E+02	-5.929E+02	1.499E+00
	EHO	-6.000E+02	**-5.991E+02**	**-5.969E+02**	-5.929E+02	**1.039E+00**
f_{10}	PSO	-5.000E+02	-4.999E+02	-4.997E+02	**-4.989E+02**	2.713E-01
	EHO	-5.000E+02	**-5.000E+02**	**-4.999E+02**	-4.984E+02	**1.449E-01**
f_{11}	PSO	-4.000E+02	-3.970E+02	-3.891E+02	-3.731E+02	5.658E+00
	EHO	-4.000E+02	**-3.972E+02**	**-3.907E+02**	**-3.781E+02**	**4.198E+00**
f_{12}	PSO	-3.000E+02	-2.970E+02	-2.861E+02	-2.682E+02	6.560E+00
	EHO	-3.000E+02	**-2.971E+02**	**-2.870E+02**	**-2.623E+02**	6.019E+00
f_{13}	PSO	-2.000E+02	-1.946E+02	-1.792E+02	-1.523E+02	9.822E+00
	EHO	-2.000E+02	**-1.992E+02**	**-1.801E+02**	**-1.617E+02**	**8.992E+00**
f_{14}	PSO	-1.000E+02	2.228E+02	7.338E+02	**1.109E+03**	2.335E+02
	EHO	-1.000E+02	**-1.419E+02**	**2.914E+02**	4.990E+02	**1.282E+02**
f_{15}	PSO	1.000E+02	4.372E+02	8.743E+02	1.705E+03	2.507E+02
	EHO	1.000E+02	**4.271E+02**	**5.695E+02**	**1.044E+03**	**2.429E+02**
f_{16}	PSO	2.000E+02	**2**.002E+02	2.005E+02	2.014E+02	2.457E-01
	EHO	2.000E+02	**2.000E+02**	**2.003E+02**	**2.007E+0**2	**1.396E-01**
f_{17}	PSO	3.000E+02	3.104E+02	3.189E+02	3.416E+02	5.873E+00
	EHO	3.000E+02	**3.098E+02**	**3.164E+02**	**3.341E+02**	3.183E+00
f_{18}	PSO	4.000E+02	**4**.125E+02	4.178E+02	4.365E+02	**4.534E+00**
	EHO	4.000E+02	**4.109E+02**	4.178E+02	**4.364E+02**	4.982E+00
f_{19}	PSO	5.000E+02	**5**.003E+02	5.009E+02	**5.019E+02**	3.886E-01
	EHO	5.000E+02	**5.001E+02**	5.009E+02	5.041E+02	**2.153E-01**
f_{20}	PSO	6.000E+02	6.020E+02	6.034E+02	6.040E+02	4.194E-01
	EHO	6.000E+02	**6.017E+02**	**6.025E+02**	6.034E+02	**4.006E-01**
f_{21}	PSO	7.000E+02	1.100E+03	1.100E+03	1.100E+03	0.00E+00
	EHO	7.000E+02	1.100E+03	1.100E+03	1.100E+03	0.00E+00
f_{22}	PSO	8.000E+02	1.206E+03	1.706E+03	2.388E+03	3.431E+02
	EHO	8.000E+02	**1.190E+03**	**1.428E+03**	**1.998E+03**	**3.083E+02**
f_{23}	PSO	9.000E+02	1.016E+03	1.810E+03	2.776E+03	**3.596E+02**
	EHO	9.000E+02	**9.991E+02**	**1.193E+03**	**1.987E+03**	5.121E+02
f_{24}	PSO	1.000E+03	1.162E+03	1.214E+03	1.222E+03	9.166E+00
	EHO	1.000E+03	**1.091E+03**	**1.179E+03**	**1.207E+03**	**6.917E+00**
f_{25}	PSO	1.100E+03	1.300E+03	1.309E+03	1.320E+03	**5.943E+00**
	EHO	1.100E+03	**1.220E+03**	**1.300E+03**	**1.312E+03**	6.152E+00
f_{26}	PSO	1.200E+03	1.307E+03	1.400E+03	1.520E+03	5.513E+01
	EHO	1.200E+03	**1.193E+03**	**1.307E+03**	**1.400E+03**	**1.131E+01**
f_{27}	PSO	1.300E+03	1.602E+03	1.636E+03	1.898E+03	7.359E+01
	EHO	1.300E+03	**1.521E+03**	**1.596E+03**	**1.705E+03**	**5.251E+01**
f_{28}	PSO	1.400E+03	**1**.500E+03	1.700E+03	2.009E+03	8.362E+01
	EHO	1.400E+03	**1.400E+03**	**1.698E+03**	**2.001E+03**	**7.672E+01**

For functions f_6, f_8 and f_{21} PSO and EHO reached the same median and the best solution. Interesting is that for f_{21} both algorithm had standard deviation 0 which means that obtained solution is probably some local optimum. For functions f_{10}, f_{16} and f_{28} EHO successfully found at least once optimal solution while PSO was not. For all other functions EHO reached better minimal, maximal and median solution and standard deviation was lower in the most cases. Smaller standard deviation around bad solution is not an advantage of the PSO algorithm. Based on the results presented in Table 1 we can conclude that EHO perform better then PSO and it was shown good characteristics.

5 Conclusion

In this paper we tested noves novel swarm optimization algorithms, elephant herding algorithm. The algorithms was tested on 28 CEC 2013 benchmark functions. Based on the experimental results we concluded that EHO has good characteristics as optimization algorithm and it perform better than PSO algorithm that was used for comparison. In further work, modification or hybridization of EHO algorithm can be proposed and tested against several other swarm optimization algorithms.

References:

[1] A. Alihodzic and M. Tuba, "Bat algorithm (BA) for image thresholding," *Recent Researches in Telecommunications, Informatics, Electronics and Signal Processing*, pp. 17–19, 2013.

[2] N. Bacanin and M. Tuba, "Firefly algorithm for cardinality constrained mean-variance portfolio optimization problem with entropy diversity constraint," *The Scientific World Journal, special issue Computational Intelligence and Metaheuristic Algorithms with Applications,*, vol. 2014, no. Article ID 721521, p. 16, 2014.

[3] ——, "Firefly algorithm for cardinality constrained mean-variance portfolio optimization problem with entropy diversity constraint," *The Scientific World Journal*, vol. 2014, 2014.

[4] I. Brajevic and M. Tuba, "An upgraded artificial bee colony algorithm (ABC) for constrained optimization problems," *Journal of Intelligent Manufacturing*, vol. 24, no. 4, pp. 729–740, August 2013.

[5] ——, *Cuckoo Search and Firefly Algorithm: Theory and Applications*. Springer International Publishing, 2014, ch. Cuckoo Search and Firefly Algorithm Applied to Multilevel Image Thresholding, pp. 115–139.

[6] I. Brajevic, M. Tuba, and M. Subotic, "Improved artificial bee colony algorithm for constrained problems," in *Proceedings of the 11th WSEAS international conference on evolutionary computing*. World Scientific and Engineering Academy and Society (WSEAS), 2010, pp. 185–190.

[7] C. Dai, W. Chen, Y. Song, and Y. Zhu, "Seeker optimization algorithm: a novel stochastic search algorithm for global numerical optimization," *Journal of Systems Engineering and Electronics*, vol. 21, no. 2, pp. 300–311, 2010.

[8] R. Jovanovic and M. Tuba, "An ant colony optimization algorithm with improved pheromone correction strategy for the minimum weight vertex cover problem," *Applied Soft Computing*, vol. 11, no. 8, pp. 5360–5366, December 2011.

[9] ——, "Ant colony optimization algorithm with pheromone correction strategy for the minimum connected dominating set problem," *Computer Science and Information Systems (ComSIS)*, vol. 10, no. 1, pp. 133–149, January 2013.

[10] D. Karaboga and B. Akay, "A modified artificial bee colony (ABC) algorithm for constrained optimization problems," *Applied Soft Computing*, vol. 11, no. 3, pp. 3021–3031, 2011.

[11] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceedings of the IEEE International Conference on Neural Networks (ICNN '95)*, vol. 4, pp. 1942–1948, 1995.

[12] J. Liang, B. Qu, P. Suganthan, and A. G. Hernández-Díaz, "Problem definitions and evaluation criteria for the cec 2013 special session on real-parameter optimization," *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report*, vol. 201212, 2013.

[13] M. Subotic and M. Tuba, "Parallelized multiple swarm artificial bee colony algorithm (MS-ABC) for global optimization," *Studies in Informatics and Control*, vol. 23, no. 1, pp. 117–126, 2014.

[14] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," *Advances in Swarm Intelligence, LNCS*, vol. 6145, pp. 355–364, June 2010.

[15] I. Tsoulos and A. Stavrakoudisb, "Enhancing pso methods for global optimization," *Applied Mathematics and Computation*, vol. 216, no. 10, pp. 2988–3001, July 2010.

[16] E. Tuba, M. Tuba, and M. Beko, *Support Vector Machine Parameters Optimization by Enhanced Fireworks Algorithm*. Springer International Publishing, 2016, vol. 9712, pp. 526–534.

[17] E. Tuba, M. Tuba, and E. Dolicanin, "Adjusted fireworks algorithm applied to retinal image registration," *Studies in Informatics and Control*, vol. 26, no. 1, pp. 33–42, 2017.

[18] E. Tuba, M. Tuba, and D. Simian, "Adjusted bat algorithm for tuning of support vector machine parameters," in *Congress on Evolutionary Computation (CEC),*. IEEE, 2016, pp. 2225–2232.

[19] ——, "Handwritten digit recognition by support vector machine optimized by bat algorithm," in *24th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision,(WSCG 2016)*, 2016, pp. 369–376.

[20] M. Tuba, "Asymptotic behavior of the maximum entropy routing in computer networks," *Entropy*, vol. 15, no. 1, pp. 361–371, January 2013.

[21] ——, "Multilevel image thresholding by nature-inspired algorithms-a short review." *The Computer Science Journal of Moldova*, vol. 22, no. 3, pp. 318–338, 2014.

[22] M. Tuba and N. Bacanin, "Artificial bee colony algorithm hybridized with firefly metaheuristic for cardinality constrained mean-variance portfolio problem," *Applied Mathematics & Information Sciences*, vol. 8, no. 6, pp. 2831–2844, November 2014.

[23] ——, "Improved seeker optimization algorithm hybridized with firefly algorithm for constrained optimization problems," *Neurocomputing*, vol. 143, pp. 197–207, 2014.

[24] ——, "Improved seeker optimization algorithm hybridized with firefly algorithm for constrained optimization problems," *Neurocomputing*, vol. 143, pp. 197–207, 2014.

[25] M. Tuba, N. Bacanin, and N. Stanarevic, "Adjusted artificial bee colony (ABC) algorithm for engineering problems," *WSEAS Transactions on Computers*, vol. 11, no. 4, pp. 111–120, 2012.

[26] M. Tuba, R. Jovanovic, and S. SERBIA, "An analysis of different variations of ant colony optimization to the minimum weight vertex cover problem," *WSEAS Transactions on Information Science and Applications*, vol. 6, no. 6, pp. 936–945, 2009.

[27] G.-G. Wang, S. Deb, X.-Z. Gao, and L. D. S. Coelho, "A new metaheuristic optimisation algorithm motivated by elephant herding behaviour," *International Journal of Bio-Inspired Computation*, vol. 8, no. 6, pp. 394–409, 2016.

[28] X.-S. Yang, "Firefly algorithms for multimodal optimization," *Stochastic Algorithms: Foundations and Applications, LNCS*, vol. 5792, pp. 169–178, 2009.

[29] ——, "A new metaheurisitic bat-inspired algorithm," *Studies in Computational Intelligence*, vol. 284, pp. 65–74, 2010.

[30] ——, "Efficiency analysis of swarm intelligence and randomization techniques," *Journal of Computational and Theoretical Nanoscience*, vol. 9, no. 2, pp. 189–198, 2012.

[31] M. Zambrano-Bigiarini, M. Clerc, and R. Rojas, "Standard particle swarm optimisation 2011 at CEC-2013: A baseline for future PSO improvements," in *IEEE Congress on Evolutionary Computation (CEC 2013)*. IEEE, 2013, pp. 2337–2344.