Modified Moth Search Algorithm for Global Optimization Problems

IVANA STRUMBERGER
Singidunum University
Faculty of Informatics and Computing
Danijelova 32, 11000 Belgrade
SERBIA
istrumberger@singidunum.ac.rs

NEBOJSA BACANIN
Singidunum University
Faculty of Informatics and Computing
Danijelova 32, 11000 Belgrade
SERBIA
nbacanin@singidunum.ac.rs

Abstract: This paper presents modified moth search algorithm for solving global optimization problems. Moth search algorithm is novel swarm intelligence metaheuristics. By analyzing original moth search approach, we noticed some deficiencies in the search process of subpopulation 2. Modified moth search addresses these weaknesses. To prove the robustness of our approach we tested our algorithm on six standard global optimization benchamarks and performed comparative analysis with original moth search, as well as with other five state-of-the-art metaheuristics. Testing results show that in average modified moth search outperforms other approaches included in comparative analysis.

Key–Words: moth search algorithm, global optimization, swarm intelligence, metaheuristics

1 Introduction

Optimization is one of the most widely used research domain since almost any real life task can be modeled as an optimization problem. Sometimes the problem is inherently a numerical optimization problem and sometimes elaborate adjustments to a mathematical model are necessary.

Numerical optimization problems can roughly be divided into combinatorial and continuous problems. Continuous problems can further be divided in two groups: unconstrained and constrained. Unconstrained (global optimization) is branch of applied mathematics and numerical analysis that tackles with the global optimization of a function or a set of functions according to some criteria.

Unconstrained (or bound constrained) optimization can be defined as D-dimensional minimization or maximization problem:

$$\min(\max) f(x),\ x = (x_1, x_2, x_3, ..., x_D) \in S, \quad (1)$$

where x is a real vector with $D \geq 1$ components and $S \in R^D$ is an D-dimensional hyper-rectangular search space constrained by lower and upper bounds:

$$lb_i \leq x_i \leq ub_i,\ \ i \in [1, D] \quad (2)$$

Many numerical problems belong to the category of NP hard optimization. One of the most well-known representatives of this class of problems is traveling salesman problem (TSP). NP hard problems can not be solved in reasonable amount of computational time with traditional deterministic methods and algorithms and for its solving many metaheuristics approaches have been evolved. Metaheuristics are capable of finding suboptimal (satisfying) solutions in acceptable amount of time.

Nature-inspired metaheuristics mimic behavior of natural systems. In its execution they use guided random search process and the mechanism that directs the search is adopted from nature. Swarm intelligence algorithms which represent the newer branch of nature inspired algorithms, simulate collective behavior of group of organisms such as flock of girds and fish, colony of bees and ants, groups of cuckoo birds and bats, herds of elephants, etc.

One of the most known representatives of swarm intelligence is artificial bee colony (ABC) algorithm that simulates behavior of bee swarm [1]. ABC has been implemented for many numerical benchmark optimization tasks [2], [3], as well as on many real life problems [4]. Firefly algorithm (FA) was originally proposed by Yang [5] for multimodal optimization. This swarm intelligence metaheuristic proved to be robust optimizer for wide variety of tasks [6]. Also, according to the literature review, FA showed great potential in hybrid approaches.

The main source of inspiration for the emergence of the fireworks algorithm (FWA) was the process of fireworks explosion. FWA was firstly proposed for global optimization [7] and since then many successful applications for benchmark [8] and real life problems [9], [10], [11], [12], [13] can be found in the literature review. Bat algorithm (BA), which is based on so-called echolocation of the bats, is a relatively new bio-inspired algorithm [14], [15]. This approach has many implementation for problems such as support vector machine parameter tuning [16], RFID network planning [17] and portfolio optimization [18]. Firefly algorithm was proposed by Yang and it was applied to numerous problems [19], [20].

Wang et al. [21] proposed an elephant herding optimization (EHO) that is inspired by a herding behavior of elephants. By investigating the relevant literature, it can be seen that EHO was applied on support vector machine parameters tuning [22], multilevel image thresholding [23], computer aided diagnostics [24], robot path planning [25] and static drone placement [26].

In this paper, we propose modified moth search algorithm (MMS) for global optimization problems. The moth search (MS) algorithm is novel nature-inspired metaheuristics proposed by Wang in 2016 [27].

The rest of the paper is structured as follows. After Introduction, basic and modified moth search algorithms are presented in Section 2. In Section 3, results of empirical tests for global optimization, as well as, comparative analysis with other approaches are given, while Section 4 concludes this paper and gives guidelines for future work.

2 Modified moth search optimization algorithm for global problems

Moth Search (MS) algorithm, proposed by Wang in 2016 for global optimization problems [27], is inspired by phototaxis phenomena and Lévy flights of the moths. Moths belong to the order Lepidoptera. Lepidoptera (which includes moths and butterflies) is the second largest order in the class Insecta. MS algorithm was compared with five state-of-the-art metaheuristic optimization algorithms through an array of experiments on fourteen basic benchmarks, eleven IEEE CEC 2005 complicated benchmarks and seven IEEE CEC 2011 real world problems, where it showed great potential for tackling global optimization tasks [27].

Moths tend to fly around and towards the light source and this phenomena is known as phototaxis. Since this behavior is still unknown, there have been various hypothesis to explain this phenomenon. One of the hypothesis is that celestial is used in transverse orientation while flying. The moths will fly in a straight line so as to remain at a fixed angle to the celestial light, like the moon [27].

Lévy flights, as one of the most important flight patterns in natural surroundings, is considered as another characteristic of moths. As an example, species like Drosophila (”fruit fly”) fly in the form of Lévy flights that can be approximated to be power law distributed over a range of scales with the feature of exponents close to 3/2 [28]. In [29], Reynolds et al. have conducted experiments which indicate that some of the complex flight patterns are in compliance with the usage of an optimal biased scale-free (Lévy flights) searching technique.

Phototaxis and Lévy flights from moths in nature were used for modeling two main processes of MS algorithm: exploitation (intensification) and exploration (diversification).

The moths that are closer to the best moth (light source) in the population will fly around the best moth in the form of Lévy flights. This type of behavior is described in the following equation [27]:

$$x_i^{t+1} = x_i^t + \alpha L(s), \qquad (3)$$

where x_i^{t+1} is updated position and x_i^t is original position of moth i in current generation t, respectively. Step drawn from Lévy distribution is denoted as $L(s)$. Parameter α is scale factor whose value depends on the optimization problem. In the original MS algorithm, α was given as [27]:

$$\alpha = S_{max}/t^2, \qquad (4)$$

where S_{max} is the maximum walk step and its value also depends on the problem in hand.

Lévy distribution given in Eq. (3) can be expressed as follows [27]:

$$L(s) = \frac{(\beta-1)\Gamma(\beta-1)\sin(\frac{\pi(\beta-1)}{2})}{\pi s^{\beta}}, \qquad (5)$$

where Γ is gamma function and s is greater than 0.

Moths that are far away from the light source (best moth in population) will fly towards the light source in line. This process can be described using the following equation [27]:

$$x_i^{t+1} = \lambda \times (x_i^t + \phi \times (x_{best}^t - x_i^t)), \qquad (6)$$

where x_{best}^t denotes best moth in generation t and ϕ and λ are acceleration and scale factors, respectively.

Also, the moth can fly in direction of the final position that is beyond the best moth in the population (light source). This flight pattern is described as [27]:

$$x_i^{t+1} = \lambda \times (x_i^t + \frac{1}{\phi} \times (x_{best}^t - x_i^t)) \qquad (7)$$

In the original paper [27], for simplicity reasons, the whole moth population is divided into two equal subpopulations according to their fitness. Positions of individuals in the subpopulation 1 (moths with greater fitness) are being updated using Lévy flights (Eq. (3)), while moth positions in the subpopulation 2 (moths with lower fitness) are being updated by using Eq. (6) or Eq. (7) with possibility of 50% [27].

By conducting empirical experiments and according to results reported in [27], MS algorithm proved to be efficient method for solving global optimization problems. However, we noticed some deficiencies in MS's execution.

Updated positions of moths in subpopulation 2 are strongly influenced by current best solution and this is good in late iterations, when we suppose that the algorithm has found right part of the search space. However, in early iterations this can lead to the premature convergence and to the worse mean values.

To overcome these deficiencies, we introduced third search equation in subpopulation 2 which performs random exploration of the search space:

$$x_{i,j} = lb_j + rand(0,1) * (ub_j - lb_j), \qquad (8)$$

where $x_{i,j}$ is the j-the parameter of th i-th moth in the subpopulation 2, $rand(0,1)$ is a random real number between 0 and 1, and ub_j and lb_j are upper and lower bounds of the j-th solution parameter respectively.

Modified MS (MMS) algorithm employs Eq. (6), Eq. (7) or Eq. (8) in subpopulation 2 with possibilities of 40%, 40% and 20%, respectively.

3 Experimental results and discussion

We conducted experiments on six standard unconstrained benchmark functions: $Ackley$ (f0), $Dixon\&Price$ (f1), $Fletcher - Powell$ (f2), $Griewank$ (f3), $Perm$ (f4) and $Step$ (f5). Comparative analysis was performed with original MS algorithm [27], as well as with five other state-of-the-art algorithms: ABC [30], biogeography-based optimization (BBO) [31], differential evolution (DE) [32], particle swarm optimization (PSO) [33] and stud genetic algorithms (SGA) [34].

Basic parameters of MMS are set as in original MS implementation [27]: population size $N = 50$, the number of moths kept in each generation 2, index $\beta = 1.5$, max walk step $S_{max} = 1.0$ and acceleration factor $\phi = (5^{1/2} - 1)/2 \cong 0.618$. Number of function evaluations (FEs) is considered as termination condition and it is set to 10^4.

The algorithm was executed in 30 independent runs, and we measured best, mean, worst and standard deviation values. Comparative analysis of best, mean, worst and standard deviation values are given in Tables 1, 2, 3 and 4, respectively.

Table 1: Comparison of best values

F.	ABC	BBO	DE	PSO	SGA	MS	MMS
f0	13.35	2.51	16.48	17.05	2.51	2.1E8	**1.8E-8**
f1	14.0E6	4.6E3	1.2E6	4.1E6	2.1E3	**0.67**	0.73
f2	1.2E5	3.7E4	1.7E5	3.3E5	3.9E4	3.4E4	**9.82E3**
f3	30.93	1.79	10.96	34.86	1.37	**1.00**	1.12
f4	1.4E45	6.0E51	3.7E37	3.7E43	6.0E51	3.0E32	**5.6E30**
f5	16.00	1.00	6.00	20.00	1.00	1.00	1.00

Table 2: Comparison of mean values

F.	ABC	BBO	DE	PSO	SGA	MS	MMS
f0	16.45	3.77	18.26	18.44	4.33	2.4E6	**9.5E-7**
f1	4.6E7	7.7E4	3.8E6	1.4E7	1.1E4	**0.67**	0.78
f2	2.7E5	**7.0E4**	2.5E5	5.0E5	8.3E4	1.6E5	9.3E4
f3	85.88	3.33	21.42	73.02	2.19	**1.00**	1.15
f4	1.2E51	6.1E51	4.5E45	4.5E47	6.0E51	2.5E37	**8.7E33**
f5	35.68	1.16	9.26	27.50	1.44	1.00	1.00

Table 3: Comparison of worst values

F.	ABC	BBO	DE	PSO	SGA	MS	MMS
f0	17.85	5.77	18.97	18.88	6.42	1.3E5	**2.1E-6**
f1	1.0E8	2.8E5	9.0E6	3.2E7	4.0E4	**0.67**	0.75
f2	4.0E5	1.2E5	3.3E5	8.2E5	1.8E5	3.9E5	**1.1E5**
f3	136.66	5.83	31.13	104.06	3.98	**1.00**	1.18
f4	6.0E51	1.0E52	6.0E46	3.1E48	6.0E51	3.5E38	**6.6E37**
f5	49.00	2.00	14.00	36.00	4.00	1.00	1.00

According to the presented results, our MMS algorithm in average outperforms all other approaches, including original MS metaheuristics. Only in the case of f1 ($Dixon\&Price$) and f3 ($Griewank$) test functions, original MS performs better than MMS.

Thus, we conclude that in average, our modification enhanced basic MS algorithm by introducing more exploration power in the search process, and in

Table 4: Comparison of standard deviation values

F.	ABC	BBO	DE	PSO	SGA	MS	MMS
f0	0.91	0.66	0.45	0.44	0.83	2.6E6	**1.5E-6**
f1	2.4E7	6.7E4	1.7E6	5.8E6	8.2E3	**5.7E5**	5.9E-5
f2	7.6E4	2.0E4	4.0E4	1.1E5	3.0E4	7.0E4	**1.92E4**
f3	24.39	0.92	5.29	12.52	0.58	**3.6E15**	5.6E-3
f4	1.6E51	5.7E50	1.1E46	6.9E47	1.3E36	5.6E37	**9.5E35**
f5	6.76	0.37	1.71	3.64	0.70	0.00	0.00

this way the trade-off between exploitation and exploration is improved.

4 Conclusion

This paper presents modified moth search algorithm adjusted for solving global optimization tasks. By empirical and theoretical analysis of original moth search algorithm, we noticed some weaknesses in the search process of subpopulation 2 that is too much oriented towards the current best solution. To address these deficiencies, we enhanced exploration power in the subpopulation 2 by introducing random search mechanism.

Comparative analysis with original moth search and five other metaheuristics on six standard global benchmarks was performed. Results of empirical tests proved that our approach has potential in tackling global optimization problems.

Acknowledgements: This research is supported by the Ministry of Education, Science and Technological Development of Republic of Serbia, Grant No. III-44006.

References:

[1] D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” *Technical Report - TR06*, pp. 1–10, 2005.

[2] I. Brajevic, M. Tuba, and M. Subotic, “Improved artificial bee colony algorithm for constrained problems,” in *Proceedings of the 11th WSEAS international conference on neural networks*, pp. 185–190, 2010.

[3] 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.

[4] N. Bacanin, M. Tuba, and I. Strumberger, “RFID network planning by ABC algorithm hybridized with heuristic for initial number and locations of readers,” in *2015 17th UKSim-AMSS International Conference on Modelling and Simulation (UKSim)*, pp. 39–44, March 2015.

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

[6] R. K. Sahu, S. Panda, and S. Padhan, “A hybrid firefly algorithm and pattern search technique for automatic generation control of multi area power systems,” *International Journal of Electrical Power & Energy Systems*, vol. 64, pp. 9–23, January 2015.

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

[8] N. Bacanin, M. Tuba, and M. Beko, “Hybridized fireworks algorithm for global optimization,” in *Mathematical Methods and Systems in Science and Engineering*, pp. 108–114, 2015.

[9] E. Tuba, M. Tuba, and M. Beko, “Node localization in ad hoc wireless sensor networks using fireworks algorithm,” in *5th International Conference on Multimedia Computing and Systems (ICMCS)*, pp. 223–229, IEEE, 2016.

[10] N. Bacanin and M. Tuba, “Fireworks algorithm applied to constrained portfolio optimization problem,” in *Proceedings of the 2015 IEEE Congress on Evolutionary Computation (CEC 2015)*, pp. 1242–1249, May 2015.

[11] E. Tuba, M. Tuba, and M. Beko, “Node localization in ad hoc wireless sensor networks using fireworks algorithm,” in *Proceedings of the 5th International Conference on Multimedia Computing and Systems (ICMCS)*, pp. 223–229, September 2016.

[12] M. Tuba, N. Bacanin, and A. Alihodzic, “Multilevel image thresholding by fireworks algorithm,” in *25th International Conference Radioelektronika*, pp. 326–330, IEEE, 2015.

[13] 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.

[14] X.-S. Yang, “A new metaheuristic bat-inspired algorithm,” *Studies in Computational Intelligence*, vol. 284, pp. 65–74, November 2010.

[15] A. Alihodzic and M. Tuba, "Improved hybridized bat algorithm for global numerical optimization," in *16th International Conference on Computer Modelling and Simulation (UKSim)*, pp. 57–62, IEEE, 2014.

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

[17] M. Tuba and N. Bacanin, "Hybridized bat algorithm for multi-objective radio frequency identification (RFID) network planning," in *Proceedings of the 2015 IEEE Congress on Evolutionary Computation (CEC 2015)*, May 2015.

[18] I. Strumberger, N. Bacanin, and M. Tuba, "Constrained portfolio optimization by hybridized bat algorithm," in *2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS)*, pp. 83–88, Jan 2016.

[19] E. Tuba, M. Tuba, and M. Beko, "Mobile wireless sensor networks coverage maximization by firefly algorithm," in *27th International Conference Radioelektronika*, pp. 1–5, IEEE, 2017.

[20] E. Tuba, M. Tuba, and M. Beko, "Two stage wireless sensor node localization using firefly algorithm," in *Smart Trends in Systems, Security and Sustainability, LNNS*, vol. 18, pp. 113–120, Springer, 2018.

[21] G.-G. Wang, S. Deb, and L. dos S. Coelho, "Elephant herding optimization," in *Proceedings of the 2015 3rd International Symposium on Computational and Business Intelligence (ISCBI)*, pp. 1–5, December 2015.

[22] E. Tuba and Z. Stanimirovic, "Elephant herding optimization algorithm for support vector machine parameters tuning," in *Proceedings of the 2017 International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pp. 1–5, June 2017.

[23] E. Tuba, A. Alihodzic, and M. Tuba, "Multilevel image thresholding using elephant herding optimization algorithm," in *Proceedings of 14th International Conference on the Engineering of Modern Electric Systems (EMES)*, pp. 240–243, June 2017.

[24] E. Tuba, I. Ribic, R. Capor-Hrosik, and M. Tuba, "Support vector machine optimized by elephant herding algorithm for erythemato-squamous diseases detection," *Procedia Computer Science*, vol. 122, pp. 916–923, 2017.

[25] A. Alihodzic, E. Tuba, R. Capor-Hrosik, E. Dolicanin, and M. Tuba, "Unmanned aerial vehicle path planning problem by adjusted elephant herding optimization," in *Proceedings of the 25th Telecommunications Forum (TELFOR)*, pp. 804–807, November 2017.

[26] I. Strumberger, N. Bacanin, M. Beko, S. Tomic, and M. Tuba, "Static drone placement by elephant herding optimization algorithm," in *Proceedings of the 25th Telecommunications Forum (TELFOR)*, pp. 808–811, November 2017.

[27] G.-G. Wang, "Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems," *Memetic Computing*, Sep 2016.

[28] A. M. Reynolds, H. B. C. Jones, J. K. Hill, A. J. Pearson, K. Wilson, S. Wolf, K. S. Lim, D. R. Reynolds, and J. W. Chapman, "Evidence for a pervasive idling-mode activity template in flying and pedestrian insects," *Open Science*, vol. 2, no. 5, 2015.

[29] A. Reynolds, D. Reynolds, A. Smith, G. Svensson, and C. Lfstedt, "Appetitive flight patterns of male agrotis segetum moths over landscape scales," *Journal of Theoretical Biology*, vol. 245, no. 1, pp. 141 – 149, 2007.

[30] 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, pp. 459–471, april 2007.

[31] D. Simon, "Biogeography-based optimization," *IEEE Transactions on Evolutionary Computation*, vol. 12, pp. 702–713, Dec 2008.

[32] 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.

[33] J. Kennedy and R. Eberhart, "Particle swarm optimization," vol. 4, pp. 1942–1948, 1995.

[34] W. Khatib and P. J. Fleming, "The stud GA: A mini revolution?," *Eiben A.E., Bck T., Schoenauer M., Schwefel HP. (eds) Parallel Problem Solving from Nature PPSN V. PPSN 1998.*, vol. 1498, pp. 683–691, 1998.