# Moth Search Algorithm for Drone Placement Problem

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

MARKO SARAC
Singidunum University
Faculty of Informatics and Computing
Danijelova 32, 11000 Belgrade
SERBIA
msarac@singidunum.ac.rs

DUSAN MARKOVIC
Singidunum University
Faculty of Informatics and Computing
Danijelova 32, 11000 Belgrade
SERBIA
dmarkovic@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 implementation of the moth search algorithm adjusted for solving static drone location problem. The optimal location of drones is one of the most important issues in this domain, and it belongs to the group of NP-hard optimization. The objective of the model applied in this paper is to establish monitoring all targets with the least possible number of drones. For testing purposes, we used problem instance with 30 uniformly distributed targets in the network domain. According to the results of simulations, where moth search algorithm established full coverage of targets, this approach shows potential in dealing with this kind of problem.

*Key–Words:* moth search algorithm, metaheuristics, NP hardness, swarm intelligence, optimization

## 1 Introduction

The applications of flexible flying drones have increased with the emerging of low energy consumption machines, processing devices with high performance and availability of light materials. Drones can be used in a wide variety of applications, such as vehicle tracking, the traffic management, fire detection, military operations,etc. [1].

Drones are mostly used to monitor targets, which are considered as points that can be static or mobile, depending on the scenario. Similar to anchor nodes targeting unknown nodes in wireless sensor network, drones deployment must be placed in a way to cover multiple targets, where each target must be covered by at least one drone [2].

The optimal placement of drones is one of the most important challenges in this domain and belong to the group of NP-hard problems [3]. For solving NP-hard problems, metaheuristics can obtain satisfying results, while standard, deterministic methods can not be applied. One of the most promising group of metaheuristics approaches is swarm intelligence.

Swarm intelligence simulate group of organisms from the nature, such as flock of birds and fish, herd of elephants, groups of bats and cuckoos, etc. Artificial bee colony (ABC) models the behavior of honey bee swarm [4], and proved to be robust optimization technique [5], [6].

Firefly algorithm (FA) emulates lighting behavior of fireflies [7], and has been implemented for a wide variety of problems [8], [9], [10]. Cuckoo search (CS) metaheuristics [11] is based on similar principles as FA and has also been applied to different real-world tasks [12], [13]. Firework algorithm (FWA) was inspired by the process of fireworks' explosion [14], and became on of the most popular algorithm with many versions [15], [16], [17], [18], [19]. Bat algorithm (BA) simulates group of bats and their characteristics of echolocation [20], and shows outstanding performance [21], [22], [23]. Brain storm optimization algorithm is based on the human idea generation process and it was applied to real world problem[24], [25], [26].

In this paper, we propose moth search (MS) algorithm adopted for solving static drone location problem. MS algorithm was proposed in 2016 by Wang for global optimization problems [27].

The structure of this paper is as follows: after Introduction, in Section 2, we show mathematical formulation of static drone placement problem, MS metaheuristics is presented in Section 3, Section 4 show empirical results, while Section 5 concludes this paper.

## 2 Formulation of static drone location problem

This section presents mathematical formulation of the static drone location problem (SDLP). In our implementation, we used similar problem formulation as in [28].

Rectangular two-dimensional terrain with length $x_{max}$ and width $y_{max}$ represents the flying region of the drone $u$. The radius $r$ and 2D coordinates $(x, y)$ determine the position of each drone $u$ in the monitoring domain. Set of available drones can be denoted as $U$, while $T$ can be used to indicate the set of targets to be monitored by the available drones.

With the assumption that the drone $u$ with radius $r^u$ is located in the terrain at coordinates $(x_u, y_u)$, and that there is a target $t_i$ with coordinates $(Y_{t_i}, Y_{t_i})$ , the distance $D_{t_i}^{x_u,y_u}$ between $u$ and $t_i$ can be calculated as:

$$D_{t_i}^{x_u,y_u} = \sqrt{(X_{t_i} - x_u)^2 + (Y_{t_i} - y_u)^2} \tag{1}$$

Moreover, Each drone $u$ with radius $r^u$ is characterized with the visibility $\theta$, that exemplifies a disk in the plane. In mathematical formulation of drone coverage of targets, two main issues should be considered. In order to monitor the targets, coordinates $(x_u, y_u)$ of each drone $u \in U$ with radius $r^u$ should be determined. With known location $(x_u, y_u)$ of the drone $u \in U$ with radius $r^u$, we need to determine which target $t_i \in T$ is monitored by the drone $u \in U$.

The mathematical formulation of two above mentioned issues can be represented as decision variables [28]:

$$\delta_{xy}^u = \begin{cases} 1, & \text{if the drone } u \text{ is located at } (x, y) \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

and

$$\gamma_{t_i}^u = \begin{cases} 1, & \text{if the target } t_i \text{ is observed by the drone } u \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

The objective function of the mathematical model employed in this paper is to monitor all targets with the least possible number of drones. This model can be expressed as follows [28]:

$$\min f(\delta) = \sum_{(x,y)} \sum_{u \in U} \delta_{xy}^u \tag{4}$$

s.t.

$$\sum_{x,y} \delta_{xy}^u \leq 1 \quad \forall u \in U \tag{5}$$

$$\gamma_{t_i}^u \leq \sum_{(x,y)} \delta_{xy}^u \left( \frac{r^u}{D_{t_i}^{uxy}} \right) \quad \forall u \in U, t_i \in T \tag{6}$$

$$\sum_{u \in U} \gamma_{t_i}^u \geq 1 \quad \forall t_i \in T \tag{7}$$

$$\delta_{xy}^u \in \{0, 1\}, \;\; \forall (x, y), \;\; 1 \leq x \leq x_{max} \tag{8}$$

$$1 \leq y \leq y_{max}, \;\; u \in U \tag{9}$$

$$\gamma_{t_i}^u \in \{0, 1\}, \;\; \forall t_i \in T, u \in U \tag{10}$$

The objective function showed in Eq.(4) deals with the minimization of the number of employed drones. Assurance that the drone $u$ is positioned in at most one location is provided by using constraint showed in Eq. (5). Condition showed in Eq. (6) is used to set the value of decision variable $\gamma_{t_i}^u$. The variable $\gamma_{t_i}^u$ takes the value of 0, if the radius of drone $u$ is lesser than the distance between the target $t_i$ and the drone $u$, and vice-versa. Condition that the each target $t_i$ is being monitored by at least one drone is specified in Eq. (7), while constraints (8) - (10) determine the domain of the variables.

## 3 Moth search algorithm

MS algorithm was inspired by the the phototaxis and Lévy flights of the moths. This relatively new algorithm was developed in 2016 by Wang [27]. MS algorithm belongs to the group of swarm intelligence metaheuristics, and was primarily implemented for global optimization problems [27].

In order to demonstrate the performance of MS algorithm, its very first implementation was compared with five state-of-the-art metaheuristics through an array of experiments on fourteen basic benchmarks, eleven IEEE CEC 2005 complicated benchmarks and seven IEEE CEC 2011 real world problems [27]. The results of comparative analysis have shown great potential of the MS algorithm for tackling global optimization tasks [27].

Moths have two distinguishing characteristics that differentiate them from other similar species. First characteristic of moths, phototaxis, represents a phenomena, where moths tend to fly around the light source [29]. The other characteristic of the moths, Lévy flights, as one of the most important flight patterns in natural surroundings, was considered for MS algorithm [27].

Lévy flights define the type of random walk which step length is drawn from Lévy distribution. The Lévy distribution which can be modeled in the form of a power-law formula [27]:

$$L(s) \sim |s|^{-\beta}, \tag{11}$$

where $\beta \in [0, 3]$ denotes an index.

According to the analysis of moths fly patterns [30], moths use Lévy flights movements with $\beta \approx 1.5$. For that reason, in our experiments, we set the value of parameter $\beta$ to 1.5.

Some other swarm intelligence approaches also use Lévy flights search, like cuckoo search (CS) [11], FA [7] and krill herd (KH) [31] metaheuristics.

Two above mentioned characteristics of moths (phototaxis and Lévy flights) were used to model two stepping stones of every swarm intelligence metaheuristics - intensification and diversification.

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

$$x_i^{t+1} = x_i^t + \alpha L(s), \tag{12}$$

where $x_i^{t+1}$ is the updated position of moth $i$ and $x_i^t$ is the original position of moth $i$ in current generation $t$, respectively. Step drawn from Lévy distribution is denoted as $L(s)$, and the parameter $\alpha$ is scale factor whose value depends on the optimization problem. In the original MS's implementation, $\alpha$ was given as [27]:

$$\alpha = S_{max}/t^2, \tag{13}$$

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

Lévy distribution given in Eq. (12) can be calculated as [27]:

$$L(s) = \frac{(\beta - 1)\Gamma(\beta - 1)\sin(\frac{\pi(\beta-1)}{2})}{\pi s^\beta}, \tag{14}$$

where $\Gamma$ is the gamma function and $s$ is greater than 0 [27].

Moths that are far from the light source (best moth in the population) will fly towards the light source with trajectory of a line.This type of fly can be mathematically expressed as [27]:

$$x_i^{t+1} = \lambda \times (x_i^t + \phi \times (x_{best}^t - x_i^t)), \tag{15}$$

where $x_{best}^t$ denotes best moth in generation $t$ and $\phi$ and $\lambda$ are acceleration and scale factors, respectively.

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]:

$$x_i^{t+1} = \lambda \times (x_i^t + \frac{1}{\phi} \times (x_{best}^t - x_i^t)) \tag{16}$$

In the original research [27], the entire moth population is separated into two equivalent subpopulations based on their fitness. In subpopulation 1 (moths with greater fitness), positions of individuals are being updated using Lévy flights (Eq. (12)), where moth positions in the subpopulation 2 (moths with lower fitness) are being updated by using Eq. (15) or Eq. (16) with possibility of 50% [27].

# 4 Experimental results

In this section, we briefly show network topology used in experiments, parameters' setup, and results of empirical tests.

In the empirical tests, we used static drone location problem instance with 30 uniformly distributed targets. Scenario with randomly distributed targets is harder to solve than scenario with clustered targets. Working domain of the network was set to 100 $m$ by 100 $m$. For all drones in the population, radius $r$ was set to 15 $m$, similar like in [28].

The number of moths in the population $N$ was set to 40, and the maximum number of generations $MaxGen$ was set to 2,000 yielding total of 80,000 objective function evaluations. The rest of parameters were adjusted as: the number of moths kept in each generation to 2, index $\beta = 1.5$, max walk step $S_{max} = 1.0$, and acceleration factor $\phi = (5^{1/2} - 1)/2 \cong 0.618$.

For testing purposes, we developed software framework using Visual Studio 2017 with .NET Framework 4.7. Algorithm was tested in 30 independent runs on Intel CoreTM i7-4770HQ processor @2.4GHz with 32GB of RAM memory.

For experimental purposes, in order to analyze how MS algorithm behaves, we conducted experiments with different number of drones (starting with only one drone). In the employed scenario, minimum number of 9 drones is necessary to cover all targets.

Experimental results for 30 uniformly distributed targets are shown in Table 1. In the presented table, we show results for different number of drones for absolute and targets coverage in percentiles, and for execution time of the MS algorithm. As performance indicators, we used best and mean results obtained in

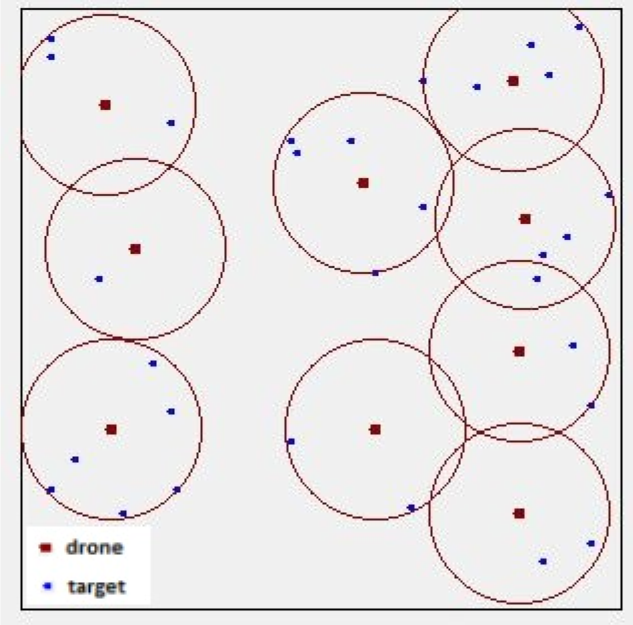


Figure 1: Examples with one drone (left), and four drones (right) in clustered target set

30 independent runs of the algorithm. In Table 1, T.C., T.C.% and E.T. are abbreviations for target coverage, target coverage in percentiles and execution time, respectively.

Table 1: Experimental results

| Drone No. | Indicator | T.C. | T.C % | E.T. |
|---|---|---|---|---|
| 1 | Best | 6 | 20% | 1.5 |
| | Mean | 5 | 16.6% | 3.2 |
| 2 | Best | 11 | 36.6% | 4.3 |
| | Mean | 10 | 33.3% | 5.1 |
| 3 | Best | 15 | 50% | 6.6 |
| | Mean | 13 | 76.6% | 7.0 |
| 4 | Best | 18 | 60% | 7.6 |
| | Mean | 16 | 53.3% | 8.3 |
| 5 | Best | 21 | 70% | 10.0 |
| | Mean | 20 | 66.6% | 11.1 |
| 6 | Best | 24 | 80% | 14.2 |
| | Mean | 22 | 73.3% | 15.2 |
| 7 | Best | 26 | 86% | 17.3 |
| | Mean | 23 | 76.6% | 18.1 |
| 8 | Best | 28 | 93% | 21.9 |
| | Mean | 27 | 90% | 24.3 |
| 9 | Best | 30 | 100% | 29.4 |
| | Mean | 29 | 96.6% | 31.6 |

From the results presented in the Table 1, we conclude that the MS algorithm generates optimal values, and establishes full coverage of targets with 9 drones. Results with 9 drones are visualized in Figure 1.

## 5 Conclusion

In this paper we showed moth search (MS) algorithm adjusted for solving static drone location problem (SDLP). MS is novel swarm intelligence metaheuristics proposed by Wang in 2016, and it was not tested on this problem before.

The MS algorithm was tested on problem instance with 30 uniformly distributed targets. In this case, MS algorithm obtained coverage of all targets with 9 drones, which is optimum solution. As a conclusion, we state that the MS algorithm shows good performance when tackling NP-hard problems such is static drone location problem.

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

*References:*

[1] H. Chen, X. min Wang, and Y. Li, "A survey of autonomous control for uav," in *Proceedings of the 09 International Conference on A Artificial Intelligence and Computational Intelligence (AICI 0́9)*, pp. 267–271, IEEE, November 2009.

[2] D. Zorbas, L. D. P. Pugliese, T. Razafindralambo, and F. Guerriero, "Optimal drone placement and cost-efficient target coverage," *Journal of Network and Computer Applications*, vol. 75, pp. 16–31, November 2016.

[3] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: A survey," *Ad Hoc Networks*, vol. 6, pp. 621–655, June 2008.

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

[5] N. Bacanin, M. Tuba, and I. Brajevic, "Performance of object-oriented software system for improved artificial bee colony optimization," *International Journal of Mathematics and Computers in Simulation*, vol. 5, no. 2, pp. 154–162, 2011.

[6] 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.

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

[8] 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.

[9] 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.

[10] M. Tuba and N. Bacanin, "JPEG quantization tables selection by the firefly algorithm," in *International Conference on Multimedia Computing and Systems (ICMCS)*, pp. 153–158, IEEE, 2014.

[11] X.-S. Yang and S. Deb, "Cuckoo search via levy flights," in *Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, pp. 210–214, 2009.

[12] I. Brajevic and M. Tuba, "Cuckoo search and firefly algorithm applied to multilevel image thresholding," in *Cuckoo Search and Firefly Algorithm: Theory and Applications* (X.-S. Yang, ed.), vol. 516 of *Studies in Computational Intelligence*, pp. 115–139, Springer International Publishing, 2014.

[13] N. Bacanin, "Implementation and performance of an object-oriented software system for cuckoo search algorithm," *International Journal of Mathematics and Computers in Simulation*, vol. 6, pp. 185–193, December 2010.

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

[15] 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.

[16] 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)*, May 2015.

[17] M. Tuba, N. Bacanin, and A. Alihodzic, "Multilevel image thresholding by fireworks algorithm," in *2015 25th International Conference Radioelektronika (RADIOELEKTRONIKA)*, pp. 326–330, April 2015.

[18] 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.

[19] E. Tuba, M. Tuba, and D. Simian, "Wireless sensor network coverage problem using modified fireworks algorithm," in *International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 696–701, IEEE, 2016.

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

[21] 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.

[22] 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.

[23] E. Tuba, M. Tuba, and D. Simian, "Handwritten digit recognition by support vector machine optimized by bat algorithm," *Proceeding of the WSCG 2016, Computer Science Research Notes*, pp. 369–376, 2016.

[24] E. Dolicanin, I. Fetahovic, E. Tuba, R. Capor-Hrosik, and M. Tuba, "Unmanned combat aerial vehicle path planning by brain storm optimization algorithm," *Studies in Informatics and Control*, vol. 27, no. 1, pp. 15–24, 2018.

[25] E. Tuba, R. Capor-Hrosik, A. Alihodzic, and M. Tuba, "Drone placement for optimal coverage by brain storm optimization algorithm," in *International Conference on Health Information Science, Advances in Intelligent Systems and Computing*, vol. 734, pp. 167–176, Springer, 2017.

[26] E. Tuba, E. Dolicanin, and M. Tuba, “Chaotic brain storm optimization algorithm,” in *Intelligent Data Engineering and Automated Learning, LNCS*, vol. 10585, (Cham), pp. 551–559, Springer International Publishing, 2017.

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

[28] D. Zorbas, T. Razafindralambo, D. P. P. Luigi, and F. Guerriero, “Energy ecient mobile target tracking using ying drones,” in *Proceedings of the 4th International Conference on Ambient Systems, Networks and Technologies (ANT 2013), Procedia Computer Science*, vol. 19, pp. 80–87, Springer, 2013.

[29] P. S. Callahan, “Moth and candle: The candle flame as a sexual mimic of the coded infrared wavelengths from a moth sex scent (pheromone),” *Applied Optics*, vol. 16, no. 12, pp. 3089–3097, 1977.

[30] 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.

[31] A. H. Gandomi and A. H. Alavi, “Krill herd: A new bio-inspired optimization algorithm,” *Commun Nonlinear Sci Numer Simulat*, vol. 17, pp. 4831–4845, 2012.