International Journal of Multidisciplinary Studies and Innovative Technologies

e-ISSN: 2602-4888
dergipark.org.tr/en/pub/ijmsit
Research Article

2024, 8 (1), 17-25
DOI: 10.36287/ijmsit.8.1.4
Received: June 7, 2024; Accepted: July 1, 2024

Comparison of Black Widow Optimization and Aquila Optimizer with Current Metaheuristics

Metin Kalyon[1*], Sibel Arslan[2]

[1*] *Computer Engineering Department, Sivas Cumhuriyet University, Sivas, Turkey (20219257012@cumhuriyet.edu.tr) (ORCID: 0000-0003-4637-836X)*

[2] *Software Engineering Department, Sivas Cumhuriyet University, Sivas, Turkey (sibelarslan@cumhuriyet.edu.tr) (ORCID: 0000-0003-3626-553X)*

Abstract – Metaheuristic optimization algorithms are an optimization approach that produces acceptable solutions in situations where it is difficult to create a mathematical model in an optimization problem or in large-scale, multivariate optimization problems. Metaheuristics play a significant role in solving optimization problems. In this study, five current meta- heuristics (Aquila Optimizer (AO), Artificial Rabbits Optimization (ARO), Black Widow Optimization (BWO), Harris Hawk Optimization (HHO) and Sooty Tern Optimization Algorithm (STOA), which are inspired by swarm intelligence and foraging behavior of creatures in nature) are compared. These algorithms are discussed in detail and information is given about their working principles. As far as is known, this is the first time that the performances of these five algorithms have been compared. The algorithms were evaluated with unimodal and multimodal test functions. The simulation results demonstrate that AO and BWO are more successful than the other algorithms. It is also evaluated that the metaheuristics used in the study can be applied to many engineering problems.

Keywords *– Metaheuristics, Aquila Optimization, Artificial Rabbit Optimization, Black Widow Optimization, Harris Hawk Optimization, Sooty Tern Optimization Algorithm, Quality Test Functions*

Citation: Kalyon, M., Arslan, S. (2024). Comparison of Black Widow Optimization and Aquila Optimizer with Current Metaheuristics. International Journal of Multidisciplinary Studies and Innovative Technologies, 8(1): 17-25.

I.INTRODUCTION

Optimization is the process of choosing the greatest solution to a problem under specific conditions [1]. Mathematical and heuristic methods are used to solve optimization problems. Mathematical methods are not preferred for complex problems due to the difficulty in deriving the model and the cost of scanning the entire solution space, while heuristic methods are not preferred for problems with overmuch of variables due to the difficulty in finding the solution [2]. Instead of these methods, metaheuristic optimization algorithms that give more successful results have been developed [3]. Metaheuristics are based on natural phenomena, social behavior of species and evolutionary concepts [4]. Furthermore, metaheuristics use exploration and exploitation phases to develop solutions. The exploration phase addresses to a further comprehensive search of the solution space. The exploitation phase performs a more local search in the space obtained by exploration [5]. Metaheuristic optimization algorithms are used in different fields such as finance [1], economics [6], energy [7], planning [8], image processing [9], and engineering design applications [10].

In this study, 5 current metaheuristic optimization algorithms are compared. These are: Harris Hawk Optimization (HHO), Sooty Tern Optimization Algorithm (STOA), Black Widow Optimization (BWO), Aquila Optimization (AO) and Artificial Rabbit Optimization (ARO). The important contribution of the study are as follows:

- As far as is known, this is the first time that the meta-heuristics used in this study have been compared using different evaluation criteria such as simulation results, convergence rates, etc.
- The performances of the algorithms are tested with various functions.
- According to the experiments, although the algorithms produce similar results, it is evaluated that AO and BWO outperform the others and STOA is less successful than the other algorithms.

The rest of the paper is arranged as follows. In the second section, the working principles and pseudo-codes are explained. In the third section, the parameters and test functions used in the experimental studies are discussed. In the fourth section, simulation results and convergence rate graphs are interpreted. In the fifth section, an overview of the study is given.

II.MATERIALS AND METHOD

This section contains summary information about the metaheuristic algorithms to be compared, the test functions used in the comparison, and the parameter values of the algorithms.

A. **Aquila Optimization (AO)**

AO is a population-based optimization algorithm inspired by the hunting and capture skills of Aquila [11]. Aquila capture their prey with 4 different hunting strategies. These are:

- **Low flight with a slow descent attack**: This is a hunting method in which rock eagles hunt by landing on the ground and then attacking their prey [14].
- **Vertical slope high flight**: It is the hunting method of aquila during flight at high altitudes. When they find prey, aquila perform a vertical dive towards the prey [12].
- **Contour flight with a short glide attack**: This is the hunting method of aquila during flight low above the ground [13].
- **Aquila catching prey on foot**: Aquila capture prey by landing on the ground for prey.

In aquila optimization, the exploration and exploitation phase is chosen based on the condition in Eq. (1) below [11].

$$t \leq \frac{2}{3} * T \tag{1}$$

where t is the instantaneous iteration and T is the maximum number of iterations. If the condition $t \leq \frac{2}{3} * T$ is satisfied, the algorithm is in the exploration phase, if not, the algorithm is in the exploitation phase. The algorithm of AO is illustrated in Fig. 1 [13]. It is initialized with a range of randomly determined candidate solutions and then performs the exploration (extended exploration, reduced exploration) and exploitation (extended exploitation, reduced exploitation) phases based on Eq. (1). X is the population, $X_{best}(t)$ is the best candidate solution in iteration. $X_M(t)$ is the average of the solutions in the step t. x and y are the spiral shape parameters in the search, and G_1, G_2 are the slope parameters used to follow the prey during escape. Levy is the flight distribution function.

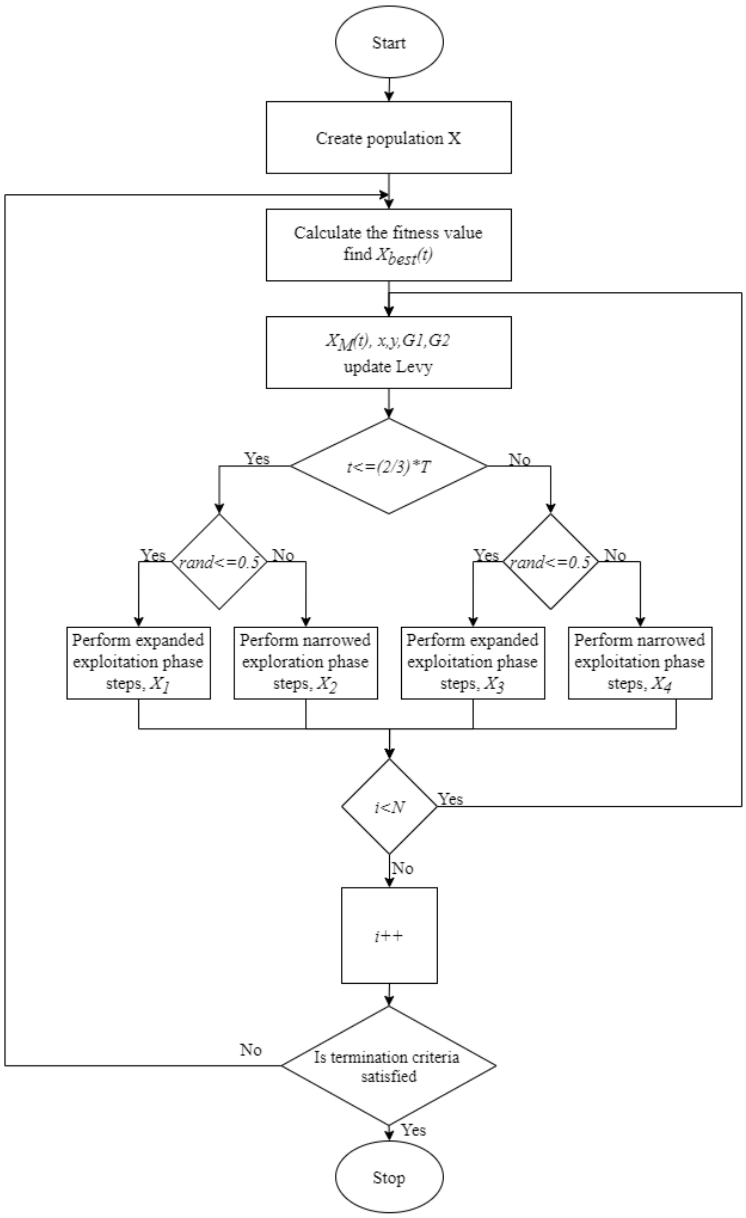


Fig. 1. Flow chart of AO.

B. Artificial Rabbit Optimization (ARO)

ARO, proposed in 2022, is an optimization algorithm that mimics he foraging, random hiding and energy minimization strategies of rabbits [15]. The exploration phase, which is the basis of optimization algorithms, corresponds to the foraging behaviour in ARO, while the exploitation phase corresponds to the random hiding phase [16]. Rabbits do not forage in their own location but in the locations of neighbouring rabbits. This movement of the rabbits is called indirect foraging (exploration) [15]. Random hiding (exploitation) is when each rabbit randomly builds a nest around itself during the iteration in order to hide against threats and randomly selects one of the nests bult by neighbouring rabbits instead of its own nest [17].

In ARO, the transition from the exploration phase to the exploitation phase is performed based on the energy reduction in the rabbit at each step [18]. The major steps of the ARO algorithm are demonstrated in Table I as follows [16]

TABLE I
PSEUDO CODE OF ARO

Step 1	Initialization
Step 2	Generate a random population of rabbits, P
Step 3	**for** i = 1 : N (number of rabbits) **do**
Step 4	Calculate the fitness value of each rabbit, f_i
Step 5	**end**
Step 6	**repeat**
Step 7	**Selection phase**
Step 8	Select P starter rabbits
Step 9	**Search phase**
Step 10	Meandering foraging (Exploration)
Step 11	Random storage (Exploitation)
Step 12	**Update phase**
Step 13	Update the population P
Step 14	**until** termination criteria met
Step 15	**end**

C. Black Widow Optimization (BWO)

BWO is a population-based algorithm inspired by the mating behaviour of black widow spiders [19]. In general terms, the algorithm mimics the reproduction and subsequent cannibalistic act of them. In other words, it reflects the ideas of Darwin's theory of evolution, namely the survival of the fittest and the superiority of the fittest. In a specific sense, mutation in a spider population refers to genetic changes [20]. BWO consists of four stages: population initialization, reproduction, cannibalization and mutation. The steps of the BWO are demonstrated in Table II [16].

TABLE II
PSEUDO CODE OF BWO

Step 1	Define initial parameters
Step 2	Generate an initial population of spiders
Step 3	Calculate the fitness value of the population
Step 4	Is the stopping criterion met?
	a. Yes:
	-Select the best spider as the solution
	b. Not satisfied:
	-Update the population
	-**Go to** Step 2

D. Harris Hawks Optimization (HHO)

HHO is a swarm-based optimization algorithm inspired by the behaviour and hunting methods of Harris hawks [21]. In HHO, Harris hawks settle in the hunting area and wait to detect their prey. During this waiting period, they detect the prey by selecting the two strategies they use for prey detection according to a random value of p in the range $[0 - 1]$. The first strategy is used when $p \geq 0.5$, while the second strategy is used when $p < 0.5$. This process for prey detection in HHO is the exploration phase [22]. In HHO, the changeover from the exploration phase to the exploitation phase depends on E (the lost energy of the prey). If $|E| \geq 1$, the exploration phase starts and if $|E| < 1$, the exploitation phase starts. Depending on the prey detected in the exploration phase and its E value, Harris's hawks move to the exploitation phase and perform a surprise attack [18]. With regard to the escape behaviour of the hunting and the following strategies of Harris hawks, 4 different types of attacks are carried out [23]. These attacks are classified as soft, hard, soft with progressive fast dives and hard with progressive fast dives [24]. The steps of the HHO algorithm are shown below in Table III. [25].

TABLE III PSEUDO CODE OF HHO

Step 1	Define initial parameters
Step 2	Generate initial hawks
Step 3	Calculate the fitness value of the falcons
Step 4	Update the positions of the falcons
Step 6	Calculate the updated fitness value
Step 7	Is the stopping criterion satisfied?
	a. Yes:
	- Select the best falcon as the solution
	b. Not satisfied:
	- Go to Step 4

E. Boosted Sooty Tern Optimization Algorithm (STOA)

STOA simulates the migration and attack behaviour of sooty terns [26]. In this algorithm, migration behaviour refers to the exploration phase and attack behaviour addresses to the exploitation phase. Sooty terns move together during migration, with different starting positions to avoid collisions. They move by changing their position in the direction of the sooty tern with the highest fitness value [27]. These animals can change their speed and angle of attack during migration and gain altitude by moving their wings. While attacking their prey, they exhibit spiral behaviours in the air below. The steps of STOA are shown below in Table IV [28].

TABLE IV PSEUDO CODE OF STOA

Input	Sooty Tern population
Output	Best sooty tern location
Step 1	Start the STOA procedure
Step 2	Initialize parameters S_A and C_B
Step 3	Calculate the position of each sooty tern
Step 4	Best sooty tern location
Step 5	**while** $(z < Max_{iterations})$ **do**
Step 6	**for** each sooty tern **do**
Step 7	Update the location of sooty terns
Step 8	**end for**
Step 9	Update parameters S_A and C_B
Step 10	Update the eligibility value of each sooty tern
Step 11	Update sooty tern location
Step 12	$z = z + 1$
Step 13	**end while**

Step 14	Get the best sooty position

Step 15	Finish the procedure

F. **Functions**

In the study, the algorithms are compared using 10 different quality test functions. The functions were selected from two main groups as unimodal and multimodal. The unimodal group functions are shown in Table V and the multimodal group functions are shown in Table VI [15]. The range given in the tables represents the limits of the research space of the functions and f_{min} represents the optimum value.

TABLE V UNIMODAL TEST FUNCTIONS

Function	**Dim**	**Rate**	$\boldsymbol{f_{\min}}$
$f_1(x) = \sum_{i=1}^{n} x_i^2$	30	$[-100,100]$	0
$f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	$[-10,10]$	0
$f_3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j\right)^2$	30	$[-100,100]$	0
$f_4(x) = \max_i\{\lvert x_i \rvert, 1 \leq i \leq n\}$	30	$[-100,100]$	0
$f_5(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	$[-30,30]$	0

TABLE VI MULTIMODAL TEST FUNCTIONS

Function	**Dim**	**Rate**	$\boldsymbol{f_{min}}$
$F_6(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	30	$[-5.12,5.12]$	0
$F_7(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$	30	$[-32,32]$	0
$F_8(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30	$[-600,600]$	0
$F_9(x) = \frac{\pi}{n}\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\}$ $+ \sum_{i=1}^{n} u(x_i, 10, 100, 4) y_i = 1 + \frac{x_i + 1}{4}$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	30	$[-50,50]$	0
$F_{10}(x) = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2[1$	30	$[-50,50]$	0

G. **Parameters**

The parameters of each algorithm are given in Table VII. The parameters P_P , C_R and P_M used in BWO are reproduction rate, cannibalization rate and mutation rate respectively. The STOA parameters C_f is a control parameter that sets the value of collision avoidance, C_B is the parameter responsible for better exploration, u and v are the parameters describing the spiral behaviour of sooty terns in the air. In the AO algorithm α is the exploration parameter, while δ is the exploitation parameter.

III. RESULTS

In the study, the whole algorithm were run 100 times independently. The algorithms were written using MATLAB 2023b platform. The computer used is a Ryzen 9 3950X with a speed of 3.5 GHz and 64 GB RAM. Simulation results are presented in Table VIII and Table IX for unimodal and multimodal functions, respectively. The mean, standard deviation, best and worst results of the runs for each function are given in the tables. Additionally, the algorithms with the best average values are bolded in the tables.

TABLE VII PARAMETERS

Control Parameters	Algorithms				
	AO	ARO	BWO	HHO	STOA
P_P	-	-	0.60	-	-
C_R	-	-	0.44	-	-
P_M	-	-	0.40	-	-
C_f	-	-	-	-	2
C_B	-	-	-	-	$[0 - 0.50]$
u	-	-	-	-	1
v	-	-	-	-	1
α	0.10	-	-	-	-
$\bar{\delta}$	0.10	-	-	-	-
Population Size	50	50	50	50	50
Maximum Iteration Number	1000	1000	1000	1000	1000

TABLE VIII
SIMULATION RESULTS FOR UNIMODAL TEST FUNCTIONS

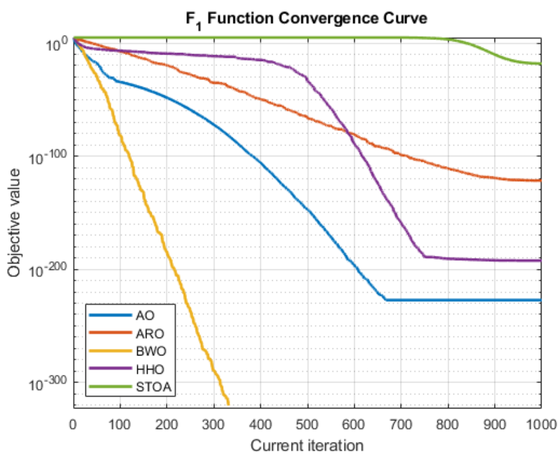
Fonksiyonlar	Kriter	AO	ARO	BWO	HHO	STOA
F_1	Ortalama	4.53E-228	2.36E-122	**0.00E+00**	4.68E-193	6.64E-19
	Standart Sapma	0.00E+00	2.36E-121	0.00E+00	0.00E+00	1.81E-18
	En İyi	0.00E+00	5.82E-141	0.00E+00	4.47E-216	3.47E-23
	En Kötü	4.53E-226	2.36E-120	0.00E+00	4.51E-191	1.36E-17
F_2	Ortalama	7.52E-110	5.35E-69	**0.00E+00**	3.53E-101	7.65E-13
	Standart Sapma	7.52E-109	2.38E-68	0.00E+00	3.10E-100	1.06E-12
	En İyi	3.41E-158	1.97E-78	0.00E+00	9.39E-113	1.18E-15
	En Kötü	7.52E-108	1.64E-67	0.00E+00	3.10E-99	6.10E-12
F_3	Ortalama	1.99E-204	2.81E-97	**0.00E+00**	2.38E-157	1.99E-09
	Standart Sapma	0.00E+00	1.62E-96	0.00E+00	2.38E-156	5.36E-09
	En İyi	0.00E+00	9.64E-119	0.00E+00	4.90E-195	2.99E-12
	En Kötü	1.92E-202	1.35E-95	0.00E+00	2.38E-155	3.50E-08
F_4	Ortalama	4.79E-138	1.47E-52	**0.00E+00**	2.79E-97	3.10E-06
	Standart Sapma	4.79E-137	8.88E-52	0.00E+00	1.44E-96	3.36E-06
	En İyi	5.58E-159	3.82E-60	0.00E+00	2.87E-109	8.77E-08
	En Kötü	4.79E-136	7.79E-51	0.00E+00	1.28E-95	2.18E-05
F_5	Ortalama	**4.74E-04**	4.93E-03	2.89E+01	1.04E-03	2.77E+01
	Standart Sapma	7.59E-04	7.61E-03	2.99E-02	1.37E-03	5.66E-01
	En İyi	8.32E-08	2.80E-05	2.89E+01	5.33E-08	2.70E+01
	En Kötü	4.29E-03	5.10E-02	2.90E+01	7.78E-03	2.88E+01

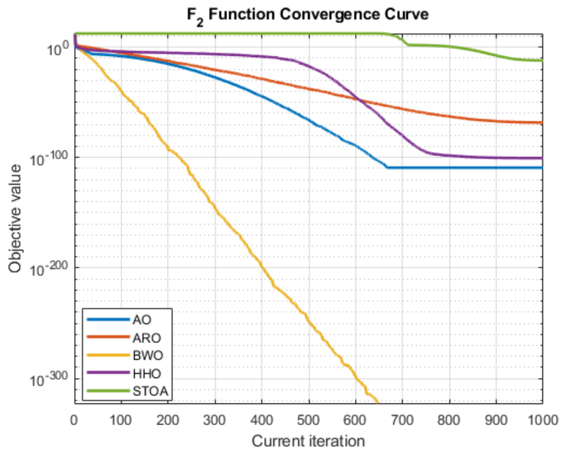
As can be seen in Table VIII, BWO gives the optimum result for the functions F_1, F_2, F_3, and F_4, while it gives the worst results among the other algorithms for the function F_4. The average values of all algorithms are quite low in F_1, F_2, F_3 and F_4 functions. When the best value and average values of F_5 function are analysed, it is seen that AO is the most successful algorithm, followed by HHO and ARO algorithms respectively. BWO reached the optimum result in 4 of the 5 unimodal functions (F_1, F_2, F_3, F_4) and AO reached the optimum result in 2 of them (F_1 and F_3). BWO and STOA algorithms failed in F_3 function. The standard deviation values are quite low in all functions of all algorithms, this is since the algorithms produce an average value in all given functions.

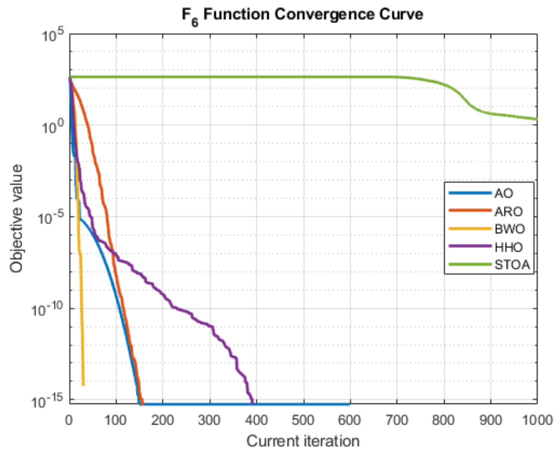
TABLE IX
SIMULATION RESULTS FOR MULTİMODAL TEST FUNCTION

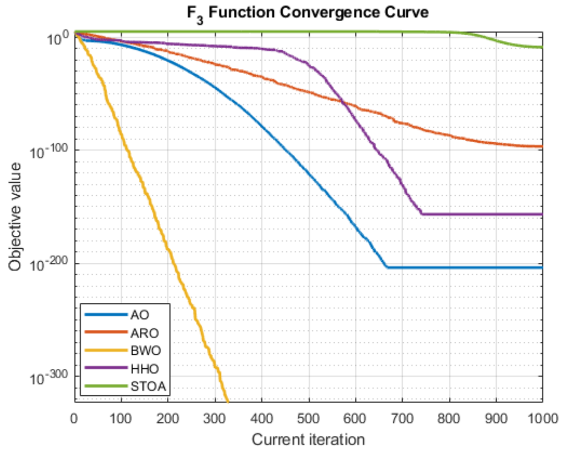
Fonksiyonlar	Kriter	AO	ARO	BWO	HHO	STOA
F_1	Ortalama	**0**	**0**	**0**	**0**	2.106416577
	Standart Sapma	0	0	0	0	6.842359055
	En İyi	0	0	0	0	0
	En Kötü	0	0	0	0	53.31885593
F_2	Ortalama	**4.44089E-16**	**4.44089E-16**	**4.44089E-16**	**4.44089E-16**	19.75922394
	Standart Sapma	0	0	0	0	1.995881757
	En İyi	4.44089E-16	4.44089E-16	4.44089E-16	4.44089E-16	3.66193E-10
	En Kötü	4.44089E-16	4.44089E-16	4.44089E-16	4.44089E-16	19.9608913
F_3	Ortalama	**0**	**0**	**0**	**0**	0.013449707
	Standart Sapma	0	0	0	0	0.022276208
	En İyi	0	0	0	0	0
	En Kötü	0	0	0	0	0.088236013
F_4	Ortalama	2.7544E-07	**6.43578E-08**	0.677557163	8.59365E-07	0.163903802
	Standart Sapma	5.33573E-07	4.74369E-08	0.22924251	1.33728E-06	0.116042462
	En İyi	1.04675E-10	4.51467E-09	0.273894913	3.58148E-11	0.057572289
	En Kötü	3.1304E-06	2.74131E-07	1.208909754	8.20476E-06	0.705247422
F_5	Ortalama	**3.37611E-06**	0.000663001	2.946764761	6.58857E-06	1.564925207
	Standart Sapma	7.40183E-06	0.00262174	0.171456051	8.6822E-06	0.229004541
	En İyi	6.15959E-09	6.70095E-09	2.126628587	5.52213E-09	0.899462787
	En Kötü	5.31504E-05	0.010988261	2.999797751	3.85219E-05	2.037382071

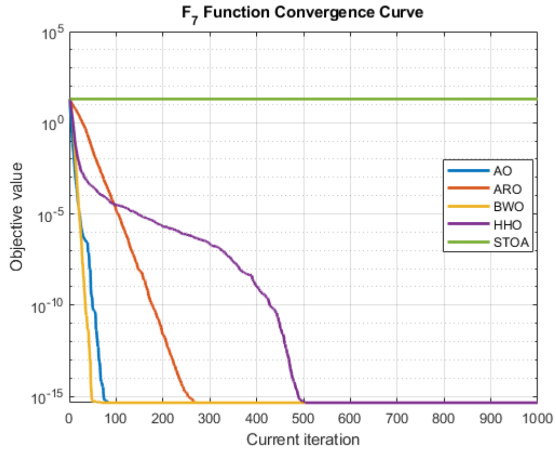
According to Table IX, except for STOA, the other algorithms give the optimum result for F_1 and F_4 functions while for F_2 function there is no remarkable difference since all values are the same. STOA has the worst average for F_1, F_2 and F_3 functions, especially for F_2 function it has a much higher average than the other algorithms, but for F_5 it reaches the optimum result at the best value. BWO is the algorithm with the worst result for the average value in the F_4 and F_5 functions. Looking at the worst value in general, STOA fails in multimodal functions with a high error value. When both unimodal and multimodal functions are evaluated together, AO, ARO and HHO produce more stable results and are stable in terms of the difference between the best and the worst, respectively. So as to comment on the convergence of the algorithms, the convergence plots of the functions F_1, F_2, F_3, F_4, F_5 for unimodal and F_6, F_7, F_8, F_9, F_{10} for multimodal are drawn and presented in Fig. 2.

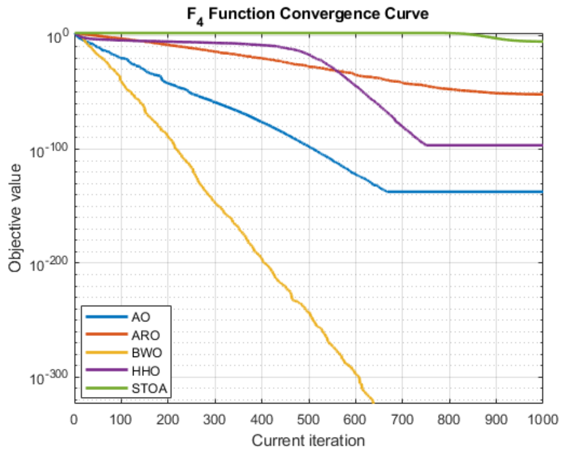


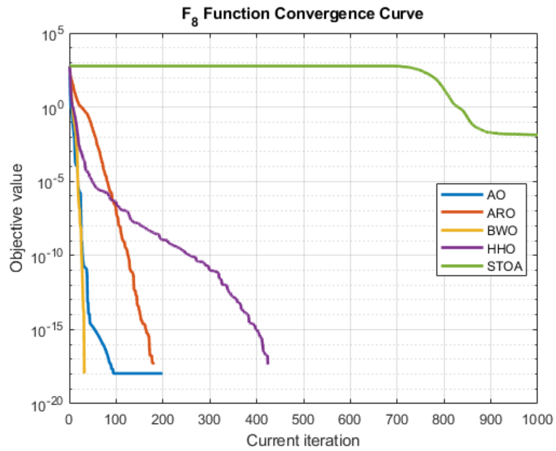

F_2 Function Convergence Curve
Objective value
Current iteration
AO
ARO
BWO
HHO
STOA

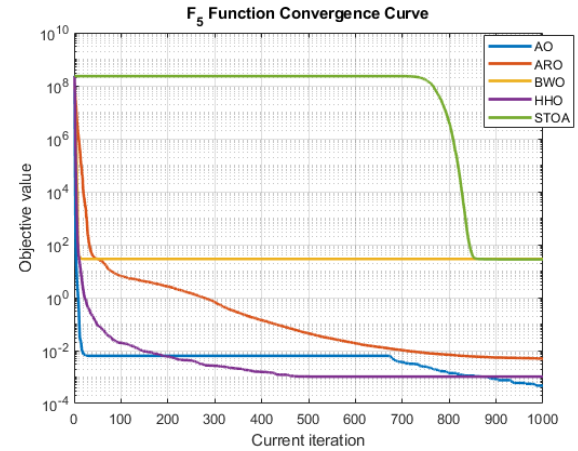

F_6 Function Convergence Curve
Objective value
Current iteration
AO
ARO
BWO
HHO
STOA

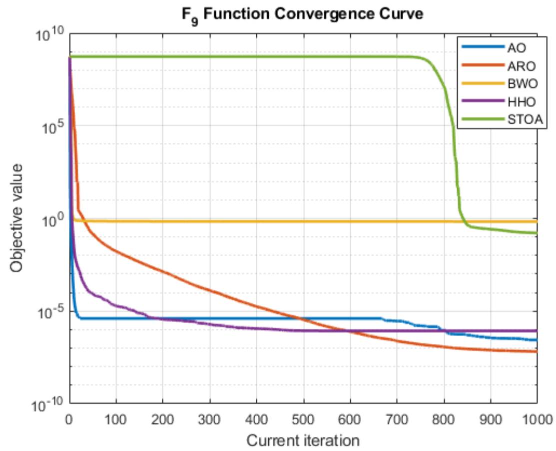

F_3 Function Convergence Curve
Objective value
Current iteration
AO
ARO
BWO
HHO
STOA


F_7 Function Convergence Curve
Objective value
Current iteration
AO
ARO
BWO
HHO
STOA


F_4 Function Convergence Curve
Objective value
Current iteration
AO
ARO
BWO
HHO
STOA


F_8 Function Convergence Curve
Objective value
Current iteration
AO
ARO
BWO
HHO
STOA


F_5 Function Convergence Curve
Objective value
Current iteration
AO
ARO
BWO
HHO
STOA


F_9 Function Convergence Curve
Objective value
Current iteration
AO
ARO
BWO
HHO
STOA

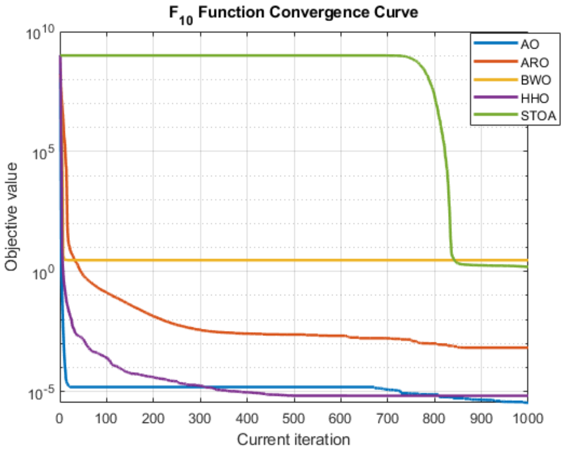


Fig. 2. Unimodal and multimodal function convergence plots

Considering the convergence plots in Fig. 2, the success of these five algorithms against the functions is as follows:

- BWO showed fast convergence, producing better results in F_1, F_2, F_3 and F_4. At the same time, BWO was followed by AO, HHO, ARO and STOA, which performed the worst in these functions.
- In F_5 and F_6, BWO has worse performance than HHO, AO and ARO, in contrast to the functions it converges better.
- •In F_7 and F_8, the convergence rates of the other algorithms are similar except STOA. The convergence rates for these two functions are BWO, AO, ARO, HHO and STOA respectively.
- While BWO has the best convergence and STOA has the worst convergence in the first four functions, STOA outperforms BWO in F_9 and F_{10}.
- STOA has the slowest convergence rate (except F_{10} and F_{10}).
- Overall, in terms of convergence rate, AO ranks first in three out of ten functions and second in seven of them. When BWO is evaluated, although it ranks first in six of the ten functions, it ranks either fourth or fifth in the other four functions. This shows that although BWO converges faster in many functions, AO has a more stable performance.
- STOA has a slow convergence rate for all functions and is less successful compared to the other algorithms

IV.DISCUSSION

This study aims to compare AO, ARO, BWO, HHO and STOA proposed in the last five years. Each meta-heuristic is briefly summarized and presented to the readers. To the best of our knowledge, this is the first time these metaheuristics have been compared in the literature. A total of 10 different test functions, 5 unimodal and 5 multimodal, were used in the experiments. In different performance criteria, AO and BWO outperform all algorithms. Among these two strong competing algorithms, AO produced more stable results. Future work will compare the performance of AO and BWO by solving engineering applications. It is also aimed to compare various metaheuristics with different evaluation criteria.

V.CONCLUSION

This study aims to compare AO, ARO, BWO, HHO and STOA proposed in the last five years. Each meta-heuristic is briefly summarized and presented to the readers. To the best of our knowledge, this is the first time these metaheuristics have been compared in the literature. A total of 10 different test functions, 5 unimodal and 5 multimodal, were used in the experiments. In different performance criteria, AO and BWO outperform all algorithms. Among these two strong competing algorithms, AO produced more stable results. Future work will compare the performance of AO and BWO by solving engineering applications. It is also aimed to compare various metaheuristics with different evaluation criteria.

VI. ACKNOWLEDGMENT

The heading of the Acknowledgment section and the References section must not be numbered.

Authors' Contributions

The authors' contributions to the paper are equal.

Statement of Conflicts of Interest

There is no conflict of interest between the authors.

Statement of Research and Publication Ethics

The authors declare that this study complies with Research and Publication Ethics

REFERENCES

[1] G. G. Emel and Ç. Taşkın, "Genetik algoritmalar ve uygulama alanları," *Uludağ Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, vol. 21, no. 1, pp. 129–152, 2002.

[2] M. Mitchell and S. Forrest, "Genetic algorithms and artificial life," *Artificial life*, vol. 1, no. 3, pp. 267–289, 1994.

[3] Z. Beheshti and S. M. H. Shamsuddin, "A review of population-based meta-heuristic algorithms," *Int. j. adv. soft comput. appl*, vol. 5, no. 1, pp. 1–35, 2013

[4] ˙I. Öznur and S. Korukoğlu, "Genetik algoritma yaklaşımı ve yöneylem ¨ arastırmasında bir uygulama," *Yonetim ve Ekonomi Dergisi ¨*, vol. 10, no. 2, pp. 191–208, 2003.

[5] M. A. Albadr, S. Tiun, M. Ayob, and F. Al-Dhief, "Genetic algorithm based on natural selection theory for optimization problems," *Symmetry*, vol. 12, no. 11, p. 1758, 2020

[6] M. Sharma and P. Kaur, "A comprehensive analysis of nature-inspired meta-heuristic techniques for feature selection problem," *Archives of Computational Methods in Engineering*, vol. 28, pp. 1103–1127, 2021.

[7] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "Gsa: a gravitational search algorithm," *Information sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[8] D. Datta, A. R. Amaral, and J. R. Figueira, "Single row facility layout problem using a permutation-based genetic algorithm," *European Journal of Operational Research*, vol. 213, no. 2, pp. 388–394, 2011.

[9] S. S. Chouhan, A. Kaul, and U. P. Singh, "Soft computing approaches for image segmentation: a survey," *Multimedia Tools and Applications*, vol. 77, no. 21, pp. 28 483–28 537, 2018

[10] E. Eyup and E. Tanyıldızı, "Güncel metasezgisel optimizasyon algoritmalarının performans karşılaştırılması," in *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*. IEEE, 2018, pp. 1–16.

[11] L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. AlQaness, and A. H. Gandomi, "Aquila optimizer: a novel meta-heuristic optimization algorithm," *Computers & Industrial Engineering*, vol. 157, p. 107250, 2021.

[12] O. E. Turgut and M. S. Turgut, "Local search enhanced aquila optimization algorithm ameliorated with an ensemble of wavelet mutation strategies for complex optimization problems," *Mathematics and Computers in Simulation*, vol. 206, pp. 302–374, 2023

[13] S. Akyol, "Global optimizasyon için yeni bir hibrit yöntem: kaya kartalı optimizasyonu-tanjant arama algoritması," *Fırat Universitesi Muhendislik Bilimleri Dergisi* , vol. 33, no. 2, pp. 721–733, 2021.
[14] Y.-J. Zhang, Y.-X. Yan, J. Zhao, and Z.-M. Gao, "Aoaao: The hybrid algorithm of arithmetic optimization algorithm with aquila optimizer," *IEEE Access*, vol. 10, pp. 10 907–10 933, 2022.
[15] L. Wang, Q. Cao, Z. Zhang, S. Mirjalili, and W. Zhao, "Artificial rabbits optimization: A new bio-inspired meta-heuristic algorithm for solving engineering optimization problems," *Engineering Applications of Artificial Intelligence*, vol. 114, p. 105082, 2022.
[16] H. Bakır, "Dynamic fitness-distance balance-based artificial rabbits optimization algorithm to solve optimal power flow problem," *Expert Systems with Applications*, vol. 240, p. 122460, 2024.
[17] Y. Wang, Y. Xiao, Y. Guo, and J. Li, "Dynamic chaotic opposition-based learning-driven hybrid aquila optimizer and artificial rabbits optimization algorithm: framework and applications," *Processes*, vol. 10, no. 12, p.2703, 2022.
[18] B. Gülmez, "Stock price prediction with optimized deep lstm network ¨with artificial rabbits optimization algorithm," *Expert Systems with Applications*, vol. 227, p. 120346, 2023.
[19] V. Hayyolalam and A. A. P. Kazem, "Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems," *Engineering Applications of Artificial Intelligence*, vol. 87, p. 103249, 2020.
[20] G. Hu, B. Du, X. Wang, and G. Wei, "An enhanced black widow optimization algorithm for feature selection," *Knowledge-Based Systems*, vol. 235, p. 107638, 2022.
[21] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future generation computer systems*, vol. 97, pp. 849–872, 2019.
[22] O. Altay, "Guncel metasezgisel yöntemlerin standart kalite testi fonksiyonlarında karşılaştırılması," *International Journal of Pure and Applied Sciences*, vol. 8, no. 2, pp. 286–301, 2022.
[23] J. C. Bednarz, "Cooperative hunting harris' hawks (parabuteo unicinctus)," *Science*, vol. 239, no. 4847, pp. 1525–1527, 1988.
[24] S. Wang, H. Jia, L. Abualigah, Q. Liu, and R. Zheng, "An improved hybrid aquila optimizer and harris hawks algorithm for solving industrial engineering optimization problems," *Processes*, vol. 9, no. 9, p. 1551, 2021.
[25] Y. Ç. Kuyu, "Optimizasyon problemleri için yeni metasezgisel yaklaşımlar," Ph.D. dissertation, Bursa Uludag University (Turkey), 2023.
[26] G. Dhiman and A. Kaur, "Stoa: a bio-inspired based optimization algorithm for industrial engineering problems," *Engineering Applications of Artificial Intelligence*, vol. 82, pp. 148–174, 2019.
[27] H. Jia, Y. Li, K. Sun, N. Cao, and H. M. Zhou, "Hybrid sooty tern optimization and differential evolution for feature selection." *Computer Systems Science & Engineering*, vol. 39, no. 3, 2021.
[28] A. Singh, A. Sharma, S. Rajput, A. K. Mondal, A. Bose, and M. Ram, "Parameter extraction of solar module using the sooty tern optimization algorithm," *Electronics*, vol. 11, no. 4, p. 564, 2022.