Actor Optimization Algorithm: A Novel Approach for Engineering Design Challenges

Widi Aribowo
Department of Electrical Engineering, Faculty of Vocational Studies, Universitas Negeri Surabaya, Surabaya, East Java 60231, Indonesia
widiaribowo@unesa.ac.id (corresponding author)

Belal Batiha
Department of Mathematics. Faculty of Science and Information Technology, Jadara University, Irbid 21110, Jordan
b.batha@jadara.edu.jo

Tareq Hamadneh
Department of Mathematics, Al Zaytoonah University of Jordan, Amman 11733, Jordan
t.hamadneh@zuj.edu.jo

Gharib Mousa Gharib
Department of Mathematics, Faculty of Science, Zarqa University, Zarqa 13110 Zarqa, Jordan
ggharib@zu.edu.jo

Hind Monadhel
Department of Cybersecurity and Cloud Computing, Technical Engineering, Uruk University, Baghdad 10001, Iraq
hindmonadhel@uruk.edu.iq

Riyadh Kareem Jawad
Department of Medical Instrumentations Techniques Engineering, Al-Rasheed University College, Baghdad 10001, Iraq
riyadh2@alrasheedcol.edu.iq

Ibraheem Kasim Ibraheem
Department of Electrical Engineering, College of Engineering, University of Baghdad, Baghdad 10001, Iraq
ibraheemki@coeng.uobaghdad.edu.iq

Zeinab Monrazeri
Department of Electrical and Electronics Engineering, Shiraz University of Technology, Shiraz, 7155713876, Iran
z.montazeri@sutech.ac.ir

Mohammad Dehghani
Department of Electrical and Electronics Engineering, Shiraz University of Technology, Shiraz, 7155713876, Iran
m.dehghani@sutech.ac.ir

Received: 8 January 2025 | Revised: 28 January 2025 | Accepted: 6 February 2025

 DOI: https://doi.org/10.48084/etasr.10162

ABSTRACT

In this paper, a novel human-based metaheuristic algorithm called Actor Optimization Algorithm (AOA) is introduced. AOA mimics the behaviors of an actor when playing a role. The main idea in designing AOA is derived from a specific behavior of the actor including (i) simulating the movements and dialogues of the given role and (ii) practicing to better present the assigned role. The theory of AOA is stated and mathematically modeled in the phases of exploration and exploitation. The performance of AOA to address real-world applications is evaluated on the CEC 2011 test suite. The optimization results show that AOA, with its high ability in exploration, exploitation, and balancing during the search process, achieved suitable results. In addition, the performance of AOA was challenged by comparing it with 12 known metaheuristic algorithms. Result comparison showed that the proposed AOA outperformed the competing algorithms by 100% (in all 22 optimization problems) of the CEC 2011 test suite. The simulation results show that AOA has a successful performance in handling optimization tasks in real-world applications by achieving better results in competition with the compared algorithms.

Keywords-optimization; human-based metaheuristic; actor optimization algorithm

I. INTRODUCTION

Optimization is one of the most important concepts in science and engineering, which refers to the process of finding the best possible solution to a problem in the presence of certain constraints [1]. This concept is of great importance in fields such as engineering, computer science, economics, and biology [2]. Approaches to solving optimization problems are generally categorized into two main groups: deterministic and stochastic methods [3]. Deterministic approaches such as linear programming algorithms and dynamic programming, although designed based on exact mathematical models and seeking to find the final optimal solution to problems, have limitations such as the need for exact models, high computational complexity, and sensitivity to small changes [4, 5]. These limitations have led researchers to look for alternative methods that can solve more complex and realistic problems more efficiently. One successful response to these challenges is the use of stochastic approaches, and in particular metaheuristic algorithms [6]. These algorithms use stochastic processes to search the response space instead of using exact models [7]. Recently published metaheuristic algorithms that can be used in various optimization applications can be mentioned: Potter Optimization Algorithm (POA) [8], Carpet Weaving Optimization (CWO) [9], Sales Training Based Optimization (STBO) [10], Dollmaker Optimization Algorithm (DOA) [11], Tailor Optimization Algorithm (TOA) [12], and Sculptor Optimization Algorithm (SOA) [13]. Metaheuristic algorithms have performed successfully in various applications such as: mobile edge computation [14], wireless networks [15], dynamic power–latency tradeoff [16], etc.

The salient features of these algorithms include high flexibility, global search capability, and scalability, which have made them one of the most widely used tools for solving complex optimization problems [17]. Metaheuristic algorithms usually start from an initial population and discover better answers through several stages of updating. Metaheuristic algorithms are built around two key concepts: exploration and exploitation. Exploration involves broadly investigating the search space without a specific direction, aiming to identify promising regions. In contrast, exploitation focuses on refining the search within these promising regions to locate the optimal solution. Global search emphasizes exploring and covering a wider space to avoid being trapped in local optima, whereas local search concentrates on exploiting the identified regions to converge on the final solution. A significant challenge in designing metaheuristic algorithms is achieving a balance between exploration and exploitation. Excessive exploration can consume substantial computational resources without necessarily advancing toward the optimal solution. Conversely, overemphasis on exploitation can result in becoming trapped in local optima, thereby missing better solutions. To address this, metaheuristic algorithms incorporate various mechanisms to maintain this crucial balance [18].

Despite the numerous metaheuristic algorithms that have been developed and introduced, the "No Free Lunch" (NFL) theorem states that no single algorithm can outperform all others across all optimization problems. This theorem highlights that the efficiency of an algorithm is contingent upon the specific characteristics of the problem at hand. Consequently, given the diversity of optimization problems across various domains and the need to address the unique conditions and constraints of each problem, the development of new algorithms remains essential. Such efforts not only enhance performance in specific fields but also contribute to the expansion of knowledge and the discovery of novel methods for solving complex problems [19].

Motivated by the NFL theorem, in this study, a new metaheuristic algorithm called Actor Optimization Algorithm (AOA) is designed to handle optimization tasks. The main contributions of this study are:

- AOA is designed by taking inspiration from the behaviors of actors when playing a role.
- The theory of AOA is stated and then its steps are mathematically modeled in two phases of exploration and exploitation.
- The performance of AOA is compared with the performance of twelve well-known metaheuristic algorithms.
- In order to evaluate the effectiveness of the proposed algorithm in dealing with real world applications, AOA is implemented on 22 constrained optimization problems from the CEC 2011 test suite.

II. ACTOR OPTIMIZATION ALGORITHM (AOA)

A. Initialization

The proposed AOA approach is a population-based algorithm that is able to achieve appropriate solutions to optimization problems based on the search power of its members in an iterative process. Each AOA member is a candidate solution to the problem that specifies the values of the decision variables of the problem. Therefore, mathematically, each AOA member can be modeled using a vector. Similarly, the community of AOA members together, which is represented by putting these vectors together from a mathematical point of view using a matrix, according to (1). The initial position of each AOA member in the search space is completely randomly initialized using (2).

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times m} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,d} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots & ⋰ & \vdots \\ x_{i,1} & \cdots & x_{i,d} & \cdots & x_{i,m} \\ \vdots & ⋰ & \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,d} & \cdots & x_{N,m} \end{bmatrix}_{N\times m} \quad (1)$$

$$x_{i,d} = lb_d + r \cdot (ub_d - lb_d) \quad (2)$$

where X is the AOA population matrix, X_i is the ith painting student (candidate solution), $x_{i,d}$ is its dth dimension in search space (decision variable), N is the number of painting students, m is the number of decision variables, r is a random number in interval $[0,1]$, lb_d, and ub_d are the lower bound and upper bound of the dth decision variable, respectively.

Corresponding to each AOA member as a candidate solution to the problem, the objective function can be evaluated. Therefore, the evaluated values for the objective function can be represented using a vector according to (3):

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix}_{N\times 1} \quad (3)$$

where F is the vector of the calculated objective function and F_i is the calculated objective function based on the ith painting student.

The evaluated values for the objective function are a suitable measure to measure the quality of the candidate solutions and consequently the AOA population members. Therefore, the best evaluated value for the objective function corresponds to the best candidate solution and consequently the best population member. Since AOA is an iteration-based approach, the positions of the population members are updated in each iteration. Following the updating of the positions of the population members, the objective function must also be re-evaluated. Based on the evaluated values for the objective function, the best population member is again identified. This updating process continues until the last iteration of the algorithm and at the end the position of the best population member is presented as the AOA solution for the given problem. In the AOA design, the positions of the AOA members are updated in two separate phases in each iteration. In the following, each of these two phases is introduced and mathematically modeled.

B. Phase 1: Simulating the Movements and Dialogues of the Given Role (Exploration)

One of the important steps in filmmaking is selecting the right actor to play the role. After the actor is selected for a role, he tries to adapt himself to the personality of the given role. This actor's behavior leads to extensive changes in his behaviors and expression techniques to play the role. Simulating these extensive changes in actor behavior leads to extensive changes in the position of AOA members in the problem-solving space and, as a result, increases the ability to explore. In the design of AOA, each member of the population is considered as an actor, and each of these actors is assigned a specific role according to (4). After determining the role, each actor tries to imitate the specified movements and dialogues and get close to the assigned role. In AOA design, based on the simulation of the actor's effort to get close to the role, a new position is calculated for each AOA member using (5). Then, if this new position leads to an improvement in the value of the objective function, it replaces the previous position of the corresponding member according to (6).

$$R_i = X_i + r \cdot (X_{best} - X_i) \quad (4)$$

$$X_i^{P1} = X_i + r \cdot (R_i - I \cdot X_i), i = 1,2, \ldots, N \quad (5)$$

$$X_i = \begin{cases} X_i^{P1}, & F_i^{P1} < F_i \\ X_i, & \text{else} \end{cases} \quad (6)$$

where R_i is the given role for ith AOA member, X_{best} is the location of the best member of AOA population, X_i^{P1} is the new suggested position of ith actor based on first phase of AOA, F_i^{P1} is its objective function value, r is a random number with a normal distribution in the range of $[0,1]$, I is random number from set $\{1,2\}$, and N is the number of population members.

C. Phase 2: Practice (Exploitation)

After understanding their role, actors try to bring their movements, behaviors, and dialogues closer to the given role through practice in order to be able to play the role in the best possible way. This behavior of actors includes small, detailed, and precise changes in their movements and expression. Simulating these small changes in the behavior of actors leads to small changes in the positions of AOA members in the problem-solving space and, as a result, increases the exploitation ability of the algorithm. In the design of AOA, it is assumed that, corresponding to the training of each actor, the positions of the population members near the location where they are located are updated. Accordingly, based on the simulation of each actor's training, a new position in the problem-solving space is calculated for each AOA member using (7). If this new position improves the objective function, it replaces the position of the corresponding member according to (8).

$$X_i^{P2} = X_i + \left(1 - 2 \cdot \sin\left(\frac{r \cdot \pi}{2}\right)\right) \cdot \frac{(ub - lb)}{t} \quad (7)$$

$$X_i = \begin{cases} X_i^{P2}, & F_i^{P2} < F_i \\ X_i, & \text{else} \end{cases} \quad (8)$$

where X_i^{P2} is the new suggested position of the ith actor based on the second phase of AOA, F_i^{P2} is its objective function value, t is the iteration counter of the algorithm, and T is the maximum number of algorithm iterations.

D. Computational Complexity of AOA

The initialization steps of AOA have a computational complexity of $O(Nm)$, where N is the population size and m is the number of decision variables of the problem. The updating process in AOA has two phases, exploration and exploitation. Therefore, the computational complexity of the updating process is $O(2NmT)$, where T is the maximum number of iterations of the algorithm. Therefore, the total computational complexity of AOA is $O(NM(1+2T))$.

III. SIMULATION STUDIES

The performance of the newly introduced AOA was rigorously analyzed to determine its efficacy in tackling real-world optimization problems. The CEC 2011 benchmark suite, which encompasses 22 constrained optimization problems derived from practical scenarios, served as the evaluation platform [20]. The CEC 2011 Test Suite of optimization problems is a collection designed to evaluate and compare global optimization algorithms. This suite, specifically created to test the performance of optimization methods in complex, real-world scenarios, consists of problems with various dimensions and characteristics such as non-linear constraints, varying scales, and the inherent complexity of objective functions. It serves as a standard reference for assessing optimization techniques in challenging environments. A detailed description of each optimization problem included in the CEC 2011 Test Suite, along with their titles follows:

- **Parameter Estimation for Frequency-Modulated (FM) Sound Waves (C11-F1):** This problem involves the estimation of parameters for Frequency-Modulated (FM) sound waves. The objective is to optimize the model parameters to best fit the input data, which is particularly useful in signal processing and communication applications.
- **Lennard-Jones Potential Problem (C11-F2):** This problem focuses on the optimization of the Lennard-Jones potential, which is used to simulate interactions between molecules or atoms in physics and chemistry. It consists of 30 variables and is specifically designed for simulating molecular interactions in complex systems.
- **The Bifunctional Catalyst Blend Optimal Control Problem (C11-F3):** This optimization problem deals with finding the optimal control for the blending of bifunctional catalysts. The goal is to determine the best combination and control for catalytic processes used in the chemical industry and energy production.
- **Optimal Control of a Non-Linear Stirred Tank Reactor (C11-F4):** This problem involves optimizing the control of a non-linear stirred tank reactor. The objective is to determine optimal conditions for chemical reactions within the reactor to maximize process efficiency and performance.
- **Tersoff Potential for Model Si (B) (C11-F5):** This problem involves optimizing the Tersoff potential for modeling silicon (Si) interactions. The goal is to optimize models related to the structural and chemical behaviors of semiconductor materials.
- **Tersoff Potential for Model Si (C) (C11-F6):** Similar to the previous problem, this optimization problem focuses on the Tersoff potential for modeling interactions in silicon (Si) in different configurations, specifically for semiconductor and related material behavior.
- **Spread Spectrum Radar Polyphase Code Design (C11-F7):** The objective of this problem is to design a polyphase code for spread spectrum radar. It is used in the design of radar and communication systems requiring high accuracy and noise resistance.
- **Transmission Network Expansion Planning (TNEP) Problem (C11-F8):** This optimization problem focuses on the expansion planning of electrical transmission networks. The goal is to optimize the development of energy networks by considering production capacities, demand, and transportation costs.
- **Large Scale Transmission Pricing Problem (C11-F9):** This problem addresses the pricing of transmission networks on a large scale. The objective is to optimize transmission pricing to minimize energy transportation costs in electricity grids.
- **Circular Antenna Array Design Problem (C11-F10):** This problem involves the design of circular antenna arrays. The goal is to optimize antenna configurations for communication and radar systems that require wide coverage.
- **The ELD Problems (C11-F11 to C11-F17):** The ELD (Economic Load Dispatch) problems are related to the economic dispatch of power in electrical systems. This includes various problems where the objective is to optimally distribute power generation across different units while considering cost constraints and operational limits.
- **Hydrothermal Scheduling Problem (C11-F18 to C11-F20):** This problem deals with the joint scheduling of hydro and thermal power plants. The objective is to optimize the generation of power from both types of plants to meet demand at the lowest possible cost.
- **Messenger: Spacecraft Trajectory Optimization Problem (C11-F21):** This problem involves optimizing the trajectory of the Messenger spacecraft for its mission to Mercury. The objective is to determine the best route to the destination while considering technical constraints and resource limitations.
- **Cassini 2: Spacecraft Trajectory Optimization Problem (C11-F22):** Similar to the previous problem, this optimization problem focuses on the trajectory of the

Cassini spacecraft for its mission to Saturn. It involves greater complexity and multiple route options.

This suite is designed to test optimization algorithms in realistic and complex conditions. These problems are widely used for industrial, scientific, and engineering applications to evaluate the efficiency and robustness of various optimization methods.

It should be noted that the penalty coefficient strategy has been used to deal with the constraints of the problems. To provide a robust comparison, the AOA is assessed alongside 12 well-established metaheuristic algorithms: Genetic Algorithm (GA) [21], Particle Swarm Optimization (PSO) [22], Gravitational Search Algorithm (GSA) [23], Teaching–Learning-Based Optimization (TLBO) [24], Multi-Verse Optimizer (MVO) [25], Grey Wolf Optimizer (GWO) [26], Whale Optimization Algorithm (WOA) [27], Marine Predators Algorithm (MPA) [28], Tunicate Swarm Algorithm (TSA) [29], Reptile Search Algorithm (RSA) [30], African Vultures Optimization Algorithm (AVOA) [31], and White Shark Optimizer (WSO) [32]. For the sake of fair comparison, the original versions of the MATLAB codes published by the original researchers have been used in the simulation studies, whereas in the case of PSO and GA, the standard codes published by Professor Mirjalili were used. The values of the control parameters of the metaheuristic algorithms are specified in Table I. Experiments have been implemented on MATLAB R2022a using 64-bit Core i7 processor with 3.20 GHz and 16 GB main memory. The proposed AOA approach and each of the competitor algorithms was implemented on the CEC-2011 functions in 25 independent implementations where each implementation contained 150,000 FEs.

TABLE I. CONTROL PARAMETERS VALUES

Algorithm	Parameter	Value
GA	Type	Real coded
	Selection	Roulette wheel (Proportionate)
	Crossover	Whole arithmetic (Probability = 0.8, $\alpha \in [-0.5, 1.5]$)
	Mutation	Gaussian (Probability = 0.05)
PSO	Topology	Fully connected
	Cognitive and social constant	$(C_1, C_2) = (2, 2)$
	Inertia weight	Linear reduction from 0.9 to 0.1
	Velocity limit	10% of dimension range
GSA	Alpha, G_0, R_{norm}, R_{power}	20, 100, 2, 1
TLBO	T_F: teaching factor	T_F = round $[(1 + rand)]$
	Random number	$rand$ is a random number between $[0 - 1]$.
GWO	Convergence parameter (a)	a: Linear reduction from 2 to 0.
MVO	Wormhole Existence Probability (WEP)	Min(WEP) = 0.2 and Max(WEP)=1.
	Exploitation accuracy over the iterations (p)	$p = 6$.
WOA	Convergence parameter (a)	a: Linear reduction from 2 to 0.
	r is a random vector in $[0 - 1]$.	
	l is a random number in $[-1,1]$.	
TSA	P_{min} and P_{max}	1, 4
	$c1, c2, c3$	random numbers lie in the range of $[0 - 1]$.
MPA	Constant number	P=0.5
	Random vector	R is a vector of uniform random numbers in $[0, 1]$.
	Fish Aggregating Devices ($FADs$)	$FADs$=0.2
	Binary vector	U= 0 or 1
RSA	Sensitive parameter	$\beta = 0.01$
	Sensitive parameter	$\alpha = 0.1$
	Evolutionary Sense (ES)	ES: randomly decreasing values between 2 and −2
AVOA	L_1, L_2	0.8, 0.2
	w	2.5
	P_1, P_2, P_3	0.6, 0.4, 0.6
WSO	F_{min} and F_{max}	0.07, 0.75
	τ, a_o, a_1, a_2	4.125, 6.25, 100, 0.0005

Table II illustrates the outcomes of the comparative evaluation of AOA and its competitors on the CEC 2011 test suite. A detailed examination of these results reveals that AOA consistently delivers superior solutions across all the benchmark problems. This persistent outperformance underscores AOA's robustness and its capability to navigate the complexities of real-world optimization tasks effectively. Notably, AOA not only outpaces its counterparts in most of the test cases but also establishes itself as the most reliable and adaptive optimization method throughout the entire suite of problems. To further substantiate the algorithm's efficiency, a statistical analysis using the Wilcoxon rank-sum test [33] was employed. This test provides critical evidence of the statistically significant superiority of AOA over other metaheuristics. The statistical findings reinforce the empirical results, highlighting AOA's consistent ability to deliver optimal or near-optimal solutions with remarkable reliability. The robust performance of AOA, demonstrated through empirical data and statistical validation, affirms its potential as a highly effective optimization tool for solving challenging problems rooted in practical applications. These findings emphasize that AOA is not only a competitive algorithm but also a standout method for addressing the intricate demands of constrained optimization tasks.

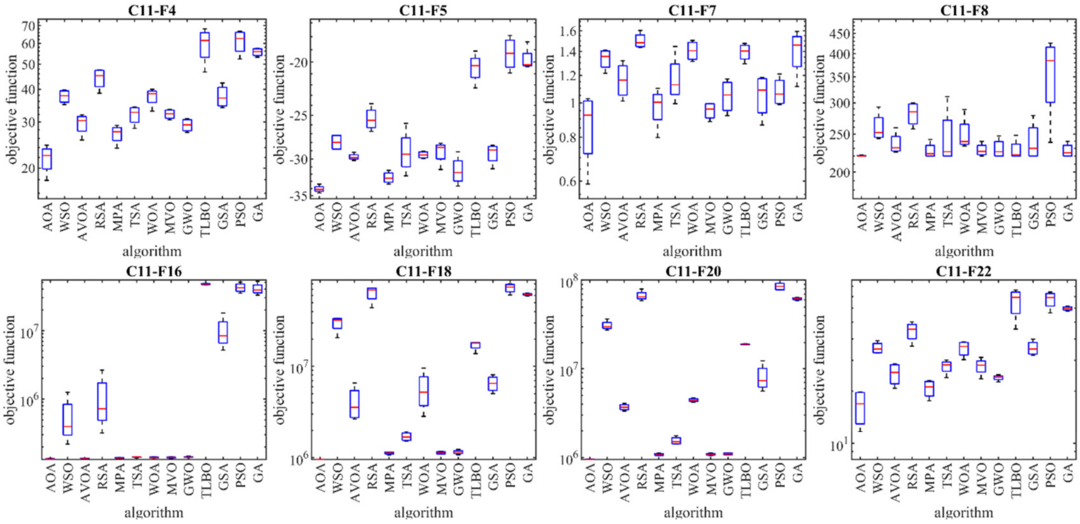

Fig. 1. Boxplot diagrams of AOA and competitor algorithms performances on CEC 2011 test suite.

TABLE II. PERFORMANCE OF METAHEURISTIC ALGORITHMS ON CEC 2011 TEST SUITE

		AOA	WSO	AVOA	RSA	MPA	TSA	WOA	MVO	GWO	TLBO	GSA	PSO	GA
C11-F1	mean	**6.060906**	14.50867	11.94661	16.82667	9.024889	14.90491	12.10467	12.51278	10.81048	14.92222	16.67552	14.65501	17.59306
	std	11.44388	5.44389	7.134568	5.016761	8.329962	3.595306	7.015798	3.884497	8.533319	**2.744263**	4.410158	6.838221	4.095731
C11-F2	mean	**-26.1505**	-17.5108	-21.0786	-16.0007	-23.2472	-15.8502	-19.7807	-14.5179	-21.9323	-15.6395	-18.1294	-21.9585	-16.7293
	std	1.123681	1.534562	0.912745	**0.471209**	1.175746	2.988094	3.698583	1.463674	2.749589	0.695852	4.05249	1.651091	2.228916
C11-F3	mean	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**	**1.15E-05**
	std	**4.22E-16**	1.89E-11	2.16E-09	4.25E-11	3.98E-15	2.5E-14	4.74E-15	8.45E-13	3.06E-15	6.5E-14	4.74E-15	4.74E-15	4.74E-15
C11-F4	mean	**21.77385**	37.55542	29.57374	44.14282	27.01992	31.97321	37.4845	32.14939	29.19959	59.44759	37.62317	61.01333	55.55381
	std	4.74272	3.7868	4.720156	7.065975	3.828154	4.525298	5.184075	**2.342945**	2.82177	15.51519	6.46463	11.1058	3.354946
C11-F5	mean	**-34.0502**	-27.9833	-29.74	-25.4021	-32.4832	-29.2205	-29.4859	-29.1461	-31.58	-20.5064	-29.3359	-19.3532	-19.8113
	std	0.865534	1.556849	**0.710277**	2.092219	1.384101	4.435949	0.733468	2.496666	3.209775	2.243928	2.21786	2.269966	1.601467
C11-F6	mean	**-23.9288**	-16.0759	-18.7358	-15.5464	-20.6404	-12.6256	-19.2266	-13.6752	-19.0549	-9.83513	-20.2544	-10.2964	-10.7793
	std	3.666169	1.7275	2.20324	**0.907972**	2.928836	6.475272	3.310772	7.638024	2.756741	1.602577	2.144739	2.847012	2.476305
C11-F7	mean	**0.865551**	1.334319	1.163964	1.500341	0.976465	1.173403	1.407092	0.95077	1.049407	1.393901	1.055783	1.07902	1.405417
	std	0.338766	0.152979	0.233072	0.128194	0.214527	0.324101	0.153903	**0.090592**	0.197853	0.134328	0.251401	0.174514	0.350253
C11-F8	mean	**220.4118**	260.2616	236.6343	281.7171	227.0622	245.5691	250.2229	227.928	229.6596	227.928	239.7541	358.2487	227.0862
	std	**1.215489**	37.81879	26.43022	33.6512	17.76889	74.3264	43.52786	14.52281	21.93127	23.12572	46.68994	142.0197	15.76169
C11-F9	mean	**9423.886**	311325.5	216263.6	579564.5	25919.09	50380.02	214315.2	86065.94	38047.19	232361.9	452731.5	590439.6	1047597
	std	**6341.454**	111673.7	27764.15	222230.7	9997.841	16561.78	176850.4	48102.25	23371.46	74877.95	69645.33	221391.4	85297.97
C11-F10	mean	**-21.4106**	-15.8331	-17.3994	-14.935	-18.5169	-16.0558	-15.2506	-16.2214	-15.9041	-14.4086	-15.3995	-14.4616	-14.3047
	std	0.792645	1.324702	0.676684	0.657044	0.555863	3.335252	0.67312	3.737495	**0.187382**	0.59553	0.397845	0.607102	0.618633
C11-F11	mean	**631002.5**	4387542	1833671	6010198	2188212	4463119	1952755	2002056	3342245	4073008	2056656	4078888	4557575
	std	421060.4	522469.2	370804.5	356859.1	349781.6	1083623	323985.9	661029.8	338246.2	272957.2	377625.2	278394.2	**234290.6**
C11-F12	mean	**1242058**	5557279	2895575	8128552	1780612	3773384	4190147	1808679	1859949	8708609	4176792	2332158	8794009
	std	71681.04	239661	59990.78	647007.9	**57794.18**	182308.8	252310.2	107248.5	110158.7	569725.8	186289.4	152006.8	97750.11
C11-F13	mean	**15444.71**	15675.41	15457.87	15915.81	15465.93	15480.28	15504.33	15489.68	15486.08	15715.53	75408.05	15480.62	23187.14
	std	**0.212583**	271.607	1.843408	623.5626	4.033786	12.26883	44.33375	24.29915	8.563772	354.9454	33842.21	22.5158	25881.87
C11-F14	mean	**18309**	68580.1	18671.42	130523.1	18718.38	19208.91	19046.6	19149.46	19050.29	173925.3	18975.95	18993.24	18986.47
	std	110.732	28755.81	**92.99736**	64842.62	145.6024	424.5493	204.8834	162.2954	228.9562	245427.9	218.3114	176.8187	303.7182
C11-F15	mean	**33123.03**	501041.5	75689.2	1036136	35882.26	47365.38	134564.8	35962.87	35951.1	8214687	177704.4	36062.81	4236664
	std	**830.3494**	823786.8	71487.17	1845868	9836.21	48281.48	118655.9	9856.168	9814.033	8074345	24199.7	9829.927	4120302
C11-F16	mean	**133677.9**	567680.4	137161.2	1101388	138479.5	142448.9	140874.2	140685.2	142845.3	47317415	10016574	42358178	40673503
	std	3806.762	786049.3	**2248.542**	1765923	3723.927	2866.251	4956.209	5564.368	4572.131	1817047	9457709	11325789	13720605
C11-F17	mean	**16064954**	4.94E+09	1.41E+09	8.42E+09	1.76E+08	8.56E+08	5.33E+09	1.77E+08	1.77E+08	1.20E+10	6.14E+09	1.12E+10	1.18E+10
	std	**3931463**	8.90E+08	1.98E+08	2.99E+09	48679702	2.37E+08	2.29E+09	47849335	49250458	6.36E+08	7.95E+08	2.26E+09	1.77E+09
C11-F18	mean	**954302.4**	29827351	4084881	63545184	1116961	1690174	5692393	1125660	1148522	17079070	6515447	72299338	61487929
	std	**3354.358**	10354738	3045857	22408431	60847.39	316879	4789714	66388.37	105585.1	3822274	2260464	14723481	3052372
C11-F19	mean	**1041232**	29476819	4246210	62343394	1312763	2017369	6139457	1492871	1432343	19626653	4039337	92494816	61798845
	std	156457.5	9162087	817967	19074481	**150460.5**	333063.2	6944347	281566.4	179580.6	7605948	2352486	16696147	2368905
C11-F20	mean	**951033.4**	31156128	3698893	67137137	1077211	1537637	4437009	1083838	1097512	18943472	8152474	85243351	61828025
	std	**10663.5**	6663657	525509.1	15002879	51428.24	261574.2	366047.8	51608.79	46365.85	542645.5	4915298	13719138	3704706
C11-F21	mean	**12.90357**	35.91799	20.70939	49.85819	17.67438	25.05979	29.82743	23.84366	21.09973	62.74046	30.84367	65.41762	63.75367
	std	3.840599	5.98077	2.507445	14.28022	3.211127	3.073882	**1.905368**	3.361642	3.012891	35.87856	2.762274	11.685	26.80165
C11-F22	mean	**16.30217**	35.46881	25.16824	44.33433	20.71999	27.66364	35.21334	27.74666	23.86415	65.14146	35.40331	67.28631	59.80752
	std	6.712288	4.840525	6.389453	10.09465	4.196121	4.484125	6.475542	5.359113	**1.586583**	23.04481	6.01575	13.15906	2.863776
Sum rank		**22**	198	112	240	56	150	151	123	99	233	165	210	234
Mean rank		**1**	9	5.090909	10.90909	2.545455	6.818182	6.863636	5.590909	4.5	10.59091	7.5	9.545455	10.63636
Total rank		**1**	9	4	13	2	6	7	5	3	11	8	10	12
Wilcoxon: *p*-value			3.73E-16	3.73E-16	3.73E-16	9.86E-16	7.99E-16	3.73E-16	8.00E-16	1.22E-15	7.99E-16	9.52E-16	4.59E-16	7.99E-16

IV. CONCLUDING REMARKS AND FUTURE WORKS

In this paper, a new metaheuristic algorithm named Actor Optimization Algorithm (AOA) designed to handle optimization applications is introduced. The main idea in designing AOA is derived from the actor's behaviors while playing a role. The theory of AOA is proposed and then mathematically modeled in two phases (i) exploration based on simulating the actor's behavior in imitating the movements and dialogues of the assigned role and (ii) exploitation based on simulating the actor's exercises to perform the assigned role as best as possible. The performance of AOA in real-world applications was tested on 22 constrained optimization problems from the CEC 2011 test suite. The optimization results showed that AOA, by balancing exploration and exploitation, has provided successful performance in handling this test suite. In order to analyze the capability of AOA, its results were compared with the performance of twelve metaheuristic algorithms. What was evident from the simulation results was that AOA, by outperforming the compared algorithms, has a suitable performance for solving optimization problems in real-world applications. Despite its advantages, AOA has its limitations. As with other stochastic methods, one of the limitations of AOA is that there is no guarantee of reaching the global optimum. Another limitation is that it is always possible to design newer algorithms that have superior performance compared to AOA.

The introduction of AOA raises several research proposals for further work in the future. The design of binary and multi-objective versions of AOA is one of the most specific research proposals of this study. In addition, the application of AOA to handle optimization tasks in various sciences and real-world applications is another research proposal for further work.

REFERENCES

[1] S. Zhao, T. Zhang, S. Ma, and M. Chen, "Dandelion Optimizer: A nature-inspired metaheuristic algorithm for engineering applications," *Engineering Applications of Artificial Intelligence*, vol. 114, Sep. 2022, Art. no. 105075, https://doi.org/10.1016/j.engappai.2022.105075.

[2] X. Wang, "Draco lizard optimizer: a novel metaheuristic algorithm for global optimization problems," *Evolutionary Intelligence*, vol. 18, no. 1, Nov. 2024, Art. no. 10, https://doi.org/10.1007/s12065-024-00998-5.

[3] V. Tomar, M. Bansal, and P. Singh, "Metaheuristic Algorithms for Optimization: A Brief Review," *Engineering Proceedings*, vol. 59, no. 1, 2024, Art. no. 238, https://doi.org/10.3390/engproc2023059238.

[4] R. Abu-Gdairi, R. Mareay, and M. Badr, "On Multi-Granulation Rough Sets with Its Applications," *Computers, Materials & Continua*, vol. 79, no. 1, pp. 1025–1038, 2024, https://doi.org/10.32604/cmc.2024.048647.

[5] H. Qawaqneh, "New contraction embedded with simulation function and cyclic (α, β)-admissible in metric-like spaces," *International Journal of Mathematics and Computer Science*, vol. 15, no. 1, pp. 1029–1044, 2020.

[6] T. Hamadneh, M. Ali, and H. AL-Zoubi, "Linear Optimization of Polynomial Rational Functions: Applications for Positivity Analysis," *Mathematics*, vol. 8, no. 2, Feb. 2020, Art. no. 283, https://doi.org/10.3390/math8020283.

[7] M. Dehghani, E. Trojovska, and P. Trojovsky, "A new human-based metaheuristic algorithm for solving optimization problems on the base of simulation of driving training process," *Scientific Reports*, vol. 12, no. 1, Jun. 2022, Art. no. 9924, https://doi.org/10.1038/s41598-022-14225-7.

[8] T. Hamadneh *et al.*, "On the Application of Potter Optimization Algorithm for Solving Supply Chain Management Application," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 5, pp. 88–99, Oct. 2024, https://doi.org/10.22266/ijies2024.1031.09.

[9] S. Alomari *et al.*, "Carpet Weaver Optimization: A Novel Simple and Effective Human-Inspired Metaheuristic Algorithm," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 4, pp. 230–242, Aug. 2024, https://doi.org/10.22266/ijies2024.0831.18.

[10] T. Hamadneh *et al.*, "Sales Training Based Optimization: A New Human-inspired Metaheuristic Approach for Supply Chain Management," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 6, pp. 1325–1334, Oct. 2024, https://doi.org/10.22266/ijies2024.1231.96.

[11] S. Al omari *et al.*, "Dollmaker Optimization Algorithm: A Novel Human-Inspired Optimizer for Solving Optimization Problems," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 3, pp. 816–828, Jun. 2024, https://doi.org/10.22266/ijies2024.0630.63.

[12] T. Hamadneh *et al.*, "On the Application of Tailor Optimization Algorithm for Solving Real-World Optimization Application," *International Journal of Intelligent Engineering and Systems*, vol. 18, no. 1, pp. 1–12, Feb. 2025, https://doi.org/10.22266/ijies2025.0229.01.

[13] T. Hamadneh *et al.*, "Sculptor Optimization Algorithm: A New Human-Inspired Metaheuristic Algorithm for Solving Optimization Problems," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 4, pp. 564–575, Aug. 2024, https://doi.org/10.22266/ijies2024.0831.43.

[14] N. Nouri and A. Tadaion, "Energy optimal resource allocation for mobile edge computation offloading in presence of computing access point," in *Iran Workshop on Communication and Information Theory*, Tehran, Iran, Apr. 2018, pp. 1–6, https://doi.org/10.1109/IWCIT.2018.8405049.

[15] N. Nouri, F. Fazel, J. Abouei, and K. N. Plataniotis, "Multi-UAV Placement and User Association in Uplink MIMO Ultra-Dense Wireless Networks," *IEEE Transactions on Mobile Computing*, vol. 22, no. 3, pp. 1615–1632, Mar. 2023, https://doi.org/10.1109/TMC.2021.3108960.

[16] N. Nouri, A. Entezari, J. Abouei, M. Jaseemuddin, and A. Anpalagan, "Dynamic Power–Latency Tradeoff for Mobile Edge Computation Offloading in NOMA-Based Networks," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2763–2776, Apr. 2020, https://doi.org/10.1109/JIOT.2019.2957313.

[17] J. de Armas, E. Lalla-Ruiz, S. L. Tilahun, and S. Voß, "Similarity in metaheuristics: a gentle step towards a comparison methodology," *Natural Computing*, vol. 21, no. 2, pp. 265–287, Jun. 2022, https://doi.org/10.1007/s11047-020-09837-9.

[18] E. Trojovska, M. Dehghani, and P. Trojovsky, "Zebra Optimization Algorithm: A New Bio-Inspired Optimization Algorithm for Solving Optimization Algorithm," *IEEE Access*, vol. 10, pp. 49445–49473, 2022, https://doi.org/10.1109/ACCESS.2022.3172789.

[19] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, Apr. 1997, https://doi.org/10.1109/4235.585893.

[20] S. Das and P. N. Suganthan, "Problem Definitions and Evaluation Criteria for CEC 2011 Competition on Testing Evolutionary Algorithms on Real World Optimization Problems," Jadavpur University and Nanyang Technological University, Technical Report, Dec. 2010.

[21] D. E. Goldberg and J. H. Holland, "Genetic Algorithms and Machine Learning," *Machine Learning*, vol. 3, no. 2, pp. 95–99, Oct. 1988, https://doi.org/10.1023/A:1022602019183.

[22] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *International Conference on Neural Networks*, Perth, WA, Australia, Dec. 1995, vol. 4, pp. 1942–1948 vol.4, https://doi.org/10.1109/ICNN.1995.488968.

[23] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A Gravitational Search Algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, Jun. 2009, https://doi.org/10.1016/j.ins.2009.03.004.

[24] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, Mar. 2011, https://doi.org/10.1016/j.cad.2010.12.015.

[25] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-Verse Optimizer: a nature-inspired algorithm for global optimization," *Neural Computing and Applications*, vol. 27, no. 2, pp. 495–513, Feb. 2016, https://doi.org/10.1007/s00521-015-1870-7.

[26] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, Mar. 2014, https://doi.org/10.1016/j.advengsoft.2013.12.007.

[27] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, May 2016, https://doi.org/10.1016/j.advengsoft.2016.01.008.

[28] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine Predators Algorithm: A nature-inspired metaheuristic," *Expert Systems with Applications*, vol. 152, Aug. 2020, Art. no. 113377, https://doi.org/10.1016/j.eswa.2020.113377.

[29] S. Kaur, L. K. Awasthi, A. L. Sangal, and G. Dhiman, "Tunicate Swarm Algorithm: A new bio-inspired based metaheuristic paradigm for global optimization," *Engineering Applications of Artificial Intelligence*, vol. 90, Apr. 2020, Art. no. 103541, https://doi.org/10.1016/j.engappai.2020.103541.

[30] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, "Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer," *Expert Systems with Applications*, vol. 191, Apr. 2022, Art. no. 116158, https://doi.org/10.1016/j.eswa.2021.116158.

[31] B. Abdollahzadeh, F. S. Gharehchopogh, and S. Mirjalili, "African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems," *Computers & Industrial Engineering*, vol. 158, Aug. 2021, Art. no. 107408, https://doi.org/10.1016/j.cie.2021.107408.

[32] M. Braik, A. Hammouri, J. Atwan, M. A. Al-Betar, and M. A. Awadallah, "White Shark Optimizer: A novel bio-inspired meta-heuristic algorithm for global optimization problems," *Knowledge-Based Systems*, vol. 243, May 2022, Art. no. 108457, https://doi.org/10.1016/j.knosys.2022.108457.

[33] F. Wilcoxon, "Individual Comparisons by Ranking Methods," in *Breakthroughs in Statistics: Methodology and Distribution*, S. Kotz and N. L. Johnson, Eds. New York, NY, USA: Springer, 1992, pp. 196–202.