Received: August 24, 2024. Revised: December 2, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Orangutan Optimization Algorithm: An Innovative Bio-Inspired Metaheuristic Approach for Solving Engineering Optimization Problems

Tareq Hamadneh[1] Belal Batiha[2] Gharib Mousa Gharib[3] Zeinab Montazeri[4] Frank Werner[5] Gaurav Dhiman[6,7,8] Mohammad Dehghani[4]* Riyadh Kareem Jawad[9] Erahid Aram[10] Ibraheem Kasim Ibraheem[11] Kei Eguchi[12]

[1]Department of Mathematics, Al Zaytoonah University of Jordan, Amman 11733, Jordan
[2]Department of Mathematics. Faculty of Science and Information Technology, Jadara University, Irbid 21110, Jordan
[3]Department of Mathematics, Faculty of Science, Zarqa University, Zarqa 13110 Zarqa, Jordan
[4]Department of Electrical and Electronics Engineering, Shiraz University of Technology, Shiraz 7155713876, Iran
[5]Faculty of Mathematics, Otto-von-Guericke University, P.O. Box 4120, 39016 Magdeburg, Germany
[6]Centre of Research Impact and Outcome, Chitkara University, Rajpura- 140417, Punjab, India
[7]Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun-248002, India
[8]Division of Research and Development, Lovely Professional University, Phagwara-144411, India
[9]Department of Medical Instrumentations Techniques Engineering, Al-Rasheed University College, Baghdad 10001, Iraq
[10]Department of Medical Instrumentation Techniques, Technical Engineering, Uruk University, Baghdad 10001, Iraq
[11]Department of Electrical Engineering, College of Engineering, University of Baghdad, Baghdad 10001, Iraq
[12]Department of Information Electronics, Fukuoka Institute of Technology, Japan
* Corresponding author's Email: adanbax@gmail.com

Abstract: In this paper, a new metaheuristic algorithm called Orangutan Optimization Algorithm (OOA) is designed, which imitates the behaviors of Orangutans in nature. The fundamental inspiration of OOA is the foraging strategy of Orangutans and the skills of these animals in nesting. The theory of OOA is explained and then the implementation steps of OOA in two phases of exploration and exploitation are mathematically modeled. The performance of OOA in dealing with real-world applications is evaluated on twenty-two constrained optimization problems from the CEC 2011 test suite. The optimization results show that the proposed OOA approach, by balancing exploration and exploitation during the search process, is able to provide suitable solutions for the benchmark functions. Also, in order to measure the quality of OOA, the results obtained from the proposed approach are compared with twelve well-known metaheuristic algorithms. Analysis of the simulation results shows that OOA has provided superior performance by providing better results in 100% of the benchmark functions compared to competitor algorithms.

Keywords: Orangutan, Nature-inspired, Optimization, Metaheuristic, Optimization algorithm, Exploration, Exploitation.

1. Introduction

Many problems in both science and real-world applications present multiple feasible solutions, which makes them complex to solve. These problems are referred to as optimization problems, and the process of identifying the most suitable solution from the set of available options is known as optimization [1]. From a mathematical perspective, optimization problems consist of three main components: decision variables, constraints, and objective functions. The goal of optimization, therefore, is to determine the optimal values for the decision variables while adhering to the constraints so that the objective function achieves its most favorable outcome, either maximum or minimum [2]. Optimization problem-solving methods are broadly classified into two completely different categories: deterministic and stochastic approaches [3]. Deterministic methods, which are further divided into gradient-based and non-gradient-based techniques, are particularly effective in solving linear, convex, continuous, differentiable, and low-dimensional optimization problems [4, 5]. However, as the complexity and dimensionality of these problems increase, deterministic methods often fail by getting trapped in suboptimal local solutions [6, 7]. This is especially true for problems that are non-linear, non-convex, discontinuous, non-differentiable, and high-dimensional, which are common in scientific and practical applications [8, 9]. Due to these limitations, researchers have developed stochastic approaches to tackle more challenging optimization problems [10].

Metaheuristic algorithms, which are one of the most widely employed stochastic methods, are highly effective in addressing complex optimization challenges. These algorithms have effective applications in various sciences such as architecture [11, 12], energy [13], protection [14], electrical engineering [15], and energy carriers [16]. Metaheuristic algorithms work by utilizing a random search mechanism within the problem space, employing random operators and a trial-and-error process to find suitable solutions. The advantages of metaheuristic algorithms include the simplicity of their concepts, ease of implementation, independence from the specific problem type, and the ability to solve non-linear, non-convex, discontinuous, non-derivative, and high-dimensional optimization problems. Furthermore, they are efficient in exploring unknown, non-linear search spaces, which explains their popularity among researchers [17].

The optimization process in metaheuristic algorithms begins by randomly generating a set of initial candidate solutions that respect the constraints of the problem. In an iterative process, these solutions are progressively refined based on the updating steps defined by the algorithm. The best solution found at each iteration is saved, and ultimately, the best overall solution is presented as the final result [18]. While the random search nature of these algorithms means that they cannot guarantee a global optimum, the solutions they produce are often near-optimal, which are referred to as quasi-optimal solutions. Consequently, when comparing the performance of multiple metaheuristic algorithms, the one that provides a better quasi-optimal solution is considered the most effective one for that particular problem [19].

The search process in metaheuristic algorithms must balance two key concepts: global exploration and local exploitation. Global exploration enables the algorithm to thoroughly scan the problem space, preventing it from getting stuck in local optima and helping to identify the most promising areas in the search space. Local exploitation, on the other hand, allows the algorithm to converge towards a global optimum by intensively searching around promising regions and refining solutions. Striking a balance between exploration and exploitation is crucial for the success of any metaheuristic algorithm in providing effective solutions [20].

A key question in metaheuristics research is whether, given the vast number of algorithms already developed, there is still a need to design completely different metaheuristic algorithms. The No Free Lunch (NFL) theorem provides an answer to this. It states that the success of a metaheuristic algorithm in solving one set of optimization problems does not guarantee its success in solving others [21]. Therefore, no single algorithm is universally optimal for all optimization tasks. This insight encourages ongoing innovation in the field of metaheuristic algorithm design, as the NFL theorem suggests that newer algorithms can offer more effective solutions for specific problem sets.

The novelty of this paper lies in the development of a new metaheuristic algorithm called the Orangutan Optimization Algorithm (OOA), designed to solve a variety of optimization problems in different scientific fields and real-world applications. The key contributions of the paper are as follows:

- OOA is inspired by the natural behavior of orangutans in the wild.
- The algorithm's core inspiration comes from the foraging strategies and nesting skills of orangutans.
- The steps of OOA are described and mathematically modeled in two phases: exploration and exploitation.
- To assess its effectiveness in real-world applications, OOA is applied to twenty-two optimization problems from the CEC 2011 test suite.
- A comparative performance analysis is conducted, comparing OOA with twelve other well-known metaheuristic algorithms.

The structure of the paper is as follows: Section 2 introduces and models the Orangutan Optimization Algorithm. Section 3 investigates the effectiveness of OOA in solving real-world applications. Finally, conclusions and suggestions for future research are discussed in Section 4.

2. Orangutan optimization algorithm

In this section, the source of inspiration behind the Orangutan Optimization Algorithm (OOA) is thoroughly explained, followed by a comprehensive description of the theory underlying the approach. Afterwards, the step-by-step implementation of the OOA is carefully modeled using mathematical formulations, ensuring it can be effectively applied to solve completely different types of optimization problems.

2.1 Algorithm initialization

The newly introduced Orangutan Optimization Algorithm (OOA) is a bio-inspired metaheuristic algorithm that draws its inspiration from the natural behaviors of orangutans. In this approach, orangutans serve as the population members, and each orangutan represents a potential solution to the given optimization problem. These solutions are completely different from one another, as each orangutan occupies a unique position within the problem's search space. The variables corresponding to each solution are determined by the orangutan's specific position, which can be mathematically modeled as a vector. As a group, these orangutans form the OOA population, which can be represented using a matrix structure using Eq. (1). This matrix is not static; it evolves as the algorithm progresses. Initially, the position of each orangutan in the search space is randomly determined. This randomness is essential in ensuring that the initial population covers diverse areas of the search space, enhancing the exploration capabilities of the algorithm. The initialization process for the population is mathematically modeled using Eq. (2), where each dimension of the orangutan's position is calculated based on random values within a predefined range. This allows for a completely different starting point for each orangutan, creating diversity in candidate solutions.

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

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

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

After initialization, each orangutan's position corresponds to a set of variables, which are evaluated using the objective function of the optimization problem. The objective function assigns a value to each candidate solution, and this set of values can be represented mathematically using a vector, as shown in Eq. (3).

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

Here F is the vector of calculated objective function and F_i is the calculated objective function based on the ith orangutan.

The calculated values of the objective function serve as a measure of the quality of each solution. Based on these evaluations, the algorithm identifies the best-performing orangutan (i.e., the candidate solution with the most optimal value) as well as the worst-performing one. In each iteration of the algorithm, the positions of the orangutans are updated, which means that their corresponding objective function values also change. As the search progresses, the best solution must be continuously updated to reflect the most optimal orangutan found so far.

This iterative process of updating orangutan positions ensures that the algorithm effectively searches the problem space, gradually moving towards an optimal or near-optimal solution.

2.2 Phase 1: foraging strategy (exploration)

Orangutans, in their natural habitat, spend a significant amount of time searching for food such as fruits, tree leaves, and other diet items. This foraging behavior leads to large-scale movements and extensive exploration in their environment, allowing them to discover completely different areas in the search of resources. The simulation of this foraging strategy within OOA enhances the algorithm's exploration capability, making it more adept at scanning and searching the global space of the problem.

In the first phase of OOA, the position of each orangutan is updated to simulate this foraging behavior. Orangutans with better objective function values are considered to represent better food sources, and each orangutan seeks out these superior positions. Eq. (4) mathematically defines the set of available food resources for each orangutan by considering all orangutans with better objective function values. The diversity of food sources allows the orangutans to explore a variety of potential solutions in completely different regions of the search space.

$$FS_i = \{X_k : F_k < F_i \; and \; k \neq i\} \tag{4}$$

Here FS_i is the set of candidate food sources' locations for the ith orangutan, X_k is the is the orangutan with a better objective function value than ith orangutan, and F_k is the its objective function value.

To model this movement mathematically, a new position is first calculated for each orangutan using Eq. (5). This movement allows the orangutan to adjust its location in a way that explores completely different regions, potentially leading to a significant improvement in the objective function value. If the objective function improves, the new position is confirmed and updated according to Eq. (6):

$$x_{i,d}^{P1} = x_{i,d} + r \cdot \left(SFS_{i,d} - I \cdot x_{i,d}\right), \tag{5}$$

$$X_i = \begin{cases} X_i^{P1}, \; F_i^{P1} \leq F_i, \\ X_i, \; else \, , \end{cases} \tag{6}$$

Here X_i^{P1} is the new suggested position of the ith orangutan based on the first phase of OOA, $x_{i,d}^{P1}$ is its dth dimension, F_i^{P1} is its objective function value, r is a random number with a normal distribution in the range of $[0,1]$, $SFS_{i,d}$ is the dth dimension of the selected food source for the ith orangutan, I is a random number from the set $\{1,2\}$, N is the number of orangutans, and m is the number of decision variables.

2.3 Phase 2: nesting skill (exploitation phase)

In addition to foraging, orangutans also demonstrate remarkable intelligence through their nesting behavior. Every day, they build nests in trees, selecting branches and leaves near their current location. This activity focuses on a more localized search, optimizing their living space. Simulating the nesting skills of orangutans in OOA enhances the algorithm's exploitation capabilities, improving the fine-tuning of solutions and allowing for more precise exploration of local regions.

During this second phase of OOA, the orangutan moves towards a nearby tree to nest. In the context of the algorithm, this nesting process is modeled by generating a new position for the orangutan based on its current location. Eq. (7) is used to simulate the movement towards the tree, and if the objective function value improves, the new position replaces the previous one, as outlined in Eq. (8):

$$x_{i,j}^{P2} = x_{i,j} + \left(1 - 2\, r_{i,j}\right) \cdot \frac{ub_j - lb_j}{t} \tag{7}$$

$$X_i = \begin{cases} X_i^{P2}, \; F_i^{P2} \leq F_i \\ X_i, \; else \end{cases} \tag{8}$$

Here X_i^{P2} is the new suggested position of the ith orangutan based on the second phase of OOA, $x_{i,d}^{P2}$ is its dth dimension, 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.

3. Simulation studies

In this section, the effectiveness of the proposed OOA approach in addressing optimization challenges in real-world applications is thoroughly examined. To achieve this, the CEC 2011 test suite, which comprises twenty-two constrained optimization problems derived from practical applications, has been employed. A comprehensive description and full details of the CEC 2011 test suite are available in [22]. The optimization results obtained by the proposed OOA approach are compared with the performance of twelve well-established metaheuristic algorithms, including: GA [23], PSO [24], GSA [25], TLBO [26], MVO [27], GWO [28], WOA [29], MPA [30], TSA [31], RSA [32], AVOA [33], and WSO [34].

The results of the OOA implementation, along with competitor algorithms, when applied to the CEC 2011 test suite, are summarized in Table 1 and Table 2. Additionally, the comparative performance of OOA and the other algorithms is illustrated through boxplots in Figure 1. From a detailed comparison of the simulation results, it becomes clear that the OOA approach consistently outperforms the competitors, emerging as the top optimizer across all the problems ranging from C11-F1 to C11-F22. This clearly highlights OOA's robustness and effectiveness in dealing with real-world optimization problems. Notably, the simulation results indicate that OOA ranks as the best optimizer in the majority of the optimization problems from the CEC 2011 test suite, showcasing its superior performance when compared to the other algorithms.

Received: August 24, 2024. Revised: December 2, 2024.

Table 1 Optimization results of CEC 2011 test suite

		OOA	WSO	AVOA	RSA	MPA	TSA	WOA
C11-F1	mean	5.920103	17.39022	13.33731	21.05706	8.715436	18.01704	13.58734
	best	2E-10	14.61193	9.495939	18.71472	1.804603	17.30358	8.520687
	worst	12.30606	19.66117	16.68954	23.05217	13.19969	19.09799	16.93418
	std	7.196379	2.400987	4.013324	1.99914	5.368972	0.818959	4.018352
	median	5.687176	17.6439	13.58188	21.23068	9.928723	17.8333	14.44725
	rank	1	7	4	12	2	9	5
C11-F2	mean	-26.3179	-14.2624	-19.9064	-11.8736	-23.3368	-11.6356	-17.8532
	best	-27.0676	-15.7508	-20.5241	-12.1168	-23.9079	-15.1534	-21.0988
	worst	-25.4328	-13.1382	-19.3344	-11.5246	-22.093	-9.58355	-14.2074
	std	0.738935	1.333128	0.544996	0.26244	0.884624	2.715589	3.570128
	median	-26.3856	-14.0804	-19.8835	-11.9266	-23.6732	-10.9027	-18.0533
	rank	1	8	5	10	2	11	6
C11-F4	mean	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05
	best	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05
	worst	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05
	std	2E-19	1.86E-11	2.13E-09	4.18E-11	6.64E-14	5.81E-14	6.62E-14
	median	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05
	rank	1	11	13	12	6	8	4
C11-F4	mean	0	0	0	0	0	0	0
	best	0	0	0	0	0	0	0
	worst	0	0	0	0	0	0	0
	std	0	0	0	0	0	0	0
	median	0	0	0	0	0	0	0
	rank	1	1	1	1	1	1	1
C11-F5	mean	-34.1274	-25.2175	-27.9963	-21.1342	-32.3358	-27.1746	-27.5944
	best	-34.7494	-26.0919	-28.9217	-22.8513	-32.8473	-30.8993	-27.7615
	worst	-33.3862	-24.4915	-27.632	-19.2053	-31.2781	-22.7089	-27.2665
	std	0.589989	0.736666	0.651398	2.066682	0.751817	3.543313	0.235926
	median	-34.1871	-25.1432	-27.7158	-21.2401	-32.609	-27.5451	-27.6749
	rank	1	9	4	10	2	7	5
C11-F6	mean	-24.1119	-14.0897	-18.2974	-13.2521	-21.3104	-8.63171	-19.0738
	best	-27.4298	-14.764	-19.1411	-14.6227	-23.6053	-16.0875	-22.4303
	worst	-23.0059	-13.5426	-17.602	-12.2858	-19.8721	-5.52345	-13.0723
	std	2.324951	0.65053	0.809346	1.065169	1.807309	5.248737	4.379777
	median	-23.0059	-14.026	-18.2232	-13.0499	-20.8821	-6.45794	-20.3964
	rank	1	7	6	8	2	10	4
C11-F7	mean	0.860699	1.558108	1.288623	1.820738	0.99202	1.303556	1.673227
	best	0.582266	1.521316	1.182697	1.605716	0.862127	1.172989	1.588548
	worst	1.025027	1.635861	1.394596	1.96165	1.059349	1.59616	1.803919
	std	0.211503	0.055246	0.118873	0.159565	0.095148	0.206503	0.098956
	median	0.91775	1.537629	1.2886	1.857793	1.023301	1.222537	1.650222
	rank	1	9	7	13	3	8	12
C11-F8	mean	220	278.3988	241.0229	312.3393	225.8807	255.1568	262.5186
	best	220	255.5256	226.2458	277.3029	222.2252	222.2252	243.9974
	worst	220	310.8709	258.4594	348.7748	231.2187	340.1051	304.4945
	std	0	25.63005	14.43071	31.00166	4.442577	59.87582	29.60888
	median	220	273.5994	239.6932	311.6397	225.0395	229.1484	250.7913
	rank	1	11	7	12	2	9	10
C11-F9	mean	8789.286	504826.7	354448.5	929153.6	53342.71	92037.41	351366.3
	best	5457.674	345338.8	322389.6	615208	32194.85	74383.89	197262.9
	worst	14042.29	584496.9	381767.1	1088516	65230.14	119930.5	582900.7
	std	3889.181	117306.7	26662.14	223901	15947.1	21113.5	183601
	median	7828.591	544735.5	356818.6	1006445	57972.91	86917.64	312650.8
	rank	1	9	7	11	2	4	6
C11-F10	mean	-21.4889	-13.6607	-16.1383	-12.2399	-17.9061	-14.0129	-12.7392
	best	-21.8299	-14.3725	-16.7205	-12.4723	-18.5767	-17.4621	-13.2762
	worst	-20.7878	-13.1969	-15.7944	-11.9599	-17.3594	-11.938	-12.3829
	std	0.498616	0.527305	0.445566	0.266005	0.585583	2.52099	0.409228
	median	-21.669	-13.5366	-16.0191	-12.2637	-17.8442	-13.3258	-12.6488
	rank	1	7	3	10	2	5	9
C11-F11	mean	571712.3	5272454	1232488	7839331	1793336	5392008	1420866
	best	260837.9	5107868	1058371	7600047	1728921	4633051	1360288
	worst	828560.9	5525958	1352575	7967240	1875084	6379519	1525324
	std	260922.1	207728.9	130596.6	180776.5	64144.21	762344.6	75885.36
	median	598725.2	5227994	1259503	7895018	1784669	5277731	1398926
	rank	1	10	2	13	6	11	3
C11-F12	mean	1199805	7567324	3356778	11634817	1593020	4745383	5404660

	best	1155937	7280380	3293066	10848229	1533606	4529257	5016677
	worst	1249353	7835976	3423927	12340363	1669648	4877249	5583985
	std	47157.58	247077.4	56234.99	642185.6	59426.91	173011	280042.1
	median	1196965	7576470	3355059	11675339	1584414	4787514	5508990
	rank	1	10	6	11	2	7	9
C11-F13	mean	15444.2	15805.61	15461.49	16185.9	15474.24	15496.94	15534.98
	best	15444.19	15647.1	15457.78	15842.51	15469.61	15485.91	15494.82
	worst	15444.21	16179.68	15467.86	17050.98	15479.23	15508.27	15590.66
	std	0.009091	264.6824	4.630429	611.0784	4.134745	13.28135	46.60689
	median	15444.2	15697.84	15460.17	15925.06	15474.06	15496.79	15527.22
	rank	1	9	2	11	3	4	8
C11-F14	mean	18295.35	97633.27	18682.8	195620.9	18757.09	19533.06	19276.3
	best	18241.58	75011.44	18613.02	144878.1	18661.23	19307.69	19122.12
	worst	18388.08	135292.4	18772.36	280518.7	18827.98	20021.3	19404.44
	std	71.59938	28333.14	85.39038	63854.81	80.72001	345.4167	134.2456
	median	18275.87	90114.63	18672.9	178543.3	18769.57	19401.62	19289.31
	rank	1	11	2	12	3	10	7
C11-F15	mean	32883.58	781176.2	108312	1627642	45341.41	63506.54	201447.2
	best	32782.17	319976.9	43595.67	678569	32904.1	33271.47	33021.41
	worst	32956.46	1941570	171262.9	4224324	51625.56	123795.5	287013.3
	std	76.94696	816676.8	70280.06	1822751	8897.118	42920.58	120525.5
	median	32897.86	431579.1	109194.6	803837.2	48417.99	48479.6	242877.1
	rank	1	10	7	11	2	6	8
C11-F16	mean	133550	817900.7	136863	1662172	138948.3	145227.4	142736.4
	best	131374.2	264960.8	136069.7	420357.3	137599.1	143439.6	136665.8
	worst	136310.8	1901072	137623.9	4096895	142513	147081.2	147681.1
	std	2392.2	773155	895.5159	1737676	2504.36	1601.356	4827.072
	median	133257.5	552785	136879.1	1065717	137840.5	145194.4	143299.4
	rank	1	8	2	9	3	6	5
C11-F17	mean	1926615	8.18E+09	2.59E+09	1.37E+10	6.48E+08	1.72E+09	8.8E+09
	best	1916953	7.28E+09	2.23E+09	1.02E+10	4.63E+08	1.35E+09	6.28E+09
	worst	1942685	8.82E+09	2.73E+09	1.64E+10	8.61E+08	2.09E+09	1.17E+10
	std	12003.53	7.26E+08	2.55E+08	2.79E+09	1.76E+08	3.26E+08	2.4E+09
	median	1923412	8.32E+09	2.71E+09	1.41E+10	6.35E+08	1.73E+09	8.61E+09
	rank	1	7	6	10	2	5	8
C11-F18	mean	942057.5	46968550	6246568	1E+08	1551619	2458384	8789489
	best	938416.2	32165899	3860025	69122918	1166395	2105030	3833110
	worst	944706.9	53337276	10680388	1.14E+08	2092481	2764259	15367297
	std	2774.139	10456522	3372738	22297257	413798.4	297588.8	5119786
	median	942553.5	51185512	5222930	1.09E+08	1473799	2482124	7978775
	rank	1	10	6	12	2	5	7
C11-F19	mean	1025341	46337780	6425512	98329374	1785097	2899712	9420437
	best	967927.7	39329617	5431525	84714139	1177061	2367354	1993479
	worst	1167142	59275780	8194119	1.24E+08	2340168	3422486	16978182
	std	99675.04	9379552	1288187	19079537	614570.1	483548.6	7410517
	median	983146.6	43372862	6038202	92318587	1811579	2904504	9355044
	rank	1	10	7	12	2	5	8
C11-F20	mean	941250.4	48967515	5532949	1.06E+08	1385714	2114060	6700575
	best	936143.2	43137685	4951268	92693847	1359767	1937660	6319847
	worst	946866.6	57917146	6133039	1.26E+08	1422491	2356836	7209610
	std	5013.552	6621981	512099.1	14830816	29465.74	198216.7	400304.6
	median	940995.9	47407614	5523745	1.02E+08	1380299	2080873	6636421
	rank	1	10	6	12	2	5	7
C11-F21	mean	12.71443	47.65386	23.5954	69.70586	18.79433	30.4773	38.01922
	best	9.974206	40.21741	22.58455	53.28962	17.08664	27.83265	35.2891
	worst	14.97499	55.7929	25.17928	86.27352	20.76606	31.91199	41.59334
	std	2.412667	7.133006	1.166637	15.39462	1.711693	1.90116	3.027254
	median	12.95425	47.30257	23.30889	69.63015	18.66231	31.08228	37.59723
	rank	1	9	3	10	2	6	7
C11-F22	mean	16.12513	45.62653	29.33206	59.65088	22.29539	33.27953	45.2224
	best	11.50133	40.43442	24.92922	44.92083	19.89289	29.90147	39.94322
	worst	19.55286	50.76417	34.23744	68.2016	24.26827	35.40932	49.68618
	std	4.197797	4.567446	4.678654	10.69338	2.342326	2.483074	4.622637
	median	16.72317	45.65375	29.08079	62.74054	22.51019	33.90365	45.63009
	rank	1	9	4	10	2	5	7
Sum rank		22	192	110	232	55	147	146
Mean rank		1	8.727273	5	10.54545	2.5	6.681818	6.636364
Total rank		1	9	4	13	2	7	6
Wilcoxon: *p*-value			2.00E-14	3.54E-17	7.12E-18	7.97E-06	2.23E-17	2.40E-17

Table 2 Optimization results of CEC 2011 test suite

		OOA	MVO	GWO	TLBO	GSA	PSO	GA
C11-F1	mean	5.920103	14.23293	11.54006	18.04442	20.81796	17.62171	22.26941
	best	2E-10	12.06721	2.440182	16.94885	18.26611	11.98458	21.09797
	worst	12.30606	16.75519	17.9142	20.10811	22.0111	23.14246	24.25607
	std	7.196379	2.542719	6.891004	1.494681	1.835857	5.329393	1.456657
	median	5.687176	14.05466	12.90292	17.56037	21.49731	17.67991	21.8618
	rank	1	6	3	10	11	8	13
C11-F2	mean	-26.3179	-9.528	-21.2568	-11.3023	-15.241	-21.2983	-13.0262
	best	-27.0676	-11.2824	-23.3867	-12.0527	-19.2658	-22.4767	-15.3514
	worst	-25.4328	-8.10438	-18.0907	-10.7637	-11.6643	-19.023	-11.3361
	std	0.738935	1.384544	2.41029	0.60266	3.68196	1.640456	1.953962
	median	-26.3856	-9.36262	-21.775	-11.1964	-15.0169	-21.8468	-12.7086
	rank	1	13	4	12	7	3	9
C11-F4	mean	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05
	best	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05
	worst	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05
	std	2E-19	9E-13	6.7E-14	1.08E-13	6.62E-14	6.62E-14	6.62E-14
	median	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05	1.15E-05
	rank	1	10	7	9	3	2	5
C11-F4	mean	0	0	0	0	0	0	0
	best	0	0	0	0	0	0	0
	worst	0	0	0	0	0	0	0
	std	0	0	0	0	0	0	0
	median	0	0	0	0	0	0	0
	rank	1	1	1	1	1	1	1
C11-F5	mean	-34.1274	-27.0569	-30.9071	-13.3897	-27.3571	-11.5656	-12.2902
	best	-34.7494	-31.0961	-33.1007	-15.1957	-30.9286	-14.6193	-13.5569
	worst	-33.3862	-25.0029	-27.5749	-12.0369	-24.7099	-10.0326	-10.9043
	std	0.589989	3.016868	2.471951	1.43243	2.882071	2.246734	1.211154
	median	-34.1871	-26.0642	-31.4763	-13.1632	-26.8949	-10.8052	-12.3498
	rank	1	8	3	11	6	13	12
C11-F6	mean	-24.1119	-10.2921	-18.8022	-4.21747	-20.6998	-4.94712	-5.71111
	best	-27.4298	-17.7564	-20.7928	-4.93827	-24.3079	-7.0585	-9.74747
	worst	-23.0059	-3.7729	-17.0597	-3.7729	-16.9161	-3.7729	-4.0188
	std	2.324951	7.752062	1.994112	0.529975	3.477442	1.570632	2.860723
	median	-23.0059	-9.81963	-18.6782	-4.07935	-20.7876	-4.47854	-4.53909
	rank	1	9	5	13	3	12	11
C11-F7	mean	0.860699	0.951373	1.107406	1.65236	1.117491	1.15425	1.670577
	best	0.582266	0.885728	0.881194	1.479679	0.968488	0.897931	1.35767
	worst	1.025027	1.02631	1.309296	1.756946	1.298184	1.369651	1.829346
	std	0.211503	0.075607	0.186797	0.131117	0.158446	0.258564	0.224931
	median	0.91775	0.946726	1.119568	1.686408	1.101647	1.17471	1.747647
	rank	1	2	4	10	5	6	11
C11-F8	mean	220	227.2504	229.9897	227.2504	245.9581	433.4045	225.9188
	best	220	222.2252	222.2252	222.2252	222.2252	246.7904	222.2252
	worst	220	236.4643	239.4366	240.8062	288.7436	521.1187	231.5722
	std	0	6.826963	9.25062	9.508314	32.95358	135.3033	4.579606
	median	220	225.156	229.1484	222.985	236.4317	482.8545	224.9389
	rank	1	5	6	4	8	13	3
C11-F9	mean	8789.286	148488.9	72528.14	379914.4	728516.7	946357	1669534
	best	5457.674	112757.3	38425.67	333470.8	624500.4	753374.8	1597320
	worst	14042.29	201975.7	104307.5	472974.3	768053.7	1146793	1778056
	std	3889.181	41402.58	28515.35	66519.34	73166.54	216193.9	88492.94
	median	7828.591	139611.4	73689.68	356606.2	760756.3	942629.9	1651381
	rank	1	5	3	8	10	12	13
C11-F10	mean	-21.4889	-14.2748	-13.7729	-11.4072	-12.9746	-11.491	-11.2428
	best	-21.8299	-20.137	-14.5805	-11.9091	-13.8309	-11.9553	-11.6709
	worst	-20.7878	-11.2489	-12.4952	-11.0793	-12.0186	-11.1889	-10.9583
	std	0.498616	4.184379	0.941716	0.371762	0.818242	0.343631	0.318359
	median	-21.669	-12.8566	-14.0078	-11.3203	-13.0246	-11.4099	-11.171
	rank	1	4	6	12	8	11	13
C11-F11	mean	571712.3	1498856	3618899	4774892	1585228	4784193	5541427
	best	260837.9	859467.4	3495887	4729579	1539942	4748182	5463937
	worst	828560.9	2772377	3928789	4846534	1677837	4846534	5679235
	std	260922.1	906932.5	217780.3	52692.61	67680.62	45577.61	99412.38
	median	598725.2	1181790	3525459	4761727	1561566	4771028	5511268
	rank	1	4	7	8	5	9	12
C11-F12	mean	1199805	1637420	1718522	12552409	5383533	2465510	12687503
	best	1155937	1473502	1584886	11856053	5104008	2332813	12602758

	worst	1249353	1780806	1844952	13104339	5562149	2663064	12761027
	std	47157.58	132583.4	112208.9	553507.6	211689.3	148250.5	68583.36
	median	1196965	1647685	1722125	12624622	5433988	2433081	12693114
	rank	1	3	4	12	8	5	13
C11-F13	mean	15444.2	15511.81	15506.11	15869.08	110296.6	15497.48	27688.41
	best	15444.19	15491.36	15497.6	15609.61	80421.19	15480.31	15478.16
	worst	15444.21	15550.36	15512.72	16342.08	150810.9	15524.7	64042.27
	std	0.009091	28.12276	6.737851	348.5	33308.79	20.1553	25474.47
	median	15444.2	15502.77	15507.06	15762.32	104977.1	15492.45	15616.61
	rank	1	7	6	10	13	5	12
C11-F14	mean	18295.35	19439.01	19282.13	264278.7	19164.53	19191.89	19181.17
	best	18241.58	19329.12	19134.19	28547.81	18912.02	19044.6	18932.51
	worst	18388.08	19534.56	19465.07	507229.8	19316.17	19321.83	19459.96
	std	71.59938	91.02157	152.7497	241544.1	188.2708	122.2792	227.9891
	median	18275.87	19446.17	19264.63	260668.5	19214.96	19200.56	19166.11
	rank	1	9	8	13	4	6	5
C11-F15	mean	32883.58	45468.92	45450.31	12983382	269689.6	45627.01	6690553
	best	32782.17	33026.54	33051.55	2736525	228340.9	33253.65	3042549
	worst	32956.46	51781.2	51703.43	19358805	291640.4	51902.61	11460012
	std	76.94696	8899.828	8868.206	7940200	30947.35	8853.229	4054593
	median	32897.86	48533.97	48523.12	14919100	279388.6	48675.9	6129825
	rank	1	4	3	13	9	5	12
C11-F16	mean	133550	142437.6	145854.6	74771248	15765094	66926236	64261251
	best	131374.2	134044.7	142569.7	72863394	8018850	55365044	51941418
	worst	136310.8	150317.3	150973.4	76922213	28504369	79970878	82187377
	std	2392.2	7126.149	3777.487	1787965	9310709	11148097	13504451
	median	133257.5	142694.1	144937.6	74649692	13268579	66184512	61458106
	rank	1	4	7	13	10	12	11
C11-F17	mean	1926615	6.49E+08	6.49E+08	1.94E+10	1.01E+10	1.82E+10	1.9E+10
	best	1916953	4.64E+08	4.63E+08	1.89E+10	8.88E+09	1.6E+10	1.79E+10
	worst	1942685	8.61E+08	8.62E+08	2E+10	1.07E+10	2.07E+10	2.13E+10
	std	12003.53	1.76E+08	1.76E+08	5.05E+08	8.76E+08	2.17E+09	1.7E+09
	median	1923412	6.35E+08	6.36E+08	1.93E+10	1.04E+10	1.8E+10	1.85E+10
	rank	1	4	3	13	9	11	12
C11-F18	mean	942057.5	1565381	1601545	26802059	10091478	1.14E+08	97052378
	best	938416.2	1208053	1185497	21028845	7729212	95817548	93529199
	worst	944706.9	2036921	2235896	29424540	13030136	1.26E+08	1E+08
	std	2774.139	368195.2	475034.6	4090135	2516208	14506058	3013638
	median	942553.5	1508275	1492394	28377426	9803282	1.17E+08	97106275
	rank	1	3	4	9	8	13	11
C11-F19	mean	1025341	2070008	1974259	30755816	6098260	1.46E+08	97467953
	best	967927.7	1210246	1380176	21455451	3147832	1.33E+08	94730931
	worst	1167142	3061406	2544052	38744416	8333924	1.68E+08	1.01E+08
	std	99675.04	845585.3	632984	7705989	2276422	16187798	2637573
	median	983146.6	2004190	1986404	31411698	6455643	1.41E+08	97257575
	rank	1	4	3	9	6	13	11
C11-F20	mean	941250.4	1396197	1417828	29648330	12578064	1.35E+08	97487351
	best	936143.2	1377810	1385798	28996475	8553436	1.23E+08	92859483
	worst	946866.6	1424318	1451737	30311072	19190127	1.46E+08	1.01E+08
	std	5013.552	23341.83	32784.24	576825.8	4891126	13500555	3641038
	median	940995.9	1391330	1416889	29642886	11284346	1.35E+08	98019702
	rank	1	3	4	9	8	13	11
C11-F21	mean	12.71443	28.5535	24.21289	90.0843	39.62681	94.3193	91.6871
	best	9.974206	25.60463	22.88643	45.69564	35.56843	81.89725	54.46949
	worst	14.97499	31.2056	26.23347	130.3305	41.73212	104.4449	110.8826
	std	2.412667	3.198601	1.538292	36.44639	3.002921	11.57281	27.60483
	median	12.95425	28.70189	23.86583	92.15553	40.60334	95.46753	100.6981
	rank	1	5	4	11	8	13	12
C11-F22	mean	16.12513	33.41087	27.26913	92.56565	45.52291	95.95859	84.12789
	best	11.50133	26.60935	26.82319	61.62802	39.50369	81.4849	83.34725
	worst	19.55286	38.15783	27.77753	108.4826	52.56221	105.311	85.4577
	std	4.197797	5.34673	0.464144	22.23837	5.653775	11.1633	0.993915
	median	16.72317	34.43814	27.2379	100.076	45.01286	98.51922	83.8533
	rank	1	6	3	12	8	13	11
Sum rank		22	119	98	222	158	199	224
Mean rank		1	5.409091	4.454545	10.09091	7.181818	9.045455	10.18182
Total rank		1	5	3	11	8	10	12
Wilcoxon: p-value			7.29E-14	8.79E-15	1.52E-17	3.67E-17	7.12E-18	1.04E-17

Received: August 24, 2024. Revised: December 2, 2024.

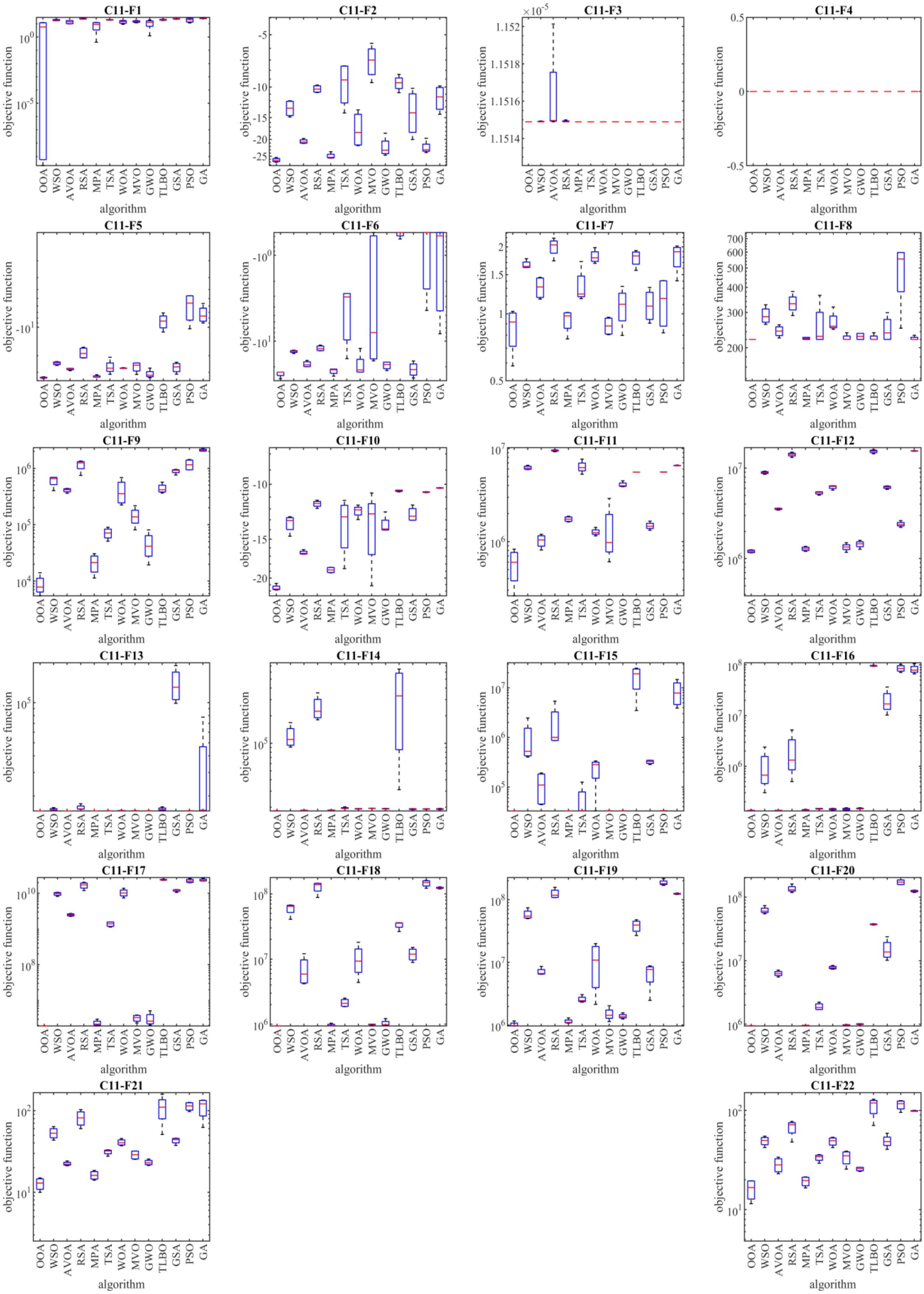

Figure 1 Boxplot diagrams of OOA and the competitor algorithms performances for the CEC 2011 test suite

Moreover, the statistical analysis conducted using the Wilcoxon rank-sum test further reinforces the advantage of the OOA approach. The results demonstrate a significant statistical superiority for OOA over the competing algorithms, confirming its enhanced ability to optimize the CEC 2011 test suite more effectively.

4. Conclusions and future Works

In this paper, a completely different bio-inspired metaheuristic algorithm called the Orangutan Optimization Algorithm (OOA) was introduced, showcasing its capability in solving optimization problems across a variety of scientific disciplines and real-world applications. This novel approach is inspired by two distinct natural behaviors of orangutans: their foraging strategies for obtaining food and their nesting behavior for resting. These behaviors formed the foundation for the design of OOA, which was mathematically structured into two key phases—exploration and exploitation. The performance of OOA was applied to twenty-two optimization problems from the CEC 2011 test suite, further validating its effectiveness in solving real-world optimization challenges. When compared to twelve other widely-recognized metaheuristic algorithms, OOA showed a superior performance, producing better optimization results for most benchmark functions. Moreover, the introduction of OOA opens several new avenues for future research. One of the most promising directions is the development of binary and multi-objective versions of OOA. Furthermore, expanding the application of OOA to address optimization challenges in completely different scientific fields and a broader range of real-world scenarios provides exciting opportunities for future studies.

Conflicts of Interest

"The authors declare no conflict of interest."

Author Contributions

Conceptualization, T.H, B.B, E.A, R.K.J, and G.M.G; methodology, TH, M.D, G.D, F.W, and K.E; software, K.E, B.B, G.D, R.K.J, and G.M.G; validation, K.E, M.D, I.K.I, and F.W; formal analysis, Z.M, M.D, and K.E; investigation, B.B, I.K.I, Z.M, E.A, and G.M.G; resources, T.H, R.K.J, Z.M, F.W, G.D, and B.B; data curation, K.E, I.K.I, and G.M.G; writing—original draft preparation, M.D, T.H, F.W, E.A, and O.P.M; writing—review and editing, G.M.G, Z.M, B.B, G.D, R.K.J, and K.E; visualization, E.A, I.K.I, and K.E; supervision, M.D; project administration, K.E, T.H, I.K.I, and F.W; funding acquisition, K.E.

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, p. 105075, 2022.

[2] Y. D. Sergeyev, D. Kvasov, and M. Mukhametzhanov, "On the efficiency of nature-inspired metaheuristics in expensive global optimization with limited budget", *Scientific reports*, Vol. 8, No. 1, pp. 1-9, 2018.

[3] L. Liberti and S. Kucherenko, "Comparison of deterministic and stochastic approaches to global optimization", *International Transactions in Operational Research*, Vol. 12, No. 3, pp. 263-285, 2005.

[4] 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. 4, pp. 1029-1044, 2020.

[5] T. Hamadneh, N. Athanasopoulos, and M. Ali, "Minimization and positivity of the tensorial rational Bernstein form", In: *Proc. of 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT - Proceedings)*, pp. 474-479, 2019.

[6] M. Dehghani, E. Trojovská, and P. Trojovský, "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, p. 9924, 2022, doi: 10.1038/s41598-022-14225-7.

[7] T. Hamadneh and R. Wisniewski, "The Barycentric Bernstein Form for Control Design", In: *Proc. of 2018 Annual American Control Conference (ACC)*, pp. 3738-3743, 2018, doi: 10.23919/ACC.2018.8431599.

[8] T. Hamadneh, A. Hioual, O. Alsayyed, Y. A. Al-Khassawneh, A. Al-Husban, and A. Ouannas, "The FitzHugh–Nagumo Model Described by Fractional Difference Equations: Stability and Numerical Simulation", *Axioms*, Vol. 12, No. 9, p. 806, 2023.

[9] T. Hamadneh, M. Ali, and H. AL-Zoubi, "Linear Optimization of Polynomial Rational Functions: Applications for Positivity Analysis", *Mathematics*, Vol. 8, No. 2, p. 283, 2020.

[10] 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

[11] S. Shafa, "Ranking of Smart Building Design Factors with Efficient Energy Management Systems and Renewable Resources", *Journal of Design Studio*, Vol. 6, No. 2, pp. 325-335, 2024, doi: 10.46474/jds.1575903.

[12] S. Shafa, "Smart Materials in Green Architecture: The Role of ETFE and Phase Change Materials in Sustainable Building Design", *Journal of Design Studio,* Vol. 6, No. 2, pp. 383-395, 2024, doi: 10.46474/jds.1556305.

[13] M. Dehghani *et al.*, "Energy Commitment for a Power System Supplied by Multiple Energy Carriers System using Following Optimization Algorithm", *Applied Sciences*, Vol. 10, No. 17, p. 5862, 2020.

[14] A. Ehsanifar, M. Dehghani, and M. Allahbakhshi, "Calculating the leakage inductance for transformer inter-turn fault detection using finite element method", In: *Proc. of 2017 Iranian Conference on Electrical Engineering (ICEE)*, Tehran, pp. 1372-1377, 2017.

[15] M. Dehghani, Z. Montazeri, and O. Malik, "Optimal sizing and placement of capacitor banks and distributed generation in distribution systems using spring search algorithm", *International Journal of Emerging Electric Power Systems,* Vol. 21, No. 1, 2020.

[16] M. Dehghani, Z. Montazeri, A. Ehsanifar, A. R. Seifi, M. J. Ebadi, and O. M. Grechko, "Planning of energy carriers based on final energy consumption using dynamic programming and particle swarm optimization", *Electrical Engineering & Electromechanics,* No. 5, pp. 62-71, 2018, doi: 10.20998/2074-272x.2018.5.10.

[17] M. Dehghani and H. Samet, "Momentum search algorithm: A new meta-heuristic optimization algorithm inspired by momentum conservation law", *SN Applied Sciences,* Vol. 2, No. 10, pp. 1-15, 2020.

[18] 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, 2022.

[19] M. Dehghani *et al.*, "A spring search algorithm applied to engineering optimization problems", *Applied Sciences,* Vol. 10, No. 18, p. 6173, 2020.

[20] I. Matoušová, P. Trojovský, M. Dehghani, E. Trojovská, and J. Kostra, "Mother optimization algorithm: a new human-based metaheuristic approach for solving engineering optimization", *Scientific Reports,* Vol. 13, No. 1, p. 10312, 2023, doi: 10.1038/s41598-023-37537-8.

[21] 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, 1997.

[22] 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, Nanyang Technological University, Kolkata,* pp. 341-359, 2010.

[23] D. E. Goldberg and J. H. Holland, "Genetic Algorithms and Machine Learning", *Machine Learning,* Vol. 3, No. 2, pp. 95-99, 1988, doi: 10.1023/A:1022602019183.

[24] J. Kennedy and R. Eberhart, "Particle swarm optimization", In: *Proc. of ICNN'95 - International Conference on Neural Networks*, Perth, WA, Australia, Vol. 4, pp. 1942-1948, 1995, doi: 10.1109/ICNN.1995.488968.

[25] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm", *Information sciences*, Vol. 179, No. 13, pp. 2232-2248, 2009.

[26] R. V. Rao, V. J. Savsani, and D. Vakharia, "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems", *Computer-Aided Design*, Vol. 43, No. 3, pp. 303-315, 2011.

[27] 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, 2016.

[28] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer", *Advances in Engineering Software,* Vol. 69, pp. 46-61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[29] S. Mirjalili and A. Lewis, "The whale optimization algorithm", *Advances in Engineering Software,* Vol. 95, pp. 51-67, 2016.

[30] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine Predators Algorithm: A nature-inspired metaheuristic", *Expert Systems with Applications,* Vol. 152, p. 113377, 2020.

[31] 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, p. 103541, 2020, doi: 10.1016/j.engappai.2020.103541.

[32] L. Abualigah, M. Abd 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, p. 116158, 2022.

[33] 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, p. 107408, 2021.

[34] 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,* p. 108457, 2022.