Testing Related to Production Technologies

Sadiq M. Sait, Pranav Mehta, Betül Sultan Yıldız and Ali Rıza Yıldız*

Structural optimization of aircraft components using chaotic pangolin optimizer

https://doi.org/10.1515/mt-2025-0343
Published online February 20, 2026

Abstract: This study introduces an enhanced chaotic pangolin optimizer (MCPO), a nature-inspired metaheuristic algorithm integrated with chaotic maps to solve complex constrained and multimodal engineering design problems. The MCPO simulates the luring and predation behavior of the pangolin in a two-stage framework, enhanced with chaos-based dynamics to improve the balance between exploration and exploitation. Aircraft components and systems are designed and optimized to be lightweight and high-load-carrying. Accordingly, the aircraft component attached to the wing is optimized for stress-carrying capacity and weight. The stress distribution and volume reduction are studied to maintain the system's overall dynamics and balance. The MCPO achieved effective results in terms of the component's overall weight and cumulative stresses. The MCPO realized a 1997 g weight of the component, which was 8.6 %, 8.5 %, and 7.8 % lower than the Ship rescue algorithm, Hiking algorithm, and Starfish algorithm, respectively. Furthermore, the weight of 585 g (from 2,582 g to 1,997 g) is reduced compared to the initial design. These results affirm the algorithm's robustness, precision, and versatility across diverse real-world design domains.

Keywords: nature-inspired algorithms; engineering design; aircraft components

1 Introduction

Optimization approaches have consistently been a focal point of research due to the ongoing need for enhancement and the management of complex constraints across many transdisciplinary designs, models, and engineering components. Classical optimization approaches are employed to address diverse design issues and achieve global optimum solutions. Nonetheless, other drawbacks, including convergence issues, computing efficiency, and the management of crucial constrained problems, are evident in these techniques. Advancements in the field of optimization research have been achieved through the development of metaheuristic algorithms, which serve as effective instruments for addressing intricate optimization challenges, especially in cases where conventional deterministic approaches prove inadequate due to high dimensionality, nonlinearity, or the multimodal characteristics of the problem space. Furthermore, metaheuristics (MHs) are overarching methods that direct the search process to efficiently navigate the solution space. These algorithms are distinguished by their adaptability and capacity to evade local optima, rendering them appropriate for many applications, such as scheduling, routing, and resource allocation [1]–[10]. These algorithms emulate natural evolutionary processes or social behaviors to progressively enhance a population of candidate solutions. Moreover, Swarm Intelligence (SI) methodologies, such as Ant Colony Optimization and Artificial Bee Colony, are derived from collective behaviors found in nature [11], [12]. Hybrid methodologies, integrating components from various metaheuristics, have been investigated to capitalize on the advantages of many techniques while alleviating their respective shortcomings.

Although metaheuristic algorithms are commonly utilized, they frequently necessitate meticulous parameter tuning and adaption to particular problem domains, which can be both labor-intensive and complex. As a result, research in this domain persists in advancing more resilient, efficient, and adaptable metaheuristic procedures that can be generalized across diverse applications. In numerous real-world scenarios, optimization encompasses various conflicting aims that necessitate simultaneous enhancement rather than being restricted to a singular goal. Multi-objective optimization (MOO) resolves this by pursuing a collection of trade-off solutions, referred to as the Pareto front, instead of a singular optimal answer. Recent

***Corresponding author: Ali Rıza Yıldız**, Department of Mechanical Engineering, Bursa Uludag University, Bursa, Türkiye,
E-mail: aliriza@uludag.edu.tr
Sadiq M. Sait, King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia
Pranav Mehta, Department of Mechanical Engineering, Dharmsinh Desai University, Nadiad, India
Betül Sultan Yıldız, Department of Mechanical Engineering, Bursa Uludag University, Bursa, Türkiye

Open Access. © 2026 the author(s), published by De Gruyter. This work is licensed under the Creative Commons Attribution 4.0 International License.

improvements in multiobjective optimization have included machine learning approaches to improve the efficacy of metaheuristic algorithms [2]–[4]. Surrogate models and reinforcement learning have been utilized to enhance the search process's efficacy. Furthermore, many-objective optimization, a subset of multiobjective optimization that involves more than three objectives, has introduced further issues that have been tackled by algorithms such as NSGA-III and MOEA/D [13], [14].

Contemporary optimization challenges – particularly in engineering design, machine learning model calibration, supply chain management, and renewable energy system planning – are progressively intricate and multifunctional. These issues often encompass nonlinear objective functions, a combination of variable types (continuous, discrete, binary), and several restrictions. Contemporary metaheuristics are explicitly formulated to address these complications with minimum domain-specific adjustments. Furthermore, the advancement of hybrid algorithms, which amalgamate the advantages of several metaheuristics or incorporate machine learning techniques, has demonstrated potential in enhancing solution quality and convergence rate. Hybrid or ensemble approaches frequently employ strategies such as local refinement, chaos theory, or fuzzy logic to enhance the efficacy of the search process.

Recent research has highlighted the significance of equitable comparative assessment, considerations of the no-free-lunch theorem, and the creation of standardized test suites for benchmarking to address these concerns. Moreover, metaheuristics are being customized for multiobjective optimization, dynamic optimization, and constrained optimization, thereby expanding their application in real-time and uncertain contexts [14]–[33].

In the present study, a recently developed pangolin optimizer has been investigated, and an improved version using chaotic maps has been developed to assess the algorithm's performance. Chaotic maps can improve the optimizer's performance metrics, particularly the exploration–exploitation balance, and the quality of solutions to constraint problems. Moreover, the modified version represents one of the early applications of chaotic enhancement within the pangolin optimizer framework for mechanical and industrial design problems.

Chaotic maps are used to replace uniform random numbers to generate deterministic yet ergodic and non-repetitive search trajectories, thereby significantly enhancing population diversity. This structured randomness helps prevent premature convergence and improves exploration–exploitation balance, particularly in constrained and multimodal engineering problems. Importantly, the chaotic modification enhances convergence robustness without introducing additional control parameters or altering the original mathematical structure of the pangolin optimizer.

2 Brief about pangolin optimization algorithm

The proposed algorithm is inspired by the hunting and luring behavior of the unique pangolin species. The algorithm is specifically inspired by the accurate sense of smell and the precise hunting behavior. The first luring behavior is associated with the identification of the prey and approaching it. While the second stage considered exploring, identifying, fast approaching, and digging to feed. All the subsections of the optimizer are mathematically structured in the present section. The initial population is generated by considering the random number ranging in the interval of 0–1, including lower and upper bound values. The same can be obtained by Equation (1) [21].

$$X_{\mathrm{ij}} = \mathrm{rand} \times \left(\mathrm{UB}_j - \mathrm{LB}_j\right) + \mathrm{LB}_{\mathrm{j}}, i = 1, 2, \ldots, n; j = 1, 2, \ldots., d \tag{1}$$

The luring and predation behaviors are simultaneously activated when the pangolin identifies surrounding ants, potentially through olfactory cues. Moreover, according to the location of the food, it decides to escalate the luring or predation behavior, which can be given by Equation (2) [21].

$$\begin{cases} C_M > 0.2 \text{ and } r_1 \le 0.5, \text{Luring behavior} \\ C_M < 0.7 \text{ and } r_1 > 0.5, \text{Predation behavior} \end{cases} \tag{2}$$

with C_M: aroma concentration in the hunting area of pangolin by ants and r_1: random number ranging in the interval of 0–1.

During the luring behavior, the pangolin attracts the nearby ants by releasing a strong aroma that lures them to the species, as if it were dead. The main strategy is known as the playing dead and attracting the ant as prey. The same can be said for Equations (3) and (4), respectively [21].

$$D_A(t) = |aX_A(t) - X_M(t)| \tag{3}$$

$$X_A(t+1) = X(t) + X_A(t) - A_1 D_A(t) \tag{4}$$

with $X_M(t)$: location of the pangolin at t-th iterations, $X_A(t)$: location of the ant at t-th iteration, and $D_A(t)$: relative distance between the pangolin and ant.

At the same stage, the pangolin may sometimes go over the nearby water reserve or river, which may be the optimum location to capture the maximum number of ants. Accordingly, once it captures the ant, it moves toward the

nearby water reserve and captures the remaining ants by a special smell. This is the subcase where the pangolin and ant are in the same place. The same can be modeled by using Equations (5) and (6), respectively [2].

$$D_M(t) = |C_1 D_A(t) - X_M(t)| + L_{\text{levy}}\left(1 - \frac{t}{T}\right) \tag{5}$$

$$X_M(t+1) = X(t) + X_M(t) - A_1 D_M(t) \tag{6}$$

with C_1: factor that indicates the responsiveness of the pangolin to move toward the river and $X_M(t)$: location of the species at the t-th iteration.

Accordingly, the mathematical modeling of the predation behavior is subdivided into three parts: exploration stage, quick approach toward the prey stage, and digging and feeding stage. In the first phase, the species identify the best possible space to attain the maximum amount of food, which can be given by Equations (7) and (8) [21].

$$D_M(t) = \left|L_{\text{levy}} X_M(t) - X(t)\right| \tag{7}$$

$$X_M(t+1) = \sin\left(C_1 X(t) + A_1\left(X_M(t) - L_{\text{levy}} D_M(t)\right)\right) \tag{8}$$

The case approaches the prey once realized during the previous stage by the strong instinct and smell. Hence, the same can be obtained by Equations (9) and (10) [21].

$$D_M(t) = |a X_M(t) - X(t)| \tag{9}$$

$$X_M(t+1) = X(t) - A_1\left(X_M(t) - \exp(-a).\ \sin(\text{rand}.\pi) D_M(t)\right) \tag{10}$$

In the last stage, the pangolin potentially identifies the ant's nest, and then it technically digs the entire location with precision and feeds itself. The same can be related to realizing the global optimum solution of the problems. Mathematical models are given by Equations (11) and (12).

$$D_M(t) = |C_1 X_M(t) - X(t)| \tag{11}$$

$$X(t+1) = X(t) + A_1\left(X_M(t) - D_M(t)\right) \tag{12}$$

Accordingly, during the predation behavior, the position updating can be obtained by Equation (13) [28].

$$X(t+1) = C_1 X_M(t) \tag{13}$$

The overall computational complexity should be given by Equation (14) [21].

$$O(\text{Pangolin Optimizer}) = O(T \times n \times (C + d)) \tag{14}$$

Figure 1 shows the insights into the incorporation of the chaotic maps with the algorithm.

The overall computational steps of the proposed modified chaotic pangolin optimizer (MCPO) are summarized in the following pseudo-code.

Pseudo-code of Modified Chaotic Pangolin Optimizer (MCPO)

```
Input:
 f(x)      objective function (minimization)
 LB, UB    lower/upper bounds of d design variables
 N         population size
 T         maximum iterations
 g(·)      chaotic map producing values in (0,1) (e.g., Logistic/Tent)
 c0        initial chaotic seed in (0,1)
Output:
xbest      best solution found
fbest      best objective value

c ← c0
// Chaotic initialization
for i = 1 to N do
   for j = 1 to d do
      c ← g(c)                    // chaos ∈ (0,1)
      X[i,j] ← LB[j] + c·(UB[j]−LB[j])
   end for
   Fi ← f(X[i])
end for
(xbest, fbest) ← argmin(Fi)
for t = 1 to T do
   for i = 1 to N do
      // Generate chaotic random numbers whenever required
      c ← g(c); r ← c
      // Compute aroma concentration / trigger terms as in original PO/CPO
      // (Use your same formulas; replace each random number by chaos)
      CM ← AromaModel(X[i], r, t)      // corresponds to your Eq.(2) logic
      // Choose behavior (same decision rule as original PO/CPO)
      if LuringCondition(CM, r, t) then
         // -------- LURING BEHAVIOR (use your Eq.(3)–(6)) --------
         // For every random number rk in Eq.(3)–(6):
         //   c ← g(c); rk ← c
         X[i] ← LuringUpdate(X[i], xbest, {rk}, t)   // Eq.(3)–(6)
      else
         // -------- PREDATION BEHAVIOR (use your Eq.(7)–(13)) ----
         // For every random number rk in Eq.(7)–(13):
         //   c ← g(c); rk ← c
         X[i] ← PredationUpdate(X[i], xbest, {rk}, t) // Eq.(7)–(13)
      end if
      // Boundary handling
      for j = 1 to d do
         if X[i,j] < LB[j] then X[i,j] ← LB[j] end if
         if X[i,j] > UB[j] then X[i,j] ← UB[j] end if
      end for
      // Fitness evaluation (and constraint handling if applicable)
      Fi ← f(X[i]) + Penalty(X[i])     // Penalty=0 if feasible (optional)
      // Update global best
      if Fi < fbest then
         xbest ← X[i]
         fbest ← Fi
      end if
   end for
end for
return xbest, fbest
```

3 Applications of the modified pangolin optimizer for global optimization of aircraft components

In this section, the modified algorithm is tested to solve a unique problem of the aircraft component used in industry.

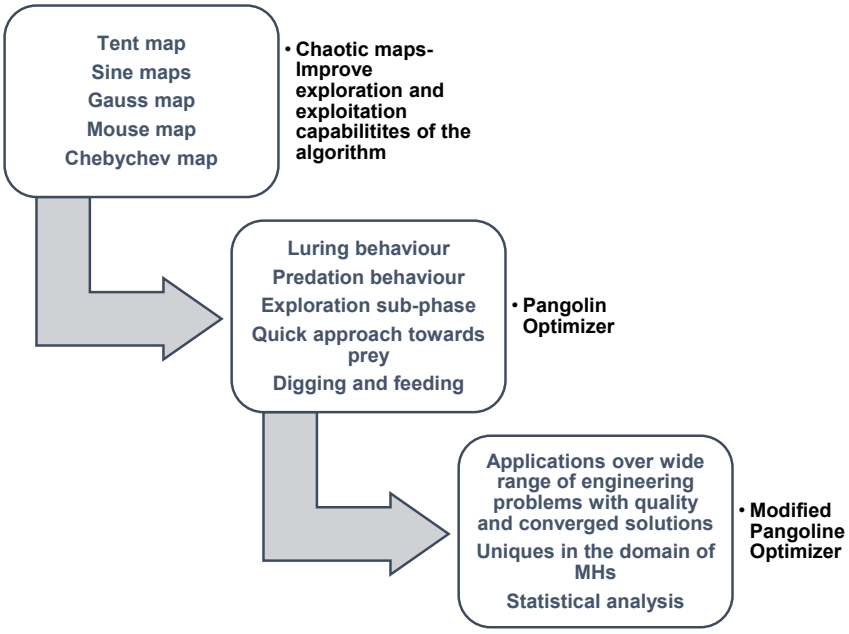


Figure 1: Modified pangolin optimizer development stages.

The results are statistically compared with published literature results to assess the performance of the modified pangolin optimizer (MCPO). Many mechanical and structural parts make up an aircraft system, and they all work together to keep the aircraft stable, efficient, and safe. These parts include main load-bearing structures, joints, couplings, and other supporting components that ensure loads are transferred and vibrations are damped throughout the system. The integration and performance of these components are essential in assessing the overall reliability and functionality of the aircraft or any vehicular system operating under dynamic conditions. This study focuses on optimizing a particular vehicle component, as depicted in Figure 2, with the main aim of minimizing its overall weight while preserving its structural integrity. Changing essential design parameters that affect the component's volume, shape, and material distribution is part of the optimization process. The study seeks to attain a balance between lightweight design and mechanical strength by methodically altering these decision factors.

In the field of vehicle dyna-aircraft design, all structural and functional components are regularly subjected to cyclic loading and changing stress conditions due to various operational forces. Over time, these cyclic stresses can cause fatigue failure, wear, and material degradation, making the entire system less safe and less efficient. To prevent failures caused by fatigue, designers usually make the part heavier or use conservative design margins. This makes the part more durable, but it also makes the system less fuel-efficient and less effective overall. To address these problems, this



Figure 2: Aircraft component at the initial stage [23].

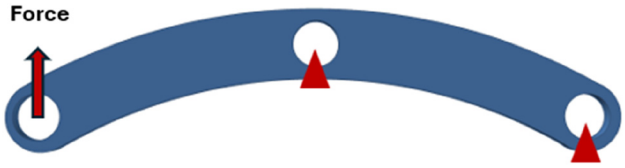


Figure 3: Boundary conditions.

work uses a systematic optimization method that focuses on weight reduction as the primary goal (fitness function), while maintaining maximum stress within safe limits. This ensures that the part not only gets lighter but also retains enough strength and stiffness to handle the loads it will encounter in the real world. The optimization framework uses advanced computational tools and numerical analysis methods to determine how changes in an object's shape affect its mass and stress distribution. The study shows that strategically reducing weight through design optimization can greatly improve the efficiency, performance, and sustainability of vehicle systems. This approach helps reduce the amount of materials and fuel used and improve a product's overall performance over its lifetime, all of which are important factors in modern aerospace engineering.

The boundary conditions of the part are shown in Figure 3. Consequently, topology optimization was conducted before the shape design optimization phase to accomplish the optimal solution. The objective of the topology optimization step is to identify the resulting material distribution that minimizes compliance while adhering to volume constraints. Outcomes of topology optimization are shown in Figure 4.

The geometry is then refined by incorporating manufacturing-related constraints derived from the topology optimization results, leading to the revised component configuration and the corresponding stress distribution illustrated in Figure 5. This study uses the MCPO algorithm to optimize the design of the aircraft bracket. The variable ranges shown in Figure 6 are confined to the specified limitations of 16–39, 17–41, and 18–42, respectively.

Figure 7 depicts the optimal design obtained using the MCPO together with the corresponding stress distribution.

Table 1 compares the MCPO results, the initial design, and three benchmark metaheuristic algorithms. Accordingly, the

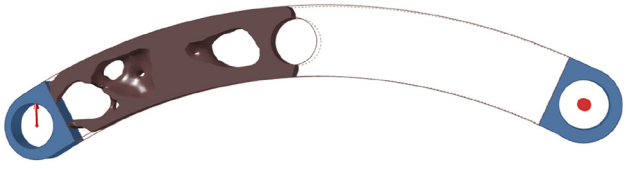


Figure 4: Material distribution based on topology optimization results.

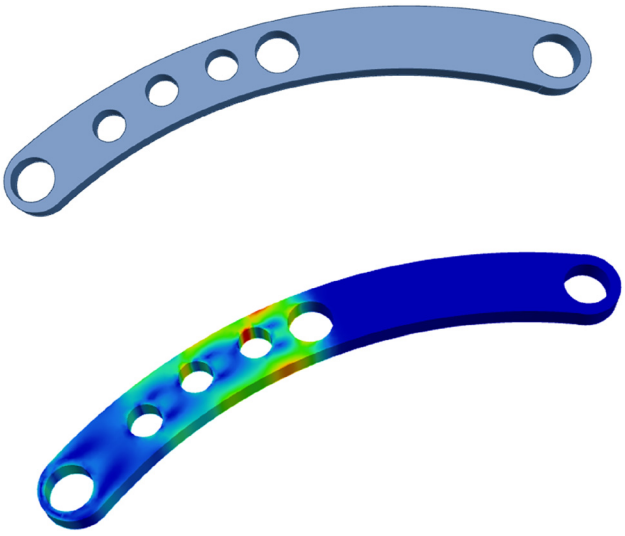


Figure 5: Structurally optimized component and stress distribution.

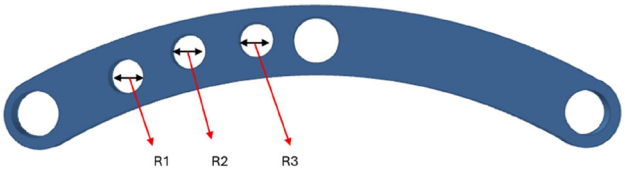


Figure 6: Optimized design variables after structural optimization.

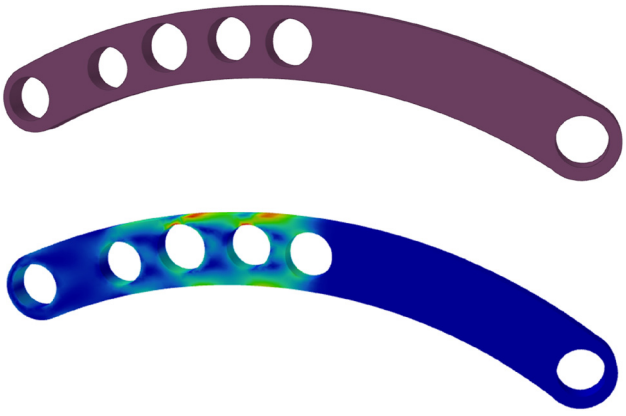


Figure 7: Final optimum design with a weight of 1,997 g.

Table 1: Proposed design of aircraft component by MCPO".

Optimizer and literature results	Fitness function (g)	Constraint	Mean	Worst	Std
Provisional design	2,582	65	–	–	–
Proposed design	2,242	82	–	–	–
Ship rescue optimizer	2,169	86	2,189	2,193	9.6
Hiking optimizer	2,168	88	2,178	2,181	8.2
Starfish optimizer	2,153	89	2,164	2,169	5.6
MCPO	1,997	95	2003	2008	2.3

MCPO achieves a final component weight of 1,997 g under the specified constraints. Constraint values denote the maximum allowable normalized stress level satisfied by each optimized design.

4 Conclusions

The integration of chaotic maps into the pangolin optimizer framework significantly enhances its global search ability and convergence characteristics. The modified chaotic pangolin optimizer (MCPO) outperformed the selected benchmark metaheuristic algorithms considered in this study in terms of accuracy, convergence reliability, and success rate.

These quantitative improvements validate MCPO as a robust and adaptable metaheuristic tool suitable for solving highly nonlinear, constrained, and high-dimensional engineering problems. Its consistent superiority and reliability position it as a strong candidate for future use in structural, mechanical, and multidisciplinary optimization scenarios.

Research ethics: Not applicable.
Informed consent: Not applicable.
Author contributions: The author has accepted responsibility for the entire content of this manuscript and approved its submission.
Use of Large Language Models, AI and Machine Learning Tools: None declared.
Conflict of interest: The author states no conflict of interest.
Research funding: None declared.
Data availability: Not applicable.

References

[1] M. Abdel-Basset, L. Abdel-Fatah, and A. K. Sangaiah, "Metaheuristic algorithms: A comprehensive review," *Comput. Intell. Multimedia Big Data Cloud Eng. Appl.*, pp. 185–231, 2018, https://doi.org/10.1016/B978-0-12-813314-9.00010-4.

[2] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, 2002, https://doi.org/10.1109/4235.996017.

[3] C. A. C. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Trans. Evol. Comput.*, 2004, https://doi.org/10.1109/TEVC.2004.826067.

[4] Q. Zhang and H. Li, "MOEA/D: A multi-objective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, 2007.

[5] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, 1994, https://doi.org/10.1162/evco.1994.2.3.221.

[6] Q. Zhang and A. Zhou, "A review on multiobjective evolutionary algorithms for many-objective optimization problems," *IEEE Trans. Evol. Comput.*, 2010.

[7] C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, India, Springer, 2007.

[8] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, 2014, https://doi.org/10.1109/TEVC.2013.2281535.

[9] J. Blank and K. Deb, "PYMOO: Multi-objective optimization in python," *IEEE Access*, 2020, https://doi.org/10.1109/ACCESS.2020.2990567.

[10] R. Wang, Y. Jin, and T. Jansen, "Data-driven evolutionary multi-objective optimization," *IEEE Trans. Evol. Comput.*, 2022.

[11] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Comput. Intell. Mag.*, vol. 1, no. 4, pp. 28–39, 2006, https://doi.org/10.1109/MCI.2006.329691.

[12] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Global Optim.*, vol. 39, pp. 459–471, 2007, https://doi.org/10.1007/s10898-007-9149-x.

[13] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, "Performance comparison of NSGA-II and NSGA-III on various many-objective test problems," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Vancouver, BC, Canada, 2016, pp. 3045–3052.

[14] P. Yan, J. Zhang, and T. Zhang, "Nature-inspired approach: A novel rat optimization algorithm for global optimization," *Biomimetics*, vol. 9, no. 12, p. 732, 2024, https://doi.org/10.3390/biomimetics9120732.

[15] M. Azizi, S. Talatahari, and A. H. Gandomi, "Fire hawk optimizer: A novel metaheuristic algorithm," *Artif. Intell. Rev.*, vol. 56, no. 1, pp. 287–363, 2023, https://doi.org/10.1007/s10462-022-10173-w.

[16] O. N. Oyelade, A. E. S. Ezugwu, T. I. A. Mohamed, and L. Abualigah, "Ebola optimization search algorithm: A new nature-inspired metaheuristic optimization algorithm," *IEEE Access*, vol. 10, pp. 16150–16177, 2022, https://doi.org/10.1109/ACCESS.2022.3147821.

[17] J. S. Pan, L. G. Zhang, R. B. Wang, V. Snášel, and S. C. Chu, "Gannet optimization algorithm: A new metaheuristic algorithm for solving engineering optimization problems," *Math. Comput. Simul.*, vol. 202, pp. 343–373, 2022, https://doi.org/10.1016/j.matcom.2022.06.007.

[18] M. Ghasemi *et al.*, "Flood algorithm (FLA): An efficient inspired meta-heuristic for engineering optimization," *J. Supercomput.*, vol. 80, no. 15, pp. 22913–23017, 2024, https://doi.org/10.1007/s11227-024-06291-7.

[19] S. Mirjalili and J. S. Dong, "Multi-objective particle swarm optimization," in *Multi-Objective Optimization using Artificial Intelligence Techniques*, Cham, Switzerland, 2020, pp. 21–36.

[20] A. Ghiaskar, A. Amiri, and S. Mirjalili, "Polar fox optimization algorithm: A novel meta-heuristic algorithm," *Neural Comput. Appl.*, pp. 1–40, 2024, https://doi.org/10.1007/s00521-024-10346-4.

[21] Z. Guo, G. Liu, and F. Jiang, "Chinese pangolin optimizer: A novel bio-inspired metaheuristic for solving optimization problems," *J. Supercomput*, vol. 81, no. 4, p. 517, 2025, https://doi.org/10.1007/s11227-025-07004-4.

[22] A. Hammoudi, S. M. Sait, and A. R. Yildiz, "Optimal design of planetary gear train for automotive transmissions using advanced meta-heuristics," *Int. J. Veh. Des.*, vol. 80, no. 2–4, pp. 121–136, 2020. https://doi.org/10.1504/IJVD.2019.109862.

[23] A. Ceruti, P. Marzocca, A. Liverani, and C. Bil, "Maintenance in aeronautics in an industry 4.0 context: The role of augmented reality and additive manufacturing," *J. Comput. Des. Eng.*, vol. 6, no. 4, pp. 516–526, 2019, https://doi.org/10.1016/j.jcde.2019.02.001.

[24] Z. K. Kocabıçak and Z. Acar, "Optimal hydraulic engine mount parameters using design of experiment (DoE) and response surface methodology," *Mater. Test.*, vol. 67, no. 4, pp. 728–736, 2025, https://doi.org/10.1515/mt-2024-0422.

[25] M. C. Tezel, N. Erdoğan, and E. Acar, "Design optimization of a multi-layer aircraft canopy transparency plate against bird strike," *Mater. Test.*, vol. 67, no. 9, pp. 1507–1519, 2025, https://doi.org/10.1515/mt-2025-0240.
[26] A. R. Yildiz, "Optimization of multi-pass turning operations using hybrid teaching learning-based approach," *Int. J. Adv. Manuf. Technol.*, vol. 66, no. 9–12, pp. 1319–1326, 2013. https://doi.org/10.1007/s00170-012-4410-y.
[27] A. Hammoudi, A. R. Yildiz, and S. M. Sait, "Mechanical engineering design optimisation using novel adaptive differential evolution algorithm," *Mater. Test.*, vol. 80, no. 2–4, pp. 285–329, 2019. https://doi.org/10.1504/IJVD.2019.109873.
[28] D. Gürses, P. Mehta, S. M. Sait, and A. R. Yildiz, "African vultures optimization algorithm for optimization of shell and tube heat exchangers," *Mater. Test.*, vol. 64, no. 8, pp. 1234–1241, 2022. https://doi.org/10.1515/mt-2022-0050.
[29] B. Ocak, Z. Evis, and F. Ozturk, "Optimization of spur gear pairs for aerospace applications," *Mater. Test.*, vol. 64, no. 11, pp. 1687–1698, 2022, https://doi.org/10.1515/mt-2022-0062.
[30] S. P. Dwivedi, N. K. Maurya, M. Maurya, A. Saxena, and A. K. Srivastava, "Optimization of casting parameters for improved mechanical properties of eggshell reinforced composites," *Mater. Test.*, vol. 63, no. 11, pp. 1041–1051, 2021, https://doi.org/10.1515/mt-2021-0044.
[31] İ. Gökdağ and E. Acar, "Application of a modular topology optimization framework to an aerospace bracket design," *Mater. Test.*, vol. 64, no. 7, pp. 1090–1102, 2022, https://doi.org/10.1515/mt-2021-2148.
[32] İ. Öztürk, "Design optimization of hybrid material B-pillar under crush loading," *Mater. Test.*, vol. 66, no. 1, pp. 36–42, 2024, https://doi.org/10.1515/mt-2023-0069.
[33] D. Gürses, "Optimum design of electric vehicle battery enclosure using the chaotic metaheuristic algorithms," *Mater. Test.*, vol. 67, no. 9, pp. 1537–1544, 2025, https://doi.org/10.1515/mt-2025-0291.

The authors of this contribution

Sadiq M. Sait
Dr. Sadiq M. Sait received his Bachelor's degree in Electronics Engineering from Bangalore University, India, in 1981, and his Master's and PhD degrees in Electrical Engineering from the King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, in 1983 and 1987, respectively. He is currently a Professor of Computer Engineering and Director of the Center for Communications and IT Research, KFUPM, Dhahran, Saudi Arabia.

Pranav Mehta
Dr. Pranav Mehta is an Assistant Professor at the Department of Mechanical Engineering, Dharmsinh Desai University, Nadiad-387,001, Gujarat, India. His major research interests include metaheuristics techniques, multiobjective optimization, solar-thermal technologies, and renewable energy.

Betül Sultan Yıldız
Dr. Betül Sultan Yıldız completed her BSc and MSc degrees at Uludağ University, Bursa, Turkey, and received her PhD in Mechanical Engineering from Bursa Technical University, Turkey. Her research interests are optimal design, shape optimization, topology optimization, topography optimization, structural optimization methods, and metaheuristic optimization algorithms, as well as applications to industrial problems.

Ali Rıza Yıldız
Dr. Ali Rıza Yıldız is a Professor and Dean of the College of Engineering, Bursa Uludağ University, Bursa, Turkey. His research interests are the finite element analysis of structural components, lightweight design, vehicle design, vehicle crashworthiness, shape and topology optimization of vehicle components, metaheuristic optimization techniques, and additive manufacturing.