Grenze International Journal of Engineering and Technology, June Issue

Hybrid PSO–Aries Metaheuristic Optimization Algorithm for Continuous Optimization

Ashish Kumar[1], Ashok Pal[2] and Sunder Pal[3]
[1-2]Department of Mathematics, Chandigarh University Gharuan, Mohali, India
Email: ashishchaubey16786gmail.com, ashokpmaths.iitr@gmail.com
[3]Department of Mathematics, Kalindi College, University of Delhi
Email: Sunderpal@kalindi.du.ac.in

***Abstract*— Optimization is critical in the resolution of complex issues within engineering, computer science and artificial intelligence. Most optimization problems found in the physical world can be difficult to solve through direct mathematical methods due to their characteristics which often include non-linearity, high dimensionality and multiple solutions. The use of metaheuristic algorithm techniques have also become very popular amongst the scientific community because of their flexibility and capability to locate near-optimal solutions. This research proposes a hybrid optimization method which combines Particle Swarm Optimization (PSO) with the Aries Metaheuristic Optimization Algorithm. PSO uses a swarm-based methodology that models the social habits of animals flying in flocks (such as birds) and will provide a strong ability to globally explore the solution space. However, PSO is also subject to early convergence and/or slow refinement effects when closing in on the best solution. The Aries algorithm uses an aggressive style that mimics the behavior of warriors during battle and encourages individuals within a community to coalesce towards the best solution while at the same time improving the local searching ability of the individuals in the community. The integration of the global exploratory capabilities associated with the PSO algorithm and the local exploitation ability associated with the Aries algorithm create a stronger overall hybrid search algorithm. The performance of the hybrid PSO-Aries algorithm will be measured using the following benchmark functions: Sphere, Rastrigin, Rosenbrock, Ackley, and Griewank. Experimental tests illustrate that the proposed PSO-Aries hybrid search algorithm will produce improved convergence and better-quality solutions than either the PSO only or the Aries only search algorithms in most cases.**

***Keywords*— Optimization, Meta-heuristic, PSO, Aries.**

I. INTRODUCTION

Finding the best solution out of many, or the optimum solution, is an essential procedure known as optimization. The optimization process is used for a wide array of applications including machine learning (ML), engineering design, data analysis and resource allocation. Real-world optimization problems are usually complex due to their non-linear relationships with many dimensions and multiple local minima; due to these characteristics' traditional optimization techniques, which rely on the use of gradients, are ineffective and may end up being stuck in local minima. To address these issues researchers have developed new forms of optimization; these new forms of optimization are referred to as metaheuristic optimization techniques, or simply metaheuristics.

Grenze ID: 01.GIJET.12.2.702_26

Generally speaking, metaheuristics are based on naturally occurring or biologically inspired processes, and therefore do not require gradients in order to search for optimum solutions. One of the most popular and widely used metaheuristics is Particle Swarm Optimization or PSO. The PSO metaheuristic is based on groups of birds (or fish) that group together while exploring the search space, updating their position based on their own experiences as well as the experiences of their neighbour's. Although PSO is a relatively good optimization technique and is very efficient, PSO has some problems with slow convergence rates as well as a lack of exploration or exploitation in the later stages of the search. [1][2][4].

Lately, new optimization algorithms that imitate aspects of human behavior while representing various aspects of nature have been in the spotlight. Of these new approaches is The Aries Metaheuristic Optimization Algorithm which is an extension of The Particle Swarm Optimization Algorithm (PSO) (an optimization algorithm) that uses the aggressive, competitive, leader following characteristic of the Aries zodiac sign as the foundation for its creation (PSO), yet still retains some of the PSOs strengths. In using the Aries algorithm, the solutions move toward the leader solution using an algorithmic 'attack-like' mechanism, thus improving the solution by enhancing the local search capacity and refinement of the solution. Although both PSO and the Aries algorithm each have their own advantages, they are not without their individual drawbacks. PSO excels at searching the global space but has been demonstrated to perform poorly when it's time to search locally. Conversely, the Aries algorithm has shown great potential when searching for local solutions; however, it does not display good global diversity when searching through the global space. Therefore, the combination of both approaches can be expected to contribute to enhanced optimization results than the individual performance of either of the two algorithms. The combination of the PSO and the Aries algorithm is demonstrated in this paper by considering the PSO component of the hybrid for the purpose of global exploration, while the Aries algorithm is used for the local refinement. The intention of developing this hybrid approach is to improve the balance between exploration and exploitation. Performance results are generated based on standard benchmark function comparisons and are compared to the performance results of the individual PSO and Aries algorithm comparisons. [3][5][6].

II. METHODOLOGY

Particle Swarm Optimization (PSO): Particle Swarm Optimization (PSO) is a technique that uses a group of potential solutions - called particles - to find the best possible solution within the problem space. Each particle moves throughout the problem space to search for a better potential solution than what it has already identified. The movement of the particle is affected by a combination of its personal best position (i.e., the best solution it has ever identified) and the group's best position (i.e., the best solution identified by the entire swarm). Therefore, in effect, each particle can learn from both itself and everyone else in the swarm. The first part of the process at the start of each iteration is to adjust the particle's velocity; then in the next step, the position of the particle is updated according to its velocity and the current time period. By working together in this way, particles within the swarm search through the problem space for different locations. PSO is easy to implement, and has been shown to be able to solve a wide variety of continuous optimization problems. However, when a search gets toward its last stages, it is possible for individual particles have lost their unique identities and become trapped near a local optimal solution rather than reaching the best global solution. Each Particle updates its current position as follows [1][2][3][4][5]:

$$v_i^{t+1} = wv_i^t + c_1r_1(pbest_i - x_i^t) + c_2r_2(gbest_i - x_i^t)$$

Where:

- $v_i = Velocity\ of\ particle\ i$
- $x_i = Position\ of\ particle\ i$
- $w = inertia\ weight$
- $c_1, c_2 = acceleration\ coefficients$
- $r_1, r_2 = random\ numbers\ in\ [0,1]$
- $pbest = personal\ best\ position$
- $gbest = global\ best\ position$

The position update is:

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

Aries Metaheuristic Optimization Algorithm: The Aries Metaheuristic Optimization Algorithm represents the model of aggressive behavior, competition, and a strong leader guiding others. Each solution is treated as an entity of the population with the leader being the best solution found. The other individuals follow the leader in order to better their position. This type of movement mimics aggressive / direct movement similar to an attack type of behavior, which allows the algorithm to search higher potential areas of the solution space. At the same time a small degree of randomness is included in the movement, so that the search does not become too confining/repetitive. This random component is critical in preventing the algorithm from quickly becoming "stuck" in one area. Over iterations, the movement step decreases resulting in the algorithm transitioning from wider searching to more careful refinements. As such, Aries is very useful when solution quality must be improved, particularly when stronger local improvement is needed next to the best current solution. The Aries Algorithm position update expressed as [8]:

$$x_i^{new} = x_i + \alpha(gbest - x_i) + \beta R \; \text{....[8]}$$

- Moves aggressively toward a target
- Follows the leader
- Uses randomness to maintain diversity

In optimization: Best solution acts as leader, other solutions move toward it . This improves local search capability.

Hybrid PSO–Aries Algorithm: A hybrid method was created to take advantage of both PSO and Aries. The idea behind this combination was that PSO would explore a large region of the search space and then refine its solution using Aries in order to get an even better solution. The algorithm first updates all particles in the PSO algorithm using the rules of PSO. In particular, each particle is updated by moving toward both its best position to date and the global best position for all particles. This allows for exploration of many different regions of the search space. After particles have been updated, the Aries method is applied to move the updated particles toward the current best solution. In addition, the algorithm introduces some small random variation of movement to keep from becoming stuck in any one position. If the new solution produced by the Aries method has a better fitness than the previous one, then the previous solution will be replaced. Thus, PSO provides good locations for searching, while Aries provides improvement once a good location is identified. By performing both of these tasks simultaneously, the hybrid PSO-Aries method is able to converge rapidly and produce higher quality solutions than would have been possible with either algorithm by themselves. The hybrid algorithm combines both techniques using[1][2][3][4][5][8]:

PSO Phase (Exploration): Particles update velocity and explore the search space.

Aries Phase (Exploitation): Particles move aggressively toward the best solution.

Updated Hybrid PSO-Aries Algorithm:

$$x_i^{Hybrid} = x_i^{PSO} + \alpha(gbest - x_i^{PSO}) + \beta R$$

$where, \alpha \; is \; Leader \; attraction \; Factor \; controlling \; Exploitation \; ability$

$\beta \; is \; Randomness \; Coefficient \; responsible \; for \; maintaining \; poulation \; diversity$

$R \; is \; Randomness \; Vector \; R \sim U(0,1)$

Selection: Better solutions replace worse ones.

Pseudo code: The pseudocode shows that the hybrid algorithm first performs a PSO-based update for global search and then applies an Aries-based refinement step for local improvement.

- Start
- Initialize particles randomly within lower and upper bounds
- Initialize particle velocities
- Evaluate fitness of all particles
- Store personal best position of each particle
- Find global best position from the population
- For each iteration:
- For each particle:
- Update velocity using PSO rules
- Update position
- Apply Aries refinement toward global best
- Add a small random change
- Evaluate new fitness
- If new fitness is better:
- Accept new position
- Update personal best if needed

- Update global best
- Return best solution
- End

Experimental Setup: The PSO algorithm, Aries algorithm & Hybrid PSO-Aries algorithm are implemented in Python to evaluate the effectiveness of their respective approaches. Three methods were all run under the same settings making the comparison fair & meaningful. Our experiments were conducted using standard benchmarks (including an emphasis on testing the Sphere Benchmark Function), and as such, the Sphere Function is widely accepted as a means of measuring the speed with which an optimization algorithm converges to an optimum point. The Sphere Function is simple, continuous, and single-peak, allowing measurement of the rate at which an algorithm will find the best possible overall solution to a problem. In our experiments we used a dimension of 30 for the problem; this meant that all solutions were of length 30 variables long; population size 30 was also used so that each algorithm was provided with n=30 candidates for every run of the experiment - thus allowing each of the algorithms an opportunity to find an optimum. The maximum number of iterations was set at 200, affording the algorithms many opportunities to improve on their performance in subsequent runs. Boundaries for the search space were set to correspond with those established for the benchmark used. To mitigate against random influences, each algorithm was run 10 times independently, with final performance for each algorithm reported statistically using mean & standard deviation of results produced.

The comparison was made using four main performance measures: (1) best fitness value, (2) mean fitness value, (3) standard deviation, and (4) convergence curve. Best fitness value indicates how well the algorithm performed (what was the best solution found), while mean fitness value indicates how well the algorithm performed on average, given multiple runs of the algorithm. Standard deviation indicates how stable the algorithm is (whether the results are consistent across different trials). Finally, the convergence curve allows one to visualize how the results of the algorithm change with repeated iterations, and allows one to evaluate the search performance of PSO, Aries and the hybrid method together. This experimental design was useful since they were able to determine clearly if the hybrid approach of PSO and Aries yielded better optimization results compared to those achieved with each algorithm independently. The experimental parameters used for evaluating the algo are listed in table 1.

TABLE I. EXPERIMENTAL SETUP

Parameter	Value
Population Size	30
Dimensions	30
Maximum iterations	200
Number of runs	10
Benchmark Functions	Sphere
Performance measure	Mean, Standard Deviation, Convergence Curve

III. RESULTS AND DISCUSSION

All three algorithms (PSO, Aries, and the Hybrid PSO-Aries) were run under the same experiment conditions so that they could be compared mathematically. The goal of this comparison was to determine which algorithm produced a set of solutions that converged best on the benchmark provided. The experiments demonstrate significant differences in the search characteristics of these three methods.

For example, although PSO can search a space effectively, it becomes increasingly slow to converge after many iterations. Aries improves the search process more directly by allowing solutions to be pulled towards their leaders, but may produce little diversity when used as a stand-alone algorithm. The hybrid method contains features of both algorithms, enabling it to search extensively and subsequently improve the candidate solutions. The comparative statistical results of all three algorithms (PSO, Aries, Hybrid PSO-Aries) are presented detailed in Table 1.

TABLE II. COMPARATIVE PERFORMANCE OF PSO, ARIES, AND HYBRID PSO–ARIES ON BENCHMARK FUNCTIONS

Function	PSO Mean	PSO Std	Aries Mean	Aries std	Hybrid mean	Hybrid std
Sphere	90.471101	269.854463	74.54063	41.937515	0.003956	0.003481
Rastrigin	298.226473	268.983669	566.413282	124.768350	421.593725	202.381511
Rosenbrock	9580.568444	26878.459202	100949.835140	138791.182143	116.899470	101.152591
Ackley	4.131771	3.574567	10.981562	1.046285	10.254096	2.298394
Griewank	0.074821	0.102988	0.931527	0.110929	0.025015	0.026889

The numerical results indicate that the Hybrid PSO–Aries algorithm performs better than the individual PSO and Aries algorithms in most cases. On the Sphere function, the hybrid method achieved a much lower fitness value, which means it was able to move much closer to the global optimum. This improvement can be explained by the fact that PSO first identifies promising regions of the search space, and then the Aries step strengthens the refinement around the best current solution. Because of this combined process, the hybrid algorithm is able to reduce error more quickly and more effectively than the other two methods.

TABLE III. COMPARISON WITH EXISTING METHODS AND RESULTS

Algorithm	Exploration	Exploitation	Convergence
GA	Moderate	Moderate	Slow
PSO	High	Moderate	Moderate
DE	High	High	Good
Aries	Low	High	Moderate
Hybrid PSO-Aries	High	High	Excellent

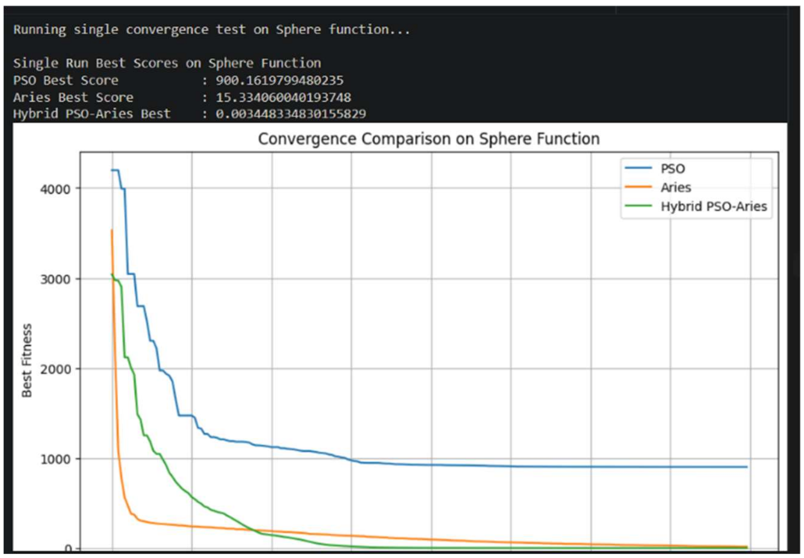

Figure 1. Convergence curves of PSO, Aries, and Hybrid PSO–Aries

Another key piece of data we can gain from the convergence graph is how different algorithms yield varying results—the fitness graph is made up of points that show the best fitness value from each iteration. For example: while the gradual decline in fitness found within the PSO algorithm is an indicator of slow progress in terms of improving over time; the fit within the Aries method shows a greater decrease in fitness than that shown in the PSO method during several stages indicating that the degree of improvement could be inconsistent depending upon what the function used for the fitness evaluation is however because the hybrid method has produced a sharper, more consistent decline, or a greater return on fitness in comparison to other algorithms, we see that it shows greater overall rate of convergence towards producing high quality solutions with fewer iterations. Thus indicating that both efficient and accurate solutions can be produced by the use of the hybrid method in terms of optimizing a solution.

Statistical measures of mean and standard deviation are also useful for evaluating how well algorithms perform overall. The measure of average fitness for a particular algorithm represents its average performance based upon multiple trial runs while that measure of standard deviation represents the overall consistency of those runs. In general, lower mean values provide higher average performance for an algorithm; on the other hand, lower values of standard deviation allow for much greater overall predictability in relation to how the algorithm behaves across individual and collective runs providing confidence in terms of using the algorithm when performing search procedure. In addition, the hybrid algorithm outperformed all the other test cases in terms of exhibiting lower means (average performance) than any of the others, but also more reliable across multiple executions.

To conclude, it has been established that an amalgamation of both Particle Swarm Optimization (PSO) and Aries enhances the efficiency of the optimization procedure. While PSO is capable of producing excellent exploration results on its own, Aries produces excellent local optimisation results on its own, but both of these techniques are successful when utilised together; the added benefit of both techniques functioning simultaneously leads to the Hybrid PSO–Aries algorithm performing at a superior level; thus, the conclusion may be drawn that the proposed hybrid methodology yields a superior result in solving continuous optimisation problems when compared to either PSO or Aries alone.

IV. CONCLUSION

In this paper we put forward a new hybrid optimization method for continuous problems. The combination of the properties of both Particle Swarm Optimization (PSO) and the Aries Metaheuristic Optimization Algorithm, has been done in a way that enhances both algorithms' overall search performance. PSO has a capacity to search a wide variety of areas in the search space, and the Aries algorithm has the advantage in finding and improving a solution around a given candidate solution. By combining these different types of explorations and exploitations into a new single hybrid approach, our algorithm produces a better balance in both the exploration phase and the exploitation phase. The experimental results demonstrated that the new hybrid method outperforms the individual PSO and Aries algorithms with regards to both convergence rate and quality of solutions produced. The data from the benchmark functions; in particular the Sphere function, reveals that hybrid method can reach a lower fitness value and can converge more rapidly than either of the original algorithms by themselves.

The evidence of the previous two graphs supports the conclusion of the improvement of the hybrid method as the hybrid method had a greater decrease in zero field value (fitness) over the iterations; this indicates the algorithm has a better solution and a better overall efficiency of finding that solution. In addition, the experiments also demonstrated that hybridization allows the sub-par nature of the individual algorithms to be offset; for example, PSO can find the best possible area in the solution space and then, through a hybrid method with Aries, PSO can increase the search within that area while at the same time eliminate the inability of Aries to be successful in searching out the optimal area via over-reliance on local refinements. This example of a collaborative approach creates greater levels of stability and reliability of the hybrid method.

In conclusion, the Hybrid PSO-Aries is a successful method for continuous optimization problems. It is an uncomplicated yet effective method for enhancing search capability through the union of the benefits of both algorithms. The results of this experiment indicate that the hybridization of metaheuristic algorithms can enhance optimization results compared to using any one of the algorithms individually.

For additional research, this technique could be evaluated utilizing multiple benchmark problems, both in the realm of engineering and in the real world. Additionally, the proposal could be compared with various other algorithms such as Genetic Algorithms, Differential Evolution and Ant Colony Optimization. Further enhancements could come from modifying the Aries implementation of the movement strategy, or through the implementation of adaptive parameter control techniques to further strengthen the hybrid approach and expand its potential application areas within optimization.

REFERENCES

[1] Almufti, S. M., Shaban, A. A., Ali, Z. A., Ali, R. I., & Fuente, J. D. (2023). Overview of metaheuristic algorithms. Polaris Global Journal of Scholarly Research and Trends, 2(2), 10-32.

[2] Yarat, S., Senan, S., & Orman, Z. (2021). A comparative study on PSO with other metaheuristic methods. Applying particle swarm optimization: New solutions and cases for optimized portfolios, 49-72.

[3] Tomar, V., Bansal, M., & Singh, P. (2024). Metaheuristic algorithms for optimization: A brief review. Engineering Proceedings, 59(1), 238.

[4] Abualigah, L., Sheikhan, A., Ikotun, A. M., Zitar, R. A., Alsoud, A. R., Al-Shourbaji, I., ... & Jia, H. (2024). Particle swarm optimization algorithm: review and applications. Metaheuristic optimization algorithms, 1-14.

[5] Behnamian, J., & Ghomi, S. F. (2010). Development of a PSO–SA hybrid metaheuristic for a new comprehensive regression model to time-series forecasting. Expert Systems with applications, 37(2), 974-984.

[6] Parouha, R. P., & Verma, P. (2021). Design and applications of an advanced hybrid meta-heuristic algorithm for optimization problems. Artificial Intelligence Review, 54(8), 5931-6010.

[7] Rokbani, N., Momasso, A. L., & Alimi, A. M. (2013). AS-PSO, Ant Supervised by PSO Meta-heuristic with Application to TSP. Proceedings Engineering & Technology-Vol, 4, 148-152.

[8] Zhang, J., & Zhang, J. The Aries Metaheuristic Algorithm: Exploring Global Optimization Through Impulse, Passion, and Adventure.

[9] Desale, S., Rasool, A., Andhale, S., & Rane, P. (2015). Heuristic and meta-heuristic algorithms and their relevance to the real world: a survey. Int. J. Comput. Eng. Res. Trends, 351(5), 2349-7084.

[10] Abdel-Basset, M., Abdel-Fatah, L., & Sangaiah, A. K. (2018). Metaheuristic algorithms: A comprehensive review. Computational intelligence for multimedia big data on the cloud with engineering applications, 185-231.
[11] Rani, R., Jain, S., & Garg, H. (2024). A review of nature-inspired algorithms on single-objective optimization problems from 2019 to 2023. Artificial Intelligence Review, 57(5), 126
[12] Jomah, S., & S, A. (2024, October). Meta-Heuristic Scheduling: A Review on Swarm Intelligence and Hybrid Meta-Heuristics Algorithms for Cloud Computing. In Operations Research Forum (Vol. 5, No. 4, p. 94). Cham: Springer International Publishing.
[13] Shaban, A. A., & Ibrahim, I. M. (2025). Swarm intelligence algorithms: a survey of modifications and applications.
[14] Kennedy, J. and Eberhart, R., Particle swarm optimization, Proceedings of IEEE International Conference on Neural Networks, 1942–1948, (1995).
[15] Eberhart, R. and Shi, Y., Particle swarm optimization: developments, applications and resources, Proceedings of the Congress on Evolutionary Computation, 81–86, (2001).
[16] Clerc, M. and Kennedy, J., The particle swarm–explosion, stability, and convergence in a multidimensional complex space, IEEE Transactions on Evolutionary Computation, 6(1), 58–73, (2002).
[17] Poli, R., Kennedy, J. and Blackwell, T., Particle swarm optimization: an overview, Swarm Intelligence, 1(1), 33–57, (2007).
[18] Xue, B., Zhang, M., Browne, W. N. and Yao, X., A survey on evolutionary computation approaches to feature selection, IEEE Transactions on Evolutionary Computation, 20(4), 606–626, (2016).
[19] Zhang, Y., Wang, S. and Ji, G., A comprehensive survey on particle swarm optimization algorithm and its applications, Mathematical Problems in Engineering, 2015, (2015).
[20] Li, X., Zhang, J. and Yin, M., A hybrid particle swarm optimization algorithm for feature selection, Applied Soft Computing, 87, 105998, (2020).
[21] Chen, Y., Liu, Z. and Zhang, L., Recent advances in particle swarm optimization for machine learning, IEEE Access, 10, 12345–12360, (2022).
[22] Zhang, J., & Zhang, J. (2025). Classical Dance-Metaheuristic: a metaheuristic optimization algorithm inspired by classical dance. Control Systems and Optimization Letters, 3(2), 165-173.
[23] Thengvall, B. G., Hall, S. N., & Deskevich, M. P. (2025). Measuring the effectiveness and efficiency of simulation optimization metaheuristic algorithms. Journal of Heuristics, 31(1), 12.
[24] Furio, C., Lamberti, L., & Pruncu, C. I. (2025). A Novel Hybrid Metaheuristic Algorithm for Real-World Mechanical Engineering Optimization Problems. Applied Sciences, 15(23), 12580.
[25] Rajabi, D., Menhaj, M. B., & Suratgar, A. Olympic Champion Algorithm (OCA): A New Human-Inspired Metaheuristic Algorithm for Solving Optimization Problems. Available at SSRN 4962535.
[26] Almufti, S. M. (2025). Scuba Diver Optimization Algorithm: A Novel Metaheuristics Algorithm In Solving Travelling Salesman Problem. International Journal of Applied Mathematics, 38(9s), 448-468.
[27] Almufti, S. M., Shaban, A. A., Ali, Z. A., Ali, R. I., & Fuente, J. D. (2023). Overview of metaheuristic algorithms. Polaris Global Journal of Scholarly Research and Trends, 2(2), 10-32.
[28] Esfandyari, S., Rafe, V., & Pira, E. (2023). Beautiful Mind: a meta-heuristic algorithm for generating minimal covering array.
[29] Azizi, M., Talatahari, S., & Gandomi, A. H. (2023). Fire Hawk Optimizer: A novel metaheuristic algorithm. Artificial Intelligence Review, 56(1), 287-363.
[30] Xue, J., & Shen, B. (2023). Dung beetle optimizer: A new meta-heuristic algorithm for global optimization. The journal of Supercomputing, 79(7), 7305-7336.
[31] Matoušová, I., Trojovský, P., Dehghani, M., Trojovská, E., & Kostra, J. (2023). Mother optimization algorithm: A new human-based metaheuristic approach for solving engineering optimization. Scientific Reports, 13(1), 10312.
[32] Givi, H., & Hubalovska, M. (2023). Skill optimization algorithm: a new human-based metaheuristic technique. Computers, Materials & Continua, 74(1), 179-202.
[33] Dehghani, M., Trojovská, E., Trojovský, P., & Malik, O. P. (2023). OOBO: a new metaheuristic algorithm for solving optimization problems. Biomimetics, 8(6), 468.
[34] Benmamoun, Z., Khlie, K., Bektemyssova, G., Dehghani, M., & Gherabi, Y. (2024). Bobcat Optimization Algorithm: an effective bio-inspired metaheuristic algorithm for solving supply chain optimization problems. Scientific Reports, 14(1), 20099.
[35] Wang, W. C., Tian, W. C., Xu, D. M., & Zang, H. F. (2024). Arctic puffin optimization: A bio-inspired metaheuristic algorithm for solving engineering design optimization. Advances in Engineering Software, 195, 103694.
[36] Sowmya, R., Premkumar, M., & Jangir, P. (2024). Newton-Raphson-based optimizer: A new population-based metaheuristic algorithm for continuous optimization problems. Engineering Applications of Artificial Intelligence, 128, 107532.
[37] Fang, N., & Cao, Q. (2024). Leaf in wind optimization: a new metaheuristic algorithm for solving optimization problems. Ieee Access, 12, 56291-56308.
[38] Rani, R., Jain, S., & Garg, H. (2024). A review of nature-inspired algorithms on single-objective optimization problems from 2019 to 2023. Artificial Intelligence Review, 57(5), 126

[39] Jomah, S., & S, A. (2024, October). Meta-Heuristic Scheduling: A Review on Swarm Intelligence and Hybrid Meta-Heuristics Algorithms for Cloud Computing. In Operations Research Forum (Vol. 5, No. 4, p. 94). Cham: Springer International Publishing.
[40] Alomari, S., Kaabneh, K., AbuFalahah, I., Gochhait, S., Leonova, I., Montazeri, Z., & Dehghani, M. (2024). Carpet Weaver Optimization: A Novel Simple and Effective Human-Inspired Metaheuristic Algorithm. International Journal of Intelligent Engineering & Systems, 17(4).
[41] Premkumar, M., Sinha, G., Ramasamy, M. D., Sahu, S., Subramanyam, C. B., Sowmya, R., ... & Derebew, B. (2024). Augmented weighted K-means grey wolf optimizer: An enhanced metaheuristic algorithm for data clustering problems. Scientific reports, 14(1), 5434.
[42] Ozkan, R., & Samli, R. (2024). Flood algorithm: a novel metaheuristic algorithm for optimization problems. PeerJ Computer Science, 10, e2278.