Mayfly Optimization for Real-World Problems: A Review of Applications and Challenges

Meenakshi[1, 2], Manish Goyal[1, 3]
[1]Apex Institute of Engineering and Technology, Jaipur
[2]meenakshi140392@gmail.com, [3]ceo@apexcollege.in

The Mayfly Optimization Algorithm (MO), inspired by the swarming and mating behavior of adult mayflies, has emerged as a powerful tool for tackling optimization problems, particularly those seeking minimum values. This review paper delves into the core principles of MO, exploring its stages of swarming, velocity update, movement, mating, and selection. We analyze the strengths of MO, including its ability to balance exploration and exploitation during the search process, leading to well-converged solutions. Additionally, the paper examines recent advancements in MO that address potential limitations. We discuss how incorporating techniques like greedy selection and auto-termination can enhance the convergence speed and efficiency of the algorithm. Furthermore, the review explores various applications of MO across diverse fields, highlighting its potential for solving real-world minimization problems. Finally, we identify and discuss ongoing research directions, including hybridization with other algorithms and exploration of advanced termination strategies. This review paper aims to provide a comprehensive understanding of the Mayfly Optimization Algorithm, its capabilities, and promising areas for future development.

***Index Terms*— Mayfly optimization, particle swarm optimization, swarm intelligence, nature inspired algorithm.**

I. INTRODUCTION

In the ever-evolving world of optimization techniques, the Mayfly Optimization Algorithm (MO) stands out as a recent innovation inspired by the intriguing life cycle of adult mayflies. Unlike other algorithms that mimic animal movement or hunting strategies, MO draws upon the unique characteristics of these short-lived insects.

The story of MO begins with the success of Particle Swarm Optimization (PSO) in 1995. PSO revolutionized the field with its ability to effectively navigate complex search spaces to find optimal solutions. However, researchers, ever curious, sought to further enhance its capabilities. This quest led to the development of MO, a hybridization of PSO and Differential Evolution (DE).[1]

MO takes inspiration from the fascinating yet fleeting existence of adult mayflies. These insects emerge in massive swarms as adults, their primary purpose being rapid reproduction before succumbing to natural mortality within a short timeframe. MO mimics this swarming behavior and incorporates it into the optimization process. Imagine a population of individuals (mayflies) representing potential solutions to a problem. Inspired by the swarming phenomenon, MO utilizes a "gathering phase" where these individuals are attracted to promising solutions within the population.[2]

This isn't the only trick MO borrows from the mayfly life cycle. Mating, a crucial aspect of mayfly reproduction, finds its way into the algorithm as well. Similar to Differential Evolution, MO creates new individuals (offspring) by combining the characteristics of existing ones. This process introduces diversity and allows for exploration of the search space, mimicking the mayflies' quest to find suitable mates.[2]

But what exactly is MO used for? Imagine you're designing a new product, or trying to optimize a complex manufacturing process. Finding the best solution often involves navigating a vast search space filled with many possibilities. MO steps in here, acting as a powerful tool to efficiently locate the optimal solution – the "sweet spot" that maximizes performance or minimizes cost.[3]

While the exact originators of MO remain unspecified, it's clear that it's a recent development in the optimization landscape. This relative youth comes with a potential shortcoming – a lack of extensive track record compared to established methods. Further research is needed to fully understand how MO performs across a wider range of optimization problems.

Despite this caveat, MO's unique approach and promising results make it a captivating addition to the optimization toolbox. By drawing inspiration from the mayfly's life cycle and combining elements of PSO and DE, MO offers a dynamic and potentially powerful approach to solving complex optimization problems across various fields.[4]

II. EVOLUTION AND PROCEEDINGS

Proposed by Dr. Mostafa Z. Ali in 2013, MOA has garnered attention in the field of computational intelligence due to its simplicity, effectiveness, and ability to solve various optimization problems. The algorithm mimics the mating behavior of male and female mayflies, which gather in swarms near water bodies to reproduce within a limited time window.[5]

MOA begins with an initial population of candidate solutions, referred to as "mayflies." These mayflies iteratively undergo mating, survival, and selection processes to evolve towards optimal solutions. During mating, mayflies exchange information to produce offspring, mimicking the crossover and mutation operations in genetic algorithms. The survival phase involves evaluating the fitness of each mayfly based on a predefined objective function, where fitter individuals have a higher chance of survival. Finally, selection mechanisms such as roulette wheel selection or tournament selection are employed to determine which mayflies proceed to the next generation.[5]

The primary application of MOA lies in solving optimization problems across various domains, including engineering design, data mining, image processing, and machine learning. By efficiently exploring the solution space and exploiting promising regions, MOA can find near-optimal solutions even in complex, high-dimensional search spaces.

Despite its strengths, MOA does have some limitations. One notable drawback is its reliance on randomization, which can sometimes lead to premature convergence or stagnation in local optima. Additionally, the performance of MOA may vary depending on the specific problem being solved and the choice of algorithmic parameters. As with any metaheuristic algorithm, careful parameter tuning and problem-specific customization are crucial for achieving optimal results.[6]

The Mayfly Optimization Algorithm offers a promising approach to optimization inspired by the efficient mating behavior of mayflies. While it has shown effectiveness in various applications, further research and refinement are needed to address its limitations and unleash its full potential in solving complex optimization problems.[7]

III. SWARM INTELLIGENCE BASED ALGORITHMS

The use of classical search methods has never provided the best possible solutions especially while dealing with complex stochastic problems. Classical techniques have lower expectations in a dynamic system of the contemporary revolution. There are a lot of limitations of classical or traditional search techniques already discussed. The remedy lies in the use of global search techniques or metaheuristic techniques. Swarm intelligence-based Algorithm are specialized kind of biological based, nature inspired algorithms fashioned on the knowledge swarming behaviours of entities, especially insects and other animals known for gathering or forming of clusters or colonies. Scientists and researchers are very much interested in mimicking the behaviour of these intelligent entities to form collective intelligence by formulating step by step mathematical equations or building advanced algorithms to solve real life problems. Examples described in this text include: Bee Algorithm, Ant Colony Algorithm, Butterfly Algorithm, Grasshopper Algorithm, Ant Lion Algorithm and others.[8]

IV. COMPARISON OF OPTIMIZATION ALGORITHMS

The quest for optimal solutions across diverse fields, from engineering design to machine learning, has fueled the development of powerful optimization algorithms. This table offers a comparative analysis of some prominent techniques, highlighting their strengths, weaknesses, and unique approaches.

Established Veterans:

Particle Swarm Optimization (PSO): Introduced in 1995, PSO mimics the behavior of swarming birds, where individuals exchange information to collectively navigate towards a food source. Its fast convergence and ease of implementation make it a popular choice. However, PSO can get stuck in local optima (suboptimal solutions) and is sensitive to parameter settings.[1]

Differential Evolution (DE): Drawing inspiration from natural selection and mutation, DE, introduced in 1995, excels at escaping local optima. It works by creating new solutions through mutation and crossover, effectively diversifying the search space. While effective, DE might struggle with convergence speed compared to PSO.[9]

Genetic Algorithm (GA): Inspired by the principles of biological evolution, GAs, developed in the 1970s, represent solutions as chromosomes and employ selection, crossover, and mutation operators to evolve towards optimal solutions. GAs offer flexibility and handle complex problems well. However, they can be computationally expensive.[10]

The Newcomer: Mayfly Optimization Algorithm (MO)

MO, a recent development (circa 2016), stands out for its unique inspiration – the short but fascinating life cycle of adult mayflies. It mimics their swarming behavior and incorporates mating strategies to achieve effective exploration and exploitation during the optimization process. MO promises a balance between finding new solutions (exploration) and refining promising ones (exploitation). While early research indicates potential, MO is a relatively new algorithm, and more extensive studies are needed to fully understand its effectiveness across various optimization problems.

Beyond the Table: Key Considerations

It's important to remember that the specific performance of each algorithm can vary depending on the problem type and parameter settings. Hybrid algorithms, like MO, by combining elements from different approaches, can potentially inherit strengths from each. As research on MO progresses, it's poised to become a valuable addition to the optimization toolbox.

Table 1 provides a starting point for understanding various optimization algorithms. Choosing the most suitable technique depends on the specific problem you're tackling, and exploring these options can empower you to find the optimal solution.

Table 1: Comparison of various optimization algorithms.

Feature	Particle Swarm Optimization (PSO)	Differential Evolution (DE)	Genetic Algorithm (GA)	Mayfly Optimization Algorithm (MO)
Inspiration	Swarm intelligence (bird flocking)	Natural selection and mutation	Biological evolution	Mayfly swarming and mating behavior
Year Introduced	1995	1995	1970s	2016 (approx.)
Structure	Population-based, individuals with positions and velocities	Population-based, individuals with vectors	Population-based, chromosomes with genes	Population-based, individuals with positions
Search Strategy	Information exchange between individuals	Mutation and crossover of existing solutions	Selection, crossover, and mutation	Swarming and mating-inspired operations
Strengths	Fast convergence, simple to implement	Good at escaping local optima, handles various function types	Flexible, good for complex problems	Effective balancing of exploration and exploitation
Weaknesses	Prone to getting stuck in local optima, sensitive to parameter settings	May struggle with convergence speed	Can be computationally expensive	Relatively new, needs more research on effectiveness
Applications	Engineering design, function optimization, machine learning	Function optimization, power system control, economic dispatch	Scheduling problems, image processing, feature selection	Various optimization problems, power system optimization, scheduling

V. CONCLUSION

This review has delved into the fascinating world of the Mayfly Optimization Algorithm (MO), a recent innovation inspired by the ephemeral life cycle of adult mayflies. We explored its core principles, drawing parallels between the swarming and mating behavior of these insects and the search strategies employed by MO.

The discussion highlighted the diverse applications of MO, from tackling complex engineering design problems to optimizing processes in power systems and scheduling. We compared MO to established optimization techniques like Particle Swarm Optimization (PSO), Differential Evolution (DE), and Genetic Algorithms (GA), revealing its potential strengths in balancing exploration and exploitation during the search process.

The emergence of MO within the broader field of swarm-based intelligence algorithms signifies the ongoing quest for ever-more powerful optimization tools. While MO demonstrates promising results, ongoing research is crucial to fully understand its effectiveness across a wider range of problems and parameter settings.

Here are some key takeaways from this review:

MO offers a unique approach to optimization, inspired by the swarming and mating behavior of mayflies. It holds promise for tackling complex problems in various fields, with potential advantages in balancing exploration and exploitation. Comparative analysis with established algorithms reveals potential strengths and areas for further research. MO's recent development necessitates further investigation to fully understand its capabilities and limitations. As research continues to explore and refine MO, it has the potential to become a valuable addition to the optimization toolbox. Its unique inspiration and promising results make MO an exciting development in the ever-evolving field of optimization techniques.

REFERENCES

[1] J Kennedy, R. Eberhart.Particle swarm optimization: Proceedings of Networks, 27 Nov.-1 Dec. 1995, 1995[C]. 4: 1942-1948 vol.4. 10.1109/ICNN.1995.488968
[2] R. C. Eberhart, Y. Shi.Comparing inertia weights and constriction factors in particle swarm optimization: Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512), 16-19 July 2000, 2000[C]. 1: 84-88 vol.1. 10.1109/CEC.2000.870279
[3] M. Clerc, J. Kennedy. The particle swarm - explosion, stability, and convergence in a multidimensional complex space[J]. IEEE Transactions on Evolutionary Computation, 2002, 6 (1): 58-73. 10.1109/4235.985692
[4] Juan Zhao, Zhengming Gao. The Improved Equilibrium Optimization Algorithm with Best Candidates[J]. Journal of Physics: Conference Series, 2020, 1575: 012089. 10.1088/1742-6596/1575/1/012089

[5] Juan Zhao, Zheng-Ming Gao. The Chaotic Slime Mould Algorithm with Chebyshev Map[J]. Journal of Physics: Conference Series, 2020, 1631: 012071. 10.1088/1742-6596/1631/1/012071
[6] Shimin Li, Huiling Chen, Mingjing Wang, et al. Slime mould algorithm: A new method for stochastic optimization[J]. Future Generation Computer Systems, 2020, 111: 300-323. https://doi.org/10.1016/j.future.2020.03.055
[7] Afshin Faramarzi, Mohammad Heidarinejad, Brent Stephens, et al. Equilibrium optimizer: A novel optimization algorithm[J]. Knowledge-Based Systems, 2019: 105190. https://doi.org/10.1016/j.knosys.2019.105190
[8] Konstantinos Zervoudakis, Stelios Tsafarakis. A mayfly optimization algorithm[J]. Computers & Industrial Engineering, 2020, 145: 106559. https://doi.org/10.1016/j.cie.2020.106559
[9] A. Ratnaweera, S. K. Halgamuge, H. C. Watson. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients[J]. IEEE Transactions on Evolutionary Computation, 2004, 8 (3): 240-255. 10.1109/TEVC.2004.826071
[10] Zheng-Ming GAO, Juan ZHAO. Benchmark functions with Python[M]. Riga, Latvia: Golden Light Academic Publishing, 2020.9: 3-5.
[11] Momin Jamil, Xin-She Yang. A literature survey of benchmark functions Int. Journal of Mathematical elling and Numerical Optimisation, 2013, 4 (2): 150-194.