

Equilibrium Optimizer Algorithm

Subjects: [Computer Science](#), [Artificial Intelligence](#)

Contributor: Hongwei Ding , Yuting Liu , Zongshan Wang , Gushen Jin , Peng Hu , Gaurav Dhiman

The equilibrium optimizer (EO) is a recently developed physics-based optimization technique for complex optimization problems.

equilibrium optimizer

metaheuristics

global optimization

nature-inspired

mobile robot path planning

1. Introduction

Optimization problems have gained significant attention in engineering and scientific domains. In general, the objective of optimization problems is to achieve the best possible outcome by minimizing the corresponding objective function while minimizing undesirable factors ^[1]. These problems may involve constraints, which means that various constraints need to be satisfied during the optimization process. Based on their characteristics, optimization problems can be classified into two categories: local optimization and global optimization. Local optimization aims to determine the optimal value within a local region ^[2]. On the other hand, global optimization aims to find the optimal value within a given region. Therefore, global optimization is more challenging compared to local optimization.

To address various types of global optimization problems, numerous optimization techniques have been developed ^[3]. Among the current optimization techniques, metaheuristic algorithms have gained widespread attention due to their advantages of being gradient-free, not requiring prior information about the problem, and offering high flexibility. Metaheuristic algorithms provide acceptable solutions with relatively fewer computational costs ^[4]. Based on their sources of inspiration, metaheuristic algorithms can be classified into four categories: swarm intelligence algorithms, evolutionary optimization algorithms, physics-inspired algorithms, and human-inspired algorithms ^{[5][6]}. Swarm intelligence optimization algorithms simulate the cooperative behavior observed in animal populations in nature. Examples of such algorithms include Artificial Bee Colony (ABC) ^[7], Particle Swarm Optimization (PSO) ^[8], Grey Wolf Optimization (GWO) ^[9], Firefly Optimization (FA) ^[10], Ant Colony Optimization (ACO) ^[11], Harris Hawks Optimization Algorithm (HHO) ^[12], Salp Swarm Algorithm (SSA) ^[13], and others. The second category draws inspiration from the concept of natural evolution. These algorithms include, but are not limited to, Evolution Strategy (ES) ^[14], Differential Evolution (DE) ^[15], Backtracking Search Algorithm (BSA) ^[16], Stochastic Fractal Search (SFS) ^[17], Wildebeests Herd Optimization (WHO) ^[18]. The third class of metaheuristic algorithms is inspired by physics concepts. The following algorithms are some examples of physics-inspired algorithms: Simulated Annealing (SA) ^[19] algorithm, Big Bang-Big Crunch (BB-BC) ^[20] algorithm, Central Force Optimization (CFO) ^[21],

Intelligent Water Drops (IWD) [22], Slime Mold Algorithm (SMA) [23], Gravitational Search Algorithm (GSA) [24], Black Hole Algorithm (BHA) [25], Water Cycle Algorithm (WCA) [26], Lightning Search Algorithm (LSA) [27]. As these physics-inspired algorithms proved to be effective in engineering and science, more similar algorithms were developed, such as Multi-Verse Optimizer (MVO) [28], Thermal Exchange Optimization (TEO) [29], Henry Gas Solubility Optimization (HGSO) [30], Equilibrium Optimizer (EO) [31], Archimedes Optimization Algorithm (AOA) [32], and Special Relativity Search (SRS) [33]. The last class of metaheuristic techniques simulates human behavior, such as Seeker Optimization Algorithm (SOA) [34], Imperialist Competitive Algorithm (ICA) [35], Brain Storm Optimization (BSO) [36], and Teaching-Learning-Based Optimization (TLBO) [37].

The most popular categories among these are swarm intelligence algorithms and physics-inspired algorithms, as they offer reliable metaphors and simple yet efficient search mechanisms. In this research, researchers consider leveraging the search behavior of swarm intelligence algorithms to enhance the performance of a physics-inspired algorithm called EO. Equilibrium optimizer (EO) simulates the dynamic equilibrium concept of mass in physics. In a container, the attempt to achieve dynamic equilibrium of mass within a controlled volume is performed by expelling or absorbing particles, which are referred to as a set of operators employed during the search in the solution space. Based on these search models, EO has demonstrated its performance across a range of real-world problems, such as solar photovoltaic parameter estimation [38], feature selection [39], multi-level threshold image segmentation [40], and so on. Despite the simple search mechanism and effective search capability of the EO algorithm, it still suffers from limitations, such as falling into local optima traps and imbalanced exploration and exploitation. To address these limitations, this research proposes a novel variant of EO called SSEO by introducing an adaptive inertia weight factor and a swarm-based spiral search mechanism. The adaptive inertia weight factor is employed to enhance population diversity and strengthen the algorithm's global exploration ability, while the spiral search mechanism is introduced to expand the search space of particles.

2. Equilibrium Optimizer

Well-established metaheuristic algorithms are equipped with reasonable mechanisms to transition between exploration and exploitation. Global exploration allows the algorithm to comprehensively search the solution space and explore unknown regions, while local exploitation aids in fine-tuning solutions within specific areas to improve solution accuracy. EO algorithm, a recently proposed physics-inspired metaheuristic algorithm, is based on metaphors from the field of physics. The efficiency and applicability of EO have been demonstrated in benchmark function optimization problems as well as real-world problems. However, despite EO's attempt to design effective search models based on reliable metaphors, the transition from exploration to exploitation during the search process is still imperfect, resulting in limitations such as getting trapped in local optima and premature convergence.

To mitigate the inherent limitations of EO and provide a viable alternative efficient optimization tool for the optimization community, many researchers have made improvements and proposed different versions of EO variants. Gupta et al. [41] introduced mutation strategies and additional search operators, referred to as mEO, into the basic EO. The mutation operation is used to overcome the problem of population diversity loss during the

search process, and the additional search operators assist the population in escaping local optima. The performance of mEO was tested on 33 commonly used benchmark functions and four engineering design problems. Experimental results demonstrated that mEO effectively enhances the search capability of the EO algorithm.

Houssein et al. [42] strengthened the balance between exploration and exploitation in the basic EO algorithm by employing the dimension hunting technique. The performance of the proposed EO variant was tested using the CEC 2020 benchmark test suite and compared with advanced metaheuristic methods. Comparative results showed the superiority of the proposed approach. Additionally, the proposed EO variant was applied to multi-level thresholding image segmentation of CT images. Comparative results with a set of popular image segmentation tools showed good performance in terms of segmentation accuracy.

Liu et al. [43] introduced three new strategies into EO to improve algorithm performance. In this version of EO, Levy flight was used to enhance particle search in unknown regions, the WOA search mechanism was employed to strengthen local exploitation tendencies, and the adaptive perturbation technique was utilized to enhance the algorithm's ability to avoid local optima. The performance of the algorithm was tested on the CEC 2014 benchmark test suite and compared with several well-known algorithms. Comparative results showed that the proposed EO variant outperformed the compared algorithms in the majority of cases. Furthermore, the algorithm's capability to solve real-world problems was investigated using engineering design cases, demonstrating its practicality in addressing real-world problems.

Tan et al. [44] proposed a hybrid algorithm called EWOA, which combines EO and WOA, aiming to compensate for the inherent limitations of the EO algorithm. Comparative results with the basic EO, WOA, and several classical metaheuristic algorithms showed that EWOA mitigates the tendency of the basic EO algorithm to get trapped in local optima to a certain extent.

Zhang et al. [45] introduced an improved EO algorithm, named ISEO, by incorporating an information exchange reinforcement mechanism to overcome the weak inter-particle information exchange capability in the basic EO. In ISEO, a global best-guided mechanism was employed to enhance the guidance towards a balanced state, a reverse learning technique was utilized to assist the population in escaping local optima, and a differential mutation mechanism was expected to improve inter-particle information exchange. These three mechanisms were simultaneously embedded in EO, resulting in an overall improved algorithm performance. The effectiveness of ISEO was demonstrated on a large number of benchmark test functions and engineering design cases.

Minocha et al. [46] proposed an EO variant called MEO, which enhances the convergence performance of the basic EO. In MEO, adjustments were made to the construction of the balance pool to strengthen the algorithm's search intensity, and the Levy flight technique was introduced to improve global search capability. To investigate the convergence performance of MEO, 62 benchmark functions with different characteristics and five engineering design cases were utilized. Experimental results demonstrated that MEO provides excellent robustness and convergence compared to other algorithms.

Balakrishnan et al. [47] introduced an improved version of EO, called LEO, for feature selection problems. LEO inherits the framework and Levy flight mechanism of EO with the expectation of providing a better search capability in comparison to the basic EO. To validate the performance of LEO, the algorithm was tested on a microarray cancer dataset and compared with several high-performing feature selection methods. Comparative results showed significant advantages of LEO in terms of accuracy and speed compared to the compared algorithms.

References

1. Gharehchopogh, F.S. Quantum-inspired metaheuristic algorithms: Comprehensive survey and classification. *Artif. Intell. Rev.* 2022, 56, 5479–5543.
2. Wang, Z.; Ding, H.; Yang, J.; Wang, J.; Li, B.; Yang, Z.; Hou, P. Advanced orthogonal opposition-based learning-driven dynamic salp swarm algorithm: Framework and case studies. *IET Control. Theory Appl.* 2022, 16, 945–971.
3. Kaveh, A.; Zaerreza, A. A new framework for reliability-based design optimization using metaheuristic algorithms. *Structures* 2022, 38, 1210–1225.
4. Wang, Z.; Ding, H.; Yang, J.; Hou, P.; Dhiman, G.; Wang, J.; Yang, Z.; Li, A. Orthogonal pinhole-imaging-based learning salp swarm algorithm with self-adaptive structure for global optimization. *Front. Bioeng. Biotechnol.* 2022, 10, 1018895.
5. Abdel-Basset, M.; Mohamed, R.; Azeem, S.A.A.; Jameel, M.; Abouhawwash, M. Kepler optimization algorithm: A new metaheuristic algorithm inspired by Kepler's laws of planetary motion. *Knowl.-Based Syst.* 2023, 268, 110454.
6. Khishe, M.; Mosavi, M.R. Chimp optimization algorithm. *Expert Syst. Appl.* 2020, 149, 113338.
7. Karaboga, D.; Basturk, B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *J. Glob. Optim.* 2007, 39, 459–471.
8. Wang, D.; Tan, D.; Liu, L. Particle swarm optimization algorithm: An overview. *Soft Comput.* 2018, 22, 387–408.
9. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* 2014, 69, 46–61.
10. Hossein, G.A.; Yang, X.-S.; Alavi, A.H. Mixed variable structural optimization using firefly algorithm. *Comput. Struct.* 2011, 89, 2325–2336.
11. Dorigo, M.; Birattari, M.; Stutzle, T. Ant colony optimization. *IEEE Comput. Intell. Mag.* 2006, 1, 28–39.
12. Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Future Gener. Comput. Syst.* 2019, 97, 849–872.

13. Mirjalili, S.; Gandomi, A.H.; Mirjalili, S.Z.; Saremi, S.; Faris, H.; Mirjalili, S.M. Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* 2017, 114, 163–191.
14. Choi, K.; Jang, D.-H.; Kang, S.-I.; Lee, J.-H.; Chung, T.-K.; Kim, H.-S. Hybrid Algorithm Combining Genetic Algorithm with Evolution Strategy for Antenna Design. *IEEE Trans. Magn.* 2015, 52, 1–4.
15. Price, K.V. Differential evolution. In *Handbook of Optimization: From Classical to Modern Approach*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 187–214.
16. Civicioglu, P. Backtracking Search Optimization Algorithm for numerical optimization problems. *Appl. Math. Comput.* 2013, 219, 8121–8144.
17. Salimi, H. Stochastic Fractal Search: A powerful metaheuristic algorithm. *Knowl.-Based Syst.* 2015, 75, 1–18.
18. Amali, D.G.B.; Dinakaran, M. Wildebeest herd optimization: A new global optimization algorithm inspired by wildebeest herding behaviour. *J. Intell. Fuzzy Syst.* 2019, 37, 8063–8076.
19. Dimitris, B.; Tsitsiklis, J. Simulated annealing. *Stat. Sci.* 1993, 8, 10–15.
20. Osman, K.E.; Eksin, I. A new optimization method: Big bang–big crunch. *Adv. Eng. Softw.* 2006, 37, 106–111.
21. Richard, A.F. Central force optimization. *Prog. Electromagn. Res.* 2007, 77, 425–491.
22. Hosseini, H.S. The intelligent water drops algorithm: A nature-inspired swarm-based optimization algorithm. *Int. J. Bio-Inspired Comput.* 2009, 1, 71–79.
23. Nguyen, T.-T.; Wang, H.-J.; Dao, T.-K.; Pan, J.-S.; Liu, J.-H.; Weng, S. An Improved Slime Mold Algorithm and its Application for Optimal Operation of Cascade Hydropower Stations. *IEEE Access* 2020, 8, 226754–226772.
24. Rashedi, E.; Nezamabadi-Pour, H.; Saryazdi, S. GSA: A Gravitational Search Algorithm. *Inf. Sci.* 2009, 179, 2232–2248.
25. Abualigah, L.; Elaziz, M.A.; Sumari, P.; Khasawneh, A.M.; Alshinwan, M.; Mirjalili, S.; Shehab, M.; Abuaddous, H.Y.; Gandomi, A.H. Black hole algorithm: A comprehensive survey. *Appl. Intell.* 2022, 52, 11892–11915.
26. Sadollah, A.; Eskandar, H.; Lee, H.M.; Yoo, D.G.; Kim, J.H. Water cycle algorithm: A detailed standard code. *SoftwareX* 2016, 5, 37–43.
27. Shareef, H.; Ibrahim, A.A.; Mutlag, A.H. Lightning search algorithm. *Appl. Soft Comput.* 2015, 36, 315–333.
28. Mirjalili, S.; Mirjalili, S.M.; Hatamlou, A. Multi-Verse Optimizer: A nature-inspired algorithm for global optimization. *Neural Comput. Appl.* 2015, 27, 495–513.

29. Kaveh, A.; Dadras, A. A novel meta-heuristic optimization algorithm: Thermal exchange optimization. *Adv. Eng. Softw.* 2017, 110, 69–84.
30. Hashim, F.A.; Houssein, E.H.; Mabrouk, M.S.; Al-Atabany, W.; Mirjalili, S. Henry gas solubility optimization: A novel physics-based algorithm. *Futur. Gener. Comput. Syst.* 2019, 101, 646–667.
31. Faramarzi, A.; Heidarinejad, M.; Stephens, B.; Mirjalili, S. Equilibrium optimizer: A novel optimization algorithm. *Knowl.-Based Syst.* 2020, 191, 105190.
32. Hashim, F.A.; Hussain, K.; Houssein, E.H.; Mabrouk, M.S.; Al-Atabany, W. Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems. *Appl. Intell.* 2021, 51, 1531–1551.
33. Goodarzimehr, V.; Shojaee, S.; Hamzehei-Javaran, S.; Talatahari, S. Special relativity search: A novel metaheuristic method based on special relativity physics. *Knowl.-Based Syst.* 2022, 257, 109484.
34. Dai, C.; Chen, W.; Zhu, Y.; Zhang, X. Seeker optimization algorithm for optimal reactive power dispatch. *IEEE Trans. Power Syst.* 2009, 24, 1218–1231.
35. Kaveh, A.; Talatahari, S. Optimum design of skeletal structures using imperialist competitive algorithm. *Comput. Struct.* 2010, 88, 1220–1229.
36. Cheng, S.; Qin, Q.; Chen, J.; Shi, Y. Brain storm optimization algorithm: A review. *Artif. Intell. Rev.* 2016, 46, 445–458.
37. Rao, R.V.; Savsani, V.J.; Vakharia, D.P. Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Comput.-Aided Des.* 2011, 43, 303–315.
38. Abdel-Basset, M.; Mohamed, R.; Mirjalili, S.; Chakraborty, R.K.; Ryan, M.J. Solar photovoltaic parameter estimation using an improved equilibrium optimizer. *Sol. Energy* 2020, 209, 694–708.
39. Ahmed, S.; Ghosh, K.K.; Mirjalili, S.; Sarkar, R. AIEOU: Automata-based improved equilibrium optimizer with U-shaped transfer function for feature selection. *Knowl.-Based Syst.* 2021, 228, 107283.
40. Wunnavu, A.; Naik, M.K.; Panda, R.; Jena, B.; Abraham, A. A novel interdependence based multilevel thresholding technique using adaptive equilibrium optimizer. *Eng. Appl. Artif. Intell.* 2020, 94, 103836.
41. Gupta, S.; Deep, K.; Mirjalili, S. An efficient equilibrium optimizer with mutation strategy for numerical optimization. *Appl. Soft Comput.* 2020, 96, 106542.
42. Houssein, E.H.; Helmy, B.E.-D.; Oliva, D.; Jangir, P.; Premkumar, M.; Elngar, A.A.; Shaban, H. An efficient multi-thresholding based COVID-19 CT images segmentation approach using an improved equilibrium optimizer. *Biomed. Signal Process. Control* 2022, 73, 103401.

43. Liu, J.; Li, W.; Li, Y. LWMEO: An efficient equilibrium optimizer for complex functions and engineering design problems. *Expert Syst. Appl.* 2022, 198, 116828.
44. Tan, W.-H.; Mohamad-Saleh, J. A hybrid whale optimization algorithm based on equilibrium concept. *Alex. Eng. J.* 2023, 68, 763–786.
45. Zhang, X.; Lin, Q. Information-utilization strengthened equilibrium optimizer. *Artif. Intell. Rev.* 2022, 55, 4241–4274.
46. Minocha, S.; Singh, B. A novel equilibrium optimizer based on levy flight and iterative cosine operator for engineering optimization problems. *Expert Syst.* 2022, 39, e12843.
47. Balakrishnan, K.; Dhanalakshmi, R.; Akila, M.; Sinha, B.B. Improved equilibrium optimization based on Levy flight approach for feature selection. *Evol. Syst.* 2023, 14, 735–746.

Retrieved from <https://encyclopedia.pub/entry/history/show/110274>