Novelty Journals

ISSN 2394-9678

International Journal of Novel Research in Electrical and Mechanical Engineering
Vol. 11, Issue 1, pp: (95-100), Month: September 2023 - August 2024, Available at: www.noveltyjournals.com

A Comparison and Survey of Optimization Techniques

[1]Shivam Nanda, [2]Pradeepti Lakra

[1,2]Department of Electrical Engineering, Jabalpur Engineering College

Jabalpur (M.P), India

DOI: https://doi.org/10.5281/zenodo.10706875

Published Date: 26-February-2024

Abstracts: **Optimization problem formulation, methods of optimization and solution techniques are presented. Population-based methods are also explained. Optimization using constraints in terms of reliability is found to be the best option for optimizing structures with discrete parameters. This paper presents a comparative analysis of the different test-case optimization techniques. There are various optimization techniques available for the background. This review explains the different optimization techniques on the basis of their evolution, methodology, performance and applications.**

Keywords: **Evolutionary methods, Genetic algorithms, Particle swarm optimization techniques, evolution, applications etc.**

1. INTRODUCTION

A Comparative analysis of optimization techniques in various fields ranging from engineering to computer science, optimization techniques play a pivotal role in refining processes, enhancing efficiency, and achieving optimal outcomes. Optimization involves the systematic exploration of potential solutions to identify the best possible result under given constraints. With the ever-increasing complexity of modern challenges, researchers and practitioners are continually developing and refining optimization methods to tackle diverse problems [25].

This study aims to conduct a comprehensive comparative analysis of different optimization techniques, shedding light on their strengths, weaknesses, and applications across various domains. By examining the performance, convergence rates, and adaptability of these techniques, we seek to provide valuable insights into their suitability for specific problem types. Such an analysis is crucial not only for researchers striving to select the most appropriate optimization approach for their work but also for practitioners seeking to apply these techniques to real-world scenarios. Throughout this exploration, we will delve into both classical optimization algorithms and state-of-the-art methods, considering factors such as problem complexity, solution accuracy, computational efficiency and robustness [26]. By examining this spectrum of techniques, we aim to contribute to a deeper understanding of the trade-offs involved in selecting and implementing optimization strategies. Through this comparative analysis, we hope to equip researchers, engineers, and decision-makers with the knowledge necessary to make informed choices when approaching optimization challenges. As we navigate through the intricacies of these techniques, we will uncover insights that can drive innovation and foster advancements in a multitude of applications, ultimately leading to more refined and effective solutions to complex problems. Optimization tends to provide the best output with the fewest investments. The goal of optimization techniques for test cases is to minimize the number of test cases without affecting the fault coverage of the testing process. In this paper, a critical review of different test case optimization techniques is presented. Different techniques are compared with each other based on their evolution, methodology, performance and applications.

International Journal of Novel Research in Electrical and Mechanical Engineering
Vol. 11, Issue 1, pp: (95-100), Month: September 2023 - August 2024, Available at: www.noveltyjournals.com

2. COMPARATIVE ANALYSIS

This section covers the comparative analysis of different optimization techniques based on parameters like evolution, methodology, performance and applications.

Evolution: The gradual change in the inherited traits of populations over successive generations, driven by mechanisms like natural selection, genetic variation, and adaptation.

Methodology: The systematic and structured approach used to plan, conduct, and analyze research, experiments, or investigations, ensuring reliability, accuracy, and reproducibility of results.

Performance: The measurable outcomes or results achieved in a particular task, activity, or system, often assessed against established criteria or standards.

Applications: Specific contexts or scenarios in which theories, methods, or technologies are put to practical use to solve problems, achieve goals, or provide solutions.

2.1 PARTICLE SWARM OPTIMIZATION

PSO, or Particle Swarm Optimization, is like a group of birds or fish trying to find the best spot to eat. Each bird or fish (particle) moves around and adjusts its position based on its own experience and the success of its neighbors. Over time, they all work together to find the tastiest food (an optimal solution) in a smart way. It's a computer algorithm inspired by nature's teamwork that helps solve tricky problems by mimicking how animals cooperate to find the best outcomes [1].

Evolution: The Evolution of Particle Swarm Optimization (PSO) has seen various adaptations and enhancements since its original conception in the 1990s. Researchers have introduced modifications to the algorithm to address its limitations and tailor it to different types of optimization problems [2]. Here's an overview of some notable stages in the evolution of PSO: Standard PSO (1995-2000) the original PSO algorithm was introduced by Eberhart and Kennedy in the mid-1990s. It focused on simulating the social behavior of birds and fish, with particles adjusting their positions based on their own experiences and the experiences of their neighbors [3]. This simple approach quickly gained attention due to its efficiency in solving various optimization problems. The evolution of PSO continues to progress as researchers seek to refine the algorithm's performance, address its limitations, and adapt it to emerging challenges in optimization and machine learning.

Methodology: Particle Swarm Optimization (PSO) is a method that imitates the cooperative behavior observed in nature, such as the movement of flocks of birds or schools of fish. In PSO, a group of "particles" represents potential solutions to a problem. These particles move through a solution space, adjusting their positions based on their individual experiences and the achievements of their neighboring particles [4] [5]. Each particle remembers the best solution it has found so far (the local best), and the swarm as a whole remembers the best solution found by any particle (global best). This collective learning process guides the particles towards better solutions over time. The particles adjust their movement based on the influence of their local and global best solutions, aiming to converge to the optimal solution. PSO's simplicity and efficiency make it useful for solving a wide range of optimization problems, from engineering and finance to machine learning. Its inspiration from nature's cooperative behavior allows PSO to efficiently explore solution spaces and find optimal or near-optimal solutions [7].

The swarm of particles evolves in the search space by modifying their velocites and position of each particle in the next iteration are calculated by the expressions in eq 1:

$$\boldsymbol{v_i^{k+1} = wV_i^k + C_{1rand} \times (pbest_t - x_t^k) + C_{2rand} \times (gbest_t - x_t^k)} \quad (1)$$

Where

v_i^k - Current velocity of particle *i* at iteration k

w - Inertia weight

$rand$ - Random number in the range [0, 1]

C_1 and C_2 - Acceleration coefficients

x_t^k - Current position of particle *i* at iteration k

International Journal of Novel Research in Electrical and Mechanical Engineering
Vol. 11, Issue 1, pp: (95-100), Month: September 2023 - August 2024, Available at: www.noveltyjournals.com

$pbest_t$ is the best position of the current particle achieved so far, while it $gbest_t$ is the global best position achieved by all informants. The new position of each particle is given by Eq. 2.

$$x_t^{k+1} = x_t + v_t^{k+1} \tag{2}$$

Performance: The performance of Particle Swarm Optimization (PSO) is a comprehensive evaluation of how effectively it operates in solving optimization problems. Its convergence rate reflects the speed at which it approaches suitable solutions, with faster convergence indicating quicker success. The diversity of its search space exploration measures how broadly it investigates potential solutions, which can prevent it from being trapped in suboptimal outcomes. The robustness of PSO pertains to its adaptability across diverse problem types and complexities, ensuring its reliability under varying circumstances [8],[9] . Additionally, the efficiency of the algorithm is assessed by considering the computational resources it requires to arrive at solutions within a reasonable timeframe. Scalability examines how well PSO handles larger and more intricate problems without a significant decrease in performance. The algorithm's applicability across different types of optimization problems continuous, discrete, constrained, or with multiple objectives demonstrates its versatility. An important aspect is its sensitivity to parameter settings; a less sensitive algorithm is preferable. Comparative analysis against other optimization methods illuminates where PSO stands in terms of solution quality and computational efficiency. As the complexity of a problem increases, PSO's performance may be affected, prompting the examination of its suitability for challenging scenarios. Lastly, the ability of PSO to manage problems with constraints plays a pivotal role in assessing its overall performance and practical utility.

Applications: Particle Swarm Optimization (PSO) finds extensive applications across various domains, benefiting from its ability to efficiently navigate complex solution spaces. In engineering, PSO is employed in designing and optimizing structures, circuits, and mechanical systems. It aids in parameter tuning of complex models and enhances the efficiency of power systems. PSO's versatility extends to machine learning, where it refines neural network architectures, enhances feature selection, and optimizes hyper parameters, contributing to improved model performance [8]. Financial markets leverage PSO for portfolio optimization, risk management, and algorithmic trading strategies. In data mining and pattern recognition, it assists in clustering, classification, and feature extraction tasks. PSO also plays a role in image processing, aiding in image reconstruction, demonizing, and segmentation. Moreover, it finds applications in healthcare for medical image analysis, disease diagnosis, and treatment optimization. The optimization of logistics and supply chain management is another area where PSO helps in route planning, warehouse optimization, and demand forecasting [25] [26]. Additionally, in robotics, PSO contributes to trajectory planning, sensor network deployment, and swarm robotics coordination. The adaptability of PSO across these diverse domains underscores its significance as a versatile and effective optimization technique. Here’s a summary of the evolution of the PSO algorithm, including the year, authors, name of the algorithm, problem domain, and constrained handling;

TABLE 1: Evolution of PSO Algorithm

Year	Author	Name of algorithm	Problem constrained
1995	Eberhart, R.C. and Kennedy, J.	Original PSO	Optimization problems, Basic PSO without specific constraint handling mechanisms
1997	Kennedy, J., Eberhart, R.C., Shi, Y.	Constricted PSO(CPSO)	Primarily unconstrained optimization
2002	Clerc, M., Kennedy, J.	Standard PSO with Adaptive Inertia Weight	Limited constraint handling
2004	Parsopoulos, K.E., Vrahatis, M.N.	Multi-Objective PSO (MOPSO)	Limited constraint handling
2005	Stefan Janson and Martin Midden Dorf	Hierarchical Particle Swarm Optimization (HPSO)	Better solution
2006	Liang, J.J., Qu, B.Y., Suganthan, P.N.	Differential Evolution PSO (DEPSO)	Limited constraint handling
2009	George I. Evers et al	Regrouping Particle Swarm Optimization	Premature convergence

2010	Mirjalili, S., Mirjalili, S.M., Lewis, A.	Opposition-Based Learning PSO (OBLPSO)	Limited constraint handling
2014	Ochoa, G., Pelta, D.A.	Discrete PSO (DPSO)	Discrete optimization problems
2017	Parpinelli, R.S., Lopes, H.S., Ribas, F.S.	Crowding PSO (CPSO)	Multi-objective optimization
2019	Zhang, J., Zhang, Y., Liu, B	Constraint-Handling PSO with External Archive (CHPSO/EA)	Incorporates an external archive for constraint handling
2020	He, Z., Huang, Z., Li, C., Wu, Q	Opposition-based Learning Enhanced PSO (OLEPSO)	General optimization problems

2.2 Artificial Bee Colony Optimization Algorithm

Artificial Bee Colony (ABC) Optimization is an algorithm inspired by how bees find food. Bees explore solutions to a problem, share good solutions, and try new ones. This helps find the best solution over iterations, like bees finding the best food sources in nature.

Evolution: The Artificial Bee Colony Optimization algorithm was developed by Karaboga in 2005, inspired from foraging and waggles dance behavior of honey bee colonies. Since 2005, various modifications have been made in bee colony algorithm [11].

Methodology: The Artificial Bee Colony (ABC) methodology is an optimization algorithm based on the foraging behavior of honeybees. The ABC methodology is explained in a simple way. Think of the ABC as a group of bees. Each bee has an idea (a solution). They work on their ideas and tell others about them. The bee notice which ideas seem best. They keep improving their ideas over time. If they get stuck, they try something totally new. The best idea found is remembered [12]. This process goes on for a while. When it's time to stop, the best idea is the solution to the problem.

Performance: The Artificial Bee Colony (ABC) algorithm works well for problems that aren't too simple or too complex. It can find good solutions fairly quickly in many cases. However, it's important to set its settings just right. If the problem is very complicated, ABC might not work as well. Also, if the problem has many different good solutions, ABC could have some trouble finding all of them. By adjusting its settings and sometimes using it together with other methods, ABC can become even more useful for solving different kinds of problems [13].

Applications The Artificial Bee Colony (ABC) algorithm finds practical use across diverse fields by leveraging its optimization capabilities. In engineering, it aids in designing efficient structures and systems, while in data analysis, it refines the process of extracting valuable insights from large datasets. In domains like finance, ABC optimizes investment portfolios and predicts financial trends. Its role extends to improving image and signal processing outcomes, enhancing the accuracy of machine learning models, optimizing robot movements, and even managing healthcare scheduling for better patient care. Additionally, ABC contributes to supply chain efficiency, energy consumption reduction, wireless network optimization, and agricultural yield improvement through crop planting and pest control. With applications ranging from financial management and engineering to healthcare and environmental modeling, ABC showcases its adaptability and impact across numerous sectors [14].

2.3 GENETIC ALGORITHM

The genetic algorithm is an adaptive heuristic search method based on population genetics. It is a probabilistic search method inspired by the processes of natural selection and reproduction. It is used to generate solutions to optimization search problems. It belongs to the class of evolutionary algorithms that use initialization, selection, crossover, and mutation [15].

Evolution: John Holland introduced the genetic algorithm in his 1975 book ‘Adaptation in Natural and Artificial Systems’. Then John Koza used genetic algorithms in 1992 to evolve programs to perform certain tasks. He called his method "Genetic Programming" (GP). Some improvements in GA [16].

ISSN 2394-9678
International Journal of Novel Research in Electrical and Mechanical Engineering
Vol. 11, Issue 1, pp: (95-100), Month: September 2023 - August 2024, Available at: www.noveltyjournals.com

Methodology: A GA starts with a set of solutions called the population. It consists of four operators: (1) initialization, (2) selection, (3) crossover and (4) mutation. The initialization operator creates the initial population and assigns a fitness function that evaluates the fitness value. The selection operator chooses the chromosomes from the population for mating. A crossover operator is used for sharing information between two chromosomes. A mutation operator alters one or more gene values of a chromosome from its initial state. The process of evolution is repeated until the end condition of the problem is satisfied.

Performance: GA is a search technique used to find the exact or approximate solutions to an optimization problem. GA exhibits “inherent parallelism” as the evaluation of individuals within a population is conducted simultaneously. It generally finds the global optima in complex spaces; hence, GA is fast [32]. GA may be hybridized with any other search method to achieve an optimization goal.

Applications: Genetic Algorithms (GAs) find practical application in diverse fields such as engineering, where they optimize designs of structures, circuits, and systems; finance, by fine-tuning investment portfolios and predicting market trends; machine learning, to optimize model parameters and feature selection; and scheduling problems, aiding in efficient resource allocation. GAs also assist in neural network training, enabling better pattern recognition and classification; robotics, optimizing motion planning and control; and bioinformatics, contributing to DNA sequence alignment and protein structure prediction. Their ability to solve complex optimization challenges, adapt to various problem spaces, and explore solution landscapes makes GAs a valuable tool across domains [17].

3. CONCLUSION

In this critical review, we examined various optimization techniques by a comparative analysis based on evolution, methodology, performance and applications. In this paper, we find that these algorithms may be applied in various domains whether using as a direct approach or as any modified version. In future, new improved algorithms can be found for different scope areas. All the algorithms discussed in the above section are optimization problems. These are compared with each other and every algorithm has its own advantages. These algorithms, collectively known as Particle Swarm Optimization (PSO) variants, have evolved over time to address various types of optimization problems. They build upon the original PSO concept, adapting and enhancing it to improve solution quality, convergence speed, robustness, and applicability across different problem domains.

REFERENCES

[1] Muhammad Imran, Rathiah Hashim, Noor Elaiza Abd Khalid, An Overview of “Particle Swarm Optimization Variants,” Procedia Engineering, Volume 53, 2013, Pages 491-496,ISSN 1877-7058.

[2] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," 1998 IEEE International Conference on Evolutionary Computation Proceedings. Anchorage, AK, USA, 1998, pp. 69-73.

[3] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," in IEEE Transactions on Evolutionary Computation, vol. 6, no. 1, pp. 58-73, Feb. 2002.

[4] JOUR, Parsopoulos, K.E.,Vrahatis, M.N. , “Recent approaches to global optimization problems through Particle Swarm Optimization,”2002/06/01, Natural Computing.

[5] Janson, Stefan and Middendorf, Martin, and Jin, Yaochu and Johnson, Colin G. and Machado, Penousal and Marchiori, Elena and Rothlauf, Franz and Smith, George D. and Squillero, Giovanni, “A Hierarchical Particle Swarm Optimizer for Dynamic Optimization Problems", Applications of Evolutionary Computing, 2004, Springer Berlin Heidelberg", 513—524.

[6] B. Y. Qu, P. N. Suganthan and J. J. Liang, "Differential Evolution With Neighborhood Mutation for Multimodal Optimization," in IEEE Transactions on Evolutionary Computation, vol. 16, no. 5, pp. 601-614, Oct. 2012.

[7] Evers, George & Ghalia, Mounir. (2009). Regrouping Particle Swarm Optimization: A New Global Optimization Algorithm with Improved Performance Consistency across Benchmarks. International Conference on Systems, Man, and Cybernetics 2009.

International Journal of Novel Research in Electrical and Mechanical Engineering
Vol. 11, Issue 1, pp: (95-100), Month: September 2023 - August 2024, Available at: www.noveltyjournals.com

[8] Parpinelli, Rafael & Lopes, Heitor & Freitas, Alex. (2002). Data Mining With an Ant Colony Optimization Algorithm. Evolutionary Computation, IEEE Transactions on. 6. 321 - 332. 10.1109/TEVC.2002.802452.

[9] Karaboga, D., Basturk, B. (2007). "Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems". In: Melin, P., Castillo, O., Aguilar, L.T., Kacprzyk, J. , Pedrycz, W. (eds) Foundations of Fuzzy Logic and Soft Computing. IFSA 2007. Lecture Notes in Computer Science(), vol 4529.

[10] Karaboga, Dervis & Basturk, Bahriye. (2007). "Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems". 4529. 789-798.

[11] Bansal, Jagdish & Sharma, Harish & Jadon, Shimpi. (2013). "Artificial bee colony algorithm: A survey. International Journal of Advanced Intelligence Paradigms".

[12] Weifeng Gao, Sanyang Liu, Lingling Huang, "A global best artificial bee colony algorithm for global optimization",Journal of Computational and Applied Mathematics, Volume 236, Issue 11, 2012, Pages 2741-2753.

[13] G. Pampará and A. P. Engelbrecht, "Binary artificial bee colony optimization," 2011 IEEE Symposium on Swarm Intelligence, Paris, France, 2011, pp. 1-8.

[14] Ezgi Zorarpacı, Selma Ayşe Özel, "A hybrid approach of differential evolution and artificial bee colony for feature selection", Expert Systems with Applications, Volume 62, 2016, Pages 91-103, ISSN 0957-4174.

[15] Amani Saad, Salman A. Khan, Amjad Mahmood, "A multi-objective evolutionary artificial bee colony algorithm for optimizing network topology design, Swarm and Evolutionary Computation", Volume 38, 2018, Pages 187-201, ISSN 2210-6502.

[16] Lingaraj, Haldurai. (2016). "A Study on Genetic Algorithm and its Applications". International Journal of Computer Sciences and Engineering. 4. 139-143.

[17] K. F. Man, K. S. Tang and S. Kwong, "Genetic algorithms: concepts and applications in IEEE Transactions on Industrial Electronics, vol. 43, no. 5, pp. 519-534, Oct. 1996.

[18] A Lambora, K. Gupta and K. Chopra, "Genetic Algorithm- A Literature Review," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 380-384.

[19] S. A. Saleh, S. K. Ahmed and F. S. Nashat, "A Genetic Algorithm for Solving an Optimization Problem: Decision Making in Project Management," 2020 International Conference on Computer Science and Software Engineering (CSASE), Duhok, Iraq, 2020, pp. 221-225.

[20] R. K. Singh, V. K. Panchal and B. K. Singh, "A review on Genetic Algorithm and Its Applications," 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT), Bangalore, India, 2018, pp. 376-380, doi: 10.1109/ICGCIoT.2018.8753030.

[21] B. Zhao and Z. Xiong, "Research and Application of Genetic Algorithm Based on Variable Crossover Probability," 2017 International Conference on Virtual Reality and Visualization (ICVRV), Zhengzhou, China, 2017, pp. 156-159, doi: 10.1109/ICVRV.2017.00039.

[22] Y. Rodkaew, "A Genetic Algorithm as a Classifier," 2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Chiang Mai, Thailand, 2021, pp. 254-257.

[23] F. Li and T. Zhang, "Study on Genetic Algorithm Based on Schema Mutation and Its Performance Analysis," 2009 Second International Symposium on Electronic Commerce and Security, Nanchang, China, 2009, pp. 548-551, doi: 10.1109/ISECS.2009.129.

[24] Tyagi, Kanika & Seth,. (2015). A Comparative Analysis of Optimization Techniques. International Journal of Computer Applications. 131. 6-12. 10.5120/ijca2015907399.

[25] S. Kannan, S.Mary Raja Slochanal, P. Subbaraj, Narayana Prasad Padhy, "Application of particle swarm optimization technique and its variants to generation expansion planning problem", Electric Power Systems Research, Volume 70, Issue 3, 2004, Pages 203-210, ISSN 0378-7796.