Grenze International Journal of Engineering and Technology, June Issue

Recent Advancements in Metaheuristic-based Approaches for Solving Multi-Objective Optimization Problems

Himpriya Kumari[1] and Ashok Pal[2]
[1-2]Department of Mathematics, Chandigarh University, Mohali, Punjab, India
Email: himpriya2003@gmail.com, ashokpmaths.iitr@gmail.com

***Abstract*— The present manuscript investigates the key notions of metaheuristics for multi-objective optimisation problems. It comprehensively analyses the foundations of multi-objective optimisation, the need to optimise multiple conflicting objectives, the constraints of traditional methods, and the role of metaheuristics in this context. In addition to these, the paper also provides an exhaustive literature review encompassing major facets of multi-objective optimisation, relevant interludes and challenges, and the rationale for the existing review. The paper aims to cover classical multi-objective optimisation techniques and their role in tackling complex real-world problems. Techniques such as PSO, NSGA-II, SPEA2, MOEA/D and many others are discussed in the paper. These approaches mentioned in the literature survey are scrutinised intensely. Recent advancements, such as hybridisation strategies, such as Hybrid EAs, adaptive and auto-tuning algorithms and parallel and distributed metaheuristics, successfully form a part of the paper. The paper also includes an important case study on an improved NSGA-II algorithm based on reinforcement learning for aircraft assembly line balancing and scheduling. This paper explores the diverse applications in engineering design, environmental management and energy optimisation, supply chain and logistics, and healthcare and medical treatment planning. Overall, the paper aims to revisit the pre-existing data, study all techniques and present future scopes.**

***Index Terms*—Multi-objective, Metaheuristics, Evolutionary Algorithm, Pareto optimality, Hybrid algorithms.**

I. INTRODUCTION

Many optimisation problems in engineering, science, finance, economics, and logistics typically have multiple conflicting objectives in the real world. Such issues are referred to as multiobjective problems [8]. Instead of being single-objective, these problems usually present several competing criteria or goals that must be addressed simultaneously. The concept of "optimum" shifts when there are multiple objective functions because, unlike global optimisation, which focuses on finding a single solution, MOPs aim to find good trade-offs. The most widely accepted definition of optimum was first put forth by Francis Ysidro Edgeworth in 1881. Vilfredo Pareto went on to generalise this idea in 1896. While some authors refer to this concept as Edgeworth-Pareto optimum, we will use the most widely used term: Pareto optimal [20].
Since no single solution can simultaneously optimise all objective functions, MOPs differ from single-objective optimisation.

Grenze ID: 01.GIJET.12.2.337_24

The conflicting nature of objective functions (improving one often results in the deterioration of another, i.e., improving one makes others worse) makes solving MOPs a difficult task. Furthermore, the set of Pareto-optimal solutions that are obtained from solving MOPs has key characteristics like the following: every solution generates trade-offs between the various objectives, and no solution is inferior to the others [8]. In today's generation, it is very important to realise why metaheuristics were required in the first place. We undoubtedly had some conventional algorithms which were developed to solve MOPs like the goal programming, the hierarchical optimisation algorithms, the ε-constraint technique, and the objective weighted method. One particular nondominated solution is produced at a time by the methods used to convert the vector of objectives in an optimisation problem with multiple objectives (MOP) into a single objective. However, there are several serious limitations with traditional algorithms. They necessitate a carefully chosen starting point, and the goal function and limitations must be differentiable and frequently require costly calculations for the Jacobian and/or Hessian. These techniques also have trouble when there is concavity, uncertainty, ambiguity, or disturbances in the exploratory space of the objectives. It is imperative to recognise that these conventional methods are not always successful in optimising complex MOPs [1]. To improve the effectiveness and relevance of optimisation techniques in addressing complex MOPs, we must seek novel solutions that get beyond these constraints [8]. Intelligent swarm algorithms and evolutionary algorithms (EAs) are two examples of these metaheuristic techniques. The Pareto-optimal solutions of MOPs, which are challenging for conventional algorithms to obtain in the allotted time, can be obtained by the majority of metaheuristic algorithms [8]. Vilfredo Pareto first proposed MOO, which is now an essential resource for choice-making in many engineering domains where the best course of action should be chosen between competing goals. For the MOO, several approaches have been put forth. Schaffer came up with the novel concept of using evolutionary optimisation algorithms for multi-objective optimisation in 1984 [9]. The Travelling Salesman Problem (TSP) is one of the NP-hard combinatorial optimisation problems that use metaheuristics. These techniques make it easier to handle competing goals, like maximising efficiency and minimising costs, in a balanced manner. Metaheuristics are used for tasks involving resource collection, scheduling, and risk assessment in a variety of industries, including logistics, healthcare, energy, telecommunications, and finance [10], [11].

Multi-objective optimisation entails optimising multiple objectives at once, as the name implies. When the goals clash, that is, when one objective function's optimal solution differs from the other, the problem becomes difficult. When such problems are solved, whether or not there are constraints, a collection of trade-off optimal solutions—also referred to as Pareto-optimal solutions—is produced. Let us discuss some basic concepts related to MOP [1]. The methodical process of finding the best possible solution to a problem among a set of feasible alternatives means maximising or minimising an objective function subject to constraints. It is formally represented as:

$$\text{Optimise } f(x), \text{ subject to } x \in S$$

Where f(x) is the objective function, and S is the feasible solution space determined by the constraints [16].

The general Multiobjective Optimisation Problem (MOP) can be formally defined as [20]: Find the vector $x^* = [x_1^*, x_2^*, \ldots, x_n^*]^T$ which will satisfy the m inequality constraints:

$$g_i(x) \leq 0, \quad i = 1, 2, \ldots, m$$

The p equality constraints:

$$h_i(x) = 0, \quad i = 1, 2, \ldots, p$$

and will optimise the vector function:

$$f(x) = [f_1(x), f_2(x), \ldots, f_k(x)]^T$$

MOPs produce a collection of Pareto-optimal solutions rather than a single ideal one, since no objective can be enhanced without compromising another. A solution is said to be Pareto Optimal if there exists no feasible vector of decision variables which would decrease some criterion without causing a simultaneous increase in at least one other criterion [4].

But it rarely gives a single solution, but rather a series of solutions called the Pareto optimal set. The vectors corresponding to the solutions included in the Pareto optimal set are called nondominated. The plot of the objective functions whose nondominated vectors are in the Pareto optimal set is called the Pareto front [20]. The aim is to approximate the Pareto front, which represents the trade-offs among the objectives. Here, the trade-offs mean that everything cannot be improved at once, so it requires sacrificing one benefit to gain another. This is why the Pareto front is important, as it is the set of solutions where no objective can be improved without sacrificing at least one other [5].

Metaheuristics—Metaheuristics are a group of approximate optimisation techniques. A metaheuristic is formally defined as a set of algorithmic concepts that can be used to represent heuristic methods applicable to a wide range of problems [11].

Benchmarking and Performance Metrics—Standardised benchmark issues and clearly stated performance measures are crucial for evaluating metaheuristic algorithms for multi-objective optimisation. These instruments guarantee that algorithm evaluations are impartial, repeatable, and educational. They also provide light on important topics like the variety of solutions throughout the front and convergence to the actual Pareto front [15].

A. Benchmark Test Suites

Several artificial standard suites have been designed to systematically experiment with the abilities of multi-objective optimisation algorithms.

- ZDT Suite (Zitzler–Deb–Thiele): One of the most widely used test sets, comprising six problems (ZDT1 to ZDT6) that vary in their objective function shapes, continuity, and complexity [10].
- DTLZ Suite (Deb–Thiele–Laumanns–Zitzler): This suite has problems like DTLZ1 to DTLZ7 that can be adjusted for different numbers of objectives and decision variables. DTLZ problems check how well an algorithm works when more objectives are added, making them ideal for studying many-objective optimisation [15].
- Walking Fish Group (WFG) Suite: These problems offer greater authority over the method, influence, and challenges. They are especially helpful for evaluating an algorithm's generalisation and adaptability across a variety of landscape characteristics [2].

B. Performance Metrics

Several quantitative indicators are employed to evaluate the calibre of solutions produced by MOO algorithms. These are frequently divided into combined, variance, and integration indicators [20].

- Hypervolume (HV): One of the most widely used and complete indicators, hypervolume calculates the extent of the space of objectives that the obtained Pareto front covers with respect to the reference point [8].
- Generational Distance (GD) and Inverted Generational Distance (IGD): With an emphasis on convergence, GD calculates the average Euclidean distance between ideas in the obtained front and the actual Pareto front. IGD captures both convergence and diversity by measuring the difference between the obtained front and the true front [6].
- Spread (Δ): This metric assesses the allocation of resolutions across the Pareto front.
- The Epsilon Indicator: Also known as the ε-indicator, it quantifies the amount of translation required to make the Pareto front produced by one algorithm dominate the front of another [16].

II. LITERATURE STUDY

A complete set of resources is offered in the book authored by Fogel, Back and Michalewicz (1997) [22], who have written a very thorough and detailed overview of evolutionary computation. The content of this book established the foundations from which evolutionary and metaheuristic methodologies evolved and are now in use for performing multi-objective optimisation [22].

An organised and thorough examination of nonlinear multi-objective optimisation can be found in the book written by Miettinen (1999) [16]. In this text, the concept of Pareto optima, an assortment of trade-off analysis techniques, and mathematical methods commonly used to solve multi-objective problems are presented. This section is based on an established theoretical foundation, which also addresses why traditional techniques for solving optimisation problems have limited applicability for problems with multiple conflicting objectives [16].

Deb (2001) [10], the creator of evolutionary algorithms for multiple objectives, proposed using an evolutionary algorithmic approach to dominance by creating early types of dominance-based approaches. This early work established the groundwork for significant algorithms that are widely used today, such as NSGA-II, and represents a major reference in the multi-objective evolutionary optimisation field [10].

Talbi (2002) [17] provides an in-depth taxonomy of hybrid metaheuristics, which he uses to classify the hybrid approaches based on how they cooperate and integrate. The classification system provided a basis for the systematic development of hybrid optimisation algorithms for complex and multi-objective problems [17].

Dynamic multi-objective optimisation problems (MOOPs), where objectives or constraints change over time, were studied by Farina, Deb and Amato (2004) [19] to identify suitable benchmark test problems and approximation methods, and to describe the necessity of developing adaptive meta-heuristics that can effectively solve non-stationary problems [19].

The survey on the use of multi-objective optimisation methods in Engineering Design has been conducted by Marler & Arora (2004) [21]. Their comparative analysis revealed that classical approaches to solving MOOPs have significant limitations. In contrast, Evolutionary Multi-Objective and Advanced Meta-Heuristic optimisation techniques present an effective approach to solving real-world engineering design optimisation problems [21].

The use of metaheuristics for multi-objective production scheduling problems has been explored extensively by Loukil, Teghem, and Tuyttens (2005) [7]. They show that Evolutionary Methods are capable of generating trade-off solutions to conflicting scheduling objectives in manufacturing environments [7].

The Comprehensive Learning Particle Swarm Optimiser (CLPSO) is a Global Optimisation technique proposed by Liang et al. (2006) [23]. While CLPSO was originally developed as a Single-Objective Optimiser, its contribution to the future development of Multi-Objective Synchronous Particle Swarm Optimisers has been highlighted by enhancing exploration and convergence capabilities [23].

Coello, Veldhuizen, and Lamont (2007) [15] offer a comprehensive overview of the use of evolutionary algorithms in addressing multi-objective issues. This book establishes the foundational concepts of dominance as well as methods for preserving diversity and measuring performance, and thus serves as a key reference source for people researching and applying multi-objective evolutionary optimisation techniques [15].

In 2008, Carlos A Coello Coello, Lamont, and Van Veldhuizen [20] wrote an authoritative resource on evolutionary algorithms used to solve multi-objective optimisation problems. The second edition updates these concepts with more detail about the algorithms that are frequently used in practice: NSGA, SPEA, etc. The book also contains many other evolutionary algorithm-related topics, such as how to evaluate performance using a variety of performance metrics and measures. It serves as an excellent introduction to the use of multi-objective metaheuristics throughout diverse engineering applications and computational sciences [20].

Talbi (2009) [11] has produced a fundamental work that will serve as a practical and theoretical reference for the implementation and design of metaheuristic algorithms. Talbi's contribution to the literature falls within the intersection of the practical and theoretical aspects of the use of metaheuristic algorithms as reusable frameworks for tackling a diverse range of multi-objective optimisation problems [11].

Ong, Lim, and Chen (2010) [18] examine the development of memetic computation, an approach that merges the use of evolutionary algorithms with local search techniques. Their findings demonstrate the roles of memetic computation in improving solution quality and convergence speed in the context of complex multi-objective optimisation problems [18].

Talbi, Basseur, Nebro, and Alba (2011) [2] have expanded on the development of non-standard multifactor metaheuristics, which include cooperative, hybrid, and parallel algorithms. The results of their experiments show that these types of algorithms provide improved ability to solve large-scale optimisations and improve overall robustness and scalability compared to traditional methods [2].

Xing, Chen, and Yang (2011) [24] developed a hybrid bat algorithm based on a differential evolution algorithm, and their findings confirm the performance improvements of hybridisation. Hybridisation is used in the subsequent development of hybrid metaheuristic algorithms intended for multi-objective problems [24].

Yalaoui & Berrichi (2013) [6] developed a generation of multi-objective meta-heuristics used to solve joint scheduling problems related to Manufacturing and Maintenance (MM) simultaneously. They emphasise how Evolutionary Techniques can handle a large number of industrial constraints while optimising conflicting operational objectives.

In Kotthoff's survey (2014) [13], he reviews algorithm selection methods used for Combinatorial Optimisation problems. He discusses how certain characteristics of a problem will dictate which algorithms should be selected, and how Machine Learning approaches to optimisation can be used to develop a method for algorithm selection in the future [13].

In 2015, members of the research team of Ishibuchi, Masuda, Tanigaki and Nojima [12] developed a method of calculating the distance metrics of generational distance and inverted generational distance. This new method is designed to improve the accuracy and reliability of multi-objective optimisation evaluation methods [12].

According to Chakraborty and Saha (2015) [26], a Binary Glowworm Swarm Optimisation Algorithm has been developed to select and classify features through the application of Swarm-based Metaheuristic methods in Multi-Objective Decision Making [26].

In their paper, Bertsimas & Dunn (2017) [14] introduced methods to create optimal classification trees through mathematical optimisation methods. Combining mathematics optimisation and machine learning produces new opportunities for researchers, which was the idea behind developing learning-guided metaheuristics [14].

The multi-objective metaheuristic research by Liu et al. (2020) [1] offered a wide-ranging study on the different types of algorithms used to solve discrete optimisation problems. The study also listed many issues associated with the scalability and performance of algorithms found within discrete search spaces [1].

A hybrid metaheuristic algorithm for multi-objective optimisation was presented by Farag et al. (2020) [8]. The results of their experimental studies produced better convergence and diversity than previous solutions, thus validating the effectiveness of using hybrid strategies [8].

The novel optimisation algorithm proposed by Sharifi et al. (2021) [9] has been created to increase the efficiency of solving multi-objective optimisation problems. The performance of this algorithm is considered competitive to established methods in terms of the results achieved on a variety of benchmark test functions [9].

A survey of metaheuristic approaches to multi-objective stochastic optimisation problems was conducted by Gannouni and Ellaia (2021) [4], which addressed pressing challenges associated with managing uncertainties, robustness, and adaptability, and called for further exploration of adaptive approaches to stochastic optimisation [4].

Zhang et al. (2024) [5] provided an overview of the recent developments in employing machine learning-based approaches to enhance multi-objective evolutionary algorithms (MOEAs) in the field of scheduling problems. This research showed the significant effects of the learning-based components on adaptability and decision-making capabilities when utilising a complex optimisation information environment [5].

Zhang et al. (2024) [3] report the impact of metaheuristic approaches on multi-objective scheduling issues in the context of Industry 4.0 & 5.0. The study also identifies challenges and future research opportunities associated with the adoption of intelligent manufacturing technologies [3].

Zhang et al. (2025) [3] include an additional discussion focused on industry impact, again highlighting how adaptively intelligent metaheuristic approaches can be leveraged for optimising industrial processes through new methodologies within Industry 4.0 & 5.0 [3].

Wen et al. (2025) [25] propose an innovative application of the NSGA-II framework combined with reinforcement learning in optimising aircraft production processes using assembly line methods. Their research indicated improved convergence and enhanced solution quality for real-world industrial scenarios compared to previously published studies [25].

III. Methodologies

This section includes a description of some key metaheuristics [5].

A. Evolutionary Algorithms

There are some types/examples of these algorithms that are widely used in different sectors.

- Genetic Algorithms (GAs)—Genetic algorithms are adaptive heuristic search algorithms premised on the evolutionary ideas of natural selection and genetics [22].
- Evolution Strategies (ES)—Evolution strategies are numerical optimisation methods inspired by the principles of natural evolution and focused on the adaptation of mutation parameters [22].
- Genetic Programming (GP)—Genetic programming is an evolutionary algorithm that automatically constructs computer programs to solve problems by simulating natural selection [5].
- Differential Evolution (DE)—Differential Evolution is a simple yet powerful evolutionary algorithm for global optimisation over continuous spaces. DE strategies are often written as DE/x/y/z, where x = base vector (rand, best, current), y = number of difference vectors used and z = crossover type (bin for binomial, exp for exponential) [10].
- Memetic Algorithms (MAs)—Memetic algorithms are hybrid evolutionary algorithms that apply local search techniques to refine individuals within the population [20].

B. Swarm-Based Algorithms

There are some types/examples of these algorithms that are widely used in different sectors [7].

- Particle Swarm Optimisation (PSO)—Particle swarm optimisation is a population-based stochastic optimisation technique inspired by the social behaviour of bird flocking or fish schooling [23].
- Ant Colony Optimisation (ACO)—Ant colony optimisation takes inspiration from the foraging behaviour of ants and applies it to solve discrete optimisation problems [21].
- Artificial Bee Colony (ABC)—Artificial bee colony algorithm is a swarm intelligence technique inspired by the foraging behaviour of honey bees [23].

- Firefly Algorithm (FA)—The firefly algorithm is based on the flashing patterns and behaviour of fireflies, and it has been successfully applied to continuous optimisation problems [8].
- Bat Algorithm (BA)—The bat algorithm uses the echolocation behaviour of bats and integrates frequency tuning with automatic zooming to balance exploration and exploitation [24].

IV. CASE STUDY

This section discusses "An Improved NSGA-II Algorithm Based on Reinforcement Learning for Aircraft Assembly Line Balancing and Scheduling" [10], [25].

A. Objective

This study offered solutions to typical problems of how the aircraft moving assembly line can work more effectively. It concentrated on the integration of line balancing in scheduling. The primary objective is to flatten the work across all stations, reduce cycle times to the lowest possible, and cut down on the number of workers needed, all while respecting task precedence relationships and supply-side constraints [25].

B. Problem Description

The problem has the full name: Aircraft Moving Assembly Line Integration Optimisation Problem (AMALIOP). It consists of assigning each task to a fixed number of workstations, then scheduling the start times for all of them. The objective function minimises cycle time, distributes workloads more evenly, and reduces the number of necessary operators [25].

C. Methodology

The authors assembled an integer linear programming model that provides AMALIOP with a suitable mathematical expression. To manage the issue, they developed a new algorithm called RLINSGA-II (A Bayesian reinforcement learning-improved NSGA-II algorithm). It extends the conventional NSGA-II and introduces reinforcement learning to enhance its strength [25].

D. RLINSGA-II Algorithm

The RLINSGA-II version employs Bayesian reinforcement learning to narrow down the process of search considerably. It introduces a new method of prioritising population and sampling. The entire process still relies on non-dominated sorting. The selection pressure is tweaked by the reinforcement learning as it proceeds, which enhances the quality of offspring directly at the crossover stage. They experimented on 5 problem cases of varying scale based on real-life information from aircraft final assembly lines [25].

E. Key Contributions

One of the major contributions is the combination of line balancing and scheduling in one unified framework. The authors devised a special decoding system suited to aircraft assembly lines, incorporating reinforcement learning as a part of NSGA-II to support dynamic modification of evolutionary steps. Compared to conventional multi-objective approaches, RLINSGA-II works better: it further decreases the cycle time and creates a more equal distribution of workloads [25].

F. Experimental Setup and Results

The experiments established five scenarios based on real-life cases with assembly tasks varying between ten and ninety. The results show that RLINSGA-II produces better Pareto fronts and converges faster than standard benchmarks. Balance is improved in factors like cycle time, equal distribution of workloads, and labour requirements [25].

The graph illustrates a Pareto Front Visualisation where the two conflicting objectives in a multi-objective optimisation problem are pointed out, which is the balancing of aircraft assembly lines. The X-axis (Cycle Time in minutes) is an indicator of the time required to complete one assembly work cycle. The Y-axis (Workload Smoothness Index) measures the distribution of workload evenly across the assembly line stations. The red dots indicate the Pareto optimal solutions found by the optimisation algorithm. The blue line depicts the approximate Pareto front, illustrating the trade-offs between the cycle time and the smoothness of the workload. The Pareto front curve shows the optimal compromises between minimisation of cycle time and the maximisation of workload smoothness, allowing decision-makers to choose based on their priorities [25].

The convergence curve graph scrutinises whether the RLINSGA-II algorithm is making strides in optimisation as compared to the regular NSGA-II over a complete evolution of 200 generations. The purple line (RLINSGA-

II) proceeds ahead at generation 10 with a best value of 56, then continues to steadily improve until generation 200, reaching 44. The blue line (NSGA-II) starts at 56 and reaches 46 by generation 200 through more gradual improvement. Every step is ahead for RLINSGA-II, indicating faster and firmer convergence to better solutions. This confirms the benefit of incorporating reinforcement learning into NSGA-II for multi-objective aircraft assembly line problems [25].

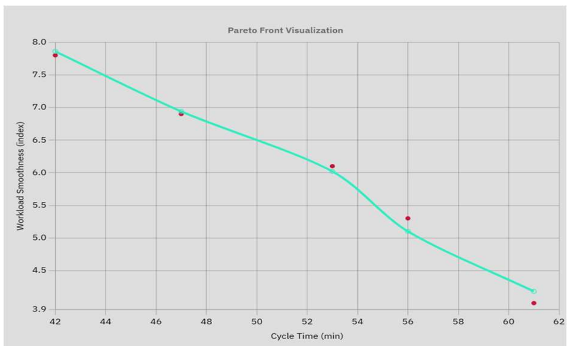

Figure 1. Pareto Front Visualisation

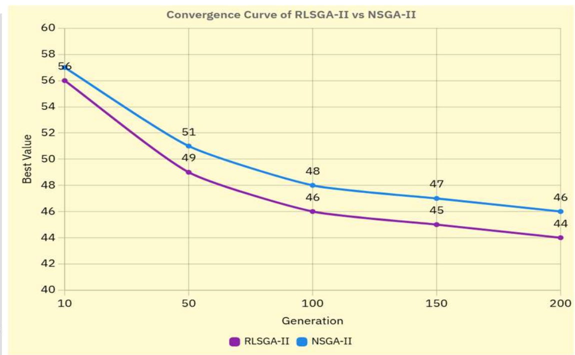

Figure 2. Convergence Curve of RLINSGA-II vs NSGA-II

V. Conclusion

The paper demonstrates how hard it might be to optimise aircraft assembly lines, where balancing and scheduling confuse themselves in complex patterns. With the reinforcement learning incorporated in their improved NSGA-II, a vigorous step forward is made, addressing these multi-objective issues by varying the search intelligence where necessary.

The metaheuristic algorithms have grown more powerful and versatile in the last few decades to solve multi-objective optimisation problems (MOPs) in diverse areas. As compared to conventional optimisation techniques, which are often afflicted by non-convexity, gaps and huge search spaces, metaheuristics offer powerful search capabilities which are highly appropriate in the complexity of real-world problems. Evolutionary algorithms have achieved good integration and diversity in approximating the Pareto front. Self-organising behaviour and decentralising search methods of swarm-based algorithms, including PSO, ACO, and ABC, have shown specific promise. In addition, domain-specific knowledge or the use of complementary strategies has enabled mixed and recent metaheuristic versions to increase performance further, allowing a better trade-off between exploration and exploitation.

Despite the great improvement, issues still occur, especially in dynamic, high-dimensional, or constrained multi-objective spaces. The importance of the choice of algorithm and optimisation of parameters to specific applications is highlighted by the fact that there is no one algorithm that can always perform better than others under all possible types of issues. Further research is expected to be dominated by more flexible, combined and learning-enhanced metaheuristics that are capable of addressing large-scale, real-time multi-objective optimisation problems. Overall, metaheuristic techniques are needed to solve MOPs both conceptually and practically. Their further evolution, enhancement of computational power, and integration with machine learning are likely to characterise the next stage of smart decision-making and optimisation.

Acknowledgement

We would like to express our sincere gratitude to our institution and mentors for their constant guidance and support throughout this research work. We also acknowledge all the researchers and authors whose valuable contributions have significantly aided in the completion of this paper.

References

[1] Liu, Q., Li, X., Liu, H., & Guo, Z. (2020). Multi-objective metaheuristics for discrete optimisation problems: A review of the state-of-the-art. Applied Soft Computing, 93, 106382. https://doi.org/10.1016/j.asoc.2020.106382

[2] Talbi, E., Basseur, M., Nebro, A. J., & Alba, E. (2011). Multi-objective optimisation using metaheuristics: Non-standard algorithms. International Transactions in Operational Research, 19(1), 283–305. https://doi.org/10.1111/j.1475-3995.2011.00808.x

[3] Zhang, W., Bao, X., Hao, X., & Gen, M. (2025). Metaheuristics for multi-objective scheduling problems in industry 4.0 and 5.0: a state-of-the-art survey. Frontiers in Industrial Engineering, 3. https://doi.org/10.3389/fieng.2025.1540022
[4] Gannouni, A., & Ellaia, R. (2021). A Survey on Metaheuristics for Solving Stochastic Multi-Objective Optimisation Problems. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3995749
[5] Zhang, W., Xiao, G., Gen, M., Geng, H., Wang, X., Deng, M., & Zhang, G. (2024). Enhancing multi-objective evolutionary algorithms with machine learning for scheduling problems: Recent advances and survey. Frontiers in Industrial Engineering. https://doi.org/10.3389/fieng.2024.1337174
[6] Berrichi, A., & Yalaoui, F. (2013). Multi-Objective Metaheuristics for the Joint Scheduling of Production and Maintenance. Metaheuristics for Production Scheduling, 283–313. https://doi.org/10.1002/9781118731598.ch11
[7] Loukil, T., Teghem, J., & Tuyttens, D. (2005). Solving multi-objective production scheduling problems using metaheuristics. European Journal of Operational Research, 161(1), 42–61. https://doi.org/10.1016/j.ejor.2003.08.029
[8] Farag, M., Mousa, A., El-Shorbagy, M., & El-Desoky, I. (2020). A New Hybrid Metaheuristic Algorithm for Multiobjective Optimisation Problems. International Journal of Computational Intelligence Systems, 13(1), 920. https://doi.org/10.2991/ijcis.d.200618.001
[9] Sharifi, M. R., Akbarifard, S., Qaderi, K., & Madadi, M. R. (2021). A new optimisation algorithm to solve multi-objective problems. Scientific Reports, 11(1). https://doi.org/10.1038/s41598-021-99617-x
[10] Deb, K. (2001). Multi-objective Optimisation Using Evolutionary Algorithms. Wiley.
[11] Talbi, E.-G. (2009). Metaheuristics: From Design to Implementation. Wiley.
[12] Ishibuchi, H., Masuda, H., Tanigaki, Y., & Nojima, Y. (2015). Modified Distance Calculation in Generational Distance and Inverted Generational Distance. EMO 2015, LNCS 9019, 110–125. https://doi.org/10.1007/978-3-319-15892-1_8
[13] Kotthoff, L. (2014). Algorithm selection for combinatorial search problems: A survey. AI Magazine, 35(3), 48–60.
[14] Bertsimas, D., & Dunn, J. (2017). Optimal classification trees. Machine Learning, 106(7), 1039–1082.
[15] Coello, C. A. C., Veldhuizen, D. A. V., & Lamont, G. B. (2007). Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd ed. Springer. https://doi.org/10.1007/978-0-387-36797-2
[16] Miettinen, K. (1999). Nonlinear Multiobjective Optimisation. Kluwer Academic Publishers.
[17] Talbi, E. G. (2002). A taxonomy of hybrid metaheuristics. Journal of Heuristics, 8(5), 541–564.
[18] Ong, Y. S., Lim, M. H., & Chen, X. (2010). Research frontier: Memetic computation—Past, present & future. IEEE Computational Intelligence Magazine, 5(2), 24–36.
[19] Farina, M., Deb, K., & Amato, P. (2004). Dynamic multiobjective optimisation problems: Test cases, approximations, and applications. IEEE Transactions on Evolutionary Computation, 8(5), 425–442.
[20] Coello Coello, C. A., Lamont, G. B., & Van Veldhuizen, D. A. (2008). Evolutionary Algorithms for Solving Multi-Objective Problems, 2nd ed. Springer. ISBN 978-0-387-33254-3.
[21] Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimisation methods for engineering. Structural and Multidisciplinary Optimisation, 26(6), 369–395.
[22] Back, T., Fogel, D. B., & Michalewicz, Z. (1997). Handbook of Evolutionary Computation. Oxford University Press.
[23] Liang, J. J., Qin, A. K., Suganthan, P. N., & Baskar, S. (2006). Comprehensive learning particle swarm optimiser for global optimisation of multimodal functions. IEEE Transactions on Evolutionary Computation, 10(3), 281–295.
[24] Xing, L., Chen, Y. W., & Yang, S. Y. (2011). Hybrid bat algorithm with differential evolution. Journal of Applied Mathematics.
[25] Wen, X., Zhang, X., Li, H., Ji, S., Wang, H., Ye, G., Xing, H., & Liu, S. (2025). An improved NSGA-II algorithm based on reinforcement learning for aircraft moving assembly line integration optimisation problem. Swarm and Evolutionary Computation, 94, 101911. https://doi.org/10.1016/j.swevo.2025.101911
[26] Chakraborty, A., & Saha, S. (2015). Binary glowworm swarm optimisation for feature selection and classification. In Advances in Swarm and Computational Intelligence.