International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 OPEN ACCESS
Available Online at : www.ijsrcseit.com
doi : https://doi.org/10.32628/CSEIT2390523

Evaluating NSGA-III : A Comprehensive Study on Multi-objective Optimization

Ms. Shilpi Jain[1], Prof Kamlesh Kumar Dubey[2], Dr. Madhur Jain[3], Divyansh Rampal[4]

[1]Assistant Professor, Department of Mathematics, ARSD College, University of Delhi, Delhi, India
[2] Professor, Department of Applied Science and Humanities, Invertis University, Bareilly, India
[3]Assistant Professor, Department of Information Technology, Bhagwan Parshuram Institute of Technology, Delhi, India
[4] Student, Department of Information Technology, Bhagwan Parshuram Institute of technology, Delhi, India

ARTICLEINFO

Article History:

Accepted: 07 Sep 2023
Published: 25 Sep 2023

Publication Issue
Volume 9, Issue 5
September-October-2023

Page Number
146-150

ABSTRACT

This research paper provides a comprehensive analysis of NSGA-III, a state-of-the-art multi-objective optimization algorithm. We delve into its principles, advancements, applications, and comparative studies to showcase its effectiveness and limitations.

Keywords: Multi-objective Optimization, NSGA-III

I. INTRODUCTION

Multi-objective optimisation problems, also known as multi-objective optimisation (MOO) or multi-criteria optimisation, are mathematical optimisation problems in which numerous competing objectives must be optimised at the same time. NSGA-III (Non-dominated Sorting Genetic Algorithm III) has emerged as a viable option among evolutionary algorithms for solving multi-objective optimisation problems. This paper seeks to offer a full overview of the NSGA-III's inner workings, current improvements, real-world applications, and comparative performance. [1]

Background

Multi-objective optimization is a fundamental field of study that addresses problems where the goal is to simultaneously optimize multiple objectives, even when these objectives may conflict with each other.

Instead of seeking a single best solution, multi-objective optimization aims to identify a set of solutions that represent diverse trade-offs among these objectives. These special solutions are known as Pareto-optimal solutions, and they form a Pareto front or Pareto frontier, showcasing the trade-off possibilities.

We delve into the principles of multi-objective optimization and focus on the application of the Non-dominated Sorting Genetic Algorithm III (NSGA-III) to solve such problems. Our investigation centres on its effectiveness, strengths, and limitations in finding Pareto-optimal solutions for complex multi-objective optimization tasks. [2]

Consider a simplified multi-objective optimization problem with two objectives: cost and performance. Imagine a scenario where a project manager is responsible for selecting a computer for a company's new software development team. The objective is to find a computer that strikes a balance between cost and performance effectively. For this scenario, we define the following parameters:

- C: Represents the cost of the computer, measured in rupees.
- P: Represents the performance score of the computer, which encompasses factors like processing speed, RAM, and graphics capabilities.

The objectives are outlined as follows:

- Objective 1: Minimize Cost (C)
- Objective 2: Maximize Performance (P)

These objectives set the stage for our research paper, where we explore the capabilities of NSGA-III and analyse its performance in solving multi-objective optimization problems, shedding light on its strengths, weaknesses, and areas of application.

II. NSGA-III Algorithm

The Non-dominated Sorting Genetic approach III (NSGA-III) is an advanced evolutionary method for addressing multi-objective optimisation problems (MOOPs). NSGA-III is built on the success of its predecessors, NSGA-I and NSGA-II, and is well-suited for MOOPs with multiple objectives. Here's a quick rundown of the NSGA-III algorithm:

Initialization: Let N be the population size. Create an initial population of solutions: $P = \{S_1, S_2, ..., S_N\}$, where each S_i represents a solution.

Non-dominated Sorting: Determine the Pareto fronts and assign a rank to each solution based on dominance. P_k represents the set of solutions in Pareto front k, where k = 1, 2, ..., K (with K being the number of Pareto fronts). Rank each solution S_i with Rank (S_i).

Crowding Distance Assignment: Calculate the crowding distance for each solution in each Pareto front to maintain diversity. $D(S_i)$ represents the crowding distance of solution S_i.

Selection: Select solutions for the next generation based on Pareto ranking and crowding distance. Solutions in less crowded regions and from lower Pareto fronts are preferred.

Crossover: Combine solutions using crossover operations. For example, if $\mathbf{S_i}$ and $\mathbf{S_j}$ are selected for crossover, create new solutions $\mathbf{S_{new\_i}}$ and $\mathbf{S_{new\_j}}$.

Mutation: Apply mutation to some offspring solutions, introducing random changes.

Replacement: Combine offspring solutions with the current population. Select solutions for the next generation using a selection mechanism that considers both Pareto ranking and crowding distance.

Termination: Repeat the above steps for multiple generations or until a termination criterion is met (e.g., a maximum number of generations).

It not only provides a systematic way to find Pareto-optimal solutions that represent trade-offs among multiple conflicting objectives but also offers improved handling of problems with a large number of objectives. Its innovative use of reference points, non-dominated sorting, and crowding distance allows it to maintain diversity and converge efficiently. [3]

Comparative Analysis: NSGA-II vs. NSGA-III in Multi-Objective Optimization

The overall behaviour of the algorithms NSGA-II and NSGA-III is similar. Both methods begin with a random population and each answer reflects a realistic scaling strategy. These algorithms utilise the same crossover and mutation operators in each generation to build an offspring population from the present population. However, the selection technique used to choose which solutions from the combined current and offspring population would comprise the new population for the future generation differs across these algorithms. Let us take a look at the comparison table to highlight the key differences between the two algorithms.

NSGA-II	NSGA-III
It handles multiple objectives effectively.	It is specifically designed for many objectives.
It uses linearly spaced reference points.	It uses a grid-based approach for partitioning the objective space.
It scales well with a moderate number of objectives and solutions.	It is designed to handle a larger number of objectives and solutions effectively.
It has limited control over convergence.	It has improved control over convergence and diversity.
It is moderately complex.	It is complex due to reference point computation.
It is sensitive to parameter settings.	It has better parameter robustness with reference points.
[4], [5], [6]	[3], [7], [8]

The NSGA-II selection procedure takes into account the nondomination level of the solutions in the combined population first, followed by the crowding distance of these solutions. The crowding distance measures the distance between a solution and its neighbours. The approach then focuses on selecting nondominated alternatives with bigger crowding distances. As a result, the method encourages the selection of varied nondominated solutions. This procedure, however, does not ensure the selection of evenly distributed nondominated solutions.

In contrast to NSGA-II, the selection procedure in NSGA-III evaluates the nondomination level of the solutions in the combined population first, followed by the linkage of these solutions to the reference points. In this sense, the procedure makes use of a collection of widely scattered reference points (i.e., a set of widely and uniformly distributed reference points). The procedure then prioritises the selection of nondominated solutions linked with each of these reference locations. Thus, to retain the diversity and dispersion of the new population, this method encourages the selection of varied and well-distributed nondominated solutions. [9]

Applications

NSGA-III algorithms have been widely used to solve optimisation issues in engineering design. These issues major revolve around structural and system design. It enables engineers to optimise many objectives at the same time, such as cost, weight, and performance, while considering numerous restrictions.

In the financial sector NSGA-III helps portfolio managers to balance several competing objectives, such as risk diversification, return maximisation, and liquidity needs.

The NSGA-III has been used to solve issues related to reservoir operation, flood control, and water allocation in situations involving water resource management. In order to balance competing goals including water supply, energy production, irrigation, and environmental sustainability, it aids in optimising water allocation techniques.

Planning and resource allocation issues in healthcare have been addressed using NSGA-III. It helps to maximise the use of healthcare resources, such as personnel, hospital beds, and medical equipment, in order to improve patient outcomes, save costs, and provide higher-quality services.

III.Challenges and Limitations

The Non-dominated Sorting Genetic Algorithm III (NSGA-III) is a powerful tool for addressing multi-objective optimization problems (MOOPs). However, like any optimization technique, it has its own set of challenges and limitations that researchers and practitioners should be aware of.

One significant challenge of NSGA-III is its computational complexity, especially when dealing with a large number of objectives or a high-dimensional search space. The algorithm involves sorting and ranking solutions in each generation, which can become computationally expensive as the number of objectives increases. While NSGA-III has introduced innovations to mitigate this issue, such as using reference points for objective space partitioning, it may still face scalability challenges in extremely high-dimensional problems. [3]

IV.Future Directions

The Non-dominated Sorting Genetic Algorithm III (NSGA-III) is positioned for major developments in its future directions. The algorithm's scalability to handle even more objectives and dimensions, its sensitivity to parameter settings, and further developing diversity management approaches to thwart early convergence are the key areas of research right now. Additionally, NSGA-III is receiving interest for combining machine learning and deep learning techniques, which would allow the algorithm to learn from previous runs and adjust to problem-specific features. The future advancements of NSGA-III will be shaped by the changing landscape of multi-objective optimisation issues, especially in complex and dynamic situations, making it a flexible and adaptable tool for tackling real-world problems.

V. REFERENCES

[1]. Deb, K. (2001). Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons.

[2]. Coello Coello, C. A., Lamont, G. B., & Van Veldhuizen, D. A. (2007). Evolutionary Algorithms for Solving Multi-Objective Problems. Springer.

[3]. Deb, K., & Jain, H. (2014). An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. IEEE Transactions on Evolutionary Computation, 18(4), 577-601.

[4]. Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In Proceedings of the Parallel Problem Solving from Nature (PPSN VII) Conference (pp. 849-858). Springer.

[5]. Srinivas, N., & Deb, K. (1994). Multiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary Computation, 2(3), 221-248.

[6]. Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. Evolutionary Computation, 8(2), 173-195.

[7]. Jain, H., & Deb, K. (2014). An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. IEEE Transactions on Evolutionary Computation, 18(4), 602-622.

[8]. Li, H., Zhang, Q., & Zheng, Y. (2015). Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II. IEEE Transactions on Evolutionary Computation, 19(2), 248-273.

[9]. Fraguela, B. B., Yannibelli, V., Pacini, E., Monge, D., Mateos, C., Rodriguez, G. (2020). A Comparative Analysis of NSGA-II and NSGA-III for Autoscaling Parameter Sweep Experiments in the Cloud. Scientific Programming, 2020, 4653204. DOI: 10.1155/2020/4653204.

Cite this article as :

Ms. Shilpi Jain, Prof Kamlesh Kumar Dubey, Dr. Madhur Jain, Divyansh Rampal, "Evaluating NSGA-III : A Comprehensive Study on Multi-objective Optimization", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 9, Issue 5, pp.146-150, September-October-2023. Available at doi : https://doi.org/10.32628/CSEIT2390523
Journal URL : https://ijsrcseit.com/CSEIT2390523