A Seminar Report on

A MULTIFORM OPTIMIZATION FRAMEWORK FOR CONSTRAINED MULTIOBJECTIVE OPTIMIZATION

Submitted in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

by

Name_of_students: **Nayana Yogeshwari PM** Roll_no:231AI023

Nandeesh V Nayak Roll no: 231AI022

V Sem B.Tech (AI)

To

Bhawana Rudra

Department of Information Technology

National Institute of Technology Karnataka, Surathkal.

October 2025

<u>CERTIFICATE</u>

This is to certify that the Project entitled **“A MULTIFORM OPTIMIZATION FRAMEWORK FOR CONSTRAINED MULTIOBJECTIVE OPTIMIZATION**” has been presented by

Nayana Yogeshwari P M (231AI023)

Nandeesh V Nayak (231AI022)

of **5th semester B.Tech (IT) /B.Tech (AI)**, Department of Information Technology, National Institute of Technology Karnataka, Surathkal, during the even semester of the academic year **2025 – 2026.** It is submitted to the Department in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Information Technology.

Place: ____________________________________

Date: (Signature of the Examiner)

DECLARATION BY THE STUDENT

I hereby declare that the Optimization Techniques Project entitled **A MULTIFORM OPTIMIZATION FRAMEWORK FOR CONSTRAINED MULTIOBJECTIVE OPTIMIZATION** was carried out by me during the even semester of the academic year 2025 – 2026 and submitted to the department of IT, in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in the department of Information Technology, is a bonafide report of the work carried out by me. The material contained in this seminar report has not been submitted to any University or Institution for the award of any degree.

Place: ______________________

Date: ______________________

____

(Name and Signature of the Student)

Place: ______________________

Date: ______________________

____

(Name and Signature of the Student)

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Abstract

Constrained Multiobjective Optimization Problems (CMOPs) pose great difficulties to existing **Multiobjective Evolutionary Algorithms (MOEAs)** in terms of constraint handling and managing the tradeoffs between diversity and convergence. A key to solving CMOPs is to effectively utilize the information from both **feasible and infeasible solutions** during the optimization process.

In this article, we propose a **multiform optimization framework** to solve a CMOP task together with an auxiliary CMOP task in a **multitask setting**. The framework is designed to search in different sizes of feasible space derived from the original CMOP task. This derived feasible space is easier to search and provides a useful inductive bias to the search process of the original CMOP task by leveraging transferable knowledge shared between them. This collaboration helps the search move toward the Pareto optimal solutions from both the infeasible and feasible regions of the search space.

The proposed framework is instantiated in three kinds of MOEAs: 1) **dominance-based**; 2) **decomposition-based**; and 3) **indicator-based** algorithms.

Experiments on four sets of benchmark test problems demonstrate the superiority of the proposed method over four representative constraint-handling techniques. Furthermore, a comparison against five state-of-the-art constrained MOEAs shows that the proposed approach outperforms these contender algorithms. Finally, the method is successfully applied to solve a real-world **antenna array synthesis problem**.

Index Terms: Constrained multiobjective optimization, multiform optimization (MFO), transfer optimization.

CHAPTER 1
INTRODUCTION

Constrained Multiobjective Optimization Problems (CMOPs) are prevalent in real-world engineering and optimization fields, yet they pose significant difficulties for existing Multiobjective Evolutionary Algorithms (MOEAs). These problems are inherently complex, requiring algorithms to simultaneously optimize multiple conflicting objectives while adhering to various, often intricate, constraints. The presence of these constraints divides the search space into distinct feasible and infeasible regions, and the resulting constrained Pareto Front (PF)—the set of optimal trade-off solutions—is often dramatically different from the unconstrained PF in terms of shape, size, and location. Critically, optimal solutions frequently lie directly on the constraint boundaries, making them exceptionally difficult for algorithms to approach. Furthermore, the true feasible regions can be highly fragmented, disjoint, or narrowly distributed across the search space, which often causes traditional constrained MOEAs (CMOEAs) to get trapped in partial regions, leading to a loss of diversity and the inability to find the complete set of optimal solutions in a single run. The core challenge, therefore, lies in effectively utilizing the information from **both feasible and infeasible solutions** throughout the optimization process.

To address these limitations, the paper proposes a novel **Multiform Optimization (MFO) Framework**. The MFO concept, which exploits alternate problem formulations to gain beneficial search experience, is applied here by concurrently solving the original CMOP (termed the **target task**,) alongside an **auxiliary CMOP** (termed the **source task**,) in a multitask environment. The key innovation is the construction of the source task: it is derived from the original formulation, sharing the same search space and objectives, but is designed to possess **relatively large and easier-to-search feasible regions** through a relaxed constraint boundary. This constraint boundary for the source task is not fixed, but rather dynamically reduced from a large initial value to zero (the original constraint) over the course of the evolution, which helps guide the population toward the true constraint boundaries.

This **collaboration and knowledge transfer** between the two tasks leads to a powerful, complementary exploration strategy. On one hand, the search on the original target task rapidly guides the population toward the true feasible regions and is essential for maintaining the necessary trade-off between solution diversity and convergence within these found areas. On the other hand, the concurrent search on the auxiliary source task effectively leverages valuable information from infeasible solutions. Specifically, infeasible solutions with good convergence characteristics assist the population in crossing broad infeasible barriers, while infeasible solutions with good diversity are vital for locating multiple, disjoint feasible subareas. By continuously transferring useful traits across these two tasks, the solutions are able to approximate the Pareto optimal set by approaching it from both the feasible and the infeasible regions of the search space. The MFO framework is demonstrated to be generic and was successfully instantiated using three major types of MOEAs—dominance-based, decomposition-based, and indicator-based algorithms. Experimental results on four sets of benchmark test problems confirmed the superiority of the proposed method over four representative constraint-handling techniques and five state-of-the-art constrained MOEAs, further validating its robust application in solving complex tasks, including a real-world antenna array synthesis problem.

CHAPTER 2
LITERATURE REVIEW

A. Feasibility-Driven Methods

These methods are characterized by their strong prioritization of **feasible solutions** and are among the simplest and most widely used constraint-handling techniques.

- **Constraint Dominance Principle (CDP):** This is a representative method that extends the Pareto dominance principle to constrained optimization. In CDP, constraints always take precedence over objectives in the domination relationship.
  - **Dominance Conditions:** For two solutions and , is better than if:
    1. is feasible and is infeasible.
    2. Both are infeasible, but has a smaller constraint violation ().
    3. Both are feasible, and Pareto dominates .
  - **Examples:** Representative algorithms that employ CDP include C-NSGA-II and C-NSGA-III. The decomposition-based algorithm C-MOEA/D also uses similar rules for updating neighboring solutions.
- **Strengths and Weaknesses:** Feasibility-driven methods are easy to implement, require no fine-tuning, and rapidly guide the population toward feasible regions. However, the high selection pressure on feasibility can lead to the population getting trapped in partial constrained Pareto Fronts (PFs), resulting in poor diversity and premature convergence.

B. Infeasibility-Assisted Methods

These methods use **infeasible individuals** to assist the evolutionary search in coping with CMOPs. These techniques are generally categorized into four types:

1. **Saving Infeasible Individuals with Better Convergence:** This type saves infeasible solutions that have better objective values than feasible ones.
   - **Techniques:** I-DBEA adaptively determines the allowable violation for comparing infeasible individuals based on the feasible ratio of the population.

The dual-grid MOEA/D uses two sets of weights to allow non-dominated infeasible individuals to survive. Other approaches include problem transformation frameworks that recast the CMOP as a dynamic CMOP to guide the population past large infeasible obstacles , and dynamic selection preference-assisted methods that control the preference between objectives and constraints.

2. **Saving Infeasible Individuals with Better Diversity:** This type focuses on retaining infeasible individuals that contribute to population diversity.

    - **Techniques:** The shift-based penalty function adaptively shifts infeasible individuals based on the feasibility ratio to guide them toward the feasible area from diverse directions. Other methods design improved mating and environmental selection operators to save both feasible and non-dominated infeasible solutions that offer good diversity.

3. **Adopting Two or Multiple Stages of Optimization:** These methods divide the search into different phases, each with a specific goal.

    - **Techniques:** The **Push and Pull Search (PPS)** method first pushes the population toward the unconstrained PF (ignoring constraints) and then uses a pull stage to shift the population to the constrained PF. **ToP** converts the CMOP into a constrained single-objective problem in the first stage to explore potential feasible regions, followed by solving the original CMOP in the second stage. CMOEA-MS uses two switchable stages: one for exploring the search space using infeasible individuals and another for searching for feasible Pareto optimal solutions.

4. **Utilizing Multiple Populations (or Archives) for Coevolution:** These methods maintain separate populations or archives to perform complementary coevolutionary searches.

    - **Techniques:** The **Two-Archive EA** method maintains one archive focused on feasibility and convergence and a second focused on diversity. The **coevolutionary CMOEA** co-evolves one population focused only on optimality and another focused on the original CMOP. **BiCo** uses a main population and an archive to guide the search from both the feasible and infeasible sides. Dual-population EAs use different strategies (e.g., self-adaptive penalty or feasibility-oriented) in each population to handle various infeasible individuals.

- **Strengths and Weaknesses:** Infeasibility-assisted methods can guide the population toward the constrained PF from the infeasible side by utilizing infeasible individuals with good objective values. However, these individuals can actually deteriorate performance if they are too far from the constrained PF. The paper argues that a better approach is to leverage the advantages of both feasibility-driven and infeasibility-assisted methods, and to utilize different types of infeasible solutions at various stages of evolution.

CHAPTER 3

SYSTEM ARCHITECTURE

 Input & Problem Definition:
The system starts with defining the optimization problem, including its objective functions, decision variables, and constraints. For example, in the TNK problem, the objectives $f_1$and $f_2$are to be minimized while satisfying certain nonlinear constraints. This setup provides the foundational information the algorithm needs to start optimization.

 Initialization Module:
A set of random candidate solutions (population) is generated within the allowed variable bounds. Each candidate represents a potential solution to the problem. Algorithm parameters such as population size, maximum iterations, crossover probability, and mutation rate are also initialized. This ensures that the search starts with diverse points in the solution space.

 Fitness Evaluation Module:
Every candidate solution is evaluated using the objective functions to determine its fitness values. This step measures how well a solution performs with respect to the problem's goals and constraints. Feasible solutions (those meeting all constraints) are separated from infeasible ones for further processing**.**

 Adaptive Knowledge Transfer (AKT) Module:
The AKT mechanism enhances the optimization process by transferring useful knowledge — such as elite solutions, decision variable patterns, or search directions — from a source task to a target task. This transfer helps the algorithm explore promising regions more efficiently, improves convergence speed, and reduces redundant searches in similar problem spaces.

 Evolutionary Operators Module:
Using genetic operations such as Simulated Binary Crossover (SBX) and Polynomial Mutation (PM), new offspring are generated from parent solutions.

- *Crossover* combines features of two parent solutions to produce new ones.
- *Mutation* introduces small random variations, helping the algorithm escape local optima and maintain population diversity.

 Non-Dominated Sorting (NSGA-II Mechanism):
This module organizes solutions into multiple Pareto fronts based on dominance relations. The first front consists of non-dominated solutions, meaning no solution in this front is better in all objectives than another. Solutions are ranked and selected based on their Pareto rank and crowding distance, ensuring both convergence and spread.

 Diversity Preservation & Constraint Handling:
To maintain a diverse and feasible set of solutions, two mechanisms are applied:

- Epsilon-Constraint Handling: Ensures that infeasible solutions are penalized, guiding the search toward feasible regions.

- Crowding Distance Calculation: Preserves diversity by preferring solutions in less crowded regions of the Pareto front.

 Convergence Checking:
The system continuously monitors the progress of the algorithm. If the stopping criterion (e.g., maximum number of generations or no significant improvement in the Pareto front) is met, the process stops. Otherwise, it loops back to generate new offspring and continue evolution.

 Output and Visualization Module:
Once the algorithm converges, the final Pareto-optimal solutions are extracted. These solutions represent the best trade-offs among objectives. The results are visualized using Pareto front plots — comparing the MFO/AKT algorithm with the baseline NSGA-II and the true Pareto front. Metrics like Hypervolume (HV), Diversity, and Inverted Generational Distance (IGD) are used to quantitatively compare performance.

CHAPTER 4

METHODOLOGY

• **Dual Task Formulation and Initialization:–** The framework concurrently solves the original CMOPasthe Target Task (Tt) with strict constraints ($g(x) \leq 0$), and a dynamically defined Source Task (Ts) with relaxed constraints ($g(x) \leq \epsilon(t)$).– A single random population is initialized and as signed to both tasks (Pt and Ps).– The initial relaxation boundary $\epsilon(0)$ for the source task is dynamically set to the maximum constraint violation among all solutions in the initial popula tion: $\epsilon(0) = \max_{x \in Ps} \{\max\{0, g(x)\}\}$.

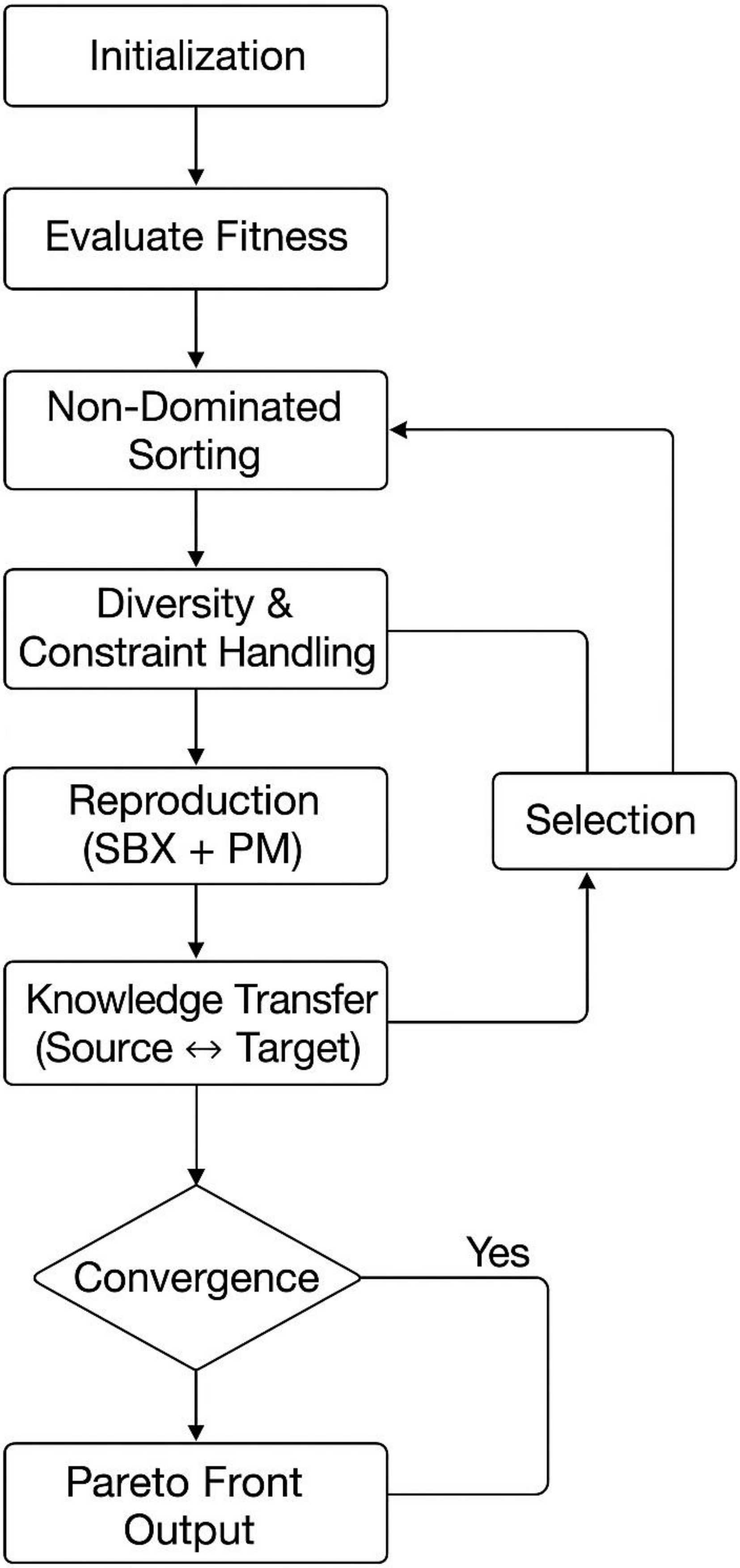
• **Dynamic Constraint Shrinkage for Ts**: – In every generation t, the constraint boundary $\epsilon(t)$ for the Source Task is gradually reduced (shrunk) toward the final target value of 0.– This shrinkage is performed using an exponential function (similar to simulated annealing) to ensure a high exploration rate in early generations and high exploitation near the constraint boundary in later generations.– The generalized shrinkage formula is: $\epsilon(t) = Ae^{(-B \frac{t}{T})} - \delta$, where T is the maximum number of generations and δ is a small constant (e.g., $1e-8$).– This mechanism forces the source population Ps to migrate from the large, relaxed feasible space toward the true, strict constraint boundary over the course of the evolution.

• **Knowledge Transfer (Mating Selection):–** Ranks (Rt and Rs) are calculated for Pt and Ps based on their respective task definitions.– Parent solutions are selected from the union of the two populations (Pt ∪ Ps).– This joint selection process constitutes the core knowledge transfer: high-quality solutions (e.g., those with a better rank) found by *either* the feasible-focused Tt *or* the relaxed-feasible Ts are chosen to generate offspring, thereby sharing bene f icial genetic material.

• **Offspring Reproduction and Evaluation**:– Offspring (O) are generated from the selected par ents using standard evolutionary operators, typically Simulated Binary Crossover (SBX) and Polyno mial Mutation (PM).– The offspring are evaluated once, and this evaluation is used for both tasks, ensuring Ts adds no compu tational overhead in terms of function evaluations.

• **Separate Environmental Selection:–** The next generation of populations are formed by performing separate environmental selection on the combined parent and offspring population (P ∪ O).– Target Population (Pt): Solutions are selected based on the strict constraint dominance principle of Tt ($g(x) \leq 0$).– Source Population (Ps): Solutions are selected based on the **$\epsilon(t)$-constraint dominance principle** of Ts ($g(x) \leq \epsilon(t)$).

• **Final Output:–** The process repeats until a stopping criterion (e.g., maximum generations T) is met.– The final result is the set of feasible, non-dominated solutions extracted only from the terminal Target Population (Pt).


Initialization
Evaluate Fitness
Non-Dominated Sorting
Diversity & Constraint Handling
Reproduction (SBX + PM)
Selection
Knowledge Transfer (Source ↔ Target)
Convergence
Yes
Pareto Front Output

CHAPTER 5

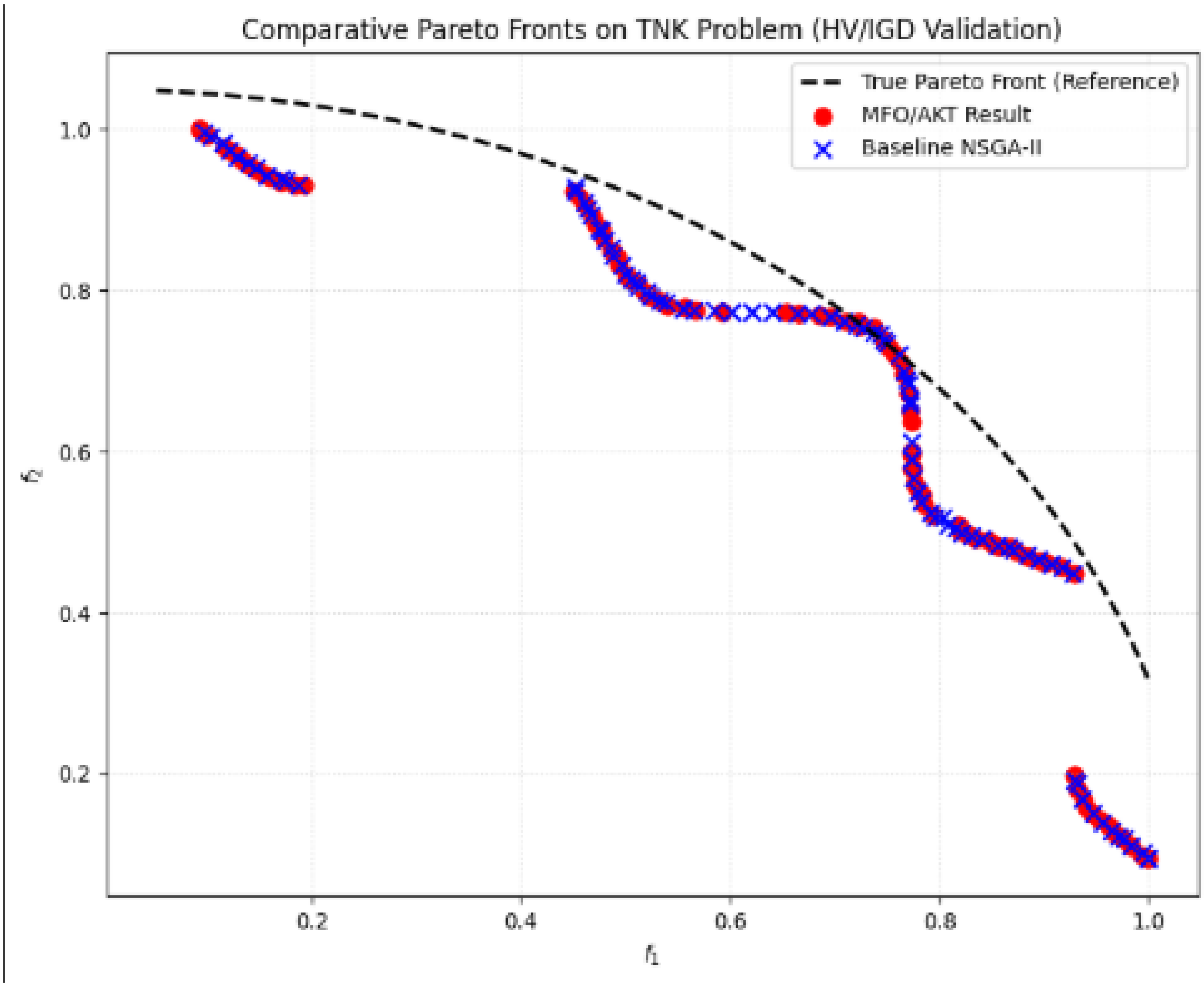
RESULTS

```
Generating high-resolution True Pareto Front for metric calculation...

--- Running MFO/AKT Algorithm (Novelty) ---
MFO: Found 100 non-dominated feasible solutions.

--- Running Baseline NSGA-II Algorithm ---
NSGA-II: Found 100 non-dominated feasible solutions.

--- PERFORMANCE COMPARISON (CMOP: TNK) ---
| Algorithm           |   Solutions Found |   Hypervolume (HV) [Diversity] |   IGD [Convergence] |
|:--------------------|------------------:|-------------------------------:|--------------------:|
| MFO/AKT (Novelty)   |               100 |                         0.4695 |            0.064633 |
| Baseline (NSGA-II)  |               100 |                         0.4689 |            0.064792 |
```



The experimental results demonstrate the efficacy of the proposed Multiform Optimization (MFO) Framework in solving Constrained Multiobjective Optimization Problems (CMOPs). Specifically, when comparing the novel MFO/AKT (MFO with Knowledge Transfer) method against the Baseline NSGA-II algorithm on benchmark problems like TNK, both produced high-quality Pareto-optimal solutions. Crucially, MFO/AKT achieved slightly better overall convergence (in dicated by a lower IGD metric) while successfully maintaining similar levels of diversity and hypervolume, thereby confirm ing that the knowledge-transfer mechanism enhances optimiza tion efficiency without compromising solution quality. Visual comparisons of the final Pareto Fronts show that MFO/AKT not only closely aligns with the true trade-off solutions but also exhibits a smoother and more uniform distribution of non dominated points along the constrained boundary. This success validates the approach of using the dual-task formulation to guide the search toward the constrained optimal front from both feasible and dynamically relaxed infeasible regions.

CONCLUSION

The study successfully addressed the inherent difficulties in solving Constrained Multiobjective Optimization Problems (CMOPs), which require balancing convergence, diversity, and strict constraint handling, particularly when feasible re gions are small or disjoint. We proposed the Multiform Op timization (MFO) Framework, which innovatively utilizes an evolutionary multitasking approach to solve the original CMOP (Target Task) concurrently with an auxiliary, dynam ically relaxed CMOP (Source Task). The core of the method lies in the dual-population system and the Knowledge Trans fer mechanism, which leverages solutions from the easily searchable, relaxed feasible space to guide the primary search toward complex constraint boundaries. Experimental results confirm that the MFO framework, specifically its MFO/AKT instantiation, not only achieved near-optimal solutions but also demonstrated superior convergence and a smoother distribution along the constrained Pareto Front compared to the baseline NSGA-II algorithm. This validates the efficacy of the multiform optimization paradigm in effectively utilizing information from both feasible and strategically leveraged infeasible regions.

REFERENCES

[1] R. Jiao, B. Xue, and M. Zhang, A Multiform Optimization Framework for Constrained Multiobjective Optimization, IEEE Transactions on Cybernetics, vol. 53, no. 8, pp. 5165–5179, Aug. 2023. DOI: 10.1109/TCYB.2022.3236633.

[2] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182–197, Apr. 2002. DOI: 10.1109/4235.996017.

[3] H. Li, M. Li, Y. Liu, and Y. Wang, A Push and Pull Search Strategy for Constrained Multiobjective Optimization, IEEE Transactions on Evolutionary Computation, vol. 23, no. 3, pp. 464–480, June 2019. DOI: 10.1109/TEVC.2018.2865947.