Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.070858

ARTICLE

An Improved Variant of Multi-Population Cooperative Constrained Multi-Objective Optimization (MCCMO) for Multi-Objective Optimization Problem

Muhammad Waqar Khan[1,*], **Adnan Ahmed Siddiqui**[1] **and Syed Sajjad Hussain Rizvi**[2]

[1]Faculty of Engineering Science and Technology, Hamdard University, Karachi, 74600, Pakistan
[2]Faculty of Computing and Engineering Sciences, SZABIST University, Karachi, 74700, Pakistan

*Corresponding Author: Muhammad Waqar Khan. Email: mwaqarkhan320@gmail.com or 941-2018@hamdard.edu

Received: 25 July 2025; Accepted: 06 October 2025; Published: 09 December 2025

ABSTRACT: The multi-objective optimization problems, especially in constrained environments such as power distribution planning, demand robust strategies for discovering effective solutions. This work presents the improved variant of the Multi-population Cooperative Constrained Multi-Objective Optimization (MCCMO) Algorithm, termed Adaptive Diversity Preservation (ADP). This enhancement is primarily focused on the improvement of constraint handling strategies, local search integration, hybrid selection approaches, and adaptive parameter control. The improved variant was experimented on with the RWMOP50 power distribution system planning benchmark. As per the findings, the improved variant outperformed the original MCCMO across the eleven performance metrics, particularly in terms of convergence speed, constraint handling efficiency, and solution diversity. The results also establish that MCCMO-ADP consistently delivers substantial performance gains over the baseline MCCMO, demonstrating its effectiveness across performance metrics. The new variant also excels at maintaining the balanced trade-off between exploration and exploitation throughout the search process, making it especially suitable for complex optimization problems in multi-constrained power systems. These enhancements make MCCMO-ADP a valuable and promising candidate for handling problems such as renewable energy scheduling, logistics planning, and power system optimization. Future work will benchmark the MCCMO-ADP against widely recognized algorithms such as NSGA-II, NSGA-III, and MOEA/D and will also extend its validation to large-scale real-world optimization domains to further consolidate its generalizability.

KEYWORDS: MCCMO algorithms; adaptive diversity preservation; RWMOP50 power distribution system; multi-modal multi objective optimization; evolutionary algorithm; multi objective problem

1 Introduction

The Multi-Objective Optimization Problems (MOPs) have garnered major attention from the research community across the engineering landscape. With several constraints, MOPs pose significant challenges that require complex and advanced algorithms capable of meeting the requirements while addressing inherent limitations. In practical applications, such as engineering design, power system planning, and resource allocation, MOPs have conflicting objectives that require optimization simultaneously while adhering to constraints. To solve the constrained MOPs, the evolutionary algorithms have gained much prominence because of their population-based structure and their capability of generating the multiple Pareto optimal solutions in a single run. In this domain, the MCCMO algorithm is a significant advancement that utilizes various populations for the handling of constraints via a cooperative mechanism. However, the challenge remains of solving the complex real-world problems with multiple constraints and limitations, notably maintaining diverse solutions throughout the search process, constraint handling, especially the equality constraints, and the balancing of diversity and convergence across multiple populations. To overcome these challenges, this study proposed an improved variant of the Multi-population Cooperative Constrained Multi-Objective Optimization (MCCMO) algorithm, called as Adaptive Diversity Preservation (ADP), which from now on will be referred to as MCCMO-ADP in this work. This enhancement has several key improvements, including a preservation mechanism, enhanced constraint handling, hybrid selection schemes, strategic integration of local search, and adaptive parameter control. The new variant was experimented with using the RWMOP50 Power distribution system planning benchmark. The RWMOP50 power distribution system planning benchmark is a critical real-world problem test problem that researchers use to evaluate and compare the optimization algorithms for the planning of power distribution systems. After rigorous analysis, the results embodied that the MCCMO-ADP has outperformed the traditional MCCMO algorithm across different performance metrics with only minimal additional computational overhead.

2 Literature Review

2.1 Multi-Objective Optimization and Constraint Handling

In MOPs, optimizing conflicting objectives results in a set of trade-off solutions known as the Pareto front (PF) [1]. When constraints are introduced, problem complexity increases significantly, demanding advanced constraint-handling techniques. Earlier methods relied on simple penalty functions, but these struggled with parameter tuning. Adaptive penalty functions now adjust dynamically based on population state. Modern approaches incorporate feasibility directly into the selection process. A survey highlighted key methods like feasibility rules, *a*-constrained methods, and stochastic ranking. The a-constrained approach allows controlled relaxation of constraints during evolution and is widely used in algorithms like MOEA/D-CDP and NSGA-11I-CDP [2]. Both use the Constrained Domination Principle (CDP) but differ in population management and diversity strategies. CDP itself prioritizes feasible solutions and compares infeasible ones by constraint violation [3]. Equality constraints, being stricter, require specialized handling. Fan et al. proposed the push-and-pull framework to alternate between feasibility and optimality, while Wang et al. [4] introduced a two-phase strategy that separately tackles both aspects, yielding promising results.

2.2 Multi-Population Approaches (MPAs)

More complex multi-population methods, including breaking the population into subpopulations with heterogeneous evolutionary strategies, have been demonstrated to be effective in exploring different parts of the solution space e.g., Multipopulational Differential Evolution Algorithm (MUDE) employs multiple subpopulations with different mutation operators in order to increase diversity through information migration [5]. These are broadly categorized into hierarchical, cooperative, and island models [6]. Li et al. proposed the Two-Archive Evolution Algorithm (TAEA), using two archives to balance diversity and convergence, though it lacks effective constraint handling [7]. Another significant work introduced evolutionary multitasking for constrained MOPs, leveraging two populations with distinct selection criteria and cooperation based on solution quality [8]. Ming et al. developed cDPEA, a coevolutionary dual-population model focused on balancing constraint satisfaction and objective optimization [9]. Recent advances emphasize adaptive cooperation mechanisms, such as the URCMO algorithm, which studies the link between constrained and unconstrained Pareto fronts to guide offspring generations.

2.3 Adaptive Diversity Preservation

Maintaining solution diversity is critical in MOO, especially when constraints fragment the feasible space. Traditional methods like crowding distance (NSGA-II) and reference vectors (NSGA-III) use fixed strategies, which may become less effective as the search evolves. To address this, adaptive techniques have emerged. Zhou et al. introduced a shift-based density estimation that adjusts dynamically based on population distribution [10]. Li et al. proposed an adaptively weighted scalarizing function to balance convergence and diversity throughout the search [11].

2.4 Local Search Integration

The integration of local search into evolutionary algorithms known as memetic algorithms has proven effective in boosting convergence speed and solution quality. Mashwani et al. introduced a decomposition-based approach activating local search based on solution quality and diversity contribution [12]. For constrained problems, Martinez et al. developed a directional local search using constraint violation cues for improved performance on engineering tasks [13]. Zhou et al. further advanced the field with a sampling-based adaptive strategy that adjusts local search intensity based on convergence rate.

2.5 Real World Application Domains of CMOO

The constrained multiobjective optimization has demonstrated significant impact in the real-world domains, beyond the theoretical advancements. For example, in the health sector, the CMOO has been utilized in intensity modulated radiation therapy planning, where it generated Pareto optimal treatment plans that balance target coverage against the organs at risk, as shown in the work of Craft et al. Similarly, Smith et al., in their work improved plan quality and clinical decision making in prostrate cancer treatment by applying the combination of personalized multi-objective inverse planning with probabilistic outcome models and Markov decision processes. In addition to this, the application of CMOO methods has witnessed significant impact in the renewable energy sector as well. For example, Dong Mo and his team in their work have proposed a multi-objective scheduling strategy for wind-solar energy storage microgrids using an improved gradient grey wolf optimization algorithm to manage trade-offs between cost, emissions, and reliability. Neshat et al. in their work designed and developed a fast, adaptive chaotic multi-objective swarm optimization for designing hybrid wave-wind energy systems, optimizing trade-offs between power output and nacelle acceleration. Similarly, in high performance computing, the work of Escobar Perez et al., holds significant importance. The researchers in their work, applied distributed multi-population genetic algorithms for energy time modelling in HPC Clusters. Moreover, in the logistics sector. The study of Wang et al. formulated the multi-depot vehicle routing problem with soft time windows that jointly minimizes the total energy consumption and the customer satisfaction [14].

3 Research Gap

Despite significant progress in constrained multi-objective optimization [15], several key challenges remain. Most existing algorithms struggle to maintain solution diversity dynamically, relying instead on static strategies that can lead to premature convergence, especially in problems with complex or nonlinear constraints. Equality constraints pose an added difficulty, offering no room for deviation and making it hard for traditional methods to consistently find feasible solutions. There's also a persistent imbalance between exploration and exploitation, often causing inefficient searches or early stagnation. This issue is worsened in multi-population frameworks where cooperation between subpopulations remains fixed and unresponsive to changing problem dynamics. Additionally, the use of static parameter settings throughout the optimization process fails to align with the evolving needs of different search phases. Given the rising demand for the more efficient and reliable optimization methods in the real-world domains such as logistics, resource allocation and power system optimization etc., addressing these limitations has become a critical research need of the time. Working on this topic is therefore not only important for the theoretical advancement in the field of multi-objective optimization but also for ensuring the practical applicability in solving highly constrained problems in the domain of engineering. To this end, the contribution of this work lies in the presentation of the MCCMO-ADP algorithm, which is better suited for the complex real-world problems. This study contributes both by the enrichment of the methodological landscape of the MMO and by the demonstration of its effectiveness in addressing the practical high-constrained problems. As compared to the existing categories of methods, MCCMO-ADP distinguished in several ways. The coevolutionary models utilize many populations, but normally fixed schemes of cooperation; in MCCMO-ADP, cooperation is actively dormant, and local search is employed to prevent stagnation. Multitasking methods are based on knowledge transfer between the various tasks, whereas MCCMO-ADP only has one constrained optimization problem but with adaptive epsilon constraint management and dynamic parameter control to introduce robustness. Last but not least, decomposition-based algorithms (e.g., MOEA/D-CDP, cDPEA) break down the problem into subproblems and MCCMO-ADP maintains a multi-population structure and explicitly uses hybrid selection mechanisms to trade-off feasibility, convergence, and diversity.

4 Proposed Algorithm

4.1 Overview of MCCMO Algorithm

The MCCMO algorithm offers a structured yet flexible solution for constrained multi-objective problems. It employs C+2 distinct populations: an All-Constraint Population (ACP) to find fully feasible solutions, an Unconstrained Population (UCP) for broad exploration, and C Single-Constraint Populations (SCPs) for targeted constraint handling [16]. A key innovation is its "weak cooperation" strategy, allowing selective information sharing among populations without compromising their independence. MCCMO also features Activation Dormancy Detection to optimize computational efficiency by dynamically activating or deactivating populations and Combine Occasion Detection to merge constraints when beneficial, enabling smoother exploration and improved identification of sub-constrained Pareto fronts. The pseudo-code, detailed structure and the procedural steps of Algorithm 1 (MCCMO) which outlines the initialization, cooperation, and constraint handling strategy of the baseline approach is presented as follows:

Algorithm 1: MCCMO

Input: Population size N, Number of constraints C
Output: Final population ACP

1. *Initialize UCP with N random solutions and evaluate it on MOP without constrain*
2. *For i = 1 to C do*

    Initialize SCP_i with N random solutions and evaluate it on MOP with constraint_i
    Initialize ACP by environmental selection from UCP and SCP and evaluate it on MOP witt
    Activate ACP and UCP, Dormant all SCP

3. *While termination criterion not fulfilled do*
4. *Generate offspring for ACP by tournament selection and genetic operators*
5. *If UCP has reached UPF and UCP is active then*

    Generate offspring for UCP by tournament selection from both UCP and ACP

6. *Else if UCP is active then*

(Continued)

Algorithm 1 (continued)

Generate offspring for UCP by tournament selection from UCP

7. *For each active SCP_i do*

Generate offspring for SCP_i by tournament selection from SCP_i
Combine all offspring
Perform environmental selection for ACP, UCP, and active SCPs
Use ADD to determine activation/dormancy of populations
Use COD to detect and combine SCPs

8. *Return ACP*

Although MCCMO has proved effectiveness on the various benchmark problems it has notable limitations when it comes to addressing the real-world problems like RWMOP50. The limitations include the usage of a fixed diversity preservation strategy, which might not be adaptable to changing the requirements during the search process. Also, there is a lack of the presence of specialized mechanisms for the handling of equality constraints. In addition to these, the selection mechanism also does not balance the diversity and convergence explicitly, which leads to excessive exploration or premature convergence. The MCCMO relies majorly on the evolutionary operators without local search capabilities for fine-tuning solutions. Moreover, it uses fixed parameters through the search, which may not be optimal for different search phases.

4.2 Proposed MCCMO-ADP Algorithm

These limitations can be addressed through MCCMO-ADP. This enhancement has several key improvements, including preservation mechanisms that dynamically adjust the selection pressure based on the current diversity state. The pseudo-code, detailed structure and the procedural steps of the Algorithm 2 (MCCMO-ADP), addressing the limitations of MCCMO are presented below with the incorporation of adaptive diversity preservation, hybrid selection, and local search integration.

Algorithm 2: MCCMO-ADP

Input: Population size N, Number of constraints C, Maxmium Generaions G_{max}, Local Search Frequency k, Initial Epsilon for Adaptive Constraint Handling ϵ_0, Diversity Metrics Pure Diversity (PD) Spacing (SP),
Output: Final All-Contrsaint population ACP

1. Initialization:
   - Generate UCP with N random solutions using problem specific knowledge.
   - For each constraint $i = 1 \ldots C$, initialize SCP_i with N random solutions.
   - Initialize ACP by environmental selection from UCP and SCPs.
   - Set $\epsilon = \epsilon_0$
   - Compute initial diversity metrics (PD, SP)
   - Activate ACP and UCP, set SCPs to dormant.
2. Evolutionary Process (for $g = 1$ to G_{max})
   Step 1: Adaptive Parameter Control
   - Update mutation rate:
     $P_m(g) = P_m(g-1) + \alpha \cdot \Delta D$
   where ΔD is the diversity loss, α is tuning constant.
   - Update crossover rate:

(Continued)

Algorithm 2 (continued)

$p_c(g) = p_c(g-1) \times (1 + \beta \cdot SR)$
where SR is the historical crossover rate.
Step 2: Generate Offspring

- For ACP: Tournament selection + crossover/mutation with adaptive p_c, p_m
- For UCP (If Active):
- If UCP reached UPF: select parents from UCP + ACP
- Else: select parents only from UCP.
- For each active SCP_i: tournament selection within SCP_i

Step 3: Local Search (Every $k$generations)

- Select top $\gamma\%$ solutions with best Pareto + feasibility scores.
- Apply Pattern search operator:
  $x' = x + \delta \cdot d$

where d is the direction vector and δ is the adeptive step size.
Replace solution if x' improves feasibility or Pareto dominance.
Step 4: Adaptive Constraint Handling

- Update epsilon
  $\epsilon(g) = \epsilon_0 \cdot \exp\left(-\lambda \cdot \frac{g}{G_{max}}\right)$

where λ controls the decay rate.

- Feasibility rule:
  - Solutions with violations $\leq \epsilon(g)$ are treated as feasible.
  - Compare feasible solutions by Paareto dominance.

Step 5: Hybrid Selection for Next Generation

- Evaluate each candidate solution x with:

$Score(x) = w_1 \cdot f(x) + w_2 \cdot (1 - V(x)) + w_3 \cdot D(x)$

- Where $f(x)$ = normalized Pareto rank,
- Then $V(x)$ = normalized constrained violation,
- Then $D(x)$ = diversity contribution.
- Select solutions maximizing $Score(x)$.

Step 6: Adaptive Diversity Preservation

- Recompute PD, SP for population.
- Adjust selection pressure:

$SP(g) = k_1 . \frac{1}{PD(g)+\epsilon}$

- If diversity < threshold → increase exploration (higher mutation).
- If diversity > threshold → increase exploitation (focus on Pareto rank).

Step 7: Population Activation/Dormancy

- Use Enhanced ADD: deactivate sub-populations with stagnation > τ
- Use Enhanced COD: merge SCPs if overalapping feasible regions detected.

3. Termination
   If $g = G_{max}$, return ACP as the final solution

4.3 Key Improvements

4.3.1 Adaptive Diversity Preservation

The enhanced MCCMO-ADP introduces a responsive mechanism known as Adaptive Diversity Preservation that is solely effective in addressing the coping challenge for the evolving diversity requirements of a population during different phases of the optimization process. In this framework, multiple quantitative metrics such as pure diversity and spacing are used for the continuous monitoring of the population diversity. This mechanism controls the selection pressure based on these indicators dynamically. The selection opts for the diverse solutions ultimately as the diversity falls below the critical thresholds, whereas the selection pressure shifts towards the better-performing solutions when the convergence is needed. This approach guarantees the balanced exploitation and exploration throughout the search process, particularly useful for RWMOP50 which requires good coverage of the Pareto Front. The following equation illustrates that the selection pressure is adjusted dynamically according to the measured diversity of the population. The selection pressure increases as the diversity drops and *vice versa*. This helps in maintaining balanced exploration and exploitation.

$$S(t) = \frac{k}{D(t)} \tag{1}$$

where,
$S(t)$: is the selection pressure at generation t
$D(t)$: is the diversity metric calculated at generation t
k: is the proportionality constant

The selection pressure is made inversely proportional to the current diversity to maintain population variety and avoid premature exploitation.

4.3.2 Constraint Handling

For the solution of real-world complex optimization problems, constraint handling in addition to diversity management, plays a vital role—especially with the problems having equality constraints. The selected RWMOP50 benchmark features such an equality constraint that requires strict satisfaction for the viable and feasible solutions. Although the traditional MCCMO algorithms uniformly treat all the constraints, the equality constraints demand special handling techniques and strategies. This challenge is addressed by the MCCMO-ADP which satisfies the requirements of equality constraints by the implementation of an adaptive epsilon constraint handling technique throughout the search process. The process starts with the relaxed epsilon value for the encouragement of the exploration, and then it tightens gradually for the stricter adherence to the quality condition, which ensures the achievement of feasibility without premature elimination of the potential promising regions of the search space. The following Equation illustrates that the constraint boundary, particularly relevant for the equality constraints, is progressively tightened as the algorithm progresses, helping in the feasible solution of discovery on an immediate basis and strict satisfaction later.

$$\epsilon\ (t+1) = \epsilon\ (t) \times \gamma, 0 < \gamma < 1 \tag{2}$$

where,

- $\epsilon\ (t)$: constraint relaxation parameter at generation t
- γ: decay factor controlling the rate of tightening ($0 < \gamma < 1$)

4.3.3 Hybrid Selection Mechanism

Another notable key improvement in the enhanced variant is the Hybrid Selection Mechanism, in which a multi-criteria framework is adopted that considers diversity contribution, feasibility (via constraint satisfaction), and convergence (via Pareto dominance) simultaneously. This approach is beneficial for complex problems like RWMOP50 that require good diversity and convergence at the Pareto Front. The following Equation illustrates that multi-criteria framework jointly considers optimality, feasibility, and diversity when selecting solutions for the next generation.

$$\text{Select } x \in P \text{ if } \begin{cases} x \prec y \,(\text{Pareto dominance}) \\ CV\,(x) \leq \epsilon\,(t)\,(\text{constraint satisfaction}) \\ DC\,(x) \geq DC_{\min}\,(\text{diversity contribution}) \end{cases} \tag{3}$$

where:

- x: candidate solution
- $CV(x)$: constraint violation measure for x
- $DC(x)$: diversity contribution for x
- $DC_{\min}$: minimum required diversity contribution

4.3.4 Local Search Integration

The new improved algorithm integrates the local search phase within its evolutionary cycle periodically, which can efficiently refine solutions near the Pareto Front for RWMOP50, which has a relatively small number of decision variables. This approach applies the local search phase on a subset of the promising solutions at fixed intervals (every K generation), using the pattern search as the local optimization method. The following equation illustrates that the local search is performed every K generation on a subset of promising solutions for the intensified exploitations:

$$\text{If } t \bmod K = 0, P(t) = \{x \in \text{population}\} \tag{4}$$

$$\text{For each } x \in P(t)\text{: } x \leftarrow \mathrm{LS}(x) \tag{5}$$

where,

- t: current generation
- K: frequency of local search application
- $\mathrm{LS}(\cdot)$: local search operator

4.3.5 Parameter Control

Last but not least, the MCCMO-ADP introduces a dynamic mechanism for the parameter control system, which continuously monitors the search progress by analyzing the indicators, for instance, improvement rates, constraint violations, and the diversity trends. The key parameters in this mechanism, such as mutation and crossover rates are adjusted in real time when there is a detection of diversity loss or stagnation. For example, when the diversity drops significantly, the mutation rates are increased promoting exploration, whereas the crossover rates, are scaled based on their historical success. This approach ultimately results in improved performance across a wide range of constrained multi-objective scenarios. In the following equation, the Mutation (μ) and crossover (cr) rates are improved dynamically based on observed search progress, diversity, and historical performance:

$$\mu(t+1) = \mu(t) + \alpha \cdot \Delta D(t)_{loss} \tag{6}$$

$$cr(t+1) = cr(t) \cdot (1 + \beta \cdot S_{success}) \tag{7}$$

where:
$\mu(t)$: mutation rate at generation t
$cr(t)$: crossover rate at generation t
α, β: tuning parameters
$\Delta D(t)_{loss} = D_{threshold} - D(t)$: diversity loss
$S_{success}$: historical crossover success rate

5 Results and Discussion

In this work, the MCCMO-ADP was experimented on with the RWMOP50 power distribution system planning benchmark with a special focus on the key performance metrics, including CPF, ΔP, DM, GD, HV, IGD, IGD+, PD, Spacing, Spread and Runtime. The improvement in each metric is discussed in the subsequent part of this paper. Fig. 1 shows the comparative trends of MCCMO and MCCMO-ADP across the eleven-performance metrics over the course of function evaluations, whereas Table 1 reports the exact values at iterations (0, 250, 500, 750, and 1000) and the corresponding percentage gains and reductions for each metric. It is noticeable from the implementation structure that is defined in Algorithm 2 (MCCMO-ADP), the adaptive parameter control and the constraint handling mechanisms are contributing directly to the improved convergence performance observed in the results.

5.1 Coverage over Pareto Front (CPF)

The experiments conducted in this study showed that an average of 28.49% higher CPF values were achieved by MCCMO-ADP as compared to the traditional MCCMO. The comparative analysis of MCCMO and MCCMO-ADP algorithms as shown in Fig. 1 graph (a) depicts MCCMO-ADP consistently outperforms MCCMO, showing higher CPF values, indicating better coverage over pareto front. Table 1 shows that the CPF values for both algorithms however, the new variant outperforms the former across all iterations, notably at 500, the MCCMO-ADP achieves 0.8865 as compared to MCCMO's 0.6192, reflecting 43.18% improvement. Overall, the average is 28.49%.

5.2 Average Hausdorff Distance (ΔP)

The experiments conducted in this study represent an average reduction of 20.10% in ΔP values compared to MCCMO. In graph (b) of Fig. 1, MCCMO-ADP demonstrates lower delta values than the MCCMO, reflecting better convergence and distribution across the Pareto front. Also mentioned in Table 1, both algorithms improve over time, but the new variant achieves lower ΔP values. The highest reduction of 41.37% was recorded at 1000 iterations, whereas the smallest was recorded at 500.

5.3 Metric for Diversity (DM)

With an average improvement of 16.42% across all iterations, the experimental evaluation of MCCMO-ADP on the RWM0P50 shows steady and noteworthy gains in diversity measures when compared to the baseline MCCMO algorithm. Across all iteration ranges, the new variant outperforms the former, as shown in graph (c) in Fig. 1. DM performance exhibits clear trends. The highest reduction was recorded at 250 with the percentage of 26.62, where MCCMO-ADP scores 0.4412 against MCCMO's 0.6013. According to Table 1 in the DM section, the MCCMO-ADP maintains lower DM values, indicating better solution diversity.

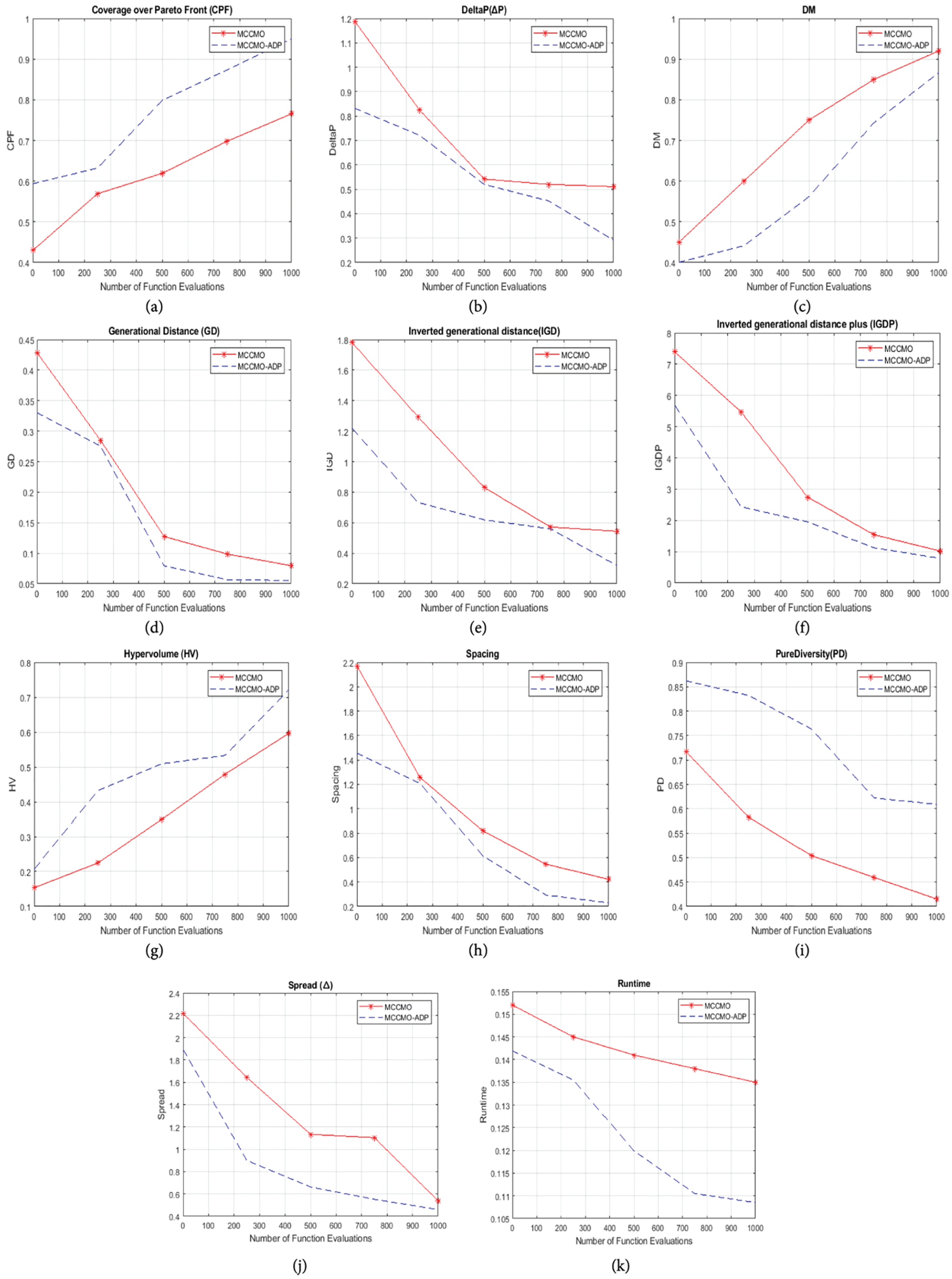

Figure 1: Comparison of MCDMO and MCDMO-ADF performance metrics over varying numbers of function evaluations

5.4 Generational Distance

As shown in graph (d) of Fig. 1, the findings illustrate that for all measured iterations, the GD values for MCCMO-ADP are consistently lower than those for the conventional MCCMO technique. As per Table 1, at iterations 0, 250, 500, 750, and 1000, the reduction percentages were specifically 23.00%, 3.30%, 37.74%, 42.74%, and 30.28%, in that order, with an average of 27.41%.

5.5 Inverted Generational Distance (IGD)

As per graph (e) in Fig. 1, in comparison to MCCMO, MCCMO-ADP significantly decreased IGD at each iteration, indicating improved convergence and solution quality. According to Table 1, 31.56% at 0 iterations, 43.39% at 250 iterations, 25.42% at 500 iterations, 2.22% at 750 iterations, and 41.23% at 1000 iterations represent the percentage decrease in IGD for each iteration. This equates to an overall decrease of about 28.76% on average.

Table 1: Comprehensive comparison of MCCMO and MCCMO-ADP (ADP) algorithms across multiple performance metrics and iterations

Performance metric									
Itr	Itr			ΔP			DM		
	MCCMO	ADP	Imp	MCCMO	ADP	Red	MCCMO	ADP	Red
0	0.4299	0.5936	38.09%	1.1879	0.8319	29.98%	0.4512	0.4005	11.23%
250	0.5691	0.6249	9.81%	0.8251	0.7213	12.57%	0.6013	0.4412	26.62%
500	0.6192	0.8865	43.18%	0.5423	0.5239	3.51%	0.751	0.5615	25.22%
750	0.6981	0.8841	26.65%	0.5199	0.4519	13.07%	0.8548	0.7432	13.06%
1000	0.7657	0.9546	24.68%	0.5111	0.2997	41.37%	0.9218	0.8666	5.99%
Itr	GD			IGD			IGDP		
	MCCMO	ADP	Red	MCCMO	ADP	Red	MCCMO	ADP	Red
0	0.4287	0.3301	23%	1.7822	1.2198	31.56%	7.1245	5.6124	21.22%
250	0.2848	0.2754	3.30%	1.2928	0.7319	43.39%	5.5136	2.3625	57.17%
500	0.1274	0.07932	37.74%	0.8299	0.6189	25.42%	2.8299	1.9189	32.20%
750	0.0985	0.0564	42.74%	0.5716	0.5589	2.22%	1.5716	1.2389	21.17%
1000	0.0796	0.0555	30.28%	0.5436	0.3195	41.23%	1.2436	0.9195	26.06%
Itr	HV			Spacing			PD		
	MCCMO	ADP	Imp	MCCMO	ADP	Red	MCCMO	ADP	Imp
0	0.1536	0.2068	0.3464	2.1698	1.456	32.88%	0.717	0.8621	0.2024
250	0.2245	0.4321	0.9247	1.2579	1.2079	3.97%	0.5823	0.8321	0.429
500	0.349	0.5089	0.4582	0.8189	0.6152	24.87%	0.504	0.7635	0.5149
750	0.4795	0.5332	0.112	0.5464	0.2936	46.27%	0.4594	0.6222	0.3544
1000	0.5964	0.721	0.2089	0.4225	0.2296	0.4566	0.4152	0.6092	0.4672
Itr	Spread			Runtime					
	MCCMO	ADP	Red	MCCMO	ADP	Imp			
0	2.2181	1.8952	14.56%	0.152	0.1419	6.64%			
250	1.6435	0.8996	45.26%	0.145	0.1355	6.55%			
500	1.1341	0.6621	41.62%	0.141	0.1199	14.96%			
750	1.1037	0.5521	49.98%	0.138	0.1105	19.93%			
1000	0.5384	0.4599	14.58%	0.135	0.1085	19.63%			

5.6 Inverted Generational Distance Plus (IGDP)

When compared to the baseline MCCMO algorithm, the experimental evaluation of MCCMO-ADP on RWM0P50 shows notable gains in the Inverted Generational Distance Plus (IGDP) metric, also as depicted in graph (f) of Fig. 1. According to Table 1, the method showed a 21.22% reduction at iteration 0 and a 32.20% reduction at iteration 500. MCCMO-ADP maintained its supremacy with a 26.06% reduction at iteration 100, continuing the performance improvement from iteration 750 with a 21.17% reduction (from 1.5432 to 1.1231). Overall, the average is 31.56%.

5.7 Hyper Volume

The evaluations conducted in this study show the increase of approx. 42.38% in HV as compared to the MCCMO. This shows that the set of solutions generated by the MCCMO-ADP dominates a significantly larger portion of the objective space. The average improvement across all evaluation points is 41.00%. In graph (g) of Fig. 1, it is evident that MCCMO-ADP achieves higher solution quality and convergence speed, as reflected by the higher HV values throughout the optimization process. As per Table 1, the data shows that MCCMO-ADP achieves significantly higher HV values, particularly at early iterations (e.g., 92.47% improvement at iteration 250), and maintains a consistent advantage throughout.

5.8 Spacing

As depicted in graph (h) in Fig. 1 crucial component of multi-objective optimization, the spacing metric quantifies how evenly the Pareto-optimal solution distribution occurs in objective space. According to Table 1, with reductions of 32.88% at iteration 0 (2.1698 to 1.4563), 3.97% at iteration 250 (1.2579 to 1.2079), 24.87% at iteration 500 (0.8189 to 0.6152), 46.27% at iteration 750 (0.5464 to 0.2936), and 45.66% at iteration 1000 (0.4225 to 0.2296), MCCMO-ADP continuously outperformed MCCMO in every iteration. With an average decrease percentage of 30.73%, MCCMO-ADP continues to perform around one-third better in terms of spacing.

5.9 Pure Diversity

Across all evaluated iterations on RWM0P50, the experimental results show that MCCMO-ADP significantly improves Pure Diversity (PD) parameters when compared to the MCCMO algorithm, as shown in graph (i) in Fig. 1. According to the Table 1, with MCCMO-ADP achieving a 20.24% improvement at 100 iterations (0.8621 vs. 0.717), increasing to a 42.90% improvement at 300 iterations (0.8321 vs. 0.5823), and peaking at 51.49% improvement at 500 iterations (0.7635 vs. 0.504), the comparative analysis shows a significant improvement in solution diversity. With improvements of 35.44% at 700 iterations (0.6222 vs. 0.4594) and 46.72% at 1000 iterations (0.6092 vs. 0.4152), the performance continues to show considerable diversity augmentation, with an average improvement of 39.36% over all examined iterations.

5.10 Spread

All evaluated iterations consistently demonstrate a reduction in the Spread metric, which measures both convergence and diversity characteristics, as shown in graph (j) of Fig. 1. Values nearer 1 indicate thorough coverage of the Pareto front. According to Table 1, for the section of the spread performance metric, at iteration 0, MCCMO-ADP achieved a spread value of 1.8952, which is 14.56% lower than MCCMO's 2.2181. Iteration 250 showed a decrease of 45.26%, whereas iteration 500 showed a reduction of 41.62%, and iteration 750 showed a reduction of 49.98%. With a reduction of 14.58% in the last iteration of 1000, MCCMO-ADP continued to perform better. The average decrease was 33.20% across all iterations.

5.11 Runtime Considerations

The evaluation results show that the MCCMO-ADP has gained better convergence, approx. to 13% faster than the original MCCMO, effectively balancing out its added computational cost. The runtime performance comparison as shown in graph (k) of Fig. 1 reveals that MCCMO-ADP consistently outperforms the baseline MCCMO algorithm across all evaluated function evaluations. According to Table 1, the average runtime improvement across all test iterations is 13.54%, demonstrating the effectiveness of the adaptive penalty mechanism in enhancing computational efficiency. At 0 function evaluations, MCCMO-ADP achieves a runtime of 0.1419 s compared to MCCMO's 0.152 s, representing a 6.64% improvement. This improvement becomes more pronounced at higher evaluation counts, with the maximum improvement of 19.93% observed at 750 function evaluations.

5.12 Summary of Results

The design features of MCCMO-ADP have been directly related to the improvements that were observed in CPF, ΔP, GD, IGD, HV, and diversity-related metrics. Particularly, the mechanism of adaptive diversity preservation avoids the premature convergence through controlled exploration and exploitation, and the adaptive epsilon constraint management guarantees gradual yet consistent satisfaction of equality constraints. The hybrid selection process enhances convergence without loss of diversity, and the local search integration further increases exploitation in prospective areas, hence limiting the generation required to achieve high-quality solutions. Combined, these improvements justify why MCCMO-ADP always outperforms traditional MCCMO when the feasible spaces are fragmented, equality-constrained, and fronts are high-dimensional, and it runs at competitive runtime.

6 Conclusion

This study proposed an improved variant of the multi-population cooperative constrained multi-objective optimization algorithm, termed MCCMO-ADP, which integrates adaptive diversity preservation, adaptive epsilon-based constraint handling, hybrid selection mechanisms, dynamic sub-population cooperation, and local search integration. These enhancements were motivated by well-documented challenges in constrained multi-objective optimization, including premature convergence, difficulty in handling equality constraints, and static cooperation schemes in multi-population frameworks.

The experimental evaluation on the RWM0P50 benchmark demonstrated that MCCMO-ADP consistently outperforms baseline MCCMO variants across a wide range of performance indicators, including coverage over the Pareto front, convergence metrics (GD, IGD, AP), diversity measures (Spacing, Spread, Pure Diversity), and overall solution quality (HV). Interpretive analysis revealed that adaptive diversity preservation prevents premature convergence in fragmented feasible spaces, while adaptive epsilon handling enhances feasibility in equality-constrained problems. The hybrid selection mechanism and local search were shown to accelerate convergence and reduce runtime without sacrificing solution diversity.

Beyond methodological contributions, this work underscores the broader implications of adaptive multi-population strategies in real-world domains. Since the RWM0P50 problem is derived from power distribution system planning, the results suggest that MCCMO-ADP has strong potential in renewable energy optimization, logistics planning, and other resource-constrained domains. At the same time, we acknowledge certain limitations: the evaluation was restricted to one benchmark class, and while comparisons were made against MCCMO variants, further benchmarking against canonical algorithms (e.g., NSGA-II, NSGA-III, MOEA/D variants, and coevolutionary approaches) remains a necessary step.

7 Future Work

Future research will therefore extend MCCMO-ADP to broader benchmark suites, including CEC-style CMOP test functions and MaF problems, and will incorporate application-specific validations in healthcare treatment planning, renewable energy scheduling, and large-scale HPC workload management. The new variant will also be tested and validated on the broader synthetic benchmark suites such as Congress on Evolutionary Computation Constrained Multi-Objective Optimization Problems (CEC CMOPs), Deb–Thiele–Laumanns–Zitzler (DTLZ) and Walking Fish Group Test Suite (WFG) to further generalize its effectiveness. These directions will strengthen the generalizability and applicability of MCCMO-ADP, further consolidating its role as a promising approach for solving constrained multi-objective optimization problems in both academic and industrial contexts.

Acknowledgement: The authors thank Hamdard University for the technical support during this research.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Muhammad Waqar Khan (Principal Author) conceived the research idea, designed the overall methodology, developed the MCCMO-ADP algorithm, performed the simulations, implemented the experiments, analyzed the results, and also drafted the initial manuscript. Adnan Ahmed Siddiqui (Co-Author) provided the supervision to the principal author during the phase of research design, reviewing the algorithm, its implementation and simulations critically and contributed in the manuscript refinement. Syed Sajjad Hussain Rizvi (Co-Author) supervised the overall research direction, validated and verified the results and the comparative analysis and also improved the clarity of the manuscript. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data used in this research are available from the corresponding author upon the reasonable request.

Ethics Approval: Not applicable. This study did not involve any human or animal subjects, and ethical approval was not required.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Cao B, Zhao J, Lv Z, Liu X, Yang S, Kang X, et al. Distributed parallel particle swarm optimization for multi-objective and many-objective large-scale optimization. IEEE Access. 2017;5:8214–21. doi:10.1109/access.2017.2702561.
2. Zhao S, Jia H, Li Y, Shi Q. A constrained multi-objective optimization algorithm with a population state discrimination model. Mathematics. 2025;13(5):688. doi:10.3390/math13050688.
3. Zhou J, Zhang Y, Yu F, Yang X, Suganthan PN. A staged fuzzy evolutionary algorithm for constrained large-scale multiobjective optimization. Appl Soft Comput. 2024;167(1):112297. doi:10.1016/j.asoc.2024.112297.
4. Wang Y, Liu Y, Zou J, Zheng J, Yang S. A novel two-phase evolutionary algorithm for solving constrained multi-objective optimization problems. Swarm Evol Comput. 2022;75(3):101166. doi:10.1016/j.swevo.2022.101166.
5. Kordestani JK, Mirsaleh MR, Rezvanian A, Meybodi MR. An overview of multi-population methods for dynamic environments. Adv Learn Autom Intell Optim. 2021;208:253–86. doi:10.1007/978-3-030-76291-9_7.
6. Limmer S, Fey D. Comparison of common parallel architectures for the execution of the island model and the global parallelization of evolutionary algorithms. Concurr Comput Pract Exp. 2017;29(9):e3797. doi:10.1002/cpe.3797.
7. Li K, Chen R, Fu G, Yao X. Two-archive evolutionary algorithm for constrained multi-objective optimization. arXiv:1711.07907. 2017.

8. Yu K, Wang L, Liang J, Wang H, Qiao K, Liang T. Constraint subsets-based evolutionary multitasking for constrained multiobjective optimization. Swarm Evol Comput. 2024;86(2):101531. doi:10.1016/j.swevo.2024.101531.
9. Ming F, Gong W, Wang L, Gao L. Constrained multiobjective optimization via multitasking and knowledge transfer. IEEE Trans Evol Comput. 2024;28(1):77–89. doi:10.1109/TEVC.2022.3230822.
10. Zhou J, Zhang Y, Zheng J, Li M. Domination-based selection and shift-based density estimation for constrained multiobjective optimization. IEEE Trans Evol Comput. 2023;27(4):993–1004. doi:10.1109/TEVC.2022.3190401.
11. Li X, Li X, Wang K, Yang S, Li Y. Achievement scalarizing function sorting for strength Pareto evolutionary algorithm in many-objective optimization. Neural Comput Appl. 2021;33(11):6369–88. doi:10.1007/s00521-020-05398-1.
12. Khan W, Salhi A, Asif M, Sulaiman M, Adeeb R, Algarni A. Evolutionary algorithms based on decomposition and indicator functions: state-of-the-art survey. Int J Adv Comput Sci Appl. 2016;7(2):583–93. doi:10.14569/ijacsa.2016.070274.
13. Martínez SZ, Coello Coello CA. A direct local search mechanism for decomposition-based multi-objective evolutionary algorithms. In: 2012 IEEE Congress on Evolutionary Computation; 2012 Jun 10–15; Brisbane, QLD, Australia. IEEE; 2012. p. 1–8. doi:10.1109/CEC.2012.6252990.
14. Escobar JJ, Sánchez-Cuevas P, Prieto B, Kızıltepe RS, Díaz-del-Río F, Kimovski D. Energy-time modelling of distributed multi-population genetic algorithms with dynamic workload in HPC clusters. Future Gener Comput Syst. 2025;167(3):107753. doi:10.1016/j.future.2025.107753.
15. Kumar A, Wu G, Ali MZ, Luo Q, Mallipeddi R, Suganthan PN, et al. A Benchmark-Suite of real-World constrained multi-objective optimization problems and some baseline results. Swarm Evol Comput. 2021;67(4):100961. doi:10.1016/j.swevo.2021.100961.
16. Zou J, Sun R, Liu Y, Hu Y, Yang S, Zheng J, et al. A multipopulation evolutionary algorithm using new cooperative mechanism for solving multiobjective problems with multiconstraint. IEEE Trans Evol Comput. 2024;28(1):267–80. doi:10.1109/TEVC.2023.3260306.