

Adaptive Neighborhood Grid-Boundary Multi-Objective Mayfly Algorithm for Improved Pareto Front Approximation

Nicholas Kwesi Prah II , Emmanuel Asuming Frimpong , and Francis Boafo Effah 

Department of Electrical and Electronic Engineering, College of Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, Ghana

Submitted: 13 July 2026
Accepted : 6 October 2026
Online First: 13 October 2026

Corresponding author
Nicholas Kwesi Prah II
bravokwesi@gmail.com

DOI:10.64470/elene.2026.42

RESEARCH ARTICLE

© Copyright, Authors,
Distributed under Creative
Commons CC-BY 4.0

Abstract: Multi-objective optimization algorithms are widely used for solving engineering design problems involving conflicting objectives. The Multi-Objective Mayfly Algorithm (MOMA) is a promising swarm-based optimizer, but its performance can be limited by fixed search parameters, premature convergence, weak archive distribution, and loss of extreme trade-off solutions during archive truncation. This paper proposes an Adaptive Neighborhood Grid-Boundary Multi-Objective Mayfly Algorithm (ANGB-MOMA) to improve the convergence and diversity behavior of the original MOMA. The proposed method introduces four mechanisms: adaptive parameter control, neighborhood-guided leader selection, grid-based archive and leader selection, and boundary-preserving archive truncation. The algorithm was evaluated using ZDT and DTLZ benchmark functions and compared with Original MOMA, Levy-flight MOMA, Stochastic-inertia MOMA, NSGA-II, MOPSO, and MOEA/D. The experimental study also included ablation analysis, parameter sensitivity analysis, Holm-corrected Wilcoxon statistical tests, metric reproducibility settings, and runtime assessment. The results show that ANGB-MOMA significantly improves the original MOMA and related MOMA variants on most tested functions. Strong improvements were observed on ZDT1, ZDT2, ZDT5, ZDT6, DTLZ4, and DTLZ7, with particularly large IGD+ reductions on ZDT5 and ZDT6. ANGB-MOMA also achieved the best overall performance on DTLZ4 among all compared algorithms. However, the comparison shows that the proposed method is not universally superior to established optimizers, as MOPSO and MOEA/D performed better on some discontinuous, discrete, and multimodal benchmark cases. The results therefore support ANGB-MOMA as a reproducible and effective improvement to the MOMA framework, while also identifying ZDT3 and ZDT4 as important limitation cases for future development.

Keywords Multi-objective optimization, Mayfly algorithm, Pareto front, Swarm intelligence

1. Introduction

Optimization sits at the core of engineering design, energy systems, scheduling, machine learning, and control, where decision variables must be chosen well despite technical and operational limits (Deb, 2001; Talbi, 2009). Real design problems rarely come down to a single objective. More often, several competing goals have to be satisfied at once. In power systems, for instance, cutting operating cost tends to work against emission reduction, voltage quality, and loss minimization (Nagarajan et al., 2024; Zhu et al., 2024). Problems of this kind are framed as multi-objective optimization problems, where the goal shifts from finding one best answer to mapping out a set of trade-offs, commonly known as the Pareto-optimal set (Coello Coello

et al., 2007; Zitzler et al., 2000).

Classical mathematical programming works well when a problem is smooth, convex, differentiable, and analytically tractable (Boyd & Vandenberghe, 2004), but most real engineering problems are not so cooperative. They tend to be nonlinear, nonconvex, discontinuous, high-dimensional, or constrained in ways that trap deterministic solvers in local optima or make them impractical as the problem dimension grows (Goldberg, 1989). This is largely why population-based metaheuristics have taken over for these harder problems: they need no gradient information and instead search through stochastic exploration and exploitation (Holland, 1975; Yang, 2010).

Within this population-based family, evolutionary algorithms and swarm-intelligence methods have drawn the most attention: Genetic Algorithm (GA), Differential Evolution (DE), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Firefly Algorithm (FA), Grey Wolf Optimizer (GWO), Whale Optimization Algorithm (WOA), and a number of other nature-inspired methods have all found use in engineering optimization (Dorigo et al., 1996; Kennedy & Eberhart, 1995; Mirjalili & Lewis, 2016; Mirjalili et al., 2014; Mirjalili et al., 2017; Storn & Price, 1997; Yang, 2009). Most draw their inspiration from biological evolution, foraging behavior, social cooperation, mating, or some form of collective intelligence, and what tends to make them effective is the balance struck between global exploration and local exploitation. Still, the No-Free-Lunch theorem is a reminder that no single algorithm wins everywhere (Wolpert & Macready, 1997) which is exactly why new variants keep appearing.

Handling several conflicting objectives at once calls for mechanisms that go beyond what single-objective algorithms offer. Because no single solution can represent the whole trade-off surface, multi-objective methods generally lean on Pareto dominance, non-dominated sorting, external archives, leader selection, and some way of preserving diversity (Coello Coello et al., 2007; Srinivas & Deb, 1994; Zitzler et al., 2000). NSGA, NSGA-II, SPEA, SPEA2, PAES, and MOPSO are the classical reference points here, each contributing a framework for ranking non-dominated solutions and keeping the Pareto front well spread (Coello Coello et al., 2004; Deb, Pratap, et al., 2002; Knowles & Corne, 2000; Srinivas & Deb, 1994; Zitzler & Thiele, 1999; Zitzler et al., 2001). NSGA-II remains influential for pairing fast non-dominated sorting with elitism and crowding distance (Deb, Pratap, et al., 2002), while MOPSO's contribution was the external repository of non-dominated particles used to steer the swarm (Coello Coello et al., 2004).

None of this makes multi-objective optimization easy, though. A good optimizer needs to converge toward the true Pareto front and keep its solutions well spread across that front at the same time (Knowles & Corne, 2002; Zitzler et al., 2000), and the two goals pull against each other: push too hard for convergence and solutions bunch up in one spot; push too hard for diversity and convergence stalls. Discontinuous, biased, multimodal, and many-objective problems make this tension worse, since the optimizer then has to hold onto several disconnected regions of solutions while still moving toward the true front (Deb, Thiele, et al., 2002; Huband et al., 2006; Knowles & Corne, 2002). That is part of why the ZDT and DTLZ benchmark suites remain a standard testing ground between them they cover a wide enough range of Pareto-front shapes and difficulties to stress-test an algorithm properly (Deb, Thiele, et al., 2002; Huband et al., 2006).

The Mayfly Algorithm (MA), proposed fairly recently, draws its search behavior from how mayflies fly and mate (Zervoudakis & Tsafarakis, 2020). It blends ideas from swarm intelligence and evolutionary computation: male and female sub-populations, velocity-based movement, personal-best memory, global-best guidance, a nuptial dance, random flight, and mating with crossover to produce offspring (Zervoudakis & Tsafarakis, 2020). Between them, these components give the algorithm both exploration and exploitation. The same original study also pushed MA into multi-objective territory by adding a repository of non-dominated solutions, which is what turned it into the Multi-Objective Mayfly Algorithm (MOMA) sometimes written elsewhere as the multi-objective mayfly algorithm (Zervoudakis & Tsafarakis, 2020).

MOMA's external repository holds the non-dominated solutions found during the search, and since multi-objective optimization has no single global best, the leader is instead drawn from this repository

(Zervoudakis & Tsafarakis, 2020). Non-dominated sorting and crowding distance handle ranking and truncation once the repository fills up, and this setup is enough to make MOMA workable for Pareto-front approximation. It is not without problems, though. Performance is sensitive to manually chosen parameters attraction coefficients, random flight factors, velocity terms; leader selection, if done randomly or weakly, tends to keep pointing mayflies back at already-crowded parts of the archive; crowding distance itself loses some of its effectiveness as the number of objectives grows; and the population can converge prematurely once it loses diversity too early in the run (Zervoudakis & Tsafarakis, 2020).

A handful of recent studies have tried to push MA or MOMA further by changing how mayflies move, explore, handle constraints, or pick leaders. Nagarajan et al. (2024) added Levy flight to MA to improve optimal power flow with integrated solar and wind generation, using the longer jumps of Levy flight to help the algorithm escape local traps and search more globally. Zhu et al. (2024) went a different route with NSMA-SF, an enhanced multi-objective Mayfly Algorithm for stochastic optimal power flow with renewable sources, combining fast non-dominated sorting, a nonlinear decreasing inertia weight, and a feasibility-based rule for handling constraints. Between them, these two studies make a reasonable case that MA/MOMA responds well to better parameter control, better movement rules, and better solution ranking.

Outside the MA literature specifically, swarm intelligence research more broadly backs up the idea that local, neighborhood-based guidance helps. PSO variants built around local-best or neighborhood structures have been used to fight premature convergence and keep swarms diverse (Kennedy, 1999; Mendes et al., 2004). One recent example uses k-means clustering to define local-best particles in a murmuration-inspired PSO, letting particles move toward a nearby local leader as well as their personal and global bests (Twumasi et al., 2024). What makes this relevant here is the underlying lesson: leaning on local leaders, not just one global one, helps a population avoid getting stuck and keeps exploration and exploitation in better balance.

Archive management is a problem in its own right. External archives are valuable precisely because they preserve elite non-dominated solutions, but a careless truncation rule can just as easily throw away sparse-region or boundary solutions that matter (Coello Coello et al., 2004; Ishibuchi et al., 2015). Crowding distance, despite being the standard tool here, does not always hold the distribution together once the objective space turns sparse, irregular, or discontinuous (Deb, Pratap, et al., 2002; Zervoudakis & Tsafarakis, 2020). Grid-based schemes offer a different angle: split the objective space into cells and bias leader selection toward the less populated ones (Coello Coello et al., 2004; Li et al., 2014). Done this way, archive control tends to cut down on clustering and leaves the final Pareto-front approximation more evenly spread.

The extreme solutions on a Pareto front matter for a simple reason: each one represents the best achievable value for a single objective, so losing them during truncation shrinks the reported trade-off range. Crowding distance does give boundary points extra weight in some implementations, but that protection can quietly fail if archive update and leader selection were never explicitly built to keep extremes around. Building in explicit protection for boundary solutions, rather than relying on crowding distance to do it implicitly, tends to give wider front coverage and a more complete set of trade-offs (Deb, Pratap, et al., 2002; Zervoudakis & Tsafarakis, 2020).

Based on the reviewed literature, the original MOMA provides a useful foundation for multi-objective optimization, but it still leaves room for improvement. Existing variants have mostly focused on one improvement direction at a time, such as Levy-flight exploration, nonlinear inertia, or stochastic search enhancement (Nagarajan et al., 2024; Zhu et al., 2024). However, fewer studies have jointly addressed adaptive search control, neighborhood-based leader diversification, archive density control, and preservation of identified extreme archive solutions within one unified MOMA framework.

To address this gap, this paper proposes an Adaptive Neighborhood Grid-Boundary Multi-Objective Mayfly Algorithm (ANGB-MOMA). The proposed algorithm modifies the original MOMA using four coordinated mechanisms. First, adaptive parameter control is introduced to adjust the search behavior according to the current success rate and population diversity. Second, neighborhood-guided leader selection

is introduced to reduce overdependence on a single global archive leader. Third, grid-based archive and leader selection is used to encourage exploration of sparse objective-space regions. Fourth, boundary-preserving archive truncation is introduced to reduce the loss of currently identified objective-wise extreme solutions during archive reduction.

The study does not claim that the proposed method universally dominates all existing multi-objective optimizers. Instead, the aim is to determine whether the MOMA framework can be strengthened through adaptive search control, neighborhood guidance, grid-based archive management, and boundary-preserving truncation. The experiments therefore evaluate both the improvement of ANGB-MOMA over the original MOMA and its competitiveness against established algorithms such as NSGA-II, MOPSO, and MOEA/D. The main contributions of this paper are summarized as follows:

1. An Adaptive Neighborhood Grid-Boundary Multi-Objective Mayfly Algorithm (ANGB-MOMA) is proposed to improve the original MOMA through four coordinated mechanisms: adaptive parameter control, neighborhood-guided leader selection, grid-based archive and leader selection, and boundary-preserving archive truncation.
2. An adaptive parameter-control strategy is introduced to adjust the nuptial dance, random flight, inertia, visibility, attraction coefficients, and local-leader probability using search success rate and population diversity.
3. A neighborhood-guided leader selection mechanism is incorporated using a ring topology to reduce overdependence on a single global archive leader and improve the balance between local exploitation and global exploration.
4. A grid-based archive and leader selection strategy is developed to bias leader selection toward sparse objective-space regions and improve Pareto-front coverage.
5. A boundary-preserving archive truncation mechanism is introduced to protect the currently identified objective-wise extreme archive solutions during archive reduction. This improves the retention of trade-off boundary solutions, while acknowledging that these are identified archive extremes rather than guaranteed true Pareto-front extremes.
6. A detailed experimental evaluation is performed using ZDT and DTLZ benchmark functions, including ZDT4 and ZDT5 clarification. The study also includes ablation experiments, parameter sensitivity analysis, statistical testing using the Wilcoxon signed-rank test with Holm-Bonferroni correction, metric reproducibility settings, and runtime analysis.
7. The proposed method is compared not only with Original MOMA and recent MOMA variants, but also with established multi-objective optimizers including NSGA-II, MOPSO, and MOEA/D. This provides a more balanced assessment of the strengths and limitations of ANGB-MOMA.

The remainder of this paper is organized as follows. Section 2 presents the theoretical background of multi-objective optimization and reviews the original MOMA and related variants. Section 3 describes the proposed ANGB-MOMA in detail, including adaptive parameter control, neighborhood-guided leader selection, grid-based archive management, boundary-preserving truncation, benchmark configuration, metric calculation, statistical testing, and computational complexity. Section 4 presents experimental results and discussion, including comparative performance analysis, ablation study, parameter sensitivity analysis, statistical significance testing, runtime analysis, and convergence behavior. Section 5 concludes the paper, while Section 6 presents the limitations and future research directions.

2. Theoretical Background

2.1. Original Mayfly Algorithm and MOMA search behavior

The Mayfly Algorithm (MA) is a population-based metaheuristic inspired by the flight and mating behavior of mayflies. The algorithm divides the population into male and female mayflies. Male mayflies are mainly responsible for directed movement toward promising regions of the search space, while female mayflies

introduce additional movement patterns that support exploration and mating-based solution generation.

In the original single-objective MA, each male mayfly updates its position using information from its personal best position and the global best solution. The personal best represents the best position previously visited by the individual mayfly, while the global best represents the best solution found by the entire population. This mechanism allows the algorithm to combine individual memory and population-level guidance.

For multi-objective optimization, however, a single global best solution cannot be directly defined because several trade-off solutions may be valid at the same time. Therefore, the Multi-Objective Mayfly Algorithm (MOMA) replaces the single global best with a leader selected from an external archive of non-dominated solutions. The archive stores elite Pareto solutions obtained during the search, and these archived solutions guide the movement of mayflies toward the Pareto front.

In MOMA, the quality of solutions is evaluated using Pareto dominance. A solution is preferred if it is non-dominated by other candidate solutions. Non-dominated sorting is used to classify the population into Pareto fronts, while a diversity-preserving mechanism such as crowding distance is usually used to maintain solution spread. The search process therefore aims to achieve two simultaneous goals: convergence toward the true Pareto front and diversity along the front.

Put simply, male mayflies head toward their personal best position and whichever archive leader they have been given, while female mayflies follow the males plus some random flight of their own. Mating then produces offspring, which get evaluated and folded back into the population, and whatever non-dominated solutions survive that round end up retained in the external archive. The general workflow of the original MOMA is summarized in Algorithm 1.

The original MOMA is a workable framework, but it carries several issues that limit it in practice. Fixed search parameters cannot give the right exploration-exploitation balance across an entire run, since what the search needs early on is not what it needs later. Picking leaders randomly, or with only weak guidance, tends to keep pulling the population back toward already-crowded archive regions. The population can converge prematurely once most mayflies end up following similar leaders. And archive truncation, if diversity isn't handled with care, can quietly discard boundary or sparse-region solutions that were worth keeping. These four problems are essentially what motivated the adaptive, neighborhood-based, grid-guided, and boundary-preserving mechanisms built into ANGB-MOMA.

2.2. Adaptive parameter control

Most metaheuristics run with the same fixed parameter values from start to finish, but what a search actually needs shifts as it progresses heavier exploration early on, to discover promising regions, and heavier exploitation later, to polish solutions that are already close to the Pareto front. Adaptive parameter control is the response to that mismatch: it lets exploration and exploitation shift in proportion to how the search is actually going, rather than locking the balance in from the start.

In the proposed ANGB-MOMA, adaptive control is used to adjust important search parameters according to the current search condition. The adaptation is guided by indicators such as improvement rate, population diversity, and iteration progress. If the population shows little improvement or begins to lose diversity, the algorithm increases exploration. If the population is improving steadily, the algorithm increases exploitation.

A general adaptive parameter update can be expressed as in Eq. (1):

$$\theta_{t+1} = \theta_t + \Delta\theta_t \quad (1)$$

where θ_t is a search parameter at iteration t , and $\Delta\theta_t$ is the adaptive adjustment applied at that iteration.

For example, if the success rate of the population is lower than the desired target, exploration-related parameters may be increased in Eq. (2):

$$\theta_{\text{exp},t+1} = \theta_{\text{exp},t} + \alpha(S_{\text{target}} - S_t) \quad (2)$$

where $\theta_{\text{exp},t}$ represents an exploration parameter, S_t is the current success rate, S_{target} is the target success rate, and α is an adaptation gain.

Similarly, when the success rate is high, exploitation-related parameters may be strengthened in Eq. (3):

$$\theta_{\text{exc},t+1} = \theta_{\text{exc},t} + \beta(S_t - S_{\text{target}}) \quad (3)$$

where $\theta_{\text{exc},t}$ represents an exploitation parameter and β controls the adaptation strength.

The effect is that the algorithm no longer leans entirely on parameters fixed in advance, it can react to whatever state the search is actually in, increasing exploration as soon as the population starts to stagnate rather than after the fact.

Algorithm 1. General Pseudocode of Original MOMA

```

Input:
  Objective functions F(x)
  Population size N
  Maximum number of iterations MaxIt
  Decision variable bounds
  Archive size

Output:
  External archive of non-dominated solutions

1. Initialize male and female mayfly populations randomly within the bounds.
2. Evaluate the objective functions of all mayflies.
3. Identify non-dominated solutions and initialize the external archive.
4. Assign personal best positions to mayflies.

5. For iteration t = 1 to MaxIt do
  5.1 Select leader solutions from the external archive.

  5.2 For each male mayfly do
    Update velocity using personal best and selected archive leader.
    Update position.
    Apply boundary control.
    Evaluate objective functions.
    Update personal best using Pareto dominance.
  End for

  5.3 For each female mayfly do
    Update movement based on attraction to male mayflies and random flight behavior.
    Apply boundary control.
    Evaluate objective functions.
    Update personal best using Pareto dominance.
  End for

  5.4 Apply mating/crossover between selected male and female mayflies.
  5.5 Evaluate offspring solutions.
  5.6 Combine parent and offspring populations.
  5.7 Perform non-dominated sorting.
  5.8 Update the external archive with non-dominated solutions.
  5.9 If archive size exceeds its limit, apply diversity-based truncation.

End for

6. Return the final archive as the approximated Pareto front.

```

2.3. Neighborhood topology

Neighborhood topology exists to break the population's dependence on one global leader. When every search agent is steered by the same elite solution, or by leaders all pulled from the same corner of the archive, convergence can speed up, but at the cost of diversity, and often premature convergence as a result.

In ANGB-MOMA, a neighborhood topology is introduced so that each mayfly can be influenced by a local leader selected from its nearby mayflies. This creates several local search regions within the population. Instead of forcing the entire population to move toward the same region, different neighborhoods can explore different parts of the search space.

A simple ring neighborhood can be defined as in Eq. (4):

$$\mathcal{N}_i = \{i - r, \dots, i, \dots, i + r\} \quad (4)$$

where $\mathcal{N}_i$ is the neighborhood of the i -th mayfly, and r determines the number of neighboring mayflies on each side. If the neighborhood size is k , then in Eq. (5):

$$r = \frac{k-1}{2} \quad (5)$$

The local leader of the i_{th} mayfly is then selected from the best or non-dominated solutions within $\mathcal{N}_i$. The mayfly movement can therefore be guided by both local and global information. A general leader-guided update may be represented as in Eq. (6):

$$\mathbf{v}_{i,t+1} = w\mathbf{v}_{i,t} + a_1 r_1 (\mathbf{p}_i^{\text{best}} - \mathbf{x}_{i,t}) + a_2 r_2 (\mathbf{L}_i - \mathbf{x}_{i,t}) \quad (6)$$

where $\mathbf{v}_{i,t}$ is the velocity of the i -th mayfly, $\mathbf{x}_{i,t}$ is its position, $\mathbf{p}_i^{\text{best}}$ is its personal best position, $\mathbf{L}_i$ is a selected leader, w is a weighting factor, a_1 and a_2 are attraction coefficients, and r_1 and r_2 are random numbers.

The leader $\mathbf{L}_i$ may be selected either from the local neighborhood or from the external archive. This improves the diversity of search directions and allows multiple regions of the Pareto front to be explored at the same time.

2.4. Grid-based archive and leader selection

The external archive holds whatever non-dominated solutions the search has found so far, but it can only hold so many, and managing it matters let too many solutions pile up in one region and the final Pareto-front approximation ends up poorly distributed. Grid-based archive management addresses this directly by improving how solutions are spread out across the objective space.

In grid-based archiving, the objective space is divided into a number of grid cells. Each archived solution is assigned to a grid cell according to its objective values. The number of solutions in each cell represents the density of that region. Sparse cells contain fewer solutions, while crowded cells contain many solutions.

If G grid divisions are used for each objective and the problem has m objectives, the number of possible grid cells is in Eq. (7):

$$G^m \quad (7)$$

Let n_c be the number of solutions in grid cell c . A leader selection probability can be assigned to each non-empty grid cell as in Eq. (8):

$$P_c = \frac{(n_c)^{-\gamma}}{\sum_{j=1}^{N_c} (n_j)^{-\gamma}} \quad (8)$$

where N_c is the number of non-empty grid cells, and γ controls the strength of preference for sparse cells.

In practice, this skews leader selection toward sparse regions of the Pareto front, which in turn pulls the

population toward the parts of the objective space that were previously underrepresented and improves how evenly non-dominated solutions end up distributed.

The same grid structure governs truncation once the archive exceeds its size limit rather than removing solutions at random, the algorithm preferentially clears out the most crowded cells, which keeps clustering down and the archive's spread intact.

The grid mechanism is therefore doing two jobs at once: steering leader selection toward sparse regions, and steering truncation away from them. Combined, these keep MOMA from concentrating its solutions in a narrow slice of objective space and give better coverage of the Pareto front overall.

2.5. Boundary preservation

Extreme solutions matter because each one is the best achievable value for a single objective, so boundary preservation exists specifically to stop them from disappearing during archive truncation. Lose them, and the reported Pareto front shrinks below what the algorithm actually found.

For a minimization problem with m objectives, the boundary solution for the j -th objective is defined as in Eq. (9):

$$\mathbf{x}^{(j)} = \operatorname{argmin}_{\mathbf{x} \in A} f_j(\mathbf{x}) \quad (9)$$

where A is the external archive.

The set of protected boundary solutions is in Eq. (10):

$$\mathcal{B} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(m)}\} \quad (10)$$

During archive truncation, the solutions in $\mathcal{B}$ are protected from deletion. After these boundary solutions are preserved, grid-based truncation is applied to the remaining archive members. This ensures that the archive maintains the extreme regions of the Pareto front while still controlling overcrowding in dense regions.

This matters because crowding distance and grid density, on their own, can still end up discarding a boundary solution if it happens to sit in a dense cell. Protecting it explicitly closes that gap, leaving ANGB-MOMA with a wider approximation of the Pareto front and a final archive that better represents the true trade-off range.

Grid control and boundary preservation are meant to work as a pair here, one manages density across the front, the other guards the extremes, so that ANGB-MOMA ends up preserving both the central and the outer regions of the Pareto front rather than trading one off against the other.

3. Methodology

3.1. Overview of the proposed ANGB-MOMA

This study proposes the Adaptive Neighborhood Grid-Boundary Multi-Objective Mayfly Algorithm (ANGB-MOMA), an improved version of the original Multi-Objective Mayfly Algorithm (MOMA). The baseline MOMA procedure is used as the reference structure, while the proposed ANGB-MOMA introduces four mechanisms into the original MOMA framework. These are adaptive parameter control, neighborhood-guided search, grid-based archive and leader selection, and boundary-preserving archive truncation.

Each mechanism targets a specific limitation of the original algorithm. Adaptive parameter control reduces dependence on fixed search parameters. Neighborhood guidance reduces overdependence on a single global archive leader. Grid-based archive control improves the distribution of non-dominated solutions in the objective space. Boundary preservation protects the identified extreme solutions currently present in the external archive during archive truncation.

The search process starts by initializing male and female mayfly populations within the decision-variable bounds. Each candidate solution is evaluated using the selected multi-objective benchmark function. Non-

dominated solutions are stored in an external archive. During each iteration, mayflies are guided by personal best positions, local neighborhood leaders, and global archive leaders. Crossover and mutation are then applied to generate additional candidate solutions. The archive is updated using Pareto dominance, grid-based density information, and boundary-preserving truncation. The final archive represents the approximated Pareto front.

3.2. Benchmark representation and objective evaluation

The proposed ANGB-MOMA was evaluated using ZDT and DTLZ benchmark problems. The ZDT functions were used to assess bi-objective optimization performance, while the DTLZ functions were used to examine the behavior of the algorithm on higher-dimensional decision spaces. All benchmark problems were treated as two-objective minimization problems.

For ZDT1, ZDT2, ZDT3, and ZDT6, ten continuous decision variables were used, with all variables bounded in the interval $[0,1]$. For ZDT4, ten decision variables were used, where the first variable was bounded in $[0,1]$ and the remaining variables were bounded in $[-5,5]$. For DTLZ1 and DTLZ4, eleven continuous decision variables were used. For DTLZ7, twenty-one continuous decision variables were used. All DTLZ decision variables were bounded in $[0,1]$.

ZDT5 requires special treatment because its original formulation is binary/discrete rather than purely continuous. In this study, ZDT5 was represented using eighty real-coded variables in $[0,1]$. Before evaluating the objective function, the real-coded vector was converted into a binary string using a threshold rule. A variable value greater than or equal to 0.5 was mapped to binary 1, while a variable value less than 0.5 was mapped to binary 0. The first thirty bits formed the first binary block, and the remaining fifty bits were divided into ten five-bit blocks.

For ZDT5, the first objective was computed as:

$$f_1 = 1 + u(b_1)$$

where $u(b_1)$ is the number of ones in the first thirty-bit block. For the remaining ten five-bit blocks, the auxiliary function was defined as:

$$v(b_i) = 2 + u(b_i), \quad i = 2, 3, \dots, 11$$

and

$$g = \sum_{i=2}^{11} v(b_i).$$

The second objective was then computed as:

$$f_2 = \frac{g}{f_1}.$$

The discrete reference Pareto front for ZDT5 was generated using:

$$f_1 \in \{1, 2, \dots, 31\}, \quad f_2 = \frac{20}{f_1}.$$

This corresponds to the case where the ten five-bit blocks are all zeros. This clarification is important because the raw objective scale of ZDT5 differs from the continuous ZDT problems. Therefore, all HV and IGD+ calculations were performed in normalized objective space, as described in Section 3.13.

3.3. Population initialization

Let the position vector of the i -th mayfly be defined as:

$$\mathbf{x}_i = [x_{i,1}, x_{i,2}, \dots, x_{i,D}]$$

where D is the number of decision variables. Each decision variable was initialized uniformly within its lower and upper bounds as:

$$x_{i,j} = x_j^{\min} + r(x_j^{\max} - x_j^{\min})$$

where $x_j^{\min}$ and $x_j^{\max}$ are the lower and upper bounds of the j -th decision variable, and r is a uniformly distributed random number in $[0,1]$.

The population was divided into male and female mayflies. In all MOMA-based algorithms, the number of male mayflies was 20 and the number of female mayflies was 20, giving a total population size of 40. The initial velocity of each mayfly was set to zero. Each mayfly was evaluated using the selected benchmark objective function, and its initial position was stored as its personal best position.

After initialization, the male and female populations were combined and non-dominated sorting was performed. The initial external archive was formed using the first non-dominated front. If the number of non-dominated solutions exceeded the archive limit, archive truncation was applied.

3.4. External archive and non-dominated sorting

The external archive stores elite non-dominated solutions discovered during the search. At each iteration, the current male population, female population, crossover offspring, mutation offspring, and previous archive members are merged. Pareto dominance is then used to identify the non-dominated solutions.

For two minimization solutions $\mathbf{x}_a$ and $\mathbf{x}_b$, $\mathbf{x}_a$ dominates $\mathbf{x}_b$ if:

$$f_j(\mathbf{x}_a) \leq f_j(\mathbf{x}_b), \quad \forall j = 1, 2, \dots, m$$

and

$$f_l(\mathbf{x}_a) < f_l(\mathbf{x}_b)$$

for at least one objective l , where m is the number of objectives.

After dominance comparison, non-dominated sorting is used to classify candidate solutions into fronts. Crowding distance is calculated within each front to support diversity preservation. The external archive is updated using the non-dominated solutions obtained from the combined population. In the original MOMA, archive reduction is based mainly on diversity ranking. In ANGB-MOMA, the archive is managed using grid-based density information and boundary-preserving truncation.

3.5. Adaptive parameter control

In the original MOMA, important search parameters are either fixed or reduced using simple damping factors. However, fixed parameters may not provide a suitable balance between exploration and exploitation throughout the optimization process. ANGB-MOMA therefore introduces adaptive parameter control based on search success and population diversity.

The success rate at iteration t is defined as:

$$S(t) = \frac{N_{\text{imp}}(t)}{N_m + N_f}$$

where $N_{\text{imp}}(t)$ is the number of mayflies whose personal-best solutions improved during iteration t , N_m is the number of male mayflies, and N_f is the number of female mayflies. The target success rate was set to:

$$S_{\text{target}} = 0.20.$$

The error between the target success rate and the current success rate is:

$$e(t) = S_{\text{target}} - S(t).$$

A positive value of $e(t)$ indicates insufficient improvement and therefore increases the exploration tendency of the algorithm. A negative value indicates that the population is improving satisfactorily, allowing stronger exploitation.

The decision-space diversity was estimated as:

$$D_{\text{pop}}(t) = \frac{1}{D} \sum_{j=1}^D \frac{\text{std}(x_{1,j}, x_{2,j}, \dots, x_{N,j})}{\text{mean}(\mathbf{x}^{\text{max}} - \mathbf{x}^{\text{min}})}.$$

The adaptive update rules used in ANGB-MOMA were:

$$\text{dance}(t+1) = \text{clamp}[\text{dance}(t)\exp(0.35e(t)), 0.05, 1.50]$$

$$fl(t+1) = \text{clamp}[fl(t)\exp(0.35e(t)), 0.05, 1.50]$$

$$g(t+1) = \text{clamp}[g(t)\exp(0.20e(t)), 0.20, 0.95]$$

$$a_q(t+1) = \text{clamp}[a_q(t)\exp(0.15e(t)), 0.50, 2.50], \quad q = 1, 2, 3.$$

The visibility coefficient β was adjusted using the population diversity condition:

$$\beta(t+1) = \begin{cases} \text{clamp}[0.98\beta(t), 0.50, 5.00], & D_{\text{pop}}(t) < 0.15, \\ \text{clamp}[1.01\beta(t), 0.50, 5.00], & D_{\text{pop}}(t) \geq 0.15. \end{cases}$$

The local leader selection probability was also adapted as:

$$p_{\text{local}}(t+1) = \text{clamp}\left[p_{\text{local}}(t) - \frac{0.20}{\text{MaxIt}} + 0.40(0.20 - D_{\text{pop}}(t)) + 0.30e(t), 0.20, 0.90\right].$$

These adaptive rules allow ANGB-MOMA to increase exploration when improvement is weak or diversity becomes low. They also allow stronger exploitation when the population is improving steadily. This reduces dependence on fixed manually selected parameters and improves the responsiveness of the search process.

3.6. Neighborhood-guided leader selection

The neighborhood-guided search mechanism is introduced to reduce the possibility that the entire mayfly population will converge toward the same archive region. In the original MOMA, the movement of mayflies is mainly guided by personal best positions and archive leaders. If most archive leaders are selected from a dense region of the objective space, the population may lose diversity and converge prematurely. ANGB-MOMA therefore combines local neighborhood guidance with global archive guidance.

A ring topology was used in this study because it is simple, deterministic, and computationally inexpensive. For the i -th mayfly, the neighborhood is defined as:

$$\mathcal{N}_i = \{i-r, \dots, i, \dots, i+r\}$$

where r is the radius of the neighborhood. If the neighborhood size is k , then:

$$r = \frac{k-1}{2}.$$

The indices are wrapped around the population so that the first and last mayflies are also connected. This prevents boundary individuals in population order from having fewer neighbors than others.

Within each neighborhood, candidate local leaders are selected from the best-ranked or non-dominated

mayflies. When several candidates are available, the one with the better diversity contribution is preferred. The leader used by the i -th mayfly is selected probabilistically as:

$$\mathbf{L}_i = \begin{cases} \mathbf{L}_i^{local}, & rand < p_{local}(t), \\ \mathbf{L}_i^{archive}, & rand \geq p_{local}(t) \end{cases}$$

where $\mathbf{L}_i^{local}$ is the local neighborhood leader, $\mathbf{L}_i^{archive}$ is the global leader selected from the external archive, and $p_{local}(t)$ is the adaptive probability of using local guidance at iteration t .

This structure allows part of the population to exploit local promising regions while other mayflies remain guided by the global archive. The use of both local and archive-based leaders therefore improves the balance between convergence and diversity.

3.7. Grid-based archive and leader selection

The external archive stores non-dominated solutions found during the search. However, if leaders are selected randomly from the archive, crowded regions may be selected more frequently simply because they contain more solutions. This can cause the population to repeatedly move toward already populated regions of the Pareto front. To reduce this effect, ANGB-MOMA uses grid-based archive management and sparse-region-biased leader selection.

At each iteration, the objective space occupied by the archive is divided into G intervals per objective. For an m -objective problem, the maximum number of possible grid cells is:

$$N_{cells} = G^m.$$

Each archive member is assigned to one grid cell according to its objective-function vector. Let n_c denote the number of archive members in the c -th non-empty grid cell. A high value of n_c indicates a crowded region, while a low value indicates a sparse region.

The probability of selecting a grid cell for leader selection is defined as:

$$P_c = \frac{1/(n_c)^\gamma}{\sum_{j=1}^{N_c} 1/(n_j)^\gamma}$$

where N_c is the number of non-empty grid cells and γ is the grid selection pressure. A larger γ increases the preference for sparse cells.

After a grid cell is selected, one solution is randomly chosen from that cell as the archive leader. This makes leader selection diversity-aware because archive members in sparse cells become more likely to guide the population than members in crowded cells.

In addition to the density-biased rule, ANGB-MOMA also uses sparse-cell enforcement during leader selection. With probability p_{sparse} , the leader is selected directly from one of the least-occupied grid cells. With probability $1 - p_{sparse}$, the probabilistic grid-density rule is used. This is expressed as:

$$\mathbf{L}^{archive} = \begin{cases} \text{leader from least-occupied grid cell,} & rand < p_{sparse}, \\ \text{leader from density-biased roulette selection,} & rand \geq p_{sparse}. \end{cases}$$

This mechanism improves archive utilization because sparse regions of the front are given stronger influence during the movement update.

3.8. Boundary-preserving archive truncation

Boundary preservation is introduced to protect the identified extreme archive solutions during truncation. In this paper, the term boundary solution refers to the best currently identified archive member for an individual objective. It does not imply that the true theoretical extreme point of the Pareto front has necessarily been found.

For a minimization problem with m objectives and archive A , the identified boundary member for the j -th objective is:

$$\mathbf{x}^{(j)} = \operatorname{argmin}_{\mathbf{x} \in A} f_j(\mathbf{x}).$$

The set of protected archive boundary members is:

$$B = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(m)}\}.$$

When the archive size exceeds the maximum archive size A_{max} , the archive is reduced using boundary-preserving grid truncation. First, the identified boundary members in B are marked as protected. Second, all archive members are assigned to grid cells. Third, deletion is applied preferentially to crowded cells. If a crowded cell contains both protected and unprotected members, only the unprotected members are considered for removal. Protected members are removed only if the number of protected solutions alone exceeds the archive size limit, which did not occur in the reported experiments.

The truncation procedure can be summarized as follows:

$$A = \begin{cases} A, & |A| \leq A_{max}, \\ \text{Truncate}(A, B, G), & |A| > A_{max} \end{cases}$$

where G is the grid structure and B is the protected boundary set.

This strategy reduces the chance that extreme trade-off solutions already discovered by the algorithm are removed during archive reduction. At the same time, grid-based deletion reduces crowding by removing excess solutions from densely populated regions.

3.9. Crossover, mutation, and boundary control

The mating stage is retained from the original Mayfly Algorithm to generate additional candidate solutions and increase search diversity. Selected male and female mayflies are paired to produce offspring using arithmetic crossover. If $\mathbf{x}^m$ and $\mathbf{x}^f$ are the selected male and female parent positions, two offspring are generated as:

$$\mathbf{o}_1 = \lambda \mathbf{x}^m + (1 - \lambda) \mathbf{x}^f$$

$$\mathbf{o}_2 = \lambda \mathbf{x}^f + (1 - \lambda) \mathbf{x}^m$$

where λ is a random number uniformly distributed in $[0,1]$.

Mutation is then applied to selected offspring to introduce additional variation. Gaussian perturbation is used:

$$\mathbf{o}' = \mathbf{o} + \sigma \mathbf{N}(0,1)$$

where σ is the mutation step size and $\mathbf{N}(0,1)$ is a normally distributed random vector.

After movement, crossover, or mutation, every candidate solution is checked against the decision-variable bounds. If any variable exceeds its lower or upper bound, it is repaired using clipping:

$$x_{i,j} = \min[\max(x_{i,j}, x_j^{min}), x_j^{max}].$$

For ZDT5, the same boundary control is first applied to the real-coded vector in $[0,1]$. The repaired real-coded vector is then converted to binary form using the threshold rule described in Section 3.2 before objective evaluation.

3.10. Complete ANGB-MOMA procedure

The complete implementation of the proposed ANGB-MOMA is given in Algorithm 2. Unlike the earlier

baseline procedure, Algorithm 2 includes adaptive parameter control, neighborhood-guided leader selection, grid-based archive leader selection, and boundary-preserving archive truncation in a single reproducible framework.

Algorithm 2. Proposed ANGB-MOMA

Input:
 Objective function $F(x)$
 Decision-variable lower and upper bounds
 Number of male mayflies N_m
 Number of female mayflies N_f
 Maximum number of iterations $MaxIt$
 Archive size A_{max}
 Grid division number G
 Grid selection pressure γ
 Neighborhood size k
 Local-leader probability limits
 Sparse-cell leader probability p_{sparse}
 Crossover and mutation settings
 Adaptive-control parameters

Output:
 Final external archive A

1. Initialize male population X_m and female population X_f randomly within bounds.
2. Initialize male and female velocities.
3. Evaluate objective vectors $F(X_m)$ and $F(X_f)$.
4. Assign each mayfly's initial position as its personal best.
5. Merge male and female populations.
6. Identify non-dominated solutions using Pareto dominance.
7. Initialize the external archive A .
8. If $|A| > A_{max}$, apply archive truncation.
9. Construct ring neighborhoods for the population.
10. For $t = 1$ to $MaxIt$ do
 - 10.1 Compute the success rate $S(t)$ from personal-best improvements.
 - 10.2 Compute the population diversity $D_{pop}(t)$.
 - 10.3 Update adaptive parameters:
 $dance(t)$, $fl(t)$, $g(t)$, $\beta(t)$,
 $a1(t)$, $a2(t)$, $a3(t)$, and $p_{local}(t)$.
 - 10.4 Update the grid structure of archive A .
 - 10.5 For each male mayfly i do
 - Select local neighborhood leader L_{local_i} .
 - Select archive leader $L_{archive_i}$ using:
 - if $rand < p_{sparse}$
 choose leader from least-occupied grid cell;
 - else
 choose leader using grid-density roulette rule;
 - end if
 - Select final leader L_i :
 - if $rand < p_{local}(t)$
 $L_i = L_{local_i}$;
 - else
 $L_i = L_{archive_i}$;
 - end if
 - Update male velocity using personal best and L_i .
 - Update male position.
 - Apply boundary control.
 - Evaluate objective vector.
 - Update personal best using Pareto dominance.
 - End for
 - 10.6 For each female mayfly i do
 - Select corresponding male and leader guidance.
 - Update female velocity using male attraction,

```

    selected leader, and adaptive parameters.
    Update female position.
    Apply boundary control.
    Evaluate objective vector.
    Update personal best using Pareto dominance.
End for

10.7 Select parent pairs from male and female populations.
10.8 Generate offspring using arithmetic crossover.
10.9 Apply Gaussian mutation to selected offspring.
10.10 Apply boundary control to offspring.
10.11 Evaluate offspring objective vectors.
10.12 Merge male population, female population, offspring,
    and previous archive A.
10.13 Perform Pareto dominance comparison and non-dominated sorting.
10.14 Update archive A with non-dominated solutions.
10.15 If  $|A| > A_{max}$  then
    Identify archive boundary set B using objective-wise minima.
    Assign archive members to grid cells.
    Remove excess unprotected members from the most crowded cells.
    Preserve members of B unless no feasible deletion alternative exists.
End if
End for

11. Return archive A as the approximated Pareto front.

```

Algorithm 2 shows that the proposed method differs from the original MOMA in four main locations: the adaptive update stage in Step 10.3, the neighborhood and archive leader selection stages in Steps 10.4 to 10.6, the grid-based archive update in Step 10.14, and the boundary-preserving truncation stage in Step 10.15. These additions are evaluated separately through the ablation study presented in Section 4.4.

3.11. Parameter settings and tuning procedure

To ensure a fair comparison, all MOMA-based algorithms were executed using the same baseline population and evaluation settings. The number of male mayflies was set to 20, and the number of female mayflies was set to 20. Thus, each MOMA-based algorithm used a total population size of 40. The maximum number of iterations was set to 1000, and the archive size was set to 50. The same number of objective-function evaluations was used for all algorithms.

The final ANGB-MOMA parameter values were selected using preliminary parameter-sensitivity experiments on representative benchmark problems. The representative problems were ZDT2, ZDT6, and DTLZ7. ZDT2 was selected as a smooth convex-front case, ZDT6 was selected because of its biased and difficult Pareto-front structure, and DTLZ7 was selected to represent a disconnected higher-dimensional benchmark structure. For each tested parameter setting, 10 independent runs were performed, and the final values were chosen by considering IGD+, HV, and runtime trade-offs. The detailed sensitivity results are reported in Section 4.5.

Table 1 presents the common baseline parameters used by the MOMA-based algorithms, while Table 2 presents the additional parameters used by ANGB-MOMA.

Table 1. Common parameter settings for MOMA-based algorithms

Parameter	Symbol	Value
Number of male mayflies	N_m	20
Number of female mayflies	N_f	20
Total population size	N	40
Maximum number of iterations	$MaxIt$	1000
External archive size	A_{max}	50
Initial inertia weight	g	0.80
Personal learning coefficient	a_1	1.00
Global learning coefficient	a_2	1.50
Female attraction coefficient	a_3	1.50
Visibility coefficient	β	2.00
Initial nuptial dance coefficient	$dance$	0.77
Initial random flight coefficient	fl	0.77
Dance damping factor	$dance_{damp}$	0.99
Random-flight damping factor	fl_{damp}	0.99
Number of crossover offspring	N_c	20
Number of mutated solutions	N_m^{mut}	10
Mutation rate	μ	0.02
Velocity upper bound	V_{max}	$5(x^{max} - x^{min})$
Velocity lower bound	V_{min}	$-V_{max}$
Number of independent runs	–	30
Random seeds	–	1 to 30

Table 2. Additional parameter settings for ANGB-MOMA

Parameter	Symbol	Value
Grid divisions per objective	G	30
Grid inflation factor	α_{grid}	0.10
Grid selection pressure	γ	1.20
Neighborhood size	k	5
Initial local-leader probability	p_{local}	0.30
Lower bound of local-leader probability	p_{local}^{min}	0.20
Upper bound of local-leader probability	p_{local}^{max}	0.90
Sparse-cell leader probability	p_{sparse}	0.30
Target success rate	S_{target}	0.20
Diversity threshold for β update	D_{thr}	0.15
Lower and upper bounds of g	$[g_{min}, g_{max}]$	[0.20, 0.95]
Lower and upper bounds of a_1, a_2, a_3	$[a_{min}, a_{max}]$	[0.50, 2.50]
Lower and upper bounds of $dance$	$[dance_{min}, dance_{max}]$	[0.05, 1.50]
Lower and upper bounds of fl	$[fl_{min}, fl_{max}]$	[0.05, 1.50]
Lower and upper bounds of β	$[\beta_{min}, \beta_{max}]$	[0.50, 5.00]
Segment-jump operator	–	Disabled

The segment-jump operator was not included in the final ANGB-MOMA configuration because the proposed method in this paper is defined by four mechanisms only: adaptive parameter control, neighborhood-guided search, grid-based archive and leader selection, and boundary-preserving archive truncation. This avoids introducing an additional hidden mechanism that is not part of the stated contribution.

3.12. Benchmark configuration

The benchmark configuration is summarized in Table 3. All problems were formulated as two-objective minimization problems. ZDT1, ZDT2, ZDT3, ZDT4, ZDT5, and ZDT6 were used to evaluate bi-objective behavior, while DTLZ1, DTLZ4, and DTLZ7 were used to test the algorithm on higher-dimensional decision spaces with two objectives.

Table 3. Benchmark configuration used in the experiments

Problem	No. of variables	No. of objectives	Lower bound	Upper bound	Notes
ZDT1	10	2	[0, ..., 0]	[1, ..., 1]	Continuous benchmark
ZDT2	10	2	[0, ..., 0]	[1, ..., 1]	Continuous benchmark
ZDT3	10	2	[0, ..., 0]	[1, ..., 1]	Discontinuous Pareto front
ZDT4	10	2	[0, -5, ..., -5]	[1, 5, ..., 5]	Multimodal continuous benchmark
ZDT5	80	2	[0, ..., 0]	[1, ..., 1]	Real-coded binary representation with 0.5 threshold
ZDT6	10	2	[0, ..., 0]	[1, ..., 1]	Biased Pareto-front benchmark
DTLZ1	11	2	[0, ..., 0]	[1, ..., 1]	Two-objective DTLZ1 with $k = 10$
DTLZ4	11	2	[0, ..., 0]	[1, ..., 1]	Two-objective DTLZ4 with $k = 10$
DTLZ7	21	2	[0, ..., 0]	[1, ..., 1]	Two-objective DTLZ7 with $k = 20$

ZDT5 was treated separately because the original problem is binary/discrete. In the present implementation, mayfly positions were stored as real-valued vectors in $[0, 1]$, but the objective evaluation was performed only after thresholding the real-valued vector into a binary string. This ensured that the search algorithm could retain its continuous movement equations while the benchmark evaluation remained consistent with the discrete structure of ZDT5.

3.13. Reproducibility of HV and IGD+ calculations

The hypervolume (HV) and IGD+ values depend on the reference Pareto front, the normalization procedure, and the HV reference point. These settings were therefore fixed explicitly for all algorithms and all runs. Table 4 summarizes the metric reproducibility settings used throughout the study.

For each benchmark, the reference Pareto front was generated analytically from the corresponding benchmark definition. A reference set of 1000 points was used for all problems except ZDT5, whose discrete Pareto front contains 31 reference points. For ZDT3, the known discontinuous Pareto-front intervals were sampled directly. The sampled intervals were:

$$[0.0000000000, 0.0830015349], [0.1822287280, 0.2577623634], [0.4093136748, 0.4538821041], \\ [0.6183967944, 0.6525117038], [0.8233317983, 0.8518328654].$$

For ZDT6, the reference front was sampled over:

$$f_1 \in [0.2807753191, 1].$$

For DTLZ7, the analytically sampled front was filtered so that only non-dominated points were retained in the reference set.

All HV and IGD+ calculations were performed in normalized objective space. Let P be the obtained non-dominated objective set and P^* be the reference Pareto-front set. The objective values were normalized using the minimum and maximum values of the reference front:

$$\tilde{f}_j = \frac{f_j - f_j^{min,*}}{f_j^{max,*} - f_j^{min,*} + \epsilon}, \quad j = 1, 2, \dots, m$$

where $f_j^{min,*}$ and $f_j^{max,*}$ are the minimum and maximum values of the j -th objective in the reference Pareto set, and ϵ is a small numerical value used to avoid division by zero.

After normalization, the HV reference point was fixed as:

$$\mathbf{r}_{HV} = [1.1, 1.1].$$

This same normalized reference point was used for all two-objective benchmark problems. For HV calculation, points outside the reference box were clipped to the reference box to avoid ill-defined dominated areas. IGD+ was computed using the normalized obtained front and the normalized reference Pareto set.

The IGD+ value was calculated as:

$$IGD^+(P, P^*) = \frac{1}{|P^*|} \sum_{\mathbf{z} \in P^*} \min_{\mathbf{p} \in P} \|\max(\mathbf{p} - \mathbf{z}, 0)\|_2$$

where P is the obtained non-dominated set, P^* is the reference Pareto set, $\mathbf{p}$ is an obtained solution, and $\mathbf{z}$ is a reference point. Smaller IGD+ values indicate better convergence and distribution.

Spread and spacing were also computed in normalized objective space. Therefore, the diversity metrics are comparable across problems with different objective-value ranges. This is especially important for ZDT5, whose raw objective values are on a different scale from the other ZDT functions.

Table 4. Metric reproducibility settings

Item	Setting
Reference Pareto front source	Analytical benchmark definitions
Reference points	1000 for continuous cases; 31 for ZDT5
Objective normalization	Reference-front minimum and maximum values
HV reference point	[1.1,1.1] after normalization
HV problem type	Two-objective minimization
IGD+ calculation	Normalized obtained front against normalized reference front
Spread and spacing	Computed in normalized objective space
ZDT3 reference set	Five analytical discontinuous intervals
DTLZ7 reference set	Analytically sampled and non-dominated filtered
ZDT5 handling	Real-coded vector thresholded to binary before evaluation

3.14. Statistical testing

Since all algorithms are stochastic, each algorithm was independently executed 30 times on each benchmark function. The same seed sequence, from 1 to 30, was used for all compared algorithms to support paired statistical testing.

The Wilcoxon signed-rank test was used to compare ANGB-MOMA against each competing algorithm. The comparison was performed pairwise using the run-level metric values. For HV, larger values indicate better performance. For IGD+, spread, and spacing, smaller values indicate better performance.

The null hypothesis for each pairwise test was that there was no statistically significant difference between ANGB-MOMA and the comparator algorithm. The alternative hypothesis was that the two algorithms produced different performance distributions. The significance level was set to:

$$\alpha = 0.05.$$

Because multiple pairwise comparisons were conducted, the Holm-Bonferroni correction was applied to control the family-wise error rate. For each benchmark and metric, the raw p-values were sorted in ascending order:

$$p_{(1)} \leq p_{(2)} \leq \dots \leq p_{(q)}$$

where q is the number of pairwise comparisons in the tested family. The i -th ordered p-value was compared with:

$$\frac{\alpha}{q - i + 1}.$$

A comparison was considered statistically significant only when it remained significant after Holm correction. The results therefore report both raw p-values and Holm-adjusted decisions.

The outcome of each comparison was classified as a win, tie, or loss for ANGB-MOMA. For HV, ANGB-MOMA was classified as a win when it had a higher mean value and the Holm-corrected test was significant. For IGD+, spread, and spacing, ANGB-MOMA was classified as a win when it had a lower mean value and the Holm-corrected test was significant. If the difference was not significant after correction, the outcome was classified as a tie.

3.15. Computational complexity analysis

The proposed ANGB-MOMA introduces additional operations beyond the original MOMA. These include adaptive parameter updates, neighborhood leader selection, grid assignment, sparse-cell leader selection, and boundary-preserving archive truncation. To clarify the computational cost, let N be the total number of mayflies, D be the number of decision variables, m be the number of objectives, A be the archive size, G be the number of grid divisions per objective, and C be the number of offspring and mutation candidates generated per iteration.

For the original MOMA, the main computational operations per iteration are solution movement, objective evaluation, crossover and mutation, non-dominated sorting, crowding-distance calculation, and archive update. Ignoring the cost of objective-function evaluation, which depends on the problem, the dominant cost is usually non-dominated sorting. If the merged candidate set has size Q , then the standard dominance comparison cost is:

$$O(mQ^2)$$

where:

$$Q \approx N + C + A.$$

The movement and variation operations have cost:

$$O(ND + CD).$$

Thus, the approximate per-iteration cost of the original MOMA is:

$$O(mQ^2 + ND + CD).$$

For ANGB-MOMA, adaptive parameter control requires calculation of success rate and population diversity. Success-rate calculation is $O(N)$, while decision-space diversity calculation is $O(ND)$. The adaptive update itself is constant time. Therefore, the adaptive-control overhead is:

$$O(ND).$$

The ring-neighborhood mechanism requires local leader selection for each mayfly. If the neighborhood size is k , the neighborhood search cost is approximately:

$$O(Nk).$$

If local dominance or ranking information is used during leader selection, the cost can be expressed as:

$$O(Nkm).$$

The grid-based archive operation assigns archive members to grid cells. This requires checking objective

values for each archive member and therefore costs:

$$O(Am).$$

The number of possible grid cells is:

$$G^m.$$

However, only non-empty grid cells are used in leader selection and archive truncation. Sparse-cell leader selection uses cell occupancy information and therefore has cost approximately:

$$O(A + N_c)$$

where N_c is the number of non-empty grid cells.

Boundary preservation requires identifying the archive member with the minimum value in each objective. This costs:

$$O(Am).$$

Archive truncation removes excess solutions from crowded cells. If R solutions must be removed, the deletion stage costs approximately:

$$O(R(A + N_c)).$$

Since the archive size is limited to A_{max} , the archive-related overhead remains bounded in practice.

Combining these terms, the approximate per-iteration complexity of ANGB-MOMA is:

$$O(mQ^2 + ND + CD + Nkm + Am + A + N_c + R(A + N_c)).$$

The dominant term remains the non-dominated sorting cost $O(mQ^2)$. Therefore, ANGB-MOMA has the same leading-order dominance-sorting complexity as the original MOMA, but with additional lower-order overhead from adaptive control, neighborhood selection, grid management, and boundary preservation.

This overhead explains why ANGB-MOMA can require higher runtime than the original MOMA even when both algorithms use the same number of function evaluations. The runtime overhead is quantified in Section 4.7 by comparing the mean runtime of ANGB-MOMA with the runtime of the original MOMA and the other comparator algorithms.

4. Results and Discussion

4.1. Overview of the experimental evaluation

This section presents the comparative evaluation of the proposed Adaptive Neighborhood Grid-Boundary Multi-Objective Mayfly Algorithm (ANGB-MOMA). Seven algorithms are compared: Original MOMA, Levy-flight MOMA, Stochastic-inertia MOMA, ANGB-MOMA, NSGA-II, MOPSO, and MOEA/D.

The benchmark set includes ZDT4 to assess performance on a highly multimodal landscape. The MOMA-based methods struggled with this problem, while NSGA-II, MOPSO, and MOEA/D achieved better convergence and HV values. ZDT4 is therefore discussed as a limitation of the present MOMA-based framework. Figure 1 compares the Pareto-front approximations for the ZDT problems. Figure 2 presents the mean IGD+ convergence curves for the selected ZDT and DTLZ problems, while Figure 3 presents the mean HV convergence curves for the ZDT problems.

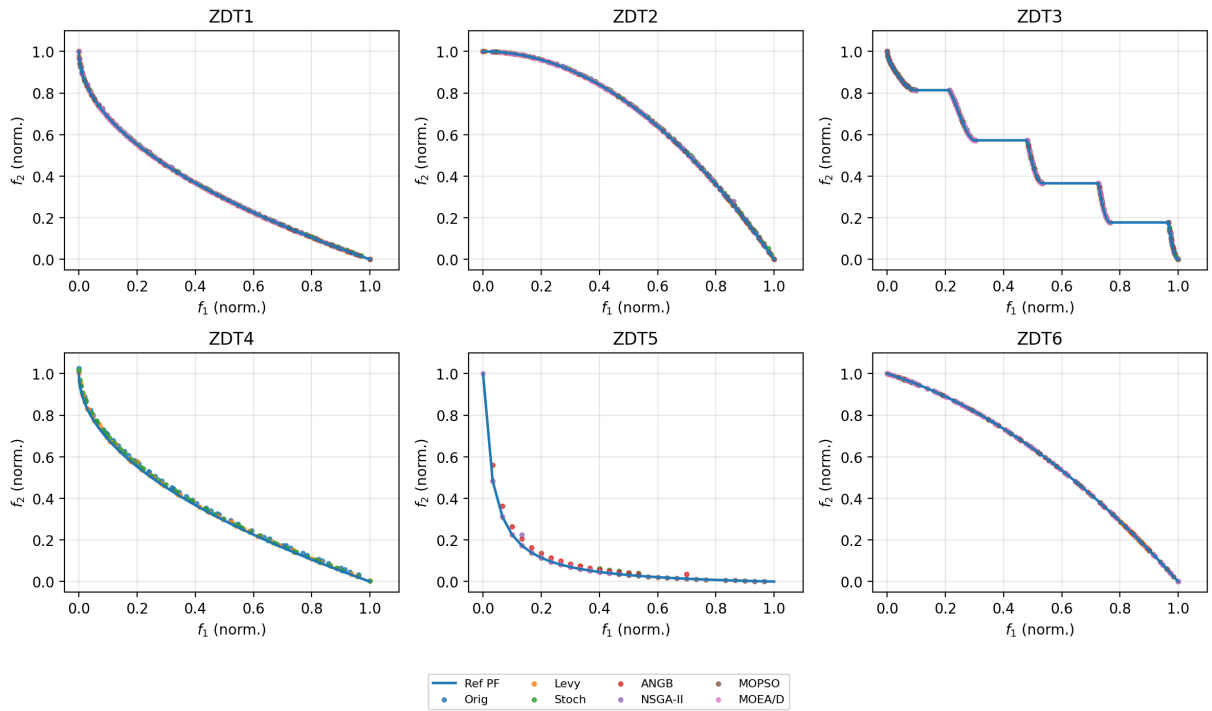


Figure 1. Pareto-front overlays for the ZDT benchmark problems

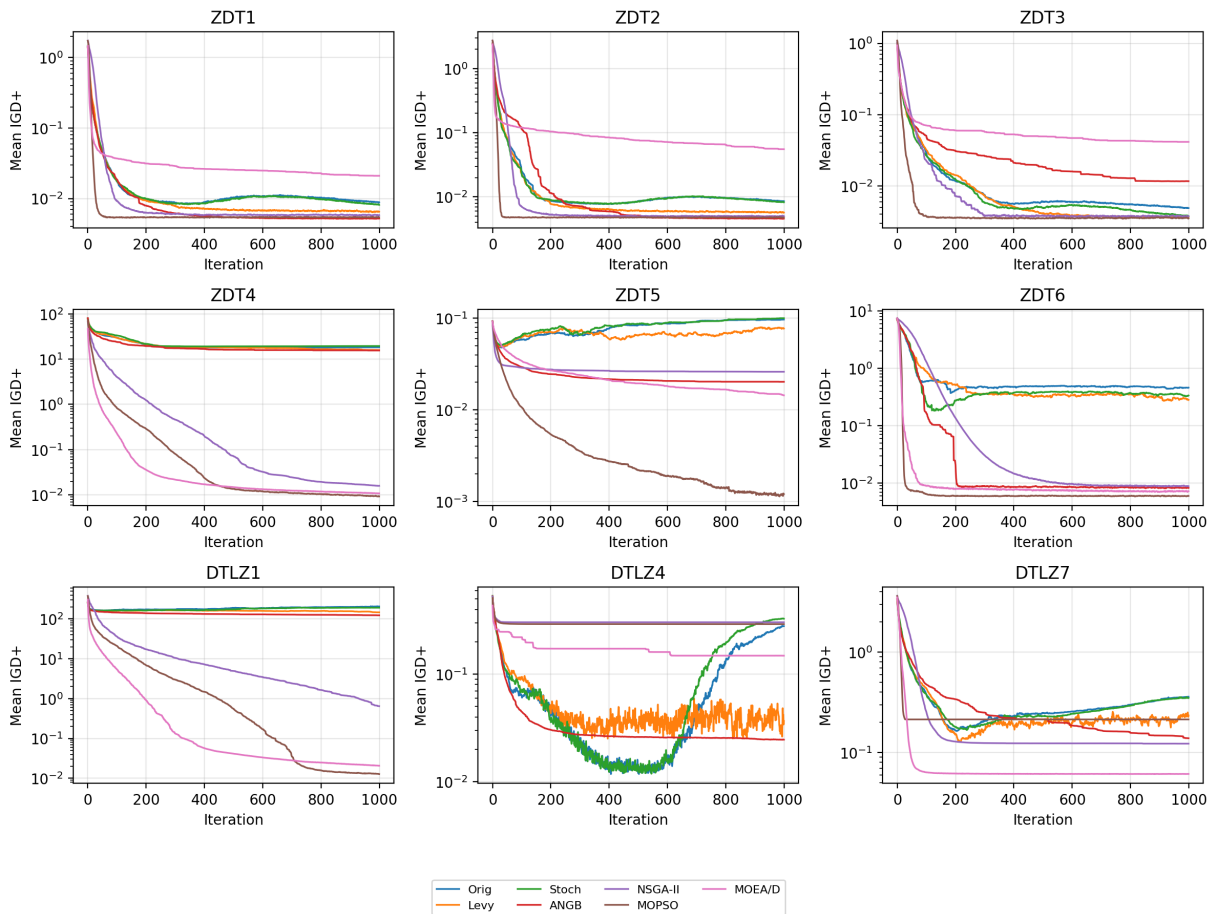


Figure 2. Mean IGD+ convergence curves for the selected ZDT and DTLZ benchmark problems

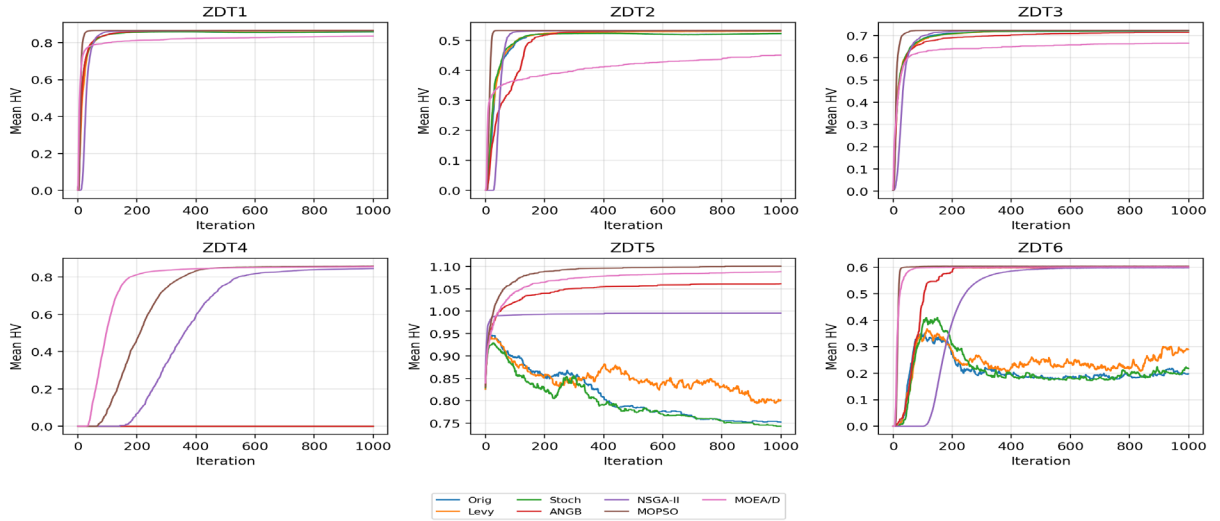


Figure 3. Mean HV convergence curves for the ZDT benchmark problems

4.2. Comparative results on ZDT benchmark problems

Table 5 presents the main convergence, and coverage results on the ZDT benchmark problems. The table reports HV and IGD+ because these two indicators provide the clearest combined evidence of Pareto-front quality. Spread and spacing are discussed separately in the diversity analysis.

Table 5. ZDT HV and IGD+ results for the compared algorithms

Problem	Metric	Original MOMA	Levy-flight MOMA	Stochastic-inertia MOMA	ANGB-MOMA	NSGA-II	MOPSO	MOEA/D	Best
ZDT1	HV	0.85840 ± 0.00618	0.86398 ± 0.00082	0.85952 ± 0.00438	0.86669 ± 0.00017	0.86553 ± 0.00035	0.86632 ± 0.00029	0.83553 ± 0.00778	ANGB-MOMA
		0.00883 ± 0.00337	0.00650 ± 0.00040	0.00813 ± 0.00210	0.00520 ± 0.00009	0.00581 ± 0.00022	0.00541 ± 0.00016	0.02103 ± 0.00622	ANGB-MOMA
ZDT2	HV	0.52260 ± 0.00996	0.53104 ± 0.00115	0.52306 ± 0.00999	0.53361 ± 0.00021	0.53272 ± 0.00039	0.53322 ± 0.00019	0.45127 ± 0.05487	ANGB-MOMA
		0.00859 ± 0.00398	0.00570 ± 0.00039	0.00826 ± 0.00397	0.00460 ± 0.00011	0.00502 ± 0.00020	0.00478 ± 0.00010	0.05541 ± 0.03671	ANGB-MOMA
ZDT3	HV	0.72077 ± 0.00396	0.72211 ± 0.00061	0.72202 ± 0.00041	0.71357 ± 0.00990	0.72284 ± 0.00026	0.72316 ± 0.00018	0.66503 ± 0.03225	MOPSO
		0.00489 ± 0.00423	0.00363 ± 0.00030	0.00377 ± 0.00022	0.01165 ± 0.01009	0.00377 ± 0.00021	0.00351 ± 0.00017	0.04148 ± 0.02167	MOPSO
ZDT4	HV	0.00000 ± 0.00000	0.00000 ± 0.00000	0.00000 ± 0.00000	0.00000 ± 0.00000	0.84566 ± 0.00815	0.85883 ± 0.00106	0.85573 ± 0.00302	MOPSO
		18.28964 ± 8.57414	15.91706 ± 6.82967	19.52139 ± 9.26718	15.55764 ± 6.06035	0.01588 ± 0.00450	0.00928 ± 0.00056	0.01081 ± 0.00165	MOPSO
ZDT5	HV	0.75253 ± 0.06338	0.80184 ± 0.07065	0.74331 ± 0.06301	1.06116 ± 0.01994	0.99565 ± 0.03970	1.10054 ± 0.01038	1.08816 ± 0.00392	MOPSO
		0.09702 ± 0.02048	0.07741 ± 0.02173	0.10038 ± 0.02324	0.02027 ± 0.00510	0.02610 ± 0.00627	0.00120 ± 0.00133	0.01442 ± 0.00340	MOPSO
ZDT6	HV	0.19805 ± 0.09181	0.28965 ± 0.13930	0.21772 ± 0.10789	0.59889 ± 0.00180	0.59770 ± 0.00136	0.60431 ± 0.00057	0.60146 ± 0.00288	MOPSO
		0.46020 ± 0.62085	0.28177 ± 0.15982	0.33547 ± 0.11631	0.00823 ± 0.00077	0.00884 ± 0.00062	0.00598 ± 0.00029	0.00724 ± 0.00130	MOPSO

Table 5 shows that ANGB-MOMA clearly improves the original MOMA on most ZDT functions. On ZDT1, ANGB-MOMA increased HV from 0.85840 to 0.86669 and reduced IGD+ from 0.00883 to 0.00520. This corresponds to a 0.97% increase in HV and a 41.13% reduction in IGD+ relative to the original MOMA. On ZDT2, ANGB-MOMA increased HV from 0.52260 to 0.53361 and reduced IGD+

from 0.00859 to 0.00460, representing a 2.11% HV improvement and a 46.46% IGD+ reduction.

The improvement over the original MOMA is more pronounced on ZDT5 and ZDT6. On ZDT5, ANGB-MOMA increased HV from 0.75253 to 1.06116 and reduced IGD+ from 0.09702 to 0.02027. This represents a 41.01% increase in HV and a 79.10% reduction in IGD+. On ZDT6, ANGB-MOMA increased HV from 0.19805 to 0.59889 and reduced IGD+ from 0.46020 to 0.00823. This corresponds to a 202.40% increase in HV and a 98.21% reduction in IGD+. These results show that the proposed mechanisms substantially improve the baseline MOMA on difficult continuous ZDT problems, especially where the original algorithm has weak convergence.

However, the comparison also shows that ANGB-MOMA is not universally superior to all established multi-objective optimizers. MOPSO achieved the best HV and IGD+ on ZDT3, ZDT4, ZDT5, and ZDT6. This means that ANGB-MOMA is best understood as a strong improvement over the original MOMA and related MOMA variants, while remaining competitive with established algorithms on some problems rather than dominating them across the full benchmark suite.

The ZDT3 result remains an important limitation. ZDT3 has a discontinuous Pareto front, and ANGB-MOMA produced lower HV and higher IGD+ than Original MOMA, Levy-flight MOMA, Stochastic-inertia MOMA, NSGA-II, and MOPSO. Its HV decreased by 1.00% relative to Original MOMA, while its IGD+ increased by 138.22%. This confirms that the present combination of adaptive control, neighborhood guidance, grid-based leader selection, and boundary preservation does not fully solve discontinuous-front preservation.

The ZDT4 results demonstrate a limitation of the MOMA-based methods. All four MOMA-based methods obtained zero normalized HV on ZDT4, while NSGA-II, MOPSO, and MOEA/D obtained high HV values above 0.84. The IGD+ values confirm the same trend: ANGB-MOMA improved the original MOMA from 18.28964 to 15.55764, but this improvement is small compared with the IGD+ values of NSGA-II, MOPSO, and MOEA/D, which were all below 0.016. This indicates that the current MOMA-based search dynamics are still weak on highly multimodal ZDT4 landscapes. ZDT4 is therefore treated as a limitation case in this study.

For ZDT5, the normalized HV values are comparable with the other ZDT results because the raw binary/discrete objective values were normalized before metric calculation. ANGB-MOMA performed strongly against the MOMA-based methods on ZDT5, but MOPSO achieved the best overall result. This suggests that the grid-boundary mechanism improves the MOMA family substantially on the discrete ZDT5 structure, but mature archive-guided swarm search remains highly competitive on this benchmark.

Overall, the ZDT results support two balanced conclusions. First, ANGB-MOMA provides large and consistent improvements over the original MOMA on ZDT1, ZDT2, ZDT5, and ZDT6. Second, the comparison shows that ANGB-MOMA is not yet as robust as MOPSO on discontinuous, discrete, and multimodal ZDT cases. This more careful interpretation avoids overclaiming and gives a clearer picture of the strengths and limitations of the proposed algorithm.

4.3. Comparative results on DTLZ benchmark problems

Table 6 presents the DTLZ results. Although HV was also computed, IGD+ is emphasized because it gives a more direct indication of convergence and distribution quality relative to the reference Pareto set. The DTLZ cases were evaluated as two-objective problems with higher-dimensional decision spaces.

Table 6. DTLZ HV and IGD+ results for the compared algorithms

Problem	Metric	Original MOMA	Levy-flight MOMA	Stochastic-inertia MOMA	ANGB-MOMA	NSGA-II	MOPSO	MOEA/D	Best
DTLZ1	HV	0.00000 ± 0.00000	0.00000 ± 0.00000	0.00000 ± 0.00000	0.00000 ± 0.00000	0.36870 ± 0.27112	0.68532 ± 0.00179	0.67088 ± 0.00716	MOPSO
		205.86309 ± 33.45936	146.53125 ± 29.71936	191.42316 ± 38.66414	123.44061 ± 10.43134	0.64640 ± 1.03677	0.01277 ± 0.00099	0.02067 ± 0.00412	
DTLZ1	IGD+	0.17501 ± 0.06295	0.36753 ± 0.02649	0.13715 ± 0.05448	0.37889 ± 0.02696	0.16016 ± 0.11408	0.17012 ± 0.12229	0.29229 ± 0.15136	ANGB-MOMA
		0.27968 ± 0.09404	0.03663 ± 0.02819	0.32799 ± 0.07107	0.02458 ± 0.01566	0.30405 ± 0.13525	0.29219 ± 0.14510	0.14851 ± 0.17853	
DTLZ4	HV	0.16772 ± 0.06501	0.27661 ± 0.10331	0.17770 ± 0.07545	0.38737 ± 0.13865	0.42308 ± 0.06778	0.35291 ± 0.00003	0.48921 ± 0.08375	MOEA/D
		0.35960 ± 0.05900	0.23394 ± 0.11459	0.34934 ± 0.06851	0.13891 ± 0.13777	0.12272 ± 0.08702	0.21386 ± 0.00003	0.06129 ± 0.09369	

Table 6 shows that ANGB-MOMA improves substantially over the original MOMA and the two MOMA variants on the DTLZ problems. On DTLZ1, ANGB-MOMA reduced IGD+ from 205.86309 for Original MOMA to 123.44061, corresponding to a 40.04% reduction. It also outperformed Levy-flight MOMA and Stochastic-inertia MOMA. However, NSGA-II, MOPSO, and MOEA/D performed much better on DTLZ1, with MOPSO obtaining the best IGD+ value of 0.01277. This indicates that, although ANGB-MOMA improves the MOMA framework, its convergence behavior on DTLZ1 remains weaker than that of mature decomposition- and archive-based optimizers.

On DTLZ4, ANGB-MOMA achieved the best result among all algorithms. It obtained the highest HV of 0.37889 and the lowest IGD+ of 0.02458. Relative to Original MOMA, this corresponds to a 116.50% increase in HV and a 91.21% reduction in IGD+. This is one of the strongest cases for the proposed algorithm because ANGB-MOMA outperformed not only the MOMA variants but also NSGA-II, MOPSO, and MOEA/D.

On DTLZ7, ANGB-MOMA again improved clearly over the MOMA-based comparators. It reduced IGD+ from 0.35960 for Original MOMA to 0.13891, corresponding to a 61.37% reduction. However, MOEA/D achieved the best overall DTLZ7 result, with an HV of 0.48921 and IGD+ of 0.06129. NSGA-II also produced a slightly lower IGD+ than ANGB-MOMA. This suggests that ANGB-MOMA is effective relative to MOMA-based methods, but decomposition-based search and mature evolutionary methods remain highly competitive on some DTLZ problems.

The DTLZ results therefore support a refined conclusion. ANGB-MOMA improves the original MOMA across all three DTLZ problems tested, but its competitiveness against established algorithms depends on the problem structure. It is strongest on DTLZ4, where it gives the best overall performance, but it is weaker than MOPSO and MOEA/D on DTLZ1 and weaker than MOEA/D on DTLZ7.

4.4. Ablation study of the proposed ANGB-MOMA components

The proposed ANGB-MOMA combines four mechanisms: adaptive parameter control, neighborhood-guided search, grid-based archive and leader selection, and boundary-preserving archive truncation. To determine the individual and combined contribution of these mechanisms, an ablation study was performed. The following variants were compared with the original MOMA:

Original MOMA; MOMA with adaptive parameter control only; MOMA with neighborhood guidance only; MOMA with grid-based archive control only; MOMA with boundary preservation only; MOMA with adaptive parameter control and neighborhood guidance; MOMA with grid-based archive control and boundary preservation; MOMA with adaptive parameter control, neighborhood guidance, and grid-based archive control; and full ANGB-MOMA.

All variants were tested using the same population size, archive size, number of iterations, function-

evaluation budget, and seed sequence. Therefore, the differences in performance are due to the added mechanisms rather than unequal computational settings.

Table 7 presents the IGD+ results of the ablation study on selected representative benchmark functions. IGD+ is emphasized because it measures convergence and distribution quality relative to the reference Pareto set, with smaller values indicating better performance. Figure 4 provides a graphical comparison of the ablation IGD+ results.

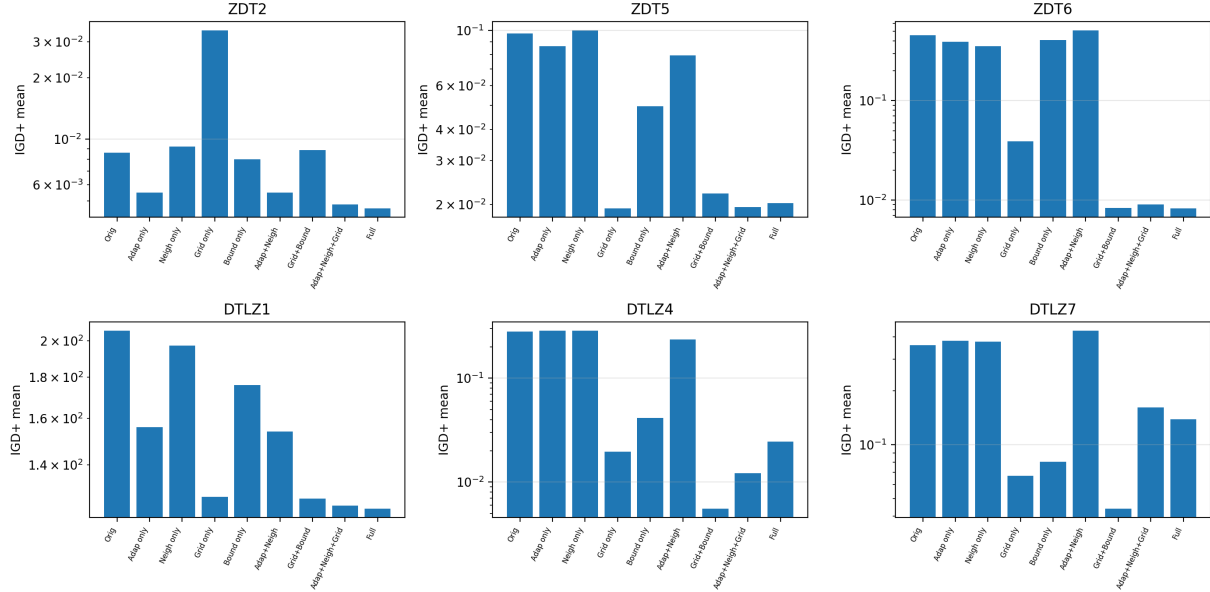


Figure 4. Ablation IGD+ comparison on selected benchmark problems

Table 7. Ablation study based on IGD+ mean values

Variant	ZDT2	ZDT3	ZDT5	ZDT6	DTLZ1	DTLZ4	DTLZ7
Original MOMA	0.00859	0.00489	0.09702	0.46020	205.86309	0.27968	0.35960
Adaptive only	0.00548	0.00712	0.08629	0.39325	155.92146	0.28513	0.38189
Neighborhood only	0.00921	0.00559	0.09989	0.35347	197.25911	0.28590	0.37760
Grid only	0.03407	0.00641	0.01933	0.03916	127.73648	0.01960	0.06719
Boundary only	0.00797	0.00482	0.04968	0.41126	176.00661	0.04132	0.08040
Adaptive + Neighborhood	0.00547	0.00876	0.07920	0.51238	154.11671	0.23477	0.43413
Grid + Boundary	0.00885	0.00638	0.02215	0.00828	127.04418	0.00556	0.04399
Adaptive + Neighborhood + Grid	0.00479	0.01031	0.01957	0.00896	124.45343	0.01218	0.16169
Full ANGB-MOMA	0.00460	0.01165	0.02027	0.00823	123.44061	0.02458	0.13891

Table 7 shows that the contribution of the four mechanisms is problem-dependent. On ZDT2, adaptive control is the most important single mechanism. The adaptive-only variant reduced IGD+ from 0.00859 to 0.00548, while the full ANGB-MOMA further reduced IGD+ to 0.00460. This represents a 46.46% reduction in IGD+ relative to the original MOMA. The result suggests that, on smooth continuous Pareto fronts, adaptive parameter control is useful for improving convergence without requiring strong archive restructuring.

On ZDT5 and ZDT6, the grid-based mechanism is the dominant source of improvement. On ZDT5, the grid-only variant reduced IGD+ from 0.09702 to 0.01933, while the full ANGB-MOMA obtained 0.02027. On ZDT6, the grid-only variant reduced IGD+ from 0.46020 to 0.03916, while the grid-boundary and full ANGB-MOMA variants further reduced IGD+ to 0.00828 and 0.00823, respectively. This confirms that the grid-based archive mechanism is especially useful when the original MOMA struggles to maintain effective objective-space coverage.

Boundary preservation also provides clear benefits on some problems. On ZDT5, boundary preservation

alone reduced IGD+ from 0.09702 to 0.04968, and on DTLZ7 it reduced IGD+ from 0.35960 to 0.08040. These results show that explicitly protecting the identified archive extremes can improve the retained trade-off range. However, the term “boundary” is used here to refer to the best currently identified archive members in each objective direction; it does not imply guaranteed preservation of the true theoretical Pareto-front extremes.

The neighborhood-only variant does not consistently improve the original MOMA when used alone. For example, on ZDT2 it produced an IGD+ of 0.00921, which is worse than the original MOMA value of 0.00859. On ZDT6, however, it reduced IGD+ from 0.46020 to 0.35347. This indicates that neighborhood guidance is not a standalone solution for all benchmark types. Its main role is to support population diversity and reduce dependence on a single global archive leader when combined with other mechanisms.

The ablation results also show that the full ANGB-MOMA does not always outperform every partial combination. For example, on DTLZ4, the grid-boundary variant achieved the lowest IGD+ value of 0.00556, while the full ANGB-MOMA achieved 0.02458. Similarly, on DTLZ7, the grid-boundary variant obtained 0.04399, while the full ANGB-MOMA obtained 0.13891. This means that the mechanisms do not contribute additively in every case. Instead, the full ANGB-MOMA gives a balanced global setting that improves the original MOMA across most tested functions, while some partial variants may perform better on specific problems.

The ablation study therefore supports a more careful interpretation of the proposed method. The results show that the improvement of ANGB-MOMA is mainly driven by adaptive control on smooth continuous problems and by grid-based archive management on difficult biased or discrete problems. Boundary preservation improves archive coverage in selected cases, while neighborhood guidance contributes more effectively when combined with the other mechanisms. However, the ablation results also confirm that the proposed mechanisms are not sufficient for ZDT3. On ZDT3, the full ANGB-MOMA increased IGD+ from 0.00489 to 0.01165, showing that discontinuous Pareto-front preservation remains a limitation of the present design.

4.5. Parameter sensitivity and tuning analysis

A parameter sensitivity study was performed to explain how the final ANGB-MOMA parameter values were selected. The tested parameters were the number of grid divisions G , neighborhood size k , local-leader probability p_{local} , archive size A_{max} , and grid selection pressure γ . The sensitivity study was carried out on ZDT2, ZDT6, and DTLZ7. These functions were selected because they represent different search conditions: ZDT2 is a smooth continuous-front problem, ZDT6 is a biased front problem, and DTLZ7 is a more difficult higher-dimensional benchmark structure.

The purpose of the sensitivity analysis was not to tune a different parameter set for each benchmark. Instead, one global ANGB-MOMA configuration was selected and used for all final experiments. This avoids overfitting the proposed algorithm to a specific test function.

Table 8 summarizes the main parameter-sensitivity observations.

Table 8. Parameter sensitivity summary

Parameter	Tested values	Final value	Main observation
Grid divisions G	10, 20, 30, 40	30	$G = 30$ gave strong ZDT2 and ZDT6 performance. $G = 40$ improved DTLZ7 but greatly increased runtime.
Neighborhood size k	3, 5, 7, 9	5	$k = 5$ gave the best DTLZ7 IGD+ and balanced ZDT2/ZDT6 performance.
Local-leader probability p_{local}	0.20, 0.30, 0.50, 0.70	0.30	$p_{local} = 0.30$ gave the best DTLZ7 and ZDT6 performance, while remaining close to the best ZDT2 result.
Archive size A_{max}	50, 100, 200	50	Larger archives improved some metrics but caused large runtime increases. $A_{max} = 50$ was retained for fair computational cost.
Grid pressure γ	0.8, 1.2, 1.8, 2.5	1.2	$\gamma = 0.8$ and $\gamma = 1.2$ were close. $\gamma = 1.2$ was retained as a moderate sparse-cell selection pressure.

For the number of grid divisions, $G = 30$ provided the best balance for the final algorithm. On ZDT2, increasing G from 10 to 30 reduced IGD+ from 0.00643 to 0.00463. On ZDT6, increasing G from 10 to 30 reduced IGD+ from 0.01255 to 0.00811. Although $G = 40$ achieved the best DTLZ7 IGD+ value of 0.03304, its mean runtime increased to 875.76 s, compared with 290.91 s for $G = 30$. Therefore, $G = 30$ was selected to provide strong convergence without the high computational cost associated with very fine grid resolution.

For neighborhood size, $k = 5$ was selected as a balanced value. On DTLZ7, $k = 5$ achieved the lowest IGD+ value of 0.09938 among the tested neighborhood sizes. On ZDT6, $k = 5$ also produced a strong IGD+ value of 0.00811, close to the best value of 0.00800 obtained with $k = 9$, but with lower runtime. On ZDT2, the differences among $k = 3$, $k = 5$, $k = 7$, and $k = 9$ were small, with IGD+ values remaining close to 0.0046. Therefore, $k = 5$ was chosen as a stable compromise.

For local leader probability, $p_{local} = 0.30$ gave the best overall balance. On ZDT6, $p_{local} = 0.30$ achieved the lowest IGD+ value of 0.00811. On DTLZ7, it also produced the best IGD+ value of 0.09938. On ZDT2, $p_{local} = 0.20$ was slightly better, with IGD+ of 0.00456 compared with 0.00463 at $p_{local} = 0.30$, but the difference was small. Therefore, $p_{local} = 0.30$ was retained because it performed well across the representative problems.

The archive size sensitivity test showed a clear trade-off between solution quality and runtime. Increasing the archive size improved IGD+ on ZDT2 and ZDT6. For example, on ZDT2, increasing A_{max} from 50 to 200 reduced IGD+ from 0.00463 to 0.00145. On ZDT6, increasing A_{max} from 50 to 200 reduced IGD+ from 0.00811 to 0.00252. However, the runtime increased substantially. On ZDT2, the mean runtime increased from 165.17 s at $A_{max} = 50$ to 2261.03 s at $A_{max} = 200$. On ZDT6, it increased from 181.66 s to 878.57 s. Because the aim was to compare algorithms under a practical and controlled archive size, $A_{max} = 50$ was retained in the final experiments.

For grid selection pressure, the results showed that very aggressive sparse-cell bias was not beneficial. Values of $\gamma = 0.8$ and $\gamma = 1.2$ produced similar results on ZDT2, ZDT6, and DTLZ7. For example, on ZDT6, the IGD+ values were 0.00799 for $\gamma = 0.8$ and 0.00811 for $\gamma = 1.2$. On DTLZ7, the IGD+ values were 0.09616 and 0.09938, respectively. Since the differences were small and $\gamma = 1.2$ provides moderate sparse-region preference, $\gamma = 1.2$ was retained for the final ANGB-MOMA configuration.

The parameter sensitivity results show that ANGB-MOMA is affected most strongly by archive size and grid resolution. Larger archive sizes and finer grids can improve solution quality, but they increase runtime. The final parameter set was therefore selected as a compromise between convergence quality, diversity preservation, and computational cost. This also explains why ANGB-MOMA generally requires more runtime than the original MOMA: its additional archive and neighborhood mechanisms provide better solution quality on many functions but introduce extra computational overhead.

4.6. Statistical significance analysis

The statistical significance of the results was examined using the Wilcoxon signed-rank test with Holm-Bonferroni correction. ANGB-MOMA was compared pairwise against Original MOMA, Levy-flight MOMA, Stochastic-inertia MOMA, NSGA-II, MOPSO, and MOEA/D. The tests were performed separately for HV and IGD+ using the 30 independent runs obtained for each benchmark function.

Table 9 presents the win/tie/loss summary after Holm correction. A win means that ANGB-MOMA obtained a better mean value and the difference remained statistically significant after Holm correction. A loss means that the comparator obtained a better mean value and the difference remained statistically significant. A tie means that the difference was not statistically significant after Holm correction.

Table 9. Holm-corrected win/tie/loss summary for ANGB-MOMA across all nine benchmark functions

Metric	Comparator	Win	Tie	Loss
HV	Original MOMA	6	2	1
HV	Levy-flight MOMA	5	3	1
HV	Stochastic-inertia MOMA	6	2	1
HV	NSGA-II	5	1	3
HV	MOPSO	3	1	5
HV	MOEA/D	3	1	5
IGD+	Original MOMA	7	1	1
IGD+	Levy-flight MOMA	6	2	1
IGD+	Stochastic-inertia MOMA	7	1	1
IGD+	NSGA-II	5	1	3
IGD+	MOPSO	4	0	5
IGD+	MOEA/D	3	1	5

Table 9 confirms that ANGB-MOMA is a statistically meaningful improvement over the original MOMA. For IGD+, ANGB-MOMA achieved seven wins, one tie, and one loss against Original MOMA. Against Stochastic-inertia MOMA, it also achieved seven wins, one tie, and one loss. Against Levy-flight MOMA, it achieved six wins, two ties, and one loss. These results support the main claim that the proposed mechanisms improve the MOMA framework.

However, Table 9 also shows that ANGB-MOMA does not dominate the established algorithms. Against NSGA-II, ANGB-MOMA achieved five IGD+ wins, one tie, and three losses. Against MOPSO, it achieved four IGD+ wins and five losses. Against MOEA/D, it achieved three IGD+ wins, one tie, and five losses. Therefore, the interpretation is that ANGB-MOMA strongly improves the MOMA family and is competitive on selected problems, but it is not universally superior to mature algorithms such as MOPSO and MOEA/D.

Table 10 gives representative Holm-corrected p-values for important benchmark cases. The full p-value table is included in the supplementary result files.

Table 10. Representative Holm-corrected p-values for ANGB-MOMA comparisons

Problem	Comparison	HV Holm p-value	HV outcome	IGD+ Holm p-value	IGD+ outcome
ZDT1	ANGB-MOMA vs Original MOMA	1.10×10^{-5}	Win	1.10×10^{-5}	Win
ZDT1	ANGB-MOMA vs MOPSO	3.26×10^{-5}	Win	1.30×10^{-5}	Win
ZDT2	ANGB-MOMA vs Original MOMA	1.10×10^{-5}	Win	1.10×10^{-5}	Win
ZDT2	ANGB-MOMA vs MOPSO	1.18×10^{-5}	Win	2.07×10^{-5}	Win
ZDT3	ANGB-MOMA vs MOPSO	1.10×10^{-5}	Loss	2.48×10^{-5}	Loss
ZDT4	ANGB-MOMA vs MOPSO	1.10×10^{-5}	Loss	1.10×10^{-5}	Loss
ZDT5	ANGB-MOMA vs Original MOMA	1.10×10^{-5}	Win	1.10×10^{-5}	Win
ZDT5	ANGB-MOMA vs MOPSO	1.10×10^{-5}	Loss	1.10×10^{-5}	Loss
ZDT6	ANGB-MOMA vs Original MOMA	1.10×10^{-5}	Win	1.10×10^{-5}	Win
ZDT6	ANGB-MOMA vs NSGA-II	7.50×10^{-3}	Win	5.00×10^{-3}	Win
DTLZ1	ANGB-MOMA vs Original MOMA	1.000	Tie	1.10×10^{-5}	Win
DTLZ1	ANGB-MOMA vs MOPSO	1.10×10^{-5}	Loss	1.10×10^{-5}	Loss
DTLZ4	ANGB-MOMA vs Original MOMA	1.10×10^{-5}	Win	1.10×10^{-5}	Win
DTLZ4	ANGB-MOMA vs Levy-flight MOMA	0.217	Tie	0.226	Tie
DTLZ7	ANGB-MOMA vs Original MOMA	2.98×10^{-5}	Win	3.29×10^{-5}	Win
DTLZ7	ANGB-MOMA vs NSGA-II	0.447	Tie	0.758	Tie
DTLZ7	ANGB-MOMA vs MOEA/D	3.45×10^{-3}	Loss	1.40×10^{-2}	Loss

Statistical results strengthen the comparative interpretation of the algorithm. On ZDT1, ZDT2, ZDT5, ZDT6, DTLZ4, and DTLZ7, ANGB-MOMA generally achieved statistically significant improvements over the original MOMA. On ZDT3 and ZDT4, the results show statistically significant weaknesses. ZDT3 exposes the difficulty of preserving separated Pareto-front segments, while ZDT4 exposes the weakness of the present MOMA-based search dynamics on strongly multimodal landscapes.

4.7. Runtime and computational cost discussion

Runtime results are reported to show the computational cost of the additional mechanisms introduced in ANGB-MOMA. All algorithms were run using the same number of function evaluations. The mean number of function evaluations was 70,040 for every algorithm on every benchmark problem. Therefore, the performance differences in HV and IGD+ are not due to unequal evaluation budgets.

Table 11 presents the mean runtime values for selected algorithms. The table focuses on Original MOMA, ANGB-MOMA, NSGA-II, MOPSO, and MOEA/D because these show the main trade-off between the proposed method and established optimizers.

Table 11. Runtime comparison in seconds

Problem	Original MOMA	ANGB-MOMA	NSGA-II	MOPSO	MOEA/D	ANGB-MOMA overhead vs Original
ZDT1	174.84	527.00	312.08	236.54	811.25	+201.4%
ZDT2	139.19	173.78	276.18	207.31	384.07	+24.9%
ZDT3	145.86	175.65	293.98	218.81	400.89	+20.4%
ZDT4	140.12	174.47	317.01	220.00	421.48	+24.5%
ZDT5	155.36	181.62	307.19	224.68	394.17	+16.9%
ZDT6	174.06	2217.90	552.51	259.59	808.97	+1174.2%
DTLZ1	721.83	162.15	312.27	207.45	433.95	-77.5%
DTLZ4	144.64	608.76	291.08	215.25	407.55	+320.9%
DTLZ7	155.36	190.60	1152.80	237.11	392.21	+22.7%

Table 11 shows that ANGB-MOMA generally requires more runtime than the original MOMA. This is expected because ANGB-MOMA adds adaptive parameter updates, neighborhood leader selection, grid-based archive management, sparse-cell leader selection, and boundary-preserving truncation to the baseline MOMA structure. These operations improve search behavior on many functions, but they introduce additional computational overhead.

The runtime increase is moderate on several problems. For example, the overhead relative to Original MOMA was 24.9% on ZDT2, 20.4% on ZDT3, 24.5% on ZDT4, 16.9% on ZDT5, and 22.7% on DTLZ7. These increases are reasonable given the improvements observed in HV and IGD+ on most of these functions.

However, some runtime results show high variability. The most noticeable case is ZDT6, where ANGB-MOMA recorded a much larger mean runtime. This high value was caused by a small number of unusually slow runs, as indicated by the large runtime standard deviation in the result file. Therefore, the runtime results should be interpreted together with the equal function-evaluation budget. The equal NFE confirms that the search budget was controlled, while the runtime values reflect implementation-level and archive-management overhead.

A second unusual case is DTLZ1, where ANGB-MOMA was faster than Original MOMA. This does not mean that the proposed mechanisms are always computationally cheaper. Rather, it shows that runtime also depends on the internal behavior of archive update, dominance comparison, and front management during the run. Because different algorithms may generate different archive structures and dominance patterns, equal NFE does not always lead to identical runtime scaling.

Overall, the runtime results show a clear trade-off. ANGB-MOMA improves the original MOMA on many convergence and coverage metrics, but this improvement is obtained at additional computational cost. The added cost is acceptable for applications where Pareto-front quality is more important than minimum execution time. However, for real-time or very large-scale problems, further code optimization or archive-size adaptation would be useful.

4.8. Discussion of convergence behavior

The ZDT6 convergence curve shows that ANGB-MOMA reduces IGD+ rapidly and then maintains a

stable low value toward the end of the search. This behavior is consistent with the grid-based archive and boundary-preserving mechanisms. The grid mechanism encourages leader selection from sparse objective-space regions, while boundary preservation reduces the probability that the currently identified extreme archive members will be deleted during truncation.

For DTLZ7, some comparator algorithms show deterioration in IGD+ after approximately 200 iterations. This behavior can occur when the external archive becomes increasingly concentrated in selected regions of the front while losing coverage in other regions. In such cases, the algorithm may continue to generate non-dominated solutions, but the distribution of those solutions may become less representative of the reference Pareto front. Since IGD+ measures the distance from reference front points to the obtained solution set, the loss of coverage in some regions can increase IGD+ even when the algorithm is still producing non-dominated solutions.

The deterioration observed in the comparator algorithms is therefore not necessarily caused by a failure to find non-dominated points. Instead, it suggests loss of distribution quality, archive imbalance, or leader-selection drift. In contrast, ANGB-MOMA reduces this effect through sparse-cell-biased leader selection and boundary-preserving archive truncation. These mechanisms do not eliminate the difficulty completely, but they help maintain a more stable approximation of the front on DTLZ7 than the original MOMA and its direct variants.

The convergence behavior on ZDT3 and ZDT4 also helps explain the limitations of ANGB-MOMA. On ZDT3, the discontinuous front requires the algorithm to preserve several separated Pareto-front segments at the same time. Although grid-based leader selection improves sparse-region exploration, it does not explicitly identify or protect disconnected segments. As a result, some front segments can still become underrepresented. On ZDT4, the multimodal landscape makes it difficult for the MOMA-based movement equations to escape local regions. The zero HV values obtained by the MOMA-based algorithms confirm that the present movement and archive mechanisms are not sufficient for this benchmark.

Therefore, the convergence analysis supports the same conclusion as the quantitative metrics. ANGB-MOMA improves convergence stability and archive distribution for many benchmark cases, especially when compared with the original MOMA. However, discontinuous and highly multimodal fronts remain challenging. These cases indicate clear directions for future improvement, including discontinuity-aware archive segmentation, adaptive restart mechanisms, and stronger multimodal exploration operators.

5. Conclusion

This paper proposed an Adaptive Neighborhood Grid-Boundary Multi-Objective Mayfly Algorithm (ANGB-MOMA) to improve the search behavior of the original Multi-Objective Mayfly Algorithm. The proposed method introduced four main mechanisms: adaptive parameter control, neighborhood-guided leader selection, grid-based archive and leader selection, and boundary-preserving archive truncation. These mechanisms were designed to improve convergence stability, reduce premature convergence, encourage sparse-region exploration, and preserve the identified extreme archive solutions during truncation.

The experimental evaluation used ZDT and DTLZ benchmark functions, including the multimodal ZDT4 problem. The comparison included Original MOMA, Levy-flight MOMA, Stochastic-inertia MOMA, ANGB-MOMA, NSGA-II, MOPSO, and MOEA/D. The study also included ablation experiments, parameter sensitivity analysis, explicit metric reproducibility settings, Holm-corrected Wilcoxon statistical tests, and runtime analysis.

The results show that ANGB-MOMA provides clear improvements over the original MOMA and the two MOMA variants on most benchmark functions. On ZDT1, ZDT2, ZDT5, ZDT6, DTLZ4, and DTLZ7, ANGB-MOMA produced improved HV and/or IGD+ values compared with the original MOMA. The improvement was especially strong on ZDT5 and ZDT6, where the original MOMA had difficulty maintaining effective convergence and objective-space coverage. On DTLZ4, ANGB-MOMA achieved the

best overall performance among all compared algorithms, showing that the proposed mechanisms can also be competitive against established optimizers on selected higher-dimensional benchmark structures.

The ablation study showed that the contribution of the proposed mechanisms is problem dependent. Adaptive parameter control was particularly useful on smooth continuous fronts such as ZDT2. Grid-based archive management was the strongest contributor on difficult biased and discrete cases such as ZDT5 and ZDT6. Boundary preservation improved the retention of identified archive extremes in selected cases, while neighborhood guidance contributed more effectively when combined with the other mechanisms rather than when used alone. These findings confirm that the proposed improvement is not caused by a single modification only, but by the combined effect of several archive and movement-control mechanisms.

The comparison also showed that ANGB-MOMA is not universally superior to all established multi-objective algorithms. MOPSO performed better on several ZDT functions, including ZDT3, ZDT4, ZDT5, and ZDT6, while MOEA/D achieved the best overall result on DTLZ7. Therefore, the main contribution of this paper is not the claim that ANGB-MOMA dominates all existing optimizers. Rather, the evidence supports the more specific conclusion that ANGB-MOMA significantly strengthens the MOMA framework and gives competitive performance on selected benchmark problems.

Statistical analysis supports this conclusion. After applying for the Holm-Bonferroni correction, ANGB-MOMA achieved more wins than losses against Original MOMA, Levy-flight MOMA, and Stochastic-inertia MOMA for both HV and IGD+. Against NSGA-II, MOPSO, and MOEA/D, the results were mixed, confirming that the proposed method is competitive but not consistently dominant. This balanced interpretation gives a more reliable assessment of the strengths and weaknesses of ANGB-MOMA.

The computational-cost analysis showed that ANGB-MOMA generally requires higher runtime than the original MOMA because of the additional adaptive-control, neighborhood-selection, grid-management, and boundary-preservation operations. However, all algorithms were evaluated using the same number of function evaluations, meaning that the comparison was based on an equal search budget. The runtime overhead is therefore a trade-off for improved convergence and archive quality on several benchmark functions.

Overall, the proposed ANGB-MOMA provides a useful and reproducible improvement to the MOMA family. Its strongest benefit is its ability to improve the convergence and archive distribution of the original MOMA on several continuous, biased, and discrete benchmark functions. At the same time, the results identify clear limitations on discontinuous and highly multimodal problems, which provide useful direction for future research.

6. Limitations and Future Work

Although ANGB-MOMA improves the original MOMA on many benchmark functions, several limitations remain.

First, ANGB-MOMA does not fully solve the discontinuous Pareto-front preservation problem. This was observed on ZDT3, where the proposed method underrepresented some separated front segments. The grid-based archive mechanism encourages sparse-region exploration, but it does not explicitly detect or preserve disconnected Pareto-front regions. Future work should therefore investigate discontinuity-aware archive segmentation, front-region clustering, or segment-preserving leader selection.

Second, the algorithm remains weak on highly multimodal landscapes such as ZDT4. Although ANGB-MOMA improved the IGD+ value of the original MOMA on ZDT4, all MOMA-based methods still produced poor HV values compared with NSGA-II, MOPSO, and MOEA/D. This shows that the current mayfly movement equations and archive mechanisms are not sufficient to overcome strong local optima in difficult multimodal spaces. Future work should consider adaptive restart strategies, opposition-based learning, stronger mutation operators, or hybrid local-search mechanisms.

Third, the ablation study showed that the full ANGB-MOMA does not always outperform every partial

variant. For example, the grid-boundary variant performed better than the full ANGB-MOMA on some DTLZ cases. This suggests that some mechanisms may interfere with one another depending on the benchmark landscape. Future work should develop problem-adaptive mechanism selection, where the algorithm can activate or reduce specific components based on online feedback from convergence rate, archive diversity, and dominance progress.

Fourth, the runtime analysis showed that ANGB-MOMA can require more computational time than the original MOMA. This is mainly due to grid assignment, sparse-cell leader selection, boundary checking, and archive truncation. Although the leading-order complexity remains dominated by non-dominated sorting, the additional archive operations increase practical runtime. Future work should therefore focus on faster archive update strategies, adaptive archive sizing, and more efficient grid-indexing structures.

Fifth, the current study used standard synthetic ZDT and DTLZ benchmark functions. These benchmarks are useful for controlled algorithm evaluation, but they do not fully represent the complexity of engineering optimization problems. Future studies should test ANGB-MOMA on real-world multi-objective engineering problems such as distributed generation planning, renewable energy system design, power-system stabilizer tuning, optimal power flow, microgrid energy management, and controller parameter optimization.

Finally, future work should extend the proposed framework to many objective optimization problems with more than two or three objectives. In such cases, grid-based archive control may become computationally expensive because the number of possible grid cells increases rapidly with the number of objectives. Decomposition-based leader selection, reference-vector guidance, or adaptive objective-space clustering may be required to make ANGB-MOMA suitable for many-objective optimization.

Declaration of Ethical Standards

The authors declare that this study complies with all ethical standards. No human participants, animal subjects, or personal data were involved in benchmark-based computational experiments.

Credit Authorship Contribution Statement

N. K. Prah II: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – Original Draft, Visualization.

E. A. Frimpong: Validation.

F. B. Effah: Validation, Writing – Review & Editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors declare that generative AI was used only for language editing and not for scientific content generation.

Funding

The authors were supported within the scope of the research by the KNUST Engineering Education Project, Power System Laboratory.

Data Availability

The MATLAB implementation, benchmark settings, and generated result files are available from the corresponding author upon reasonable request.

References

- Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511804441>

- Coello Coello, C. A., Pulido, G. T., & Lechuga, M. S. (2004). Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3), 256–279. <https://doi.org/10.1109/TEVC.2004.826067>
- Coello Coello, C. A., Van Veldhuizen, D. A., & Lamont, G. B. (2007). *Evolutionary algorithms for solving multi-objective problems* (2nd ed.). Springer. <https://doi.org/10.1007/978-0-387-36797-2>
- Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. Wiley.
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197. <https://doi.org/10.1109/4235.996017>
- Deb, K., Thiele, L., Laumanns, M., & Zitzler, E. (2002). Scalable multi-objective optimization test problems. In *Proceedings of the IEEE Congress on Evolutionary Computation* (pp. 825–830). <https://doi.org/10.1109/CEC.2002.1007032>
- Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 26(1), 29–41. <https://doi.org/10.1109/3477.484436>
- Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley.
- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press.
- Huband, S., Hingston, P., Barone, L., & While, L. (2006). A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, 10(5), 477–506. <https://doi.org/10.1109/TEVC.2005.861417>
- Ishibuchi, H., Masuda, H., Tanigaki, Y., & Nojima, Y. (2015). Modified distance calculation in generational distance and inverted generational distance. In *Evolutionary Multi-Criterion Optimization* (pp. 110–125). Springer. https://doi.org/10.1007/978-3-319-15892-1_8
- Kennedy, J. (1999). Small worlds and mega-minds: Effects of neighborhood topology on particle swarm performance. In *Proceedings of the Congress on Evolutionary Computation* (pp. 1931–1938). <https://doi.org/10.1109/CEC.1999.785509>
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks* (pp. 1942–1948). <https://doi.org/10.1109/ICNN.1995.488968>
- Knowles, J., & Corne, D. (2002). On metrics for comparing nondominated sets. In *Proceedings of the IEEE Congress on Evolutionary Computation* (pp. 711–716). <https://doi.org/10.1109/CEC.2002.1007013>
- Knowles, J. D., & Corne, D. W. (2000). Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary Computation*, 8(2), 149–172. <https://doi.org/10.1162/106365600568167>
- Li, M., Yang, S., & Liu, X. (2014). Shift-based density estimation for Pareto-based algorithms in many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 18(3), 348–365. <https://doi.org/10.1109/TEVC.2013.2262178>
- Mendes, R., Kennedy, J., & Neves, J. (2004). The fully informed particle swarm: Simpler, maybe better. *IEEE Transactions on Evolutionary Computation*, 8(3), 204–210. <https://doi.org/10.1109/TEVC.2004.826074>
- Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. *Advances in Engineering Software*, 114, 163–191. <https://doi.org/10.1016/j.advengsoft.2017.07.002>
- Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in Engineering Software*, 95, 51–67. <https://doi.org/10.1016/j.advengsoft.2016.01.008>
- Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 69, 46–61. <https://doi.org/10.1016/j.advengsoft.2013.12.007>
- Nagarajan, K., Reddy, K. B. N. K., Rajagopalan, A., Kumar, N. M. G., & Bajaj, M. (2024). Improved mayfly algorithm for optimizing power flow with integrated solar and wind energy. *International Journal of Electrical and Electronics Research*, 12(2), 415–420. <https://doi.org/10.37391/IJEER.120212>
- Srinivas, N., & Deb, K. (1994). Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, 2(3), 221–248. <https://doi.org/10.1162/evco.1994.2.3.221>
- Storn, R., & Price, K. (1997). Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4), 341–359. <https://doi.org/10.1023/A:1008202821328>
- Talbi, E.-G. (2009). *Metaheuristics: From design to implementation*. Wiley. <https://doi.org/10.1002/9780470496916>

- Twumasi, E., Frimpong, E. A., Prah II, N. K., & Gyasi, D. B. (2024). A novel improvement of particle swarm optimization using an improved velocity update function based on local best murmuration particle. *Journal of Electrical Systems and Information Technology*, 11, Article 42. <https://doi.org/10.1186/s43067-024-00168-8>
- Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1), 67–82. <https://doi.org/10.1109/4235.585893>
- Yang, X.-S. (2009). Firefly algorithms for multimodal optimization. In *Stochastic Algorithms: Foundations and Applications* (pp. 169–178). Springer. https://doi.org/10.1007/978-3-642-04944-6_14
- Yang, X.-S. (2010). *Nature-inspired metaheuristic algorithms* (2nd ed.). Luniver Press.
- Zervoudakis, K., & Tsafarakis, S. (2020). A mayfly optimization algorithm. *Computers & Industrial Engineering*, 145, 106559. <https://doi.org/10.1016/j.cie.2020.106559>
- Zhu, J., Zhou, Y., Wei, Y., Luo, Q., & Huang, H. (2024). Optimal stochastic power flow using enhanced multi-objective mayfly algorithm. *Heliyon*, 10(4), e26427. <https://doi.org/10.1016/j.heliyon.2024.e26427>
- Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, 8(2), 173–195. <https://doi.org/10.1162/106365600568202>
- Zitzler, E., Laumanns, M., & Thiele, L. (2001). *SPEA2: Improving the strength Pareto evolutionary algorithm* (TIK-Report 103). ETH Zurich. <https://doi.org/10.3929/ethz-a-004284029>
- Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4), 257–271. <https://doi.org/10.1109/4235.797969>