

An Improved Multi-objective Centered Collision Optimization Algorithm IMOCCO and Its Engineering Applications

Yanyin Sun

7114124037@email2.ncu.edu.cn

Nanchang University

Yuqi Chen

Nanchang University

Research Article

Keywords:

Posted Date: October 21st, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11252490/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

An Improved Multi-objective Centered Collision Optimization Algorithm IMOCCO and Its Engineering Applications

Yanyin Sun^{1*} and Yuqi Chen¹

^{1*}Jiluan College, Nanchang University, Nanchang, 330031, China.

*Corresponding author(s). E-mail(s): 7114124037@email2.ncu.edu.cn;

Abstract

This paper proposes an improved multi-objective centered collision optimization algorithm (IMOCCO), based on the Centered Collision Optimizer (CCO). From the multi-objective framework extended from CCO, IMOCCO introduces three improvements: manifold-feature collision, mass-restitution collision, and angular-sparsity guidance, acting on the search-operator, individual-interaction, and environmental-selection layers, respectively. IMOCCO is validated on 31 standard test functions, an ablation study, and two engineering problems against six comparators, using inverted generational distance, generational distance, spacing, hypervolume, and Wilcoxon rank-sum and Friedman tests. IMOCCO ranks first with a Friedman mean rank of 1.73; on the four-bar truss and disc-brake problems its Wilcoxon result is 2/0/0, significantly outperforming all competitors. The ablation results indicate that all three components contribute to performance, with angular-sparsity guidance producing the largest deterioration in overall rank when removed.

Keywords: multi-objective optimization, IMOCCO, centered collision optimization, engineering design problem, metaheuristic

1 Introduction

In practical engineering design, objectives such as structural weight, displacement, stiffness, cost, and natural frequency often conflict with one another. Unlike single-objective optimization, multi-objective optimization (MOO)[1] does not seek a single optimum but returns a set of mutually compromising Pareto-optimal solutions[2] for decision makers. Developing an efficient multi-objective algorithm with good convergence, solution diversity, and low computational cost is of both theoretical and engineering significance.

Methods for solving MOO problems fall into two categories. The first comprises mathematical programming methods, such as the weighted-sum

and epsilon-constraint methods[1, 3], which transform a multi-objective problem into a sequence of single-objective problems; however, they are sensitive to problem structure and struggle with complex or irregular Pareto fronts. The second category consists of population-based metaheuristics, which are gradient-free and obtain an approximate Pareto front in a single run. Deb et al.[4] proposed NSGA-II with fast non-dominated sorting and crowding distance, yet its diversity is insufficient on disconnected fronts. Zhang and Li[5] proposed MOEA/D through decomposition, but the subproblem weights are hard to adapt to irregular front shapes. Coello Coello et al.[6] introduced particle swarm into multi-objective optimization, and Tripathi et al.[7] proposed a

MOPSO with time-varying inertia and acceleration coefficients to improve best-particle selection and diversity maintenance. Lang et al.[8] proposed CCO and Kaveh and Mahdavi[9] proposed CBO, both drawing on physical collision laws. Taken together, existing approaches still face a persistent convergence-diversity trade-off on irregular or disconnected Pareto fronts, while constrained engineering applications also place pressure on computational efficiency[10–17].

The Centered Collision Optimizer (CCO), proposed by Lang et al.[8], updates individual positions via the centered collision equation and constructs a rotation-invariant feature search space through eigendecomposition of the population covariance matrix, with a collision coefficient decaying cosine-wise from 0.87 to 0.2. However, its original scalar-fitness ordering does not directly provide the dominance relations, diversity control, or archive management required for multi-objective optimization. This paper extends CCO into the multi-objective centered-collision framework MOCCO, using fast non-dominated sorting, crowding distance, an external archive, and polynomial mutation. Nevertheless, MOCCO has three drawbacks: it does not exploit the manifold structure of the Pareto set and cannot distinguish "spreading along the front" from "converging toward the front"; its collision weight is fixed and ignores the physical mass ratio and restitution coefficient; and crowding distance has poor resolution on irregular and disconnected fronts, while the archive only stores solutions passively without guiding the search.

To address these drawbacks, this paper proposes the improved multi-objective algorithm IMOCCO with three non-overlapping improvements: manifold-feature collision (S1) estimates the tangent and normal spaces of the Pareto set at the search-operator layer and applies anisotropic scaling to the collision displacement; mass-restitution collision (S2) defines individual mass by Pareto quality and restitution coefficient by objective-space distance at the individual-interaction layer, so that a single collision rule carries both convergence and diversity pressure; angular-sparsity guidance (S3) uses a unified sparsity measure in normalized objective space at the environmental-selection layer to drive both archive pruning and collision-center selection,

thereby converting the external archive from passive storage into an active source of search guidance.

The remainder of this paper is organized as follows. Section 2 presents the multi-objective framework MOCCO extended from CCO. Section 3 describes IMOCCO and its three improvements. Section 4 reports the standard-test-function experiments, the ablation study, and the engineering examples. Section 5 concludes the paper.

2 The Multi-objective Framework MOCCO

2.1 Basic Definitions of Multi-objective Optimization

As mentioned, a multi-objective problem optimizes a set of conflicting objectives simultaneously, generally expressed as:

$$\begin{aligned} \min \mathbf{F}(\mathbf{X}) &= \{f_1(\mathbf{X}), f_2(\mathbf{X}), \dots, \\ &\quad f_M(\mathbf{X})\}, \\ \text{s.t. } g_j(\mathbf{X}) &\leq 0, \quad j = 1, \dots, m, \\ x_{i,\min} &\leq x_i \leq x_{i,\max} \end{aligned} \quad (1)$$

where $\mathbf{X} = [x_1, x_2, \dots, x_n]$ is the n -dimensional decision vector, g_j is the j -th inequality constraint, f_i is the i -th objective, and $x_{i,\min}$ and $x_{i,\max}$ are the lower and upper bounds of x_i . Since objectives conflict, reducing one may worsen another, and the goal is to find Pareto-optimal solutions under constraints.

For a minimization problem, solution $\mathbf{u}$ dominates solution $\mathbf{v}$ (denoted $\mathbf{u} \prec \mathbf{v}$) as defined below:

$$\begin{aligned} \mathbf{u} \prec \mathbf{v} &\iff \\ \forall i, \quad f_i(\mathbf{u}) &\leq f_i(\mathbf{v}), \\ \wedge \exists k, \quad f_k(\mathbf{u}) &< f_k(\mathbf{v}). \end{aligned} \quad (2)$$

If no solution in the search space dominates $\mathbf{x}$, then $\mathbf{x}$ is called a Pareto-optimal (non-dominated) solution. The set of all Pareto-optimal solutions in decision space is the Pareto Set (PS), and its image in objective space is the Pareto Front (PF).

Throughout this paper, $\odot$ denotes element-wise multiplication; N is the population size, N_r the external archive size, T the maximum number of iterations, t the current iteration, n the number

of decision variables, and m the number of objectives; α is the CCO collision coefficient, decaying cosine-wise from 0.87 to 0.2 and scheduling the overall collision strength.

2.2 The Centered Collision Optimizer CCO

CCO, proposed by Lang et al.[8], is a physics-based metaheuristic. Each individual is viewed as a colliding body with mass and velocity and updates its position through a centered collision with a collision center, the post-collision velocity being given by the one-dimensional centered collision equation. Compared with other algorithms, CCO has two distinctive features: it uses the classical mechanics centered collision equation as the position-update rule, and it constructs a rotation-invariant feature space via eigendecomposition of the covariance matrix of the inferior half of the population.

2.3 The Multi-objective Centered-collision Framework MOCCO

The three mechanisms of CCO all rely on a total order of scalar fitness. The key to multi-objective extension is to rebuild such a total order and provide a corresponding acceptance criterion. Individuals are sorted in ascending order; a smaller rank means a better individual.

2.3.1 Pareto Dominance and Population Total Order

For a minimization problem, individual a dominates individual b , denoted $a \prec b$, if:

$$\begin{aligned} \mathbf{x}_a \prec \mathbf{x}_b &\iff \\ \forall i, & f_i(\mathbf{x}_a) \leq f_i(\mathbf{x}_b), \\ \wedge \exists k, & f_k(\mathbf{x}_a) < f_k(\mathbf{x}_b). \end{aligned} \quad (3)$$

MOCCO uses fast non-dominated sorting[4] to divide the population into non-dominated fronts, then sorts individuals within the same front by crowding distance in descending order, yielding a total order consistent with CCO: lower rank first, and among equal rank larger crowding distance first. The crowding distance of individual i is:

$$d_i = \sum_{k=1}^m \frac{f_k^{i+1} - f_k^{i-1}}{f_k^{\max} - f_k^{\min}} \quad (4)$$

where $a_{i,j}$ and $b_{i,j}$ are the two nearest neighbors of individual i on either side along the j -th objective; boundary individuals are assigned infinity.

2.3.2 Collision Center and External Archive

In CCO the collision center is drawn from better individuals. MOCCO maintains an external archive A of size N_r storing the non-dominated solutions found so far, and the collision center is randomly drawn from the archive. Archive members are all non-dominated and hence naturally satisfy the requirement of being "better than the current individual", while also enriching the search direction. When the archive overflows, it is truncated by crowding distance; at termination the archive is returned as the approximate Pareto front.

2.3.3 Acceptance Criterion and Mutation Operator

CCO uses greedy acceptance, which has no counterpart in the multi-objective case. MOCCO adopts the Pareto acceptance criterion:

$$\mathbf{x}_j^{t+1} = \begin{cases} \mathbf{x}_j^{\text{new}}, & \mathbf{x}_j^{\text{new}} \prec \mathbf{x}_j^t \\ & \vee (\mathbf{x}_j^{\text{new}} \parallel \mathbf{x}_j^t \\ & \wedge \text{rand} < 0.5), \\ \mathbf{x}_j^t, & \text{otherwise.} \end{cases} \quad (5)$$

where $\parallel$ denotes mutual non-dominance. When the offspring and current solution are mutually non-dominated, neither is strictly preferred by Pareto dominance. We therefore use an unbiased Bernoulli tie-break ($p = 0.5$), assigning equal probability to the two incomparable candidates rather than tuning this value as a performance parameter. A recent MOPSO study likewise uses random selection with probability 0.5 specifically for mutually non-dominated ties and notes its role in reducing regional bias and preserving diversity [18].

Moreover, consistent with mainstream algorithms such as NSGA-II and MOEA/D[4, 5], MOCCO uses standard polynomial mutation:

$$x'_k = x_k + \delta_k (\text{ub}_k - \text{lb}_k) \quad (6)$$

$$\delta_k = \begin{cases} (2u)^{\frac{1}{\eta_m+1}} - 1, & u < 0.5, \\ 1 - (2(1-u))^{\frac{1}{\eta_m+1}}, & u \geq 0.5 \end{cases} \quad (7)$$

where u is a random number on $[0,1]$, the mutation probability is $p_m = 1/n$, and the distribution index is $\eta_m = 20$. Both MOCCO and IMOCCO include this operator.

3 The Proposed Multi-objective Algorithm IMOCCO

3.1 Overall Framework

IMOCCO takes CCO as the search engine and MOCCO as the multi-objective framework, and introduces three non-overlapping improvements acting on three layers: S1 on the search-operator layer, handling the geometry of decision space; S2 on the individual-interaction layer, rebuilding the physical meaning of collision; and S3 on the environmental-selection layer, improving both archive pruning and collision-center selection. The separation of these mechanisms is intentional: each component modifies a different stage of the optimization process, which also permits their contributions to be examined independently in the ablation study.

3.2 Manifold-feature Collision (S1)

The Pareto set of a continuous multi-objective problem is an $(m-1)$ -dimensional piecewise continuous manifold in decision space, a regularity widely exploited by estimation-of-distribution algorithms[19]. CCO constructs a feature space from the covariance of the inferior half but treats all feature directions equally, neither exploiting this manifold structure nor distinguishing movement along the front from movement toward it. S1 estimates the local tangent and normal spaces via a rank-weighted covariance of the elite half and scales the two types of movement differently. This decomposition is intended to decouple diversity-oriented motion along the front from convergence-oriented motion toward the front.

(1) Elite rank-weighted covariance. Taking the sorted elite half of size $\mu = \lceil N/2 \rceil$, the rank weight is:

$$w_i = \frac{\ln(\mu + 0.5) - \ln i}{\sum_{l=1}^{\mu} (\ln(\mu + 0.5) - \ln l)} \quad (8)$$

The weighted mean and weighted covariance matrix are constructed as:

$$\bar{\mathbf{x}}_w = \sum_{i=1}^{\mu} w_i \mathbf{x}_i \quad (9)$$

$$\mathbf{C} = \sum_{i=1}^{\mu} w_i (\mathbf{x}_i - \bar{\mathbf{x}}_w)^T (\mathbf{x}_i - \bar{\mathbf{x}}_w) + \varepsilon \mathbf{I} \quad (10)$$

Eigendecomposition of $\mathbf{C}$ yields the orthogonal basis $\mathbf{V}$ and eigenvalues sorted in descending order:

$$\begin{aligned} \mathbf{C} &= \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T, \\ \mathbf{\Lambda} &= \text{diag}(\lambda_1, \dots, \lambda_n), \\ \lambda_1 &\geq \dots \geq \lambda_n \end{aligned} \quad (11)$$

Since the elite set lies near the Pareto set, the principal eigenvectors estimate the local tangent space of the manifold. Using rank-decreasing weights reduces the influence of lower-ranked elite points and limits the extent to which a single outlying point can dominate the covariance estimate.

(2) Tangent and normal space decomposition. The first $(m-1)$ eigenvectors span the tangent space and the rest span the normal space:

$$\begin{aligned} \mathcal{T} &= \text{span}(\mathbf{v}_1, \dots, \mathbf{v}_{m-1}), \\ \mathcal{N} &= \text{span}(\mathbf{v}_m, \dots, \mathbf{v}_n) \end{aligned} \quad (12)$$

Tangent directions correspond to changes of the individual along the front (i.e., trade-off directions), while normal directions correspond to changes in the individual's distance to the front.

(3) Anisotropic scaling. The collision displacement is decomposed into an attractive component d_{pull} toward the collision center and a random differential component d_{rand} , which are scaled separately in the feature coordinate system $\mathbf{y} = \mathbf{xV}$:

$$\begin{aligned} \mathbf{y}_j^{\text{new}} &= \mathbf{y}_j + \mathbf{s}_{\text{pull}} \odot \mathbf{d}_{\text{pull}} \\ &\quad + \mathbf{s}_{\text{rand}} \odot \mathbf{d}_{\text{rand}} \end{aligned} \quad (13)$$

Table 1 Correspondence between the mechanisms of CCO and MOCCO

CCO mechanism	Corresponding design in MOCCO
Sort by scalar fitness ascending	Fast non-dominated sorting + crowding distance gives total order
Collision center from better individuals	Randomly sampled from archive; archive members are non-dominated
Greedy acceptance	Pareto acceptance; accept with prob 0.5 when mutually non-dominated
No archive	Archive + crowding-distance truncation; return archive at end
Random reset out of bounds	Truncation out of bounds, consistent with other algorithms
Feature-space search, dynamic space allocation	Kept unchanged

The scaling coefficients transition linearly from isotropic to anisotropic with iteration progress τ , avoiding misleading estimates at an early stage when the feature directions are poorly estimated:

$$s_k = 1 + \tau(c_k - 1), \tau = \frac{t}{T} \quad (14)$$

The four coefficients and their effects are listed in Table 2. Tangent attraction is weakened and tangent perturbation amplified so that individuals do not cluster at the same point on the front; normal attraction is over-relaxed and normal perturbation contracted so that individuals collapse onto the front without wasteful off-manifold wandering. Thus, the anisotropic scaling explicitly assigns different search roles to tangent and normal directions instead of applying a common update strength to all coordinates.

(4) Normal-stagnation sparse resampling. A stagnation counter cnt_j is maintained for each individual, incremented on rejection and reset on acceptance. When cnt_j exceeds a threshold T_{cnt} , the individual keeps its tangent coordinates and resamples only a subset of normal coordinates with probability p_r :

$$y_{j,k} \leftarrow \bar{y}_k + (2\text{rand} - 1)\beta \text{ext}_k, \quad k \in \mathcal{N}, \quad \text{rand} < p_{\text{res}} \quad (15)$$

where h_k is the half-side length of the search box along direction k :

$$\text{ext}_k = \sum_{i=1}^n |V_{ik}| \frac{\text{ub}_i - \text{lb}_i}{2} \quad (16)$$

This scale is determined by the search box and does not vanish as the population collapses, so escape capability remains after population contraction. The multimodality of ZDT4, DTLZ1, and DTLZ3 lies in their distance functions, whose variables span the normal space of the front; resampling only part of the normal coordinates

lets the individual escape the local normal basin while preserving its tangent trade-off position.

In addition, a global escape magnitude σ_E is maintained, with feedback taken as the fraction f_r of archive members dominated by the offspring of the current generation, and adapted as:

$$\sigma_E \leftarrow \text{clip}(\sigma_E \exp(c_E(\rho^* - \rho_t)), 10^{-4}, \sigma_E^{\text{max}}) \quad (17)$$

This directly measures front progress: σ_E contracts when progress is fast and expands when progress slows. σ_E only acts on the normal Gaussian perturbation of the inferior half, leaving the elite half undisturbed, so escape capability is provided without destroying existing convergence. Using archive improvement as feedback links the escape magnitude to observed front progress rather than to iteration count alone.

3.3 Mass-restitution Collision (S2)

CCO uses a single deterministic coefficient α so that all individuals move toward their own centers, which is disconnected from the physics of collision, whose outcome depends on the mass ratio and restitution coefficient. In the multi-objective case these two quantities can carry convergence and diversity information. S2 defines mass by Pareto quality and restitution coefficient by objective-space distance. Consequently, pairwise collision intensity becomes state-dependent: Pareto quality influences the relative mass, whereas objective-space proximity controls collision elasticity.

$$v'_1 = \frac{(m_1 - em_2)v_1 + (1 + e)m_2v_2}{m_1 + m_2} \quad (18)$$

where m is mass and e is the restitution coefficient, with $e=0$ for a perfectly inelastic collision and $e=1$ for a perfectly elastic one. Accordingly:

Table 2 Tangential and normal anisotropic scaling coefficients

Direction	Attraction coefficient	Perturbation coefficient	Effect
Tangential ($k \leq m-1$)	$c_{PT} = 0.5$, decayed	$c_{RT} = 1.5$, amplified	Prevent clustering on the front, enhance spreading
Normal ($k > m-1$)	$c_{PN} = 1.2$, over-relaxed	$c_{RN} = 0.2$, contracted	Accelerate collapse to front, suppress off-manifold wandering

(1) Mass is given by Pareto quality. Let j be the rank of an individual in the population total order, and define:

$$q_j = 1 - \frac{j-1}{N-1}, \quad (19)$$

$$m_j = m_{\min} + (1 - m_{\min})q_j$$

Archive members take $m_c = 1$. Better individuals are "heavy" and worse ones are "light", so in a collision the worse individual is always pushed more.

(2) The restitution coefficient is given by the objective-space distance:

$$e = \exp\left(-\frac{\|\hat{\mathbf{f}}_j - \hat{\mathbf{f}}_c\|}{\theta \bar{d}}\right) \quad (20)$$

where $\hat{\mathbf{f}}$ is the normalized objective vector, $\bar{d}$ is the average pairwise distance of the current population, and θ is a scale parameter. The closer two solutions are in objective space, the closer e is to 1 and the more elastic the collision.

(3) Collision weights and position update:

$$w_c = (1 - \alpha) \cdot \frac{(1 + e)m_c}{m_j + m_c}, \quad (21)$$

$$w_j = 1 - w_c$$

$$\mathbf{x}_j^{\text{new}} = \mathbf{x}_j + w_c(\mathbf{x}_c - \mathbf{x}_j) + \mathbf{d}_{\text{rand}} \quad (22)$$

This rule has a clear dual physical and algorithmic interpretation. When two solutions are far apart in objective space, the collision is nearly inelastic, the weights reduce to a simple attraction, and the worse solution is "absorbed" by the better one, producing convergence pressure. When they are very close, the collision is nearly elastic and the weights can exceed 1, bouncing the individual to the other side of the collision center and preserving diversity. In this way, the same collision law shifts continuously between attraction-dominated convergence and elasticity-driven local

spreading according to the relationship between the paired solutions.

α is retained as a global schedule of collision strength; the collision law only modulates it. Mass ratio and restitution coefficient determine the relative strength between different pairs within one generation, while α controls the overall strength across iterations; the two roles are clearly separated.

3.4 Angular-sparsity Guidance (S3)

Crowding distance has poor resolution on three-objective and irregular/disconnected fronts, while grid-based methods depend on the sensitive grid-size parameter. More importantly, in most swarm-based multi-objective algorithms the archive only stores solutions passively without guiding the search. S3 uses a unified sparsity measure in normalized objective space to drive both archive pruning and collision-center selection. The objective is to use one consistent geometric criterion both to maintain archive coverage and to bias subsequent search toward underrepresented trade-off directions.

$$\hat{\mathbf{f}} = \frac{\mathbf{f} - \mathbf{z}^{\min}}{\mathbf{z}^{\max} - \mathbf{z}^{\min}} \quad (23)$$

(1) Archive pruning. When the archive exceeds N_r , the minimum solution on each objective is always retained to preserve front width; then the candidate with the largest minimum distance to the already selected set is repeatedly added until the archive reaches size N_r :

$$\mathbf{a}^* = \arg \max_{\mathbf{a} \in A \setminus S} \min_{\mathbf{s} \in S} \|\hat{\mathbf{f}}_{\mathbf{a}} - \hat{\mathbf{f}}_{\mathbf{s}}\| \quad (24)$$

This farthest-first rule explicitly maximizes the minimum spacing of the solution set, yielding an approximately uniform distribution that naturally suits disconnected and degenerate fronts without any grid parameter. Because it does not require a predefined grid resolution, the pruning rule avoids

introducing an additional discretization parameter into archive maintenance.

(2) Collision-center selection. For individual j , an angular neighborhood A_j is formed from the archive members whose directions are closest to it, of size $\lceil 0.1 \cdot N_r \rceil$. A roulette selection is then performed within this neighborhood based on angular sparsity:

$$sp_i = 1 - \max_{i' \neq i} \frac{\langle \hat{\mathbf{f}}_i, \hat{\mathbf{f}}_{i'} \rangle}{\|\hat{\mathbf{f}}_i\| \|\hat{\mathbf{f}}_{i'}\|} \quad (25)$$

$$P(c = i) = \frac{sp_i}{\sum_{l \in N_{\text{ang}}(j)} sp_l} \quad (26)$$

Selecting the center only within the individual's own angular neighborhood ensures convergence along its own trade-off direction, preventing it from being pulled toward the globally sparsest region and disrupting distribution. Within the neighborhood, preferring sparse members drives the population to fill gaps in the front. The probability P_A of choosing an archive member rather than a peer as the collision center increases linearly with iteration:

$$p_A = 0.2 + 0.6\tau \quad (27)$$

Early on, collisions occur mostly with peers to preserve exploration; later, collisions occur mostly with the front to refine convergence.

4 Numerical Examples

This section evaluates IMOCCO on 31 standard test functions and two engineering structural optimization problems to verify its effectiveness and robustness. The four-bar truss and disc brake are widely studied practical problems, further validating the algorithm on real constrained engineering instances. All algorithms are implemented in MATLAB; the structural analysis uses the direct stiffness method, and all computations are performed on a Core 2 Duo 2.53 GHz machine with runtime measured in clock time.

4.1 Experimental Setup and Performance Metrics

4.1.1 Performance Metrics

Let P be the approximate Pareto front obtained by an algorithm and P^* a set of reference points

sampled uniformly on the true Pareto front. Four widely used metrics are adopted.

(1) Inverted generational distance (IGD)[20]. It measures the average distance from each reference point in P^* to P , reflecting both convergence and distribution; smaller is better:

$$\text{IGD}(P^*, P) = \frac{1}{|P^*|} \sum_{\mathbf{v} \in P^*} \min_{\mathbf{u} \in P} d(\mathbf{u}, \mathbf{v}) \quad (28)$$

(2) Generational distance (GD)[21]. It measures the average distance from each solution in P to P^* , mainly reflecting convergence; smaller is better:

$$\text{GD}(P, P^*) = \left[\frac{1}{|P|} \sum_{\mathbf{u} \in P} \left(\min_{\mathbf{v} \in P^*} d(\mathbf{u}, \mathbf{v}) \right)^2 \right]^{1/2} \quad (29)$$

(3) Spacing[22]. It measures the consistency of neighboring spacing on P , reflecting distribution uniformity; smaller is better:

$$\text{Spacing} = \left[\frac{1}{|P| - 1} \sum_{i=1}^{|P|} (d_i - \bar{d})^2 \right]^{1/2} \quad (30)$$

(4) Hypervolume (HV)[23, 24]. It measures the volume enclosed by P and a reference point in objective space, reflecting both convergence and diversity; larger is better.

4.1.2 Compared Algorithms and Parameters

The compared algorithms are MOCCO together with five multi-objective comparators: NSGA-II, MOPSO, MOSCSO, MOJS, and MOAVOA[4, 6, 25–27]. Here, MOSCSO denotes the archive-based multi-objective implementation of the sand cat search rule[25], using non-dominated solution storage, uniformly sampled archive leaders, and grid-based archive truncation. All algorithms use a population size of 100, an archive size of 100, a maximum of 300 iterations, and 30 independent runs; the parameter settings are listed in Table 3.

Table 3 Unified parameter settings of all algorithms

Algorithm	Population size	Archive size	Max iterations	Independent runs
IMOCCO	100	100	300	30
MOCCO	100	100	300	30
NSGA-II	100	—	300	30
MOPSO	100	100	300	30
MOSCSO	100	100	300	30
MOJS	100	—	300	30
MOAVOA	100	100	300	30

4.1.3 Statistical Tests

To assess the significance of differences, IMOCCO is used as the baseline. The Wilcoxon rank-sum test[28] at significance level 0.05 is applied problem by problem, counting the numbers of "better/equal/worse" functions between IMOCCO and each competitor, and the Friedman test[29] is used to rank the algorithms overall. Together, these two analyses distinguish problem-wise pairwise significance from the algorithms' aggregate rank across the complete benchmark set.

4.2 Standard Test Functions

4.2.1 Test Function Suite

Thirty-one standard test functions with different Pareto-front characteristics are selected: five bi-objective problems ZDT1, ZDT2, ZDT3, ZDT4, and ZDT6[30] covering convex, non-convex, and disconnected fronts; seven scalable problems DTLZ1–DTLZ7[31]; nine WFG problems[32]; and ten CEC2009 UF problems[33].

4.2.2 Results and Statistics

Tables 4–7 give the mean and standard deviation of IGD, GD, Spacing, and HV for the seven algorithms on the 31 functions, with the best mean in each row shown in bold. Across the IGD results in Table 4, IMOCCO obtains the lowest mean value on the majority of the test functions, although several difficult functions remain competitive or favor other methods.

Table 8 summarizes the "better/equal/worse" counts of the Wilcoxon rank-sum test. On IGD, IMOCCO is significantly better than the six competitors on 26, 26, 29, 27, 28, and 28 functions, respectively; the counts for GD, Spacing, and HV likewise show more favorable than unfavorable comparisons against each competitor, while also

indicating that the magnitude of the advantage is metric-dependent.

Table 9 gives the Friedman mean ranks on the four metrics and overall. IMOCCO's mean ranks on IGD, GD, Spacing, and HV are 1.42, 2.10, 1.77, and 1.65, with an overall mean rank of 1.73, ranking first among the seven algorithms.

To visualize the relative standing of the algorithms, Fig. 1 shows a heat map of the Friedman mean ranks on the four metrics and overall.

4.2.3 Convergence and Distribution Analysis

Fig. 2 compares the Pareto fronts obtained by all algorithms on the representative UF1 problem with the true front. The IMOCCO solutions visually track the reference front closely over most of its extent and remain comparatively well distributed, whereas MOJS and MOAVOA display more evident deviation or clustering in several regions.

Fig. 3 shows the approximate fronts of all algorithms, and Fig. 4 summarizes the four metrics on the representative problem, further illustrating front coverage.

The box plots in Fig. 5 show that IMOCCO has a lower median IGD and smaller dispersion, supporting a comparatively concentrated IGD distribution across repeated runs rather than relying only on mean performance.

IGD only reflects the combined level of convergence and distribution. Figs. 6–8 further give box plots of GD, Spacing, and HV across functions, providing evidence from convergence accuracy, distribution uniformity, and front coverage, respectively.

Figs. 9 and 10 summarize the overall Friedman ranks and the Wilcoxon better/equal/worse

Table 4 IGD results of all algorithms on the 31 test functions (mean $\pm$ std, 30 independent runs)

Function	IMOCCO	MOCCO	NSGA2	MOPSO	MOSCSO	MOJS	MOAVOA
ZDT1	5.068e-03 $\pm$ 2.284e-04	9.381e-03 $\pm$ 5.095e-04	1.885e-01 $\pm$ 1.222e-02	9.675e-03 $\pm$ 1.479e-03	6.371e-03 $\pm$ 3.363e-04	2.632e-02 $\pm$ 4.094e-02	8.996e-03 $\pm$ 7.341e-04
ZDT2	4.905e-03 $\pm$ 2.001e-04	9.316e-03 $\pm$ 7.979e-04	3.729e-01 $\pm$ 3.713e-02	1.079e-02 $\pm$ 1.754e-03	4.674e-02 $\pm$ 1.532e-01	7.414e-02 $\pm$ 2.042e-01	9.453e-03 $\pm$ 9.122e-04
ZDT3	5.670e-03 $\pm$ 3.825e-04	1.000e-02 $\pm$ 6.953e-04	1.792e-01 $\pm$ 1.159e-02	1.142e-02 $\pm$ 1.552e-03	6.810e-03 $\pm$ 3.190e-04	8.673e-02 $\pm$ 7.772e-02	1.017e-02 $\pm$ 1.233e-03
ZDT4	3.676e-01 $\pm$ 4.844e-01	6.016e-01 $\pm$ 6.117e-01	7.809e-01 $\pm$ 1.173e-01	7.485e-02 $\pm$ 1.802e-01	6.492e-03 $\pm$ 3.515e-04	4.948e-01 $\pm$ 1.953e-01	9.328e-03 $\pm$ 8.354e-04
ZDT6	3.475e-03 $\pm$ 1.091e-04	3.602e-03 $\pm$ 1.210e-04	2.351e+00 $\pm$ 7.769e-02	7.267e-03 $\pm$ 1.825e-03	5.267e-03 $\pm$ 2.958e-04	1.027e-01 $\pm$ 4.287e-01	6.952e-03 $\pm$ 6.607e-04
DTLZ1	7.366e-01 $\pm$ 5.313e-01	9.794e-01 $\pm$ 4.853e-01	1.101e+00 $\pm$ 4.919e-01	1.691e+01 $\pm$ 2.925e+00	1.815e+01 $\pm$ 3.101e+00	3.739e+00 $\pm$ 2.808e+00	8.008e+00 $\pm$ 4.148e+00
DTLZ2	5.685e-02 $\pm$ 7.978e-04	7.376e-02 $\pm$ 3.248e-03	6.914e-02 $\pm$ 2.958e-03	1.019e-01 $\pm$ 5.056e-03	1.235e-01 $\pm$ 4.403e-03	8.916e-02 $\pm$ 4.914e-03	1.775e-01 $\pm$ 1.434e-02
DTLZ3	5.977e+01 $\pm$ 1.310e+01	7.160e+01 $\pm$ 1.583e+01	1.559e+01 $\pm$ 6.180e+00	1.859e+02 $\pm$ 1.093e+01	1.687e+02 $\pm$ 1.290e+01	8.809e+01 $\pm$ 3.354e+01	1.056e+02 $\pm$ 7.679e+01
DTLZ4	9.804e-02 $\pm$ 9.639e-02	7.062e-02 $\pm$ 2.465e-03	7.627e-02 $\pm$ 1.011e-02	1.672e-01 $\pm$ 5.571e-02	1.104e-01 $\pm$ 4.963e-02	8.412e-02 $\pm$ 8.841e-03	2.019e-01 $\pm$ 5.355e-02
DTLZ5	4.801e-03 $\pm$ 1.507e-04	6.429e-03 $\pm$ 3.813e-04	5.580e-03 $\pm$ 3.064e-04	2.540e-02 $\pm$ 3.130e-03	1.373e-02 $\pm$ 2.388e-03	9.060e-03 $\pm$ 1.149e-03	5.445e-02 $\pm$ 1.145e-02
DTLZ6	4.516e-03 $\pm$ 1.289e-04	4.913e-03 $\pm$ 5.932e-04	7.072e+00 $\pm$ 1.767e-01	6.404e-03 $\pm$ 8.796e-04	6.939e-03 $\pm$ 4.044e-04	1.862e+00 $\pm$ 1.141e+00	9.877e-03 $\pm$ 1.179e-03
DTLZ7	6.721e-02 $\pm$ 2.335e-03	1.709e-01 $\pm$ 1.846e-01	7.426e-01 $\pm$ 5.732e-02	1.419e-01 $\pm$ 2.032e-02	1.261e-01 $\pm$ 9.554e-03	3.116e-01 $\pm$ 2.040e-01	1.601e-01 $\pm$ 6.247e-02
wfg1	1.011e+00 $\pm$ 1.437e-02	1.096e+00 $\pm$ 1.409e-02	1.551e+00 $\pm$ 8.213e-02	1.216e+00 $\pm$ 2.086e-02	1.155e+00 $\pm$ 6.621e-03	1.208e+00 $\pm$ 2.209e-02	1.250e+00 $\pm$ 2.503e-02
wfg2	1.148e-02 $\pm$ 2.744e-04	1.362e-02 $\pm$ 4.923e-04	2.047e-01 $\pm$ 3.619e-02	1.887e-02 $\pm$ 2.846e-03	1.673e-02 $\pm$ 1.001e-03	4.162e-02 $\pm$ 2.098e-02	4.698e-02 $\pm$ 2.209e-02
wfg3	1.307e-02 $\pm$ 1.102e-04	1.742e-02 $\pm$ 6.227e-04	1.621e-02 $\pm$ 1.501e-03	2.149e-02 $\pm$ 9.322e-04	2.616e-02 $\pm$ 1.016e-03	3.125e-02 $\pm$ 5.538e-03	4.708e-02 $\pm$ 3.751e-03
wfg4	6.951e-02 $\pm$ 1.658e-03	7.007e-02 $\pm$ 4.178e-03	7.510e-02 $\pm$ 4.186e-03	7.278e-02 $\pm$ 1.518e-03	7.089e-02 $\pm$ 1.989e-03	7.198e-02 $\pm$ 1.273e-03	8.757e-02 $\pm$ 1.019e-02
wfg5	6.872e-02 $\pm$ 2.967e-03	7.025e-02 $\pm$ 7.025e-02	7.632e-02 $\pm$ 2.064e-03	7.207e-02 $\pm$ 1.439e-03	7.065e-02 $\pm$ 1.829e-03	7.234e-02 $\pm$ 9.419e-04	8.381e-02 $\pm$ 6.042e-03
wfg6	1.407e-02 $\pm$ 2.384e-04	1.735e-02 $\pm$ 6.991e-04	2.489e-02 $\pm$ 1.118e-02	2.006e-02 $\pm$ 1.101e-03	2.544e-02 $\pm$ 1.887e-03	3.310e-02 $\pm$ 6.720e-03	4.794e-02 $\pm$ 9.737e-03
wfg7	1.423e-02 $\pm$ 2.289e-04	1.787e-02 $\pm$ 9.069e-04	1.752e-02 $\pm$ 1.287e-03	1.985e-02 $\pm$ 1.075e-03	2.451e-02 $\pm$ 1.497e-03	2.875e-02 $\pm$ 4.096e-03	4.455e-02 $\pm$ 5.307e-03
wfg8	2.589e-01 $\pm$ 3.499e-02	2.794e-01 $\pm$ 1.120e-02	2.843e-01 $\pm$ 5.282e-02	3.092e-01 $\pm$ 6.643e-03	2.689e-01 $\pm$ 1.376e-02	2.700e-01 $\pm$ 3.565e-02	2.514e-01 $\pm$ 4.841e-02
wfg9	2.011e-02 $\pm$ 7.404e-04	2.191e-02 $\pm$ 6.299e-04	2.383e-02 $\pm$ 2.268e-03	2.566e-02 $\pm$ 6.326e-04	2.523e-02 $\pm$ 1.177e-03	2.707e-02 $\pm$ 1.916e-03	3.756e-02 $\pm$ 3.812e-03
UF1	3.782e-02 $\pm$ 1.289e-02	5.188e-02 $\pm$ 1.708e-02	6.367e-02 $\pm$ 2.306e-02	8.162e-02 $\pm$ 5.744e-03	4.942e-02 $\pm$ 9.680e-03	4.621e-02 $\pm$ 5.358e-03	7.490e-02 $\pm$ 1.171e-02
UF2	1.744e-02 $\pm$ 4.213e-03	1.661e-02 $\pm$ 2.848e-03	2.506e-02 $\pm$ 8.795e-03	5.276e-02 $\pm$ 6.383e-03	1.978e-02 $\pm$ 2.591e-03	3.007e-02 $\pm$ 3.296e-03	3.798e-02 $\pm$ 4.779e-03
UF3	3.175e-01 $\pm$ 4.152e-02	3.607e-01 $\pm$ 2.455e-02	2.443e-01 $\pm$ 5.272e-02	3.411e-01 $\pm$ 7.423e-02	5.727e-01 $\pm$ 1.730e-01	3.543e-01 $\pm$ 8.419e-02	3.359e-01 $\pm$ 3.742e-02
UF4	4.478e-02 $\pm$ 2.780e-03	4.852e-02 $\pm$ 4.273e-03	3.843e-02 $\pm$ 2.565e-03	4.795e-02 $\pm$ 1.573e-03	4.925e-02 $\pm$ 2.265e-03	5.200e-02 $\pm$ 3.798e-03	5.817e-02 $\pm$ 3.732e-03
UF5	1.911e-01 $\pm$ 1.011e-01	2.205e-01 $\pm$ 1.510e-01	2.210e-01 $\pm$ 3.956e-02	5.859e-01 $\pm$ 3.210e-01	5.359e-01 $\pm$ 1.826e-01	7.530e-01 $\pm$ 3.065e-01	7.932e-01 $\pm$ 2.305e-01
UF6	2.634e-01 $\pm$ 1.158e-01	2.635e-01 $\pm$ 1.577e-01	2.066e-01 $\pm$ 7.776e-02	7.979e-01 $\pm$ 1.452e-01	4.636e-01 $\pm$ 1.387e-01	5.979e-01 $\pm$ 2.157e-01	7.892e-01 $\pm$ 1.365e-01
UF7	2.493e-02 $\pm$ 5.119e-03	2.900e-02 $\pm$ 4.747e-03	1.972e-01 $\pm$ 1.092e-01	5.208e-02 $\pm$ 4.801e-03	2.192e-02 $\pm$ 4.186e-03	6.634e-02 $\pm$ 5.940e-02	6.803e-02 $\pm$ 2.928e-02
UF8	1.080e-01 $\pm$ 1.031e-02	2.033e-01 $\pm$ 9.612e-02	2.286e-01 $\pm$ 8.453e-02	2.201e-01 $\pm$ 4.766e-02	3.028e-01 $\pm$ 1.130e-01	2.579e-01 $\pm$ 5.946e-02	1.916e-01 $\pm$ 5.060e-02
UF9	1.198e-01 $\pm$ 5.005e-02	1.434e-01 $\pm$ 6.478e-02	2.348e-01 $\pm$ 8.241e-02	2.834e-01 $\pm$ 3.882e-02	3.063e-01 $\pm$ 1.566e-01	3.140e-01 $\pm$ 7.756e-02	1.850e-01 $\pm$ 2.704e-02
UF10	3.603e-01 $\pm$ 1.208e-01	4.126e-01 $\pm$ 5.954e-02	4.573e-01 $\pm$ 6.586e-02	2.032e+00 $\pm$ 4.782e-01	9.237e-01 $\pm$ 1.786e-01	1.237e+00 $\pm$ 4.812e-01	7.511e-01 $\pm$ 2.228e-01

Table 5 GD results of all algorithms on the 31 test functions (mean $\pm$ std, 30 independent runs)

Function	IMOCCO	MOCCO	NSGA2	MOPSO	MOSCSO	MOJS	MOAVOA
ZDT1	3.121e-04 $\pm$ 3.603e-05 2.453e-04	8.452e-04 $\pm$ 6.815e-05 8.275e-04	2.879e-02 $\pm$ 1.219e-02 4.253e-02	9.666e-04 $\pm$ 1.865e-04 1.031e-03	1.282e-04 $\pm$ 4.339e-05 4.999e-05	2.291e-03 $\pm$ 4.367e-03 5.280e-03	1.280e-03 $\pm$ 6.013e-03 2.735e-03
ZDT2	$\pm$ 4.130e-05 2.945e-04	$\pm$ 8.763e-05 7.359e-04	$\pm$ 9.898e-03 3.036e-02	$\pm$ 1.822e-04 7.999e-04	$\pm$ 1.424e-05 1.765e-04	$\pm$ 1.747e-02 8.496e-03	$\pm$ 1.422e-02 2.010e-04
ZDT3	$\pm$ 4.579e-05 3.979e-02	$\pm$ 7.388e-05 6.571e-02	$\pm$ 1.007e-02 1.074e-01	$\pm$ 1.447e-04 7.832e-03	$\pm$ 1.675e-05 6.852e-05	$\pm$ 1.045e-02 5.295e-02	$\pm$ 3.836e-05 1.051e-01
ZDT4	$\pm$ 5.371e-02 2.523e-02	$\pm$ 6.883e-02 1.820e-02	$\pm$ 1.732e-02 2.875e-01	$\pm$ 1.941e-02 3.651e-02	$\pm$ 1.059e-05 2.840e-03	$\pm$ 2.156e-02 9.611e-02	$\pm$ 5.704e-01 7.784e-02
ZDT6	$\pm$ 1.231e-02 1.537e-01	$\pm$ 1.311e-02 1.872e-01	$\pm$ 2.823e-02 3.282e-01	$\pm$ 2.090e-02 4.857e+00	$\pm$ 1.111e-02 2.797e+00	$\pm$ 4.976e-02 8.084e-01	$\pm$ 5.214e-02 3.570e+00
DTLZ1	$\pm$ 1.455e-01 2.248e-03	$\pm$ 8.905e-02 2.972e-03	$\pm$ 1.216e-01 2.041e-03	$\pm$ 4.952e-01 1.178e-02	$\pm$ 2.571e-01 6.539e-03	$\pm$ 5.380e-01 7.327e-03	$\pm$ 6.062e-01 2.061e-02
DTLZ2	$\pm$ 1.491e-04 1.366e+01	$\pm$ 2.412e-04 1.360e+01	$\pm$ 6.824e-04 3.363e+00	$\pm$ 2.831e-03 3.703e+01	$\pm$ 1.068e-03 2.052e+01	$\pm$ 9.846e-04 1.299e+01	$\pm$ 3.867e-03 2.556e+01
DTLZ3	$\pm$ 2.056e+00 2.747e-03	$\pm$ 2.336e+00 2.730e-03	$\pm$ 8.487e-01 1.760e-03	$\pm$ 4.018e+00 9.283e-03	$\pm$ 8.891e-01 6.136e-03	$\pm$ 4.277e+00 4.929e-03	$\pm$ 4.203e+00 2.444e-02
DTLZ4	$\pm$ 1.267e-03 1.998e-04	$\pm$ 1.977e-04 3.679e-04	$\pm$ 8.704e-05 1.302e-04	$\pm$ 2.821e-03 4.598e-03	$\pm$ 3.641e-03 1.368e-03	$\pm$ 1.017e-03 8.535e-04	$\pm$ 8.559e-03 8.859e-03
DTLZ5	$\pm$ 7.179e-05 1.809e-04	$\pm$ 5.385e-05 5.338e-05	$\pm$ 1.240e-04 7.590e-01	$\pm$ 2.285e-03 4.033e-04	$\pm$ 3.743e-04 4.836e-05	$\pm$ 1.805e-04 3.743e-01	$\pm$ 3.058e-03 4.754e-05
DTLZ6	$\pm$ 8.989e-05 4.886e-03	$\pm$ 2.620e-05 8.760e-03	$\pm$ 1.254e-02 1.840e-01	$\pm$ 3.181e-04 1.012e-02	$\pm$ 3.066e-06 5.017e-03	$\pm$ 1.733e-01 1.400e-02	$\pm$ 3.421e-06 1.985e-02
DTLZ7	$\pm$ 3.767e-04 1.020e-01	$\pm$ 1.654e-03 1.099e-01	$\pm$ 3.626e-02 7.960e-02	$\pm$ 1.032e-03 1.461e-01	$\pm$ 2.150e-03 1.151e-01	$\pm$ 2.120e-02 1.207e-01	$\pm$ 5.478e-02 1.497e-01
wfg1	$\pm$ 3.168e-03 5.088e-04	$\pm$ 1.298e-03 5.825e-04	$\pm$ 1.094e-02 4.952e-04	$\pm$ 9.434e-03 1.043e-03	$\pm$ 5.905e-04 6.691e-04	$\pm$ 1.956e-03 2.367e-03	$\pm$ 1.271e-02 2.615e-03
wfg2	$\pm$ 6.137e-05 3.235e-04	$\pm$ 6.704e-05 1.070e-03	$\pm$ 8.552e-05 5.728e-04	$\pm$ 1.765e-04 1.778e-03	$\pm$ 8.761e-05 1.717e-03	$\pm$ 1.127e-03 3.026e-03	$\pm$ 6.676e-04 3.800e-03
wfg3	$\pm$ 5.184e-05 6.436e-03	$\pm$ 9.494e-05 6.511e-03	$\pm$ 2.215e-04 6.665e-03	$\pm$ 1.563e-04 6.849e-03	$\pm$ 1.100e-04 6.409e-03	$\pm$ 7.029e-04 6.583e-03	$\pm$ 4.361e-04 6.673e-03
wfg4	$\pm$ 2.641e-05 6.374e-03	$\pm$ 4.018e-04 6.509e-03	$\pm$ 4.134e-04 6.807e-03	$\pm$ 7.379e-05 6.798e-03	$\pm$ 3.910e-05 6.399e-03	$\pm$ 6.738e-05 6.601e-03	$\pm$ 1.413e-04 6.676e-03
wfg5	$\pm$ 2.695e-04 2.616e-04	$\pm$ 3.951e-04 8.236e-04	$\pm$ 9.084e-05 1.584e-03	$\pm$ 9.885e-05 1.264e-03	$\pm$ 3.682e-05 1.381e-03	$\pm$ 6.347e-05 2.638e-03	$\pm$ 1.820e-04 3.219e-03
wfg6	$\pm$ 3.940e-05 3.145e-04	$\pm$ 9.709e-05 9.323e-04	$\pm$ 1.401e-03 3.819e-04	$\pm$ 1.476e-04 1.222e-03	$\pm$ 1.197e-04 1.277e-03	$\pm$ 7.322e-04 1.990e-03	$\pm$ 4.359e-04 2.877e-03
wfg7	$\pm$ 6.223e-05 2.920e-02	$\pm$ 1.078e-04 3.223e-02	$\pm$ 7.268e-05 2.923e-02	$\pm$ 1.132e-04 3.434e-02	$\pm$ 9.934e-05 3.209e-02	$\pm$ 3.326e-04 2.445e-02	$\pm$ 2.662e-04 2.238e-02
wfg8	$\pm$ 5.823e-03 1.003e-03	$\pm$ 1.770e-03 1.221e-03	$\pm$ 5.423e-03 9.305e-04	$\pm$ 1.934e-03 1.764e-03	$\pm$ 3.072e-03 1.195e-03	$\pm$ 4.085e-03 1.839e-03	$\pm$ 6.355e-03 2.087e-03
wfg9	$\pm$ 1.086e-04 1.939e-03	$\pm$ 9.886e-05 1.736e-03	$\pm$ 3.636e-04 3.054e-04	$\pm$ 7.256e-05 3.091e-02	$\pm$ 1.071e-04 8.393e-03	$\pm$ 1.745e-04 1.416e-02	$\pm$ 2.127e-04 1.857e-02
UF1	$\pm$ 5.131e-04 1.337e-03	$\pm$ 7.830e-04 1.373e-03	$\pm$ 5.870e-04 4.796e-04	$\pm$ 2.880e-02 1.211e-02	$\pm$ 9.836e-03 2.235e-03	$\pm$ 1.450e-02 5.131e-03	$\pm$ 2.195e-02 6.269e-03
UF2	$\pm$ 5.871e-04 6.781e-03	$\pm$ 6.640e-04 5.971e-03	$\pm$ 1.749e-04 1.143e-02	$\pm$ 6.197e-03 2.266e-02	$\pm$ 1.023e-03 5.196e-02	$\pm$ 6.250e-03 3.167e-02	$\pm$ 5.752e-03 1.098e-02
UF3	$\pm$ 4.558e-03 4.827e-03	$\pm$ 4.558e-03 4.924e-03	$\pm$ 6.082e-03 3.708e-03	$\pm$ 6.276e-02 5.520e-03	$\pm$ 7.439e-02 5.573e-03	$\pm$ 1.576e-02 5.982e-03	$\pm$ 8.499e-03 6.468e-03
UF4	$\pm$ 2.288e-04 1.786e-02	$\pm$ 2.809e-04 2.552e-02	$\pm$ 4.182e-04 3.303e-02	$\pm$ 2.754e-04 3.538e-01	$\pm$ 4.460e-04 6.209e-02	$\pm$ 5.715e-04 1.070e-01	$\pm$ 5.636e-04 1.834e-01
UF5	$\pm$ 5.004e-03 2.215e-02	$\pm$ 6.397e-03 4.408e-02	$\pm$ 1.287e-02 3.772e-02	$\pm$ 2.680e-01 3.780e-01	$\pm$ 4.176e-02 8.143e-02	$\pm$ 5.563e-02 1.025e-01	$\pm$ 6.188e-02 2.336e-01
UF6	$\pm$ 1.594e-02 1.628e-03	$\pm$ 2.113e-02 1.052e-03	$\pm$ 1.631e-02 1.913e-04	$\pm$ 2.842e-01 1.005e-02	$\pm$ 8.637e-02 3.319e-03	$\pm$ 7.694e-02 1.735e-02	$\pm$ 1.419e-01 1.450e-02
UF7	$\pm$ 2.290e-03 3.352e-02	$\pm$ 6.912e-04 1.392e-01	$\pm$ 2.213e-04 1.301e-02	$\pm$ 6.370e-03 2.474e-01	$\pm$ 3.405e-03 3.040e-01	$\pm$ 1.669e-02 1.377e-01	$\pm$ 1.180e-02 9.175e-02
UF8	$\pm$ 2.095e-02 4.893e-02	$\pm$ 1.257e-01 7.472e-02	$\pm$ 1.433e-02 3.328e-02	$\pm$ 1.161e-01 2.696e-01	$\pm$ 1.659e-01 2.788e-01	$\pm$ 1.005e-01 2.682e-01	$\pm$ 1.025e-01 1.339e-01
UF9	$\pm$ 4.637e-02 1.248e-01	$\pm$ 7.634e-02 1.693e-01	$\pm$ 3.534e-02 7.920e-02	$\pm$ 9.808e-02 1.228e+00	$\pm$ 1.843e-01 3.250e-01	$\pm$ 1.505e-01 5.167e-01	$\pm$ 1.314e-01 9.140e-01
UF10	$\pm$ 4.922e-02	$\pm$ 6.813e-02	$\pm$ 2.761e-02	$\pm$ 5.585e-01	$\pm$ 6.343e-01	$\pm$ 6.016e-01	$\pm$ 1.291e+00

Table 6 Spacing results of all algorithms on the 31 test functions (mean $\pm$ std, 30 independent runs)

Function	IMOCCO	MOCCO	NSGA2	MOPSO	MOSCSO	MOJS	MOAVOA
ZDT1	4.815e-03 $\pm 3.735e-04$	6.896e-03 $\pm 6.611e-04$	1.868e-02 $\pm 1.269e-02$	5.264e-03 $\pm 4.324e-04$	1.131e-02 $\pm 8.713e-04$	1.101e-02 $\pm 3.728e-03$	1.222e-02 $\pm 1.850e-03$
ZDT2	4.937e-03 $\pm 4.301e-04$	7.367e-03 $\pm 7.135e-04$	1.468e-02 $\pm 5.657e-03$	7.791e-03 $\pm 2.359e-03$	1.048e-02 $\pm 2.922e-03$	1.102e-02 $\pm 5.941e-03$	1.441e-02 $\pm 1.041e-02$
ZDT3	4.754e-03 $\pm 5.288e-04$	7.735e-03 $\pm 7.877e-04$	1.910e-02 $\pm 1.011e-02$	7.277e-03 $\pm 1.149e-03$	1.172e-02 $\pm 8.582e-04$	1.463e-02 $\pm 9.009e-03$	1.332e-02 $\pm 2.130e-03$
ZDT4	5.800e-03 $\pm 1.165e-03$	8.366e-03 $\pm 1.533e-03$	7.641e-02 $\pm 8.747e-02$	5.638e-03 $\pm 5.011e-04$	1.111e-02 $\pm 8.543e-04$	9.181e-03 $\pm 3.213e-03$	1.430e-02 $\pm 8.421e-03$
ZDT6	9.926e-02 $\pm 7.822e-02$	6.401e-02 $\pm 5.277e-02$	4.214e-02 $\pm 2.364e-02$	1.206e-01 $\pm 1.299e-01$	2.712e-02 $\pm 9.643e-02$	1.383e-01 $\pm 1.823e-01$	7.353e-02 $\pm 1.142e-01$
DTLZ1	1.206e-01 $\pm 8.230e-02$	1.957e-01 $\pm 9.632e-02$	5.123e-01 $\pm 5.621e-01$	3.453e+00 $\pm 7.551e-01$	1.859e+00 $\pm 4.717e-01$	5.541e-01 $\pm 3.609e-01$	2.656e+00 $\pm 1.610e+00$
DTLZ2	3.521e-02 $\pm 2.388e-03$	6.040e-02 $\pm 5.254e-03$	5.181e-02 $\pm 4.715e-03$	6.394e-02 $\pm 1.017e-02$	6.323e-02 $\pm 1.465e-02$	5.780e-02 $\pm 6.129e-03$	1.075e-01 $\pm 1.774e-02$
DTLZ3	1.085e+01 $\pm 2.131e+00$	1.108e+01 $\pm 2.732e+00$	4.985e+00 $\pm 2.075e+00$	2.755e+01 $\pm 5.730e+00$	1.280e+01 $\pm 2.885e+00$	9.178e+00 $\pm 5.653e+00$	1.971e+01 $\pm 2.123e+01$
DTLZ4	3.938e-02 $\pm 1.108e-02$	5.774e-02 $\pm 5.387e-03$	5.051e-02 $\pm 5.186e-03$	7.502e-02 $\pm 2.124e-02$	7.095e-02 $\pm 1.469e-02$	5.951e-02 $\pm 7.594e-03$	8.395e-02 $\pm 2.126e-02$
DTLZ5	6.745e-03 $\pm 6.104e-04$	9.355e-03 $\pm 8.208e-04$	9.237e-03 $\pm 8.367e-04$	3.412e-02 $\pm 3.078e-02$	1.408e-02 $\pm 1.443e-03$	9.000e-03 $\pm 1.196e-03$	4.516e-02 $\pm 2.461e-02$
DTLZ6	6.359e-03 $\pm 5.263e-04$	7.417e-03 $\pm 1.324e-03$	4.235e-01 $\pm 8.376e-02$	9.883e-03 $\pm 1.921e-03$	1.281e-02 $\pm 1.033e-03$	3.317e-01 $\pm 1.758e-01$	1.452e-02 $\pm 1.517e-03$
DTLZ7	5.030e-02 $\pm 4.734e-03$	7.076e-02 $\pm 1.633e-02$	1.089e-01 $\pm 2.650e-02$	6.501e-02 $\pm 6.160e-03$	5.166e-02 $\pm 1.181e-02$	8.010e-02 $\pm 2.821e-02$	1.332e-01 $\pm 1.299e-01$
wfg1	8.516e-02 $\pm 3.857e-03$	8.319e-02 $\pm 3.906e-03$	8.212e-03 $\pm 8.116e-03$	1.544e-01 $\pm 2.115e-02$	8.870e-02 $\pm 7.367e-03$	1.079e-01 $\pm 8.725e-03$	1.857e-01 $\pm 4.784e-02$
wfg2	1.075e-02 $\pm 1.142e-03$	1.629e-02 $\pm 1.648e-03$	1.300e-02 $\pm 1.668e-03$	1.800e-02 $\pm 3.597e-03$	2.394e-02 $\pm 1.673e-03$	2.027e-02 $\pm 7.002e-03$	2.710e-02 $\pm 5.583e-03$
wfg3	1.526e-02 $\pm 9.021e-04$	1.836e-02 $\pm 2.131e-03$	1.981e-02 $\pm 2.454e-03$	1.497e-02 $\pm 1.426e-03$	3.299e-02 $\pm 2.159e-03$	1.815e-02 $\pm 2.049e-03$	3.493e-02 $\pm 4.220e-03$
wfg4	1.765e-02 $\pm 1.442e-03$	2.021e-02 $\pm 1.874e-03$	2.064e-02 $\pm 2.377e-03$	1.846e-02 $\pm 2.120e-03$	3.251e-02 $\pm 2.668e-03$	2.106e-02 $\pm 1.898e-03$	3.744e-02 $\pm 9.036e-03$
wfg5	1.655e-02 $\pm 2.219e-03$	2.027e-02 $\pm 1.952e-03$	2.048e-02 $\pm 2.054e-03$	1.845e-02 $\pm 1.816e-03$	3.303e-02 $\pm 2.345e-03$	1.989e-02 $\pm 1.733e-03$	3.666e-02 $\pm 5.311e-03$
wfg6	1.758e-02 $\pm 1.691e-03$	2.100e-02 $\pm 2.503e-03$	2.056e-02 $\pm 2.658e-03$	2.155e-02 $\pm 3.640e-03$	3.315e-02 $\pm 4.383e-03$	2.475e-02 $\pm 6.472e-03$	3.549e-02 $\pm 7.929e-03$
wfg7	1.643e-02 $\pm 1.916e-03$	2.186e-02 $\pm 2.299e-03$	2.044e-02 $\pm 2.650e-03$	2.181e-02 $\pm 3.399e-03$	3.154e-02 $\pm 1.749e-03$	2.271e-02 $\pm 6.712e-03$	3.781e-02 $\pm 7.649e-03$
wfg8	2.374e-02 $\pm 2.961e-02$	1.991e-02 $\pm 4.464e-03$	2.210e-02 $\pm 1.629e-02$	3.111e-02 $\pm 1.058e-02$	3.117e-02 $\pm 9.916e-03$	4.054e-02 $\pm 3.387e-02$	4.219e-02 $\pm 3.000e-02$
wfg9	1.610e-02 $\pm 1.926e-03$	1.935e-02 $\pm 2.223e-03$	2.078e-02 $\pm 1.639e-03$	1.748e-02 $\pm 1.908e-03$	3.007e-02 $\pm 1.956e-03$	1.741e-02 $\pm 1.906e-03$	3.483e-02 $\pm 4.131e-03$
UF1	5.345e-03 $\pm 2.610e-03$	8.321e-03 $\pm 6.057e-03$	1.023e-02 $\pm 1.166e-02$	1.060e-01 $\pm 1.208e-01$	1.091e-02 $\pm 2.493e-02$	2.595e-02 $\pm 3.211e-02$	1.307e-02 $\pm 7.714e-03$
UF2	5.511e-03 $\pm 3.478e-03$	6.982e-03 $\pm 2.543e-03$	1.642e-02 $\pm 6.900e-03$	3.509e-02 $\pm 2.632e-02$	9.710e-03 $\pm 6.746e-04$	1.257e-02 $\pm 7.156e-03$	1.313e-02 $\pm 2.827e-03$
UF3	8.041e-03 $\pm 1.223e-02$	8.655e-02 $\pm 7.987e-02$	3.619e-02 $\pm 4.405e-02$	7.166e-02 $\pm 8.230e-02$	9.982e-03 $\pm 1.596e-02$	1.277e-02 $\pm 3.086e-02$	3.472e-03 $\pm 2.712e-03$
UF4	6.747e-03 $\pm 2.326e-03$	7.676e-03 $\pm 1.386e-03$	1.338e-02 $\pm 4.427e-03$	8.219e-03 $\pm 2.163e-03$	1.050e-02 $\pm 1.212e-03$	9.862e-03 $\pm 1.562e-03$	1.348e-02 $\pm 4.667e-03$
UF5	8.990e-03 $\pm 1.908e-02$	5.423e-02 $\pm 5.099e-02$	5.968e-02 $\pm 4.804e-02$	5.347e-01 $\pm 5.599e-01$	6.670e-03 $\pm 1.368e-02$	6.443e-02 $\pm 1.629e-01$	1.008e-01 $\pm 5.390e-02$
UF6	2.360e-02 $\pm 3.147e-02$	7.775e-02 $\pm 4.889e-02$	7.960e-02 $\pm 9.749e-02$	5.409e-01 $\pm 8.997e-01$	9.389e-03 $\pm 1.045e-02$	6.276e-02 $\pm 1.674e-01$	8.040e-02 $\pm 7.513e-02$
UF7	6.664e-03 $\pm 5.835e-03$	5.810e-03 $\pm 3.200e-03$	9.086e-03 $\pm 9.036e-03$	3.524e-02 $\pm 3.912e-02$	8.662e-03 $\pm 8.059e-04$	3.126e-02 $\pm 6.753e-02$	1.580e-02 $\pm 8.523e-03$
UF8	1.598e-01 $\pm 1.443e-01$	1.347e-01 $\pm 8.023e-02$	4.615e-02 $\pm 1.950e-02$	7.925e-01 $\pm 3.718e-01$	1.162e-01 $\pm 5.550e-02$	4.189e-01 $\pm 3.312e-01$	1.865e-01 $\pm 1.902e-01$
UF9	1.305e-01 $\pm 8.948e-02$	1.132e-01 $\pm 5.360e-02$	4.415e-02 $\pm 2.188e-02$	7.343e-01 $\pm 3.856e-01$	9.242e-02 $\pm 3.877e-02$	6.314e-01 $\pm 4.241e-01$	2.951e-01 $\pm 2.704e-01$
UF10	2.472e-01 $\pm 2.044e-01$	2.940e-01 $\pm 1.259e-01$	1.428e-01 $\pm 1.002e-01$	2.984e+00 $\pm 1.373e+00$	5.767e-02 $\pm 1.086e-01$	6.648e-01 $\pm 8.103e-01$	3.158e-01 $\pm 2.785e-01$

Table 7 HV results of all algorithms on the 31 test functions (mean $\pm$ std, 30 independent runs)

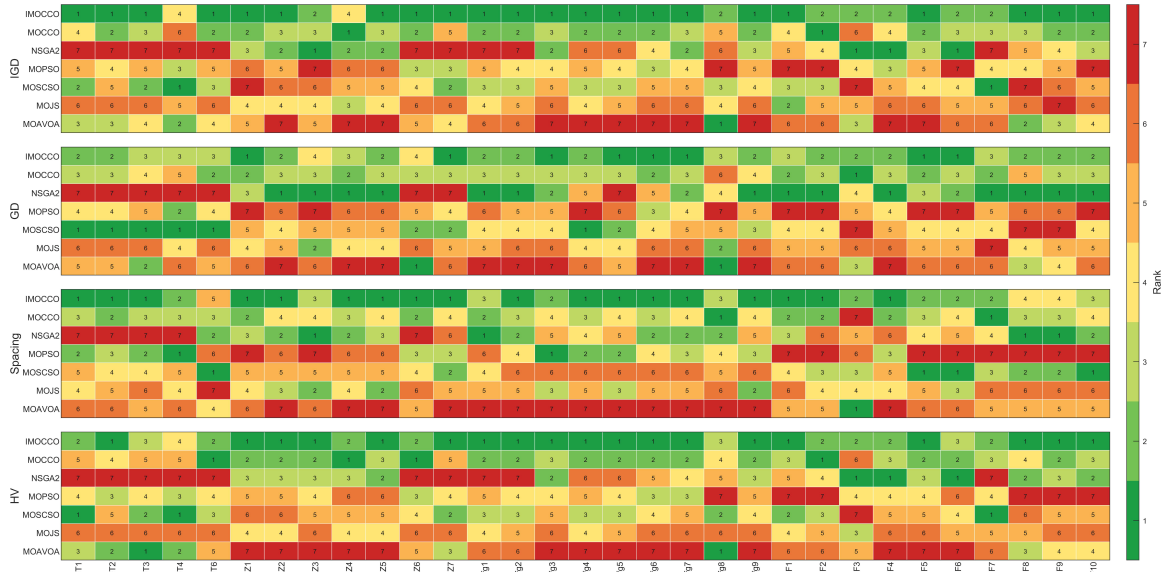
Function	IMOCO	MOCCO	NSGA2	MOPSO	MOSCSO	MOJS	MOAVOA
ZDT1	7.174e-01 $\pm$ 4.110e-04 4.426e-01	7.106e-01 $\pm$ 7.764e-04 4.349e-01	4.909e-01 $\pm$ 1.394e-02 9.885e-02	7.111e-01 $\pm$ 2.003e-03 4.350e-01	7.177e-01 $\pm$ 3.742e-04 4.188e-01	6.902e-01 $\pm$ 5.464e-02 3.962e-01	7.148e-01 $\pm$ 8.249e-04 4.395e-01
ZDT2	$\pm$ 3.926e-04 5.982e-01	$\pm$ 1.267e-03 5.935e-01	$\pm$ 1.971e-02 4.955e-01	$\pm$ 1.824e-03 5.957e-01	$\pm$ 8.914e-02 5.991e-01	$\pm$ 1.139e-01 5.598e-01	$\pm$ 1.492e-03 5.999e-01
ZDT3	$\pm$ 1.033e-03 4.419e-01	$\pm$ 1.060e-03 2.973e-01	$\pm$ 1.634e-02 4.325e-02	$\pm$ 3.700e-03 6.450e-01	$\pm$ 1.149e-04 7.178e-01	$\pm$ 6.858e-02 2.209e-01	$\pm$ 4.608e-03 7.146e-01
ZDT4	$\pm$ 3.101e-01 3.881e-01	$\pm$ 2.895e-01 3.883e-01	$\pm$ 3.679e-02 0	$\pm$ 1.882e-01 3.849e-01	$\pm$ 3.644e-04 3.868e-01	$\pm$ 1.448e-01 3.642e-01	$\pm$ 9.385e-04 3.846e-01
ZDT6	$\pm$ 1.733e-04 7.054e-02	$\pm$ 1.232e-04 1.602e-02	$\pm$ 0 1.049e-02	$\pm$ 1.626e-03 0	$\pm$ 2.702e-04 0	$\pm$ 8.751e-02 5.051e-04	$\pm$ 7.708e-04 0
DTLZ1	$\pm$ 1.655e-01 5.466e-01	$\pm$ 4.648e-02 5.184e-01	$\pm$ 4.801e-02 5.154e-01	$\pm$ 0 4.502e-01	$\pm$ 0 4.391e-01	$\pm$ 2.493e-03 4.685e-01	$\pm$ 0 3.382e-01
DTLZ2	$\pm$ 2.207e-03 0	$\pm$ 4.708e-03 0	$\pm$ 8.501e-03 0	$\pm$ 1.063e-02 0	$\pm$ 8.475e-03 0	$\pm$ 1.094e-02 0	$\pm$ 2.118e-02 0
DTLZ3	$\pm$ 0 5.217e-01	$\pm$ 0 5.262e-01	$\pm$ 0 5.169e-01	$\pm$ 0 4.378e-01	$\pm$ 0 4.821e-01	$\pm$ 0 5.061e-01	$\pm$ 0 3.685e-01
DTLZ4	$\pm$ 6.372e-02 1.993e-01	$\pm$ 4.348e-03 1.979e-01	$\pm$ 1.407e-02 1.988e-01	$\pm$ 2.979e-02 1.774e-01	$\pm$ 2.517e-02 1.936e-01	$\pm$ 1.055e-02 1.947e-01	$\pm$ 5.635e-02 1.427e-01
DTLZ5	$\pm$ 1.795e-04 1.998e-01	$\pm$ 2.763e-04 2.000e-01	$\pm$ 3.309e-04 0	$\pm$ 6.536e-03 1.991e-01	$\pm$ 6.414e-04 1.988e-01	$\pm$ 1.002e-03 2.962e-02	$\pm$ 1.687e-02 1.966e-01
DTLZ6	$\pm$ 8.997e-05 2.669e-01	$\pm$ 3.664e-04 2.340e-01	$\pm$ 0 4.216e-02	$\pm$ 4.976e-04 2.375e-01	$\pm$ 2.186e-04 2.513e-01	$\pm$ 6.984e-02 2.184e-01	$\pm$ 1.643e-03 2.482e-01
DTLZ7	$\pm$ 1.659e-03 2.493e-01	$\pm$ 1.190e-02 2.210e-01	$\pm$ 7.756e-03 2.039e-02	$\pm$ 4.838e-03 1.725e-01	$\pm$ 4.395e-03 2.050e-01	$\pm$ 5.405e-02 1.782e-01	$\pm$ 1.240e-02 1.565e-01
wfg1	$\pm$ 4.878e-03 6.332e-01	$\pm$ 3.653e-03 6.318e-01	$\pm$ 2.785e-02 6.132e-01	$\pm$ 5.131e-03 6.289e-01	$\pm$ 1.595e-03 6.304e-01	$\pm$ 6.843e-03 6.203e-01	$\pm$ 8.612e-03 6.175e-01
wfg2	$\pm$ 9.860e-05 5.811e-01	$\pm$ 1.606e-04 5.777e-01	$\pm$ 3.680e-03 5.779e-01	$\pm$ 1.008e-03 5.749e-01	$\pm$ 3.154e-04 5.725e-01	$\pm$ 6.645e-03 5.682e-01	$\pm$ 4.073e-03 5.584e-01
wfg3	$\pm$ 9.623e-05 3.083e-01	$\pm$ 4.106e-04 3.077e-01	$\pm$ 1.256e-03 3.051e-01	$\pm$ 5.746e-04 3.058e-01	$\pm$ 6.026e-04 3.072e-01	$\pm$ 3.480e-03 3.064e-01	$\pm$ 2.772e-03 2.998e-01
wfg4	$\pm$ 1.122e-03 3.087e-01	$\pm$ 2.366e-03 3.076e-01	$\pm$ 2.218e-03 3.043e-01	$\pm$ 1.042e-03 3.064e-01	$\pm$ 1.339e-03 3.073e-01	$\pm$ 7.194e-04 3.061e-01	$\pm$ 2.588e-03 3.005e-01
wfg5	$\pm$ 1.705e-03 3.466e-01	$\pm$ 2.033e-03 3.441e-01	$\pm$ 1.059e-03 3.360e-01	$\pm$ 9.974e-04 3.424e-01	$\pm$ 1.159e-03 3.398e-01	$\pm$ 6.090e-04 3.351e-01	$\pm$ 2.252e-03 3.282e-01
wfg6	$\pm$ 1.349e-04 3.465e-01	$\pm$ 3.557e-04 3.439e-01	$\pm$ 7.228e-03 3.416e-01	$\pm$ 5.716e-04 3.426e-01	$\pm$ 5.995e-04 3.405e-01	$\pm$ 3.497e-03 3.385e-01	$\pm$ 5.078e-03 3.301e-01
wfg7	$\pm$ 1.151e-04 2.395e-01	$\pm$ 4.109e-04 2.377e-01	$\pm$ 1.555e-03 2.372e-01	$\pm$ 4.770e-04 2.240e-01	$\pm$ 4.994e-04 2.399e-01	$\pm$ 1.605e-03 2.300e-01	$\pm$ 3.322e-03 2.432e-01
wfg8	$\pm$ 4.819e-03 3.372e-01	$\pm$ 9.865e-04 3.363e-01	$\pm$ 1.248e-02 3.351e-01	$\pm$ 1.310e-03 3.334e-01	$\pm$ 1.064e-03 3.347e-01	$\pm$ 1.142e-02 3.324e-01	$\pm$ 2.002e-02 3.283e-01
wfg9	$\pm$ 4.948e-04 6.744e-01	$\pm$ 4.708e-04 6.563e-01	$\pm$ 1.698e-03 6.464e-01	$\pm$ 3.828e-04 6.062e-01	$\pm$ 6.221e-04 6.583e-01	$\pm$ 1.116e-03 6.500e-01	$\pm$ 1.743e-03 6.170e-01
UF1	$\pm$ 1.656e-02 7.040e-01	$\pm$ 2.205e-02 7.041e-01	$\pm$ 2.291e-02 6.990e-01	$\pm$ 8.082e-03 6.545e-01	$\pm$ 1.264e-02 6.999e-01	$\pm$ 7.232e-03 6.843e-01	$\pm$ 1.534e-02 6.763e-01
UF2	$\pm$ 4.028e-03 2.730e-01	$\pm$ 2.869e-03 2.105e-01	$\pm$ 5.682e-03 3.985e-01	$\pm$ 7.958e-03 2.607e-01	$\pm$ 2.514e-03 1.896e-01	$\pm$ 4.109e-03 2.692e-01	$\pm$ 5.356e-03 2.444e-01
UF3	$\pm$ 6.757e-02 3.859e-01	$\pm$ 4.290e-02 3.828e-01	$\pm$ 4.621e-02 3.900e-01	$\pm$ 5.424e-02 3.790e-01	$\pm$ 6.822e-02 3.783e-01	$\pm$ 8.321e-02 3.714e-01	$\pm$ 5.495e-02 3.639e-01
UF4	$\pm$ 2.888e-03 3.206e-01	$\pm$ 3.832e-03 3.031e-01	$\pm$ 2.591e-03 2.960e-01	$\pm$ 2.308e-03 6.161e-02	$\pm$ 2.387e-03 5.540e-02	$\pm$ 6.149e-03 4.473e-02	$\pm$ 4.227e-03 1.061e-02
UF5	$\pm$ 5.720e-02 1.901e-01	$\pm$ 7.431e-02 2.120e-01	$\pm$ 3.463e-02 2.727e-01	$\pm$ 6.411e-02 3.812e-03	$\pm$ 5.470e-02 6.856e-02	$\pm$ 7.180e-02 5.870e-02	$\pm$ 2.716e-02 3.288e-04
UF6	$\pm$ 9.322e-02 5.506e-01	$\pm$ 1.017e-01 5.463e-01	$\pm$ 4.301e-02 4.176e-01	$\pm$ 1.184e-02 5.086e-01	$\pm$ 4.882e-02 5.536e-01	$\pm$ 6.874e-02 5.005e-01	$\pm$ 1.203e-03 4.848e-01
UF7	$\pm$ 7.136e-03 4.374e-01	$\pm$ 5.771e-03 3.384e-01	$\pm$ 7.436e-02 3.984e-01	$\pm$ 6.749e-03 2.471e-01	$\pm$ 6.117e-03 2.529e-01	$\pm$ 5.855e-02 2.599e-01	$\pm$ 4.375e-02 3.580e-01
UF8	$\pm$ 1.163e-02 6.765e-01	$\pm$ 8.782e-02 6.330e-01	$\pm$ 3.378e-02 6.096e-01	$\pm$ 3.745e-02 4.012e-01	$\pm$ 9.765e-02 4.350e-01	$\pm$ 5.688e-02 4.028e-01	$\pm$ 3.739e-02 5.611e-01
UF9	$\pm$ 4.374e-02 2.002e-01	$\pm$ 7.215e-02 1.219e-01	$\pm$ 5.701e-02 1.415e-01	$\pm$ 5.248e-02 0	$\pm$ 1.772e-01 8.094e-02	$\pm$ 8.137e-02 3.234e-03	$\pm$ 3.971e-02 1.162e-01
UF10	$\pm$ 6.751e-02	$\pm$ 4.608e-02	$\pm$ 2.853e-02	$\pm$ 0	$\pm$ 3.339e-02	$\pm$ 9.266e-03	$\pm$ 6.310e-02

Table 8 Summary of Wilcoxon rank-sum test results (better/equal/worse)

Metric	MOCCO	NSGA-II	MOPSO	MOSCSO	MOJS	MOAVOA
IGD	26/4/1	26/1/4	29/2/0	27/3/1	28/2/1	28/3/0
GD	21/7/3	16/3/12	29/2/0	23/1/7	28/2/1	26/2/3
Spacing	24/4/3	22/2/7	27/4/0	23/6/2	27/3/1	27/3/1
HV	25/3/3	27/1/3	28/3/0	25/2/4	28/3/0	26/5/0

Table 9 Friedman mean ranks

Metric	IMOCCO	MOCCO	NSGA2	MOPSO	MOSCSO	MOJS	MOAVOA
IGD	1.42	2.87	4.39	4.9	4.1	5.13	5.19
GD	2.1	3	3.23	5.45	3.71	5.03	5.48
Spacing	1.77	3.1	3.84	4.71	4.06	4.55	5.97
HV	1.65	2.84	4.45	4.81	3.84	5.19	5.23
Mean rank	1.73	2.95	3.98	4.97	3.93	4.98	5.47

**Fig. 1** Heat map of Friedman mean ranks on the four metrics and overall

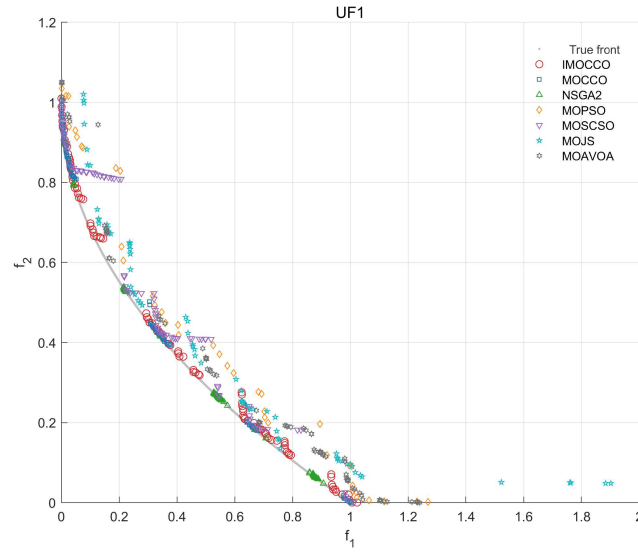


Fig. 2 Comparison of Pareto fronts on the UF1 problem

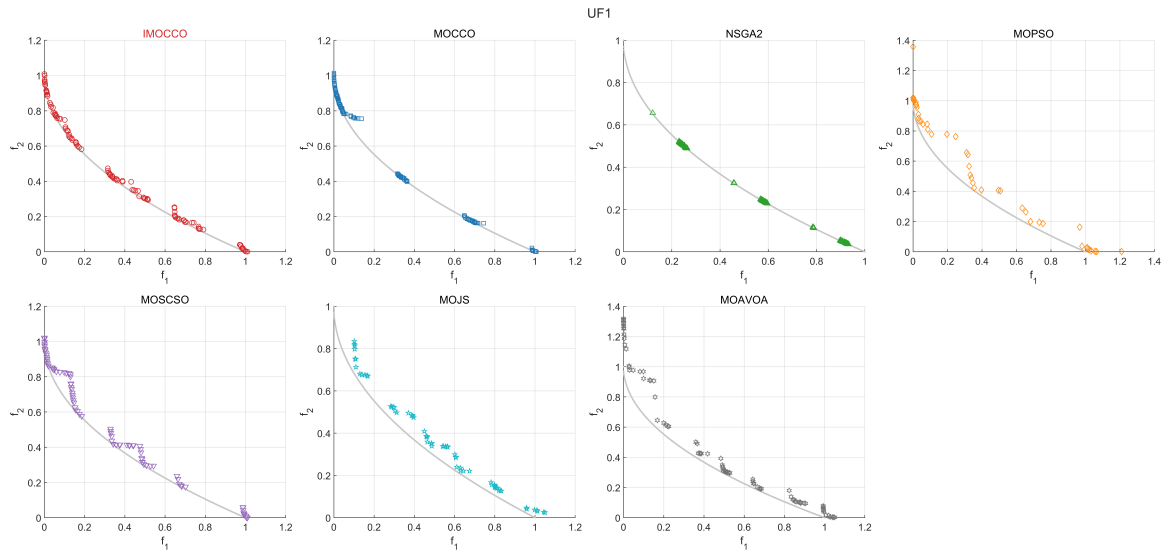


Fig. 3 Approximate Pareto fronts of all algorithms on UF1

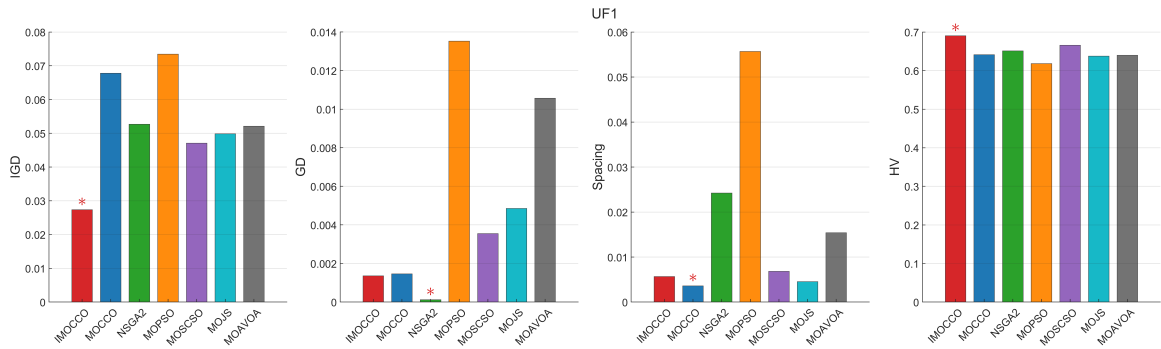


Fig. 4 Comparison of four metrics on UF1

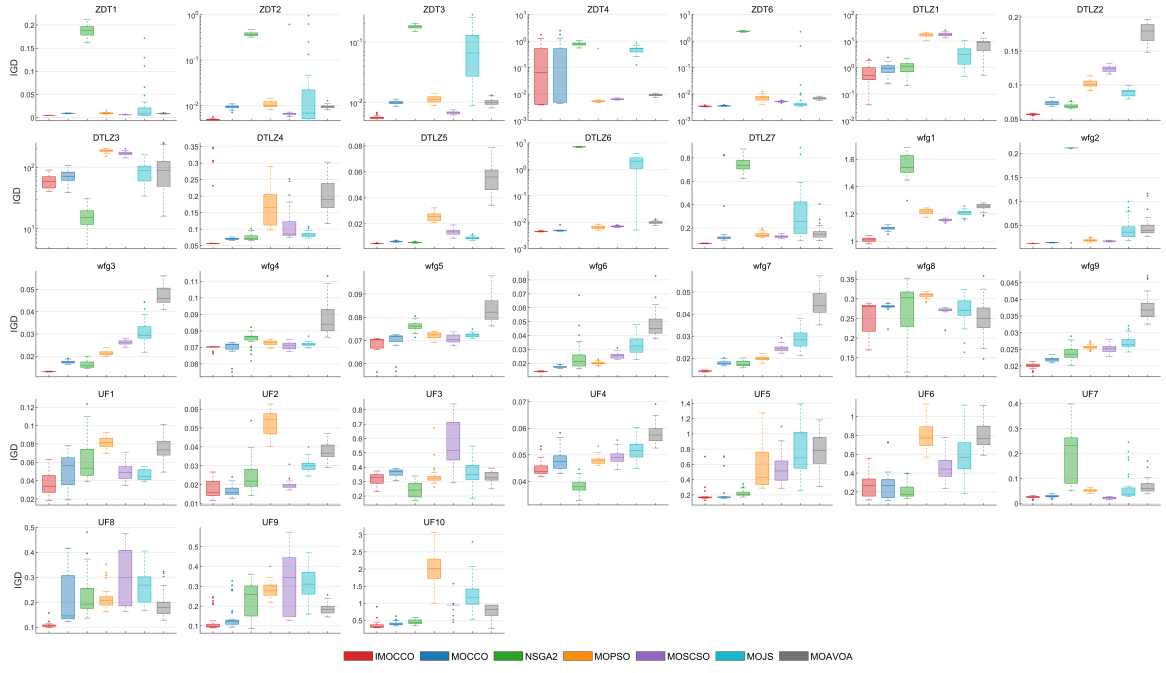


Fig. 5 Box plots of IGD values on the 31 test functions

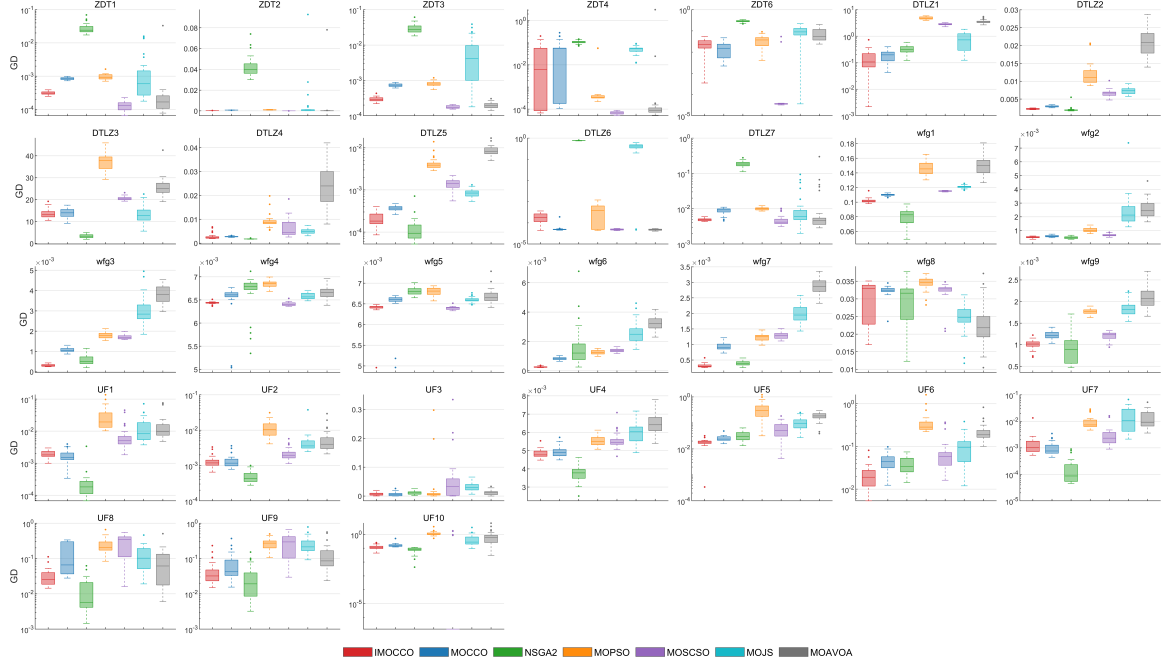


Fig. 6 Box plots of GD values on the 31 test functions

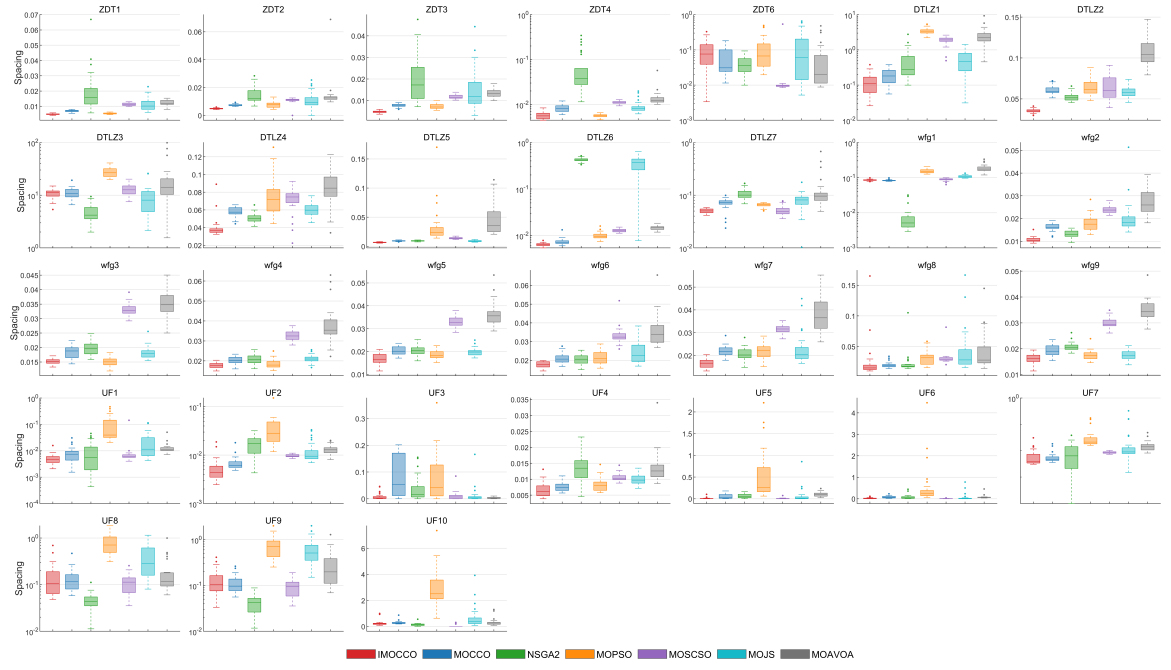


Fig. 7 Box plots of Spacing values on the 31 test functions

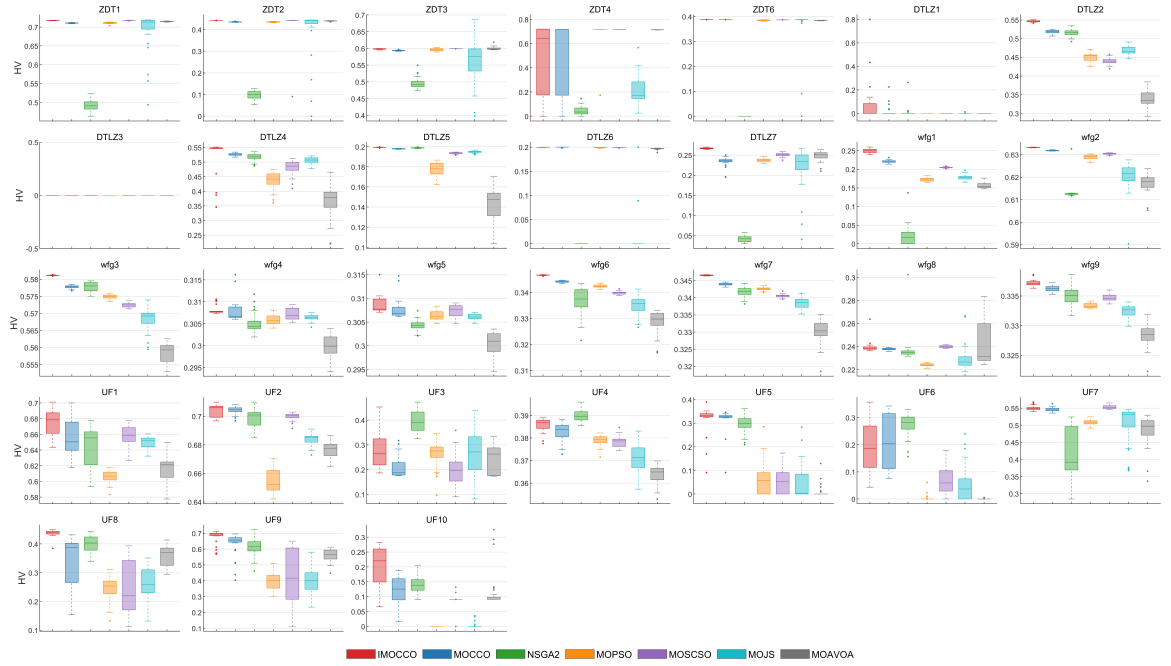


Fig. 8 Box plots of HV values on the 31 test functions

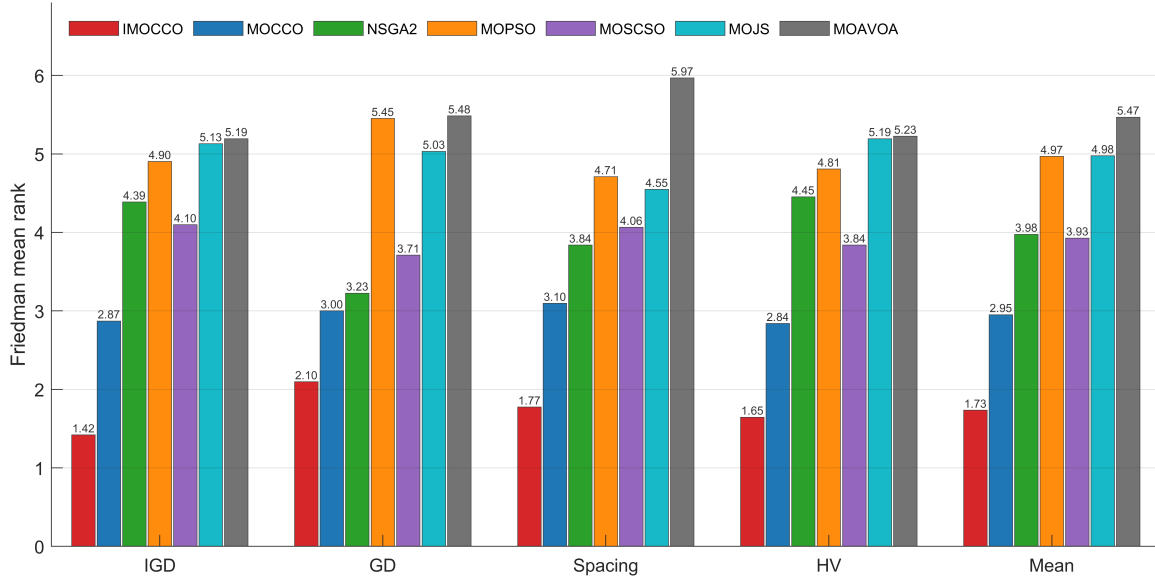


Fig. 9 Friedman mean ranks on the four metrics and overall

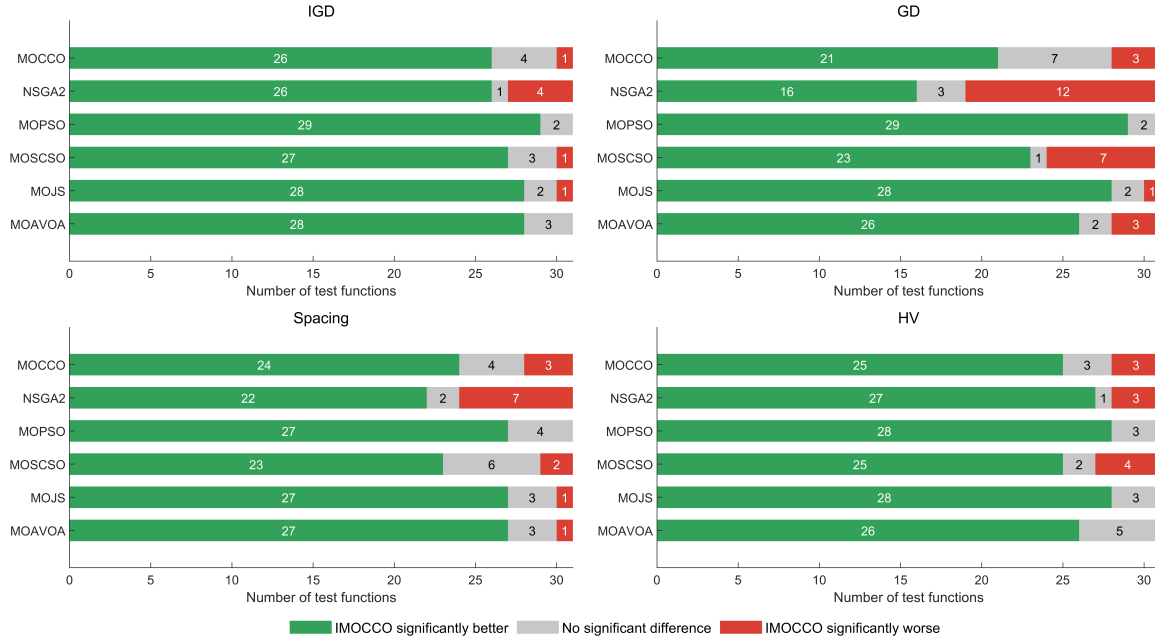


Fig. 10 Better/equal/worse counts of the Wilcoxon test on each metric

counts. In Fig. 9, IMOCCO has the lowest Friedman rank on all four metrics and overall, with an overall rank of 1; Fig. 10 complements this result by showing the problem-wise balance of significant wins, ties, and losses against each comparator.

4.3 Ablation Study

To verify the independent contribution of each improvement, five variants are configured: IMOCCO (full), w/o S1 (manifold-feature collision off), w/o S2 (mass-restitution collision off), w/o S3 (angular-sparsity guidance off), and "all off" (degraded to MOCCO). All variants run under the same settings.

Table 10 Friedman mean ranks of the ablation study

Metric	IMOCCO	w/o S1	w/o S2	w/o S3	All off
IGD	2.16	2.42	2.16	3.94	4.32
GD	2.10	2.94	2.55	3.10	4.32
Spacing	2.16	2.55	2.19	4.16	3.94
HV	2.26	2.55	2.23	3.71	4.26
Mean rank	2.17	2.61	2.28	3.73	4.21

The results show that the full IMOCCO has an overall mean rank of 2.17. Removing S3 increases the mean rank to 3.73, a larger deterioration than removing S1 (2.61) or S2 (2.28), which identifies angular-sparsity guidance as the most influential component under the present benchmark settings. The "all off" variant has the highest mean rank (4.21), supporting the interpretation that the three mechanisms provide complementary contributions rather than a gain attributable to only one modification.

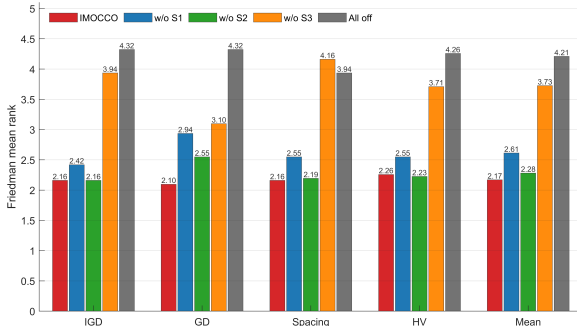


Fig. 11 Friedman mean ranks of the ablation study

4.4 Engineering Applications

To verify IMOCCO on real constrained engineering problems, two classic multi-objective engineering problems are selected: a four-bar truss design and a disc-brake design. Both use a population size of 100, archive size of 100, 300 iterations, and 30 independent runs. Constraint violations are handled by a penalty function.

4.4.1 Four-bar Truss Design

The four-bar truss takes the cross-sectional areas x_1, x_2, x_3, x_4 of the four bars as design variables and simultaneously minimizes the structural volume f_1 and the node displacement f_2 , with box constraints only[34–37]. The load is $F = 10$ kN, the elastic modulus $E = 2 \times 10^5$ kN/cm², and the bar length $L = 200$ cm. The objectives are:

$$f_1 = L (2x_1 + \sqrt{2}x_2 + \sqrt{x_3} + x_4) \quad (31)$$

$$f_2 = \frac{FL}{E} \left(\frac{2}{x_1} + \frac{2\sqrt{2}}{x_2} - \frac{2\sqrt{2}}{x_3} + \frac{2}{x_4} \right) \quad (32)$$

The lower bounds are $[a, \sqrt{2}a, \sqrt{2}a, a]$ and the upper bounds are $[3a, 3a, 3a, 3a]$. Larger cross sections increase the structural volume but reduce the node displacement, so the two objectives conflict.

4.4.2 Disc Brake Design

The disc-brake problem takes the inner radius r , outer radius R , braking force F , and number of friction surfaces s as design variables, simultaneously minimizing the disc mass f_1 and the braking time f_2 . The variable ranges are $r \in [55, 80]$, $R \in [75, 110]$, $F \in [1000, 3000]$, and $s \in [2, 10]$. The objectives are:

$$f_1 = 4.9 \times 10^{-5} (R^2 - r^2)(s - 1) \quad (33)$$

$$f_2 = \frac{9.82 \times 10^6 (R^2 - r^2)}{Fs(R^3 - r^3)} \quad (34)$$

The problem includes five constraints on minimum radial thickness, maximum thickness, maximum pressure, maximum temperature rise, and minimum braking torque:

$$\begin{aligned}
g_1 &= 20 - (R - r); \\
g_2 &= 2.5(s + 1) - 30; \\
g_3 &= \frac{F}{3.14(R^2 - r^2)} - 0.4
\end{aligned} \tag{35}$$

$$\begin{aligned}
g_4 &= \frac{2.22 \times 10^{-3} F(R^3 - r^3)}{(R^2 - r^2)^2} - 1; \\
g_5 &= 900 - \frac{0.0266 F s(R^3 - r^3)}{R^2 - r^2}
\end{aligned} \tag{36}$$

The sum of constraint violations is multiplied by a penalty factor of 10^6 and added to each objective.

4.4.3 Engineering Results and Discussion

As Tables 11–12 and Figs. 12–13 show, IMOCCO achieves the best IGD, GD, Spacing, and HV on both engineering problems: on the four-bar truss, $IGD = 0.003742$, $GD = 8.82 \times 10^{-5}$, $Spacing = 0.005120$, and $HV = 0.73004$; on the disc brake, $IGD = 0.004343$, $GD = 1.017 \times 10^{-4}$, $Spacing = 0.004135$, and $HV = 0.89834$. The Wilcoxon test is 2/0/0 on both problems. These benchmark results provide an application-level check that the gains observed on synthetic test functions are retained under constrained engineering design formulations.

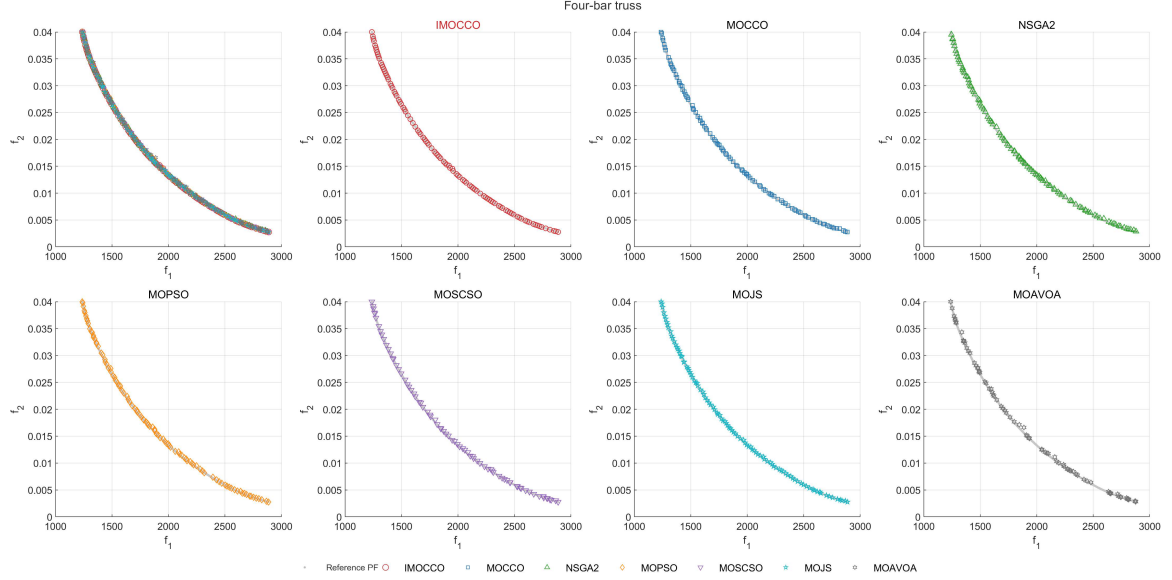
Fig. 14 shows that on both engineering problems IMOCCO has the lowest mean IGD, GD, and Spacing, the highest mean HV, and short and narrow error bars, indicating concentrated and stable results. MOCCO remains the strongest baseline on several of these comparisons, while IMOCCO's further improvement supports the added value of the three proposed mechanisms.

5 Conclusions

This paper proposes IMOCCO, a novel multi-objective optimizer built on the CCO strategy. Compared with existing multi-objective algorithms, IMOCCO has the following features. First, it introduces manifold-feature collision, mass-restitution collision, and angular-sparsity guidance on top of CCO and the MOCCO framework, acting respectively on the search-operator, individual-interaction, and environmental-selection layers. Second, across 31 standard test functions, IMOCCO attains the best overall Friedman mean rank (1.73) and shows predominantly favorable Wilcoxon better/equal/worse counts against the six comparators. Third, the ablation study confirms that all three improvements are effective, with angular-sparsity guidance being the most influential. Fourth, on the four-bar truss and disc-brake problems, IMOCCO achieves the best IGD, GD, Spacing, and HV with a Wilcoxon result of 2/0/0, supporting its performance on two constrained engineering design benchmarks. The current implementation fixes several mechanism-specific coefficients in addition to the population and archive sizes; systematic sensitivity analysis of these coefficients remains an important direction for future work. Future work will extend IMOCCO to larger three-dimensional trusses, combine it with surrogate models to reduce computational cost, and apply it to more practical engineering structures.

Table 11 Metrics of all algorithms on the four-bar truss (mean $\pm$ std)

Metric	IMOCCO	MOCCO	NSGA2	MOPSO	MOSCSO	MOJS	MOAVOA
IGD	3.74e-03 $\pm$ 8.73e-05	5.14e-03 $\pm$ 2.06e-04	5.63e-03 $\pm$ 3.27e-04	5.31e-03 $\pm$ 1.99e-04	6.60e-03 $\pm$ 2.68e-04	5.06e-03 $\pm$ 1.83e-04	1.15e-02 $\pm$ 2.28e-03
GD	8.82e-05 $\pm$ 2.40e-05	2.10e-04 $\pm$ 2.42e-05	2.84e-04 $\pm$ 3.72e-05	3.22e-04 $\pm$ 3.25e-05	2.37e-04 $\pm$ 2.86e-05	2.82e-04 $\pm$ 3.28e-05	5.20e-04 $\pm$ 1.55e-04
Spacing	5.12e-03 $\pm$ 4.31e-04	6.33e-03 $\pm$ 8.16e-04	5.88e-03 $\pm$ 7.38e-04	5.40e-03 $\pm$ 6.29e-04	1.09e-02 $\pm$ 6.82e-04	5.76e-03 $\pm$ 6.32e-04	1.25e-02 $\pm$ 1.49e-03
HV	7.30e-01 $\pm$ 7.54e-05	7.28e-01 $\pm$ 3.03e-04	7.27e-01 $\pm$ 6.05e-04	7.28e-01 $\pm$ 3.06e-04	7.26e-01 $\pm$ 3.99e-04	7.28e-01 $\pm$ 3.02e-04	7.21e-01 $\pm$ 1.69e-03

**Fig. 12** Comparison of Pareto fronts on the four-bar truss**Table 12** Metrics of all algorithms on the disc brake (mean $\pm$ std)

Metric	IMOCCO	MOCCO	NSGA2	MOPSO	MOSCSO	MOJS	MOAVOA
IGD	4.34e-03 $\pm$ 1.01e-04	5.31e-03 $\pm$ 1.36e-04	4.62e-02 $\pm$ 1.53e-02	6.38e-03 $\pm$ 6.31e-04	6.81e-03 $\pm$ 3.02e-04	9.20e-03 $\pm$ 2.58e-03	1.74e-02 $\pm$ 4.75e-03
GD	1.02e-04 $\pm$ 2.33e-05	1.84e-04 $\pm$ 3.13e-05	2.32e-03 $\pm$ 5.36e-03	2.71e-04 $\pm$ 3.33e-05	2.89e-04 $\pm$ 4.51e-05	4.74e-04 $\pm$ 2.58e-04	5.79e-04 $\pm$ 1.58e-04
Spacing	4.14e-03 $\pm$ 3.25e-04	6.96e-03 $\pm$ 5.70e-04	6.99e-03 $\pm$ 9.25e-03	6.21e-03 $\pm$ 9.55e-04	1.03e-02 $\pm$ 7.03e-04	8.13e-03 $\pm$ 4.65e-03	1.25e-02 $\pm$ 3.55e-03
HV	8.98e-01 $\pm$ 1.68e-04	8.97e-01 $\pm$ 2.50e-04	8.81e-01 $\pm$ 8.84e-03	8.97e-01 $\pm$ 3.09e-04	8.96e-01 $\pm$ 3.81e-04	8.95e-01 $\pm$ 1.16e-03	8.90e-01 $\pm$ 1.14e-03

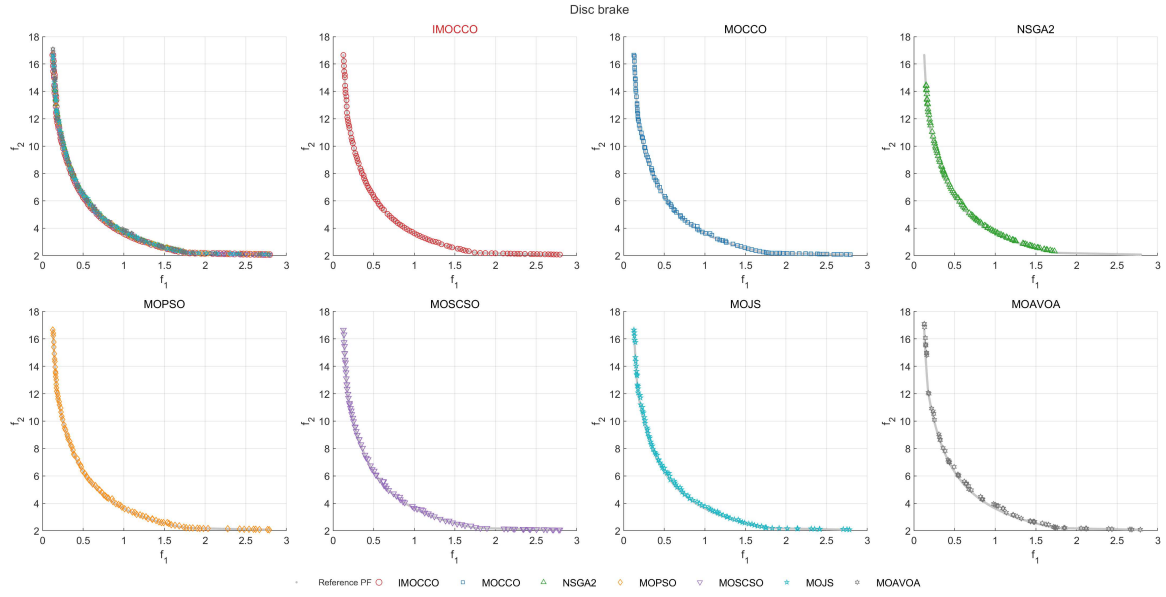


Fig. 13 Comparison of Pareto fronts on the disc brake

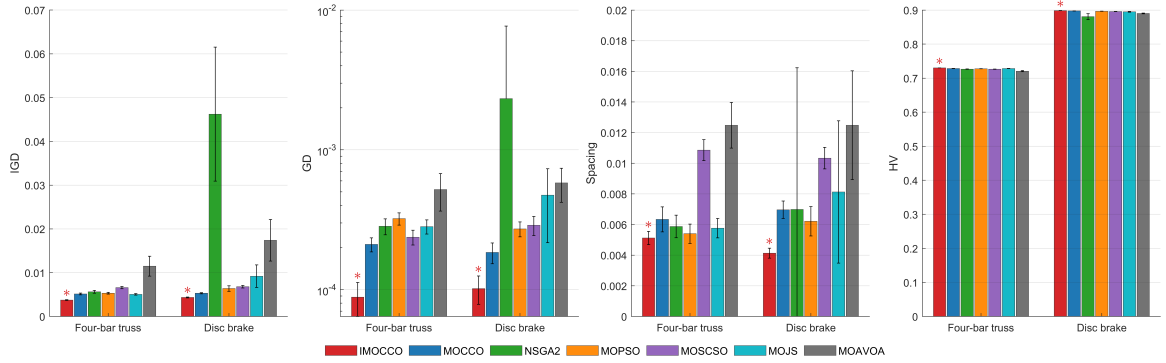


Fig. 14 Comparison of four metrics on the two engineering problems

References

- [1] Marler, R.T., Arora, J.S.: Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization* **26**(6), 369–395 (2004) <https://doi.org/10.1007/s00158-003-0368-6>
- [2] Miettinen, K.: *Nonlinear Multiobjective Optimization*. Springer, New York (1999). <https://doi.org/10.1007/978-1-4615-5563-6>
- [3] Das, I., Dennis, J.E.: A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation. *Structural Optimization* **14**(1), 63–69 (1997) <https://doi.org/10.1007/BF01197559>
- [4] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* **6**(2), 182–197 (2002) <https://doi.org/10.1109/4235.996017>
- [5] Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation* **11**(6), 712–731 (2007) <https://doi.org/10.1109/TEVC.2007.892759>
- [6] Coello Coello, C.A., Pulido, G.T., Lechuga, M.S.: Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation* **8**(3), 256–279 (2004) <https://doi.org/10.1109/TEVC.2004.826067>
- [7] Tripathi, P.K., Bandyopadhyay, S., Pal, S.K.:

- Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients. *Information Sciences* **177**(22), 5033–5049 (2007) <https://doi.org/10.1016/j.ins.2007.06.018>
- [8] Lang, Y., Gao, Y., Chen, T., Wang, H.: Centered collision optimizer: A novel and efficient physics-based metaheuristic optimization algorithm for solving complex real-world engineering optimization problems. *Computer Methods in Applied Mechanics and Engineering* **448**, 118491 (2026) <https://doi.org/10.1016/j.cma.2025.118491>
- [9] Kaveh, A., Mahdavi, V.R.: Colliding bodies optimization: A novel meta-heuristic method. *Computers & Structures* **139**, 18–27 (2014) <https://doi.org/10.1016/j.compstruc.2014.04.005>
- [10] Zhou, A., Qu, B.-Y., Li, H., Zhao, S.-Z., Suganthan, P.N., Zhang, Q.: Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation* **1**(1), 32–49 (2011) <https://doi.org/10.1016/j.swevo.2011.03.001>
- [11] Reyes-Sierra, M., Coello Coello, C.A.: Multi-objective particle swarm optimizers: A survey of the state-of-the-art. *International Journal of Computational Intelligence Research* **2**(3), 287–308 (2006) <https://doi.org/10.5019/j.ijcir.2006.68>
- [12] Talbi, E.-G.: *Metaheuristics: From Design to Implementation*. Wiley, Hoboken (2009). <https://doi.org/10.1002/9780470496916>
- [13] Eiben, A.E., Smith, J.E.: *Introduction to Evolutionary Computing*, 2nd edn. Springer, Berlin (2015). <https://doi.org/10.1007/978-3-662-44874-8>
- [14] Konak, A., Coit, D.W., Smith, A.E.: Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety* **91**(9), 992–1007 (2006) <https://doi.org/10.1016/j.res.2005.11.018>
- [15] Coello Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A.: *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd edn. Springer, New York (2007). <https://doi.org/10.1007/978-0-387-36797-2>
- [16] Fonseca, C.M., Fleming, P.J.: Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In: *Proceedings of the Fifth International Conference on Genetic Algorithms*, pp. 416–423. Morgan Kaufmann, San Mateo (1993)
- [17] Van Veldhuizen, D.A.: *Multiobjective evolutionary algorithms: Classifications, analyses, and new innovations*. PhD thesis, Air Force Institute of Technology, Dayton (1999)
- [18] Kim, M., Seo, S., Choi, S., Kim, Y.J., Kim, Y., Kim, T.-H.: Model order reduction using dual ν -gap metrics: A multi-objective optimization approach. *Electronic Research Archive* **34**(1), 433–462 (2026) <https://doi.org/10.3934/era.2026021>
- [19] Hauschild, M., Pelikan, M.: An introduction and survey of estimation of distribution algorithms. *Swarm and Evolutionary Computation* **1**(3), 111–128 (2011) <https://doi.org/10.1016/j.swevo.2011.08.003>
- [20] Sierra, M.R., Coello Coello, C.A.: Improving PSO-based multi-objective optimization using crowding, mutation and epsilon-dominance. In: *Evolutionary Multi-Criterion Optimization. Lecture Notes in Computer Science*, vol. 3410, pp. 505–519. Springer, Berlin (2005). https://doi.org/10.1007/978-3-540-31880-4_35
- [21] Van Veldhuizen, D.A., Lamont, G.B.: Multi-objective evolutionary algorithms: Analyzing the state-of-the-art. *Evolutionary Computation* **8**(2), 125–147 (2000) <https://doi.org/10.1162/106365600568158>
- [22] Schott, J.R.: *Fault tolerant design using single and multicriteria genetic algorithm optimization*. Master’s thesis, Massachusetts Institute of Technology, Cambridge (1995)
- [23] Zitzler, E.: *Evolutionary algorithms for multiobjective optimization: Methods and applications*. PhD thesis, ETH Zurich, Zurich (1999)
- [24] While, L., Hingston, P., Barone, L., Huband, S.: A faster algorithm for calculating hypervolume. *IEEE Transactions on Evolutionary Computation* **10**(1), 29–38 (2006) <https://doi.org/10.1109/TEVC.2005.851275>

- [25] Seyyedabbasi, A., Kiani, F.: Sand cat swarm optimization: A nature-inspired algorithm to solve global optimization problems. *Engineering with Computers* **39**, 2627–2651 (2023) <https://doi.org/10.1007/s00366-022-01604-x>
- [26] Chou, J.-S., Truong, D.-N.: Multiobjective optimization inspired by behavior of jellyfish for solving structural design problems. *Chaos, Solitons & Fractals* **135**, 109738 (2020) <https://doi.org/10.1016/j.chaos.2020.109738>
- [27] Khodadadi, N., Gharehchopogh, F.S., Mirjalili, S.: MOAVOA: A new multi-objective artificial vultures optimization algorithm. *Neural Computing and Applications* **34**(24), 20791–20829 (2022) <https://doi.org/10.1007/s00521-022-07557-y>
- [28] Wilcoxon, F.: Individual comparisons by ranking methods. *Biometrics Bulletin* **1**(6), 80–83 (1945) <https://doi.org/10.2307/3001968>
- [29] Friedman, M.: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association* **32**(200), 675–701 (1937) <https://doi.org/10.1080/01621459.1937.10503522>
- [30] Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation* **8**(2), 173–195 (2000) <https://doi.org/10.1162/106365600568202>
- [31] Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable test problems for evolutionary multiobjective optimization. In: *Evolutionary Multiobjective Optimization*, pp. 105–145. Springer, London (2005). https://doi.org/10.1007/1-84628-137-7_6
- [32] Huband, S., Hingston, P., Barone, L., While, L.: A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation* **10**(5), 477–506 (2006) <https://doi.org/10.1109/TEVC.2005.861417>
- [33] Zhang, Q., Zhou, A., Zhao, S., Suganthan, P.N., Liu, W., Tiwari, S.: Multiobjective optimization test instances for the CEC 2009 special session and competition. Technical Report CES-487, University of Essex, Colchester (2008)
- [34] Ray, T., Liew, K.M.: A swarm metaphor for multiobjective design optimization. *Engineering Optimization* **34**(2), 141–153 (2002) <https://doi.org/10.1080/03052150210915>
- [35] Coello Coello, C.A., Christiansen, A.D.: Multiobjective optimization of trusses using genetic algorithms. *Computers & Structures* **75**(6), 647–660 (2000) [https://doi.org/10.1016/S0045-7949\(99\)00110-8](https://doi.org/10.1016/S0045-7949(99)00110-8)
- [36] Luh, G.-C., Chueh, C.-H.: Multi-objective optimal design of truss structure with immune algorithm. *Computers & Structures* **82**(11–12), 829–844 (2004) <https://doi.org/10.1016/j.compstruc.2004.03.003>
- [37] Perez, R.E., Behdinan, K.: Particle swarm approach for structural design optimization. *Computers & Structures* **85**(19–20), 1579–1588 (2007) <https://doi.org/10.1016/j.compstruc.2006.10.013>

Statements and Declarations

Funding. The authors received no funding for this work.

Competing interests. The authors declare that they have no competing interests.

Data and code availability. The data and code supporting this study are available from the corresponding author on reasonable request.



Yanyin Sun is currently pursuing a bachelor's degree in electrical engineering and automation at Nanchang University. Research interests include multi-objective optimization, metaheuristic algorithms, and engineering design optimization.



Yuqi Chen is currently pursuing a bachelor's degree in artificial intelligence at Nanchang University. Research interests include swarm intelligence, evolutionary computation, and multi-objective optimization.