An Improved Pareto Front Modeling Algorithm for Large-scale Many-Objective Optimization

Panichella, A.

DOI
10.1145/3512290.3528732

Publication date
2022

Document Version
Final published version

Published in
The Genetic and Evolutionary Computation Conference

An Improved Pareto Front Modeling Algorithm for Large-scale Many-Objective Optimization

Annibale Panichella
Delft University of Technology
Delft, The Netherlands
a.panichella@tudelft.nl

ABSTRACT

A key idea in many-objective optimization is to approximate the optimal Pareto front using a set of representative non-dominated solutions. The produced solution set should be close to the optimal front (convergence) and well-diversified (diversity). Recent studies have shown that measuring both convergence and diversity depends on the shape (or curvature) of the Pareto front. In recent years, researchers have proposed evolutionary algorithms that model the shape of the non-dominated front to define environmental selection strategies that adapt to the underlying geometry. This paper proposes a novel method for non-dominated front modeling using the Newton-Raphson iterative method for roots finding. Second, we compute the distance (diversity) between each pair of non-dominated solutions using geodesics, which are generalizations of the distance on Riemann manifolds (curved topological spaces). We have introduced an evolutionary algorithm within the Adaptive Geometry Estimation based MOEA (AGE-MOEA) framework, which we called AGE-MOEA-II. Computational experiments with 17 problems from the WFG and SMOP benchmarks show that AGE-MOEA-II outperforms its predecessor AGE-MOEA as well as other state-of-the-art many-objective algorithms, i.e., NSGA-III, MOEA/D, VaEA, and LMEA.

CCS CONCEPTS

• **Theory of computation → Evolutionary algorithms**.

KEYWORDS

Many-objective optimization, Geodesics, Newton-Raphson method

ACM Reference Format:
Annibale Panichella. 2022. An Improved Pareto Front Modeling Algorithm for Large-scale Many-Objective Optimization. In *Genetic and Evolutionary Computation Conference (GECCO '22), July 9–13, 2022, Boston, MA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3512290.3528732

1 INTRODUCTION

Many engineering and real-world problems are inherently multi-objective as they require addressing multiple objectives [4, 16, 24]. However, there is no single solution that can satisfy all objectives if they are conflicting. In these scenarios, there are multiple Pareto optimal trade-offs between the various objectives, which are critical for the decision-makers when choosing the solution that best fits their constraints. Thus, finding these Pareto optimal trade-offs is the primary goal of multi- and many-objective optimization algorithms.

Many-objective Evolutionary algorithms (MOEAs) have been successfully used in literature to solve multi and many-objective optimization problems (MOPs) with conflicting objectives. Their goal is twofold: (1) produce a set of trade-offs that is as close to the true Pareto front as possible (*convergence*); and (2) obtain well-distributed trade-offs (*diversity*) for a more complete and informed decision-making process. The success of MOEAs strongly depends on their ability to balance convergence and diversity.

Over the last two decades, several MOEAs have been proposed in the literature. They differ in the main mechanisms used to select new solutions for the next generations (*environmental selection*) such that both convergence and diversity are promoted. For example, Pareto-based MOEAs like `NSGA-II` [14] divide candidate solutions into several non-dominance fronts (for convergence) and then use a distance function (e.g., *crowding* distance [14]) to promote more diverse solutions within the same convergence level (front). Reference-points based MOEAs (e.g., `NSGA-III` [13]) use a set of well-distributed reference points (diversity) and select the best solution for each of the reference points (convergence). Finally, indicator-based MOEAs (e.g., IBEA [41]) rely on quality indicators, such as the hypervolume (HV), that measure both solutions diversity and convergence. Other widely-used quality indicators include the Averaged Hausdorff distance (Δ_p) [32], the inverted generational distance plus (IGD^+) [20], and R2 [7].

As highlighted in recent studies [25, 29, 36], these well-known MOEAs use fixed diversity and convergence methods that achieve different performances depending on the shape/geometry of the Pareto front to approximate. For example, Das and Dennis's systematic approach [11] used in `NSGA-III` and `MOEA/D` places the reference points on a normalized hyper-plane, which is optimal for flat Pareto fronts. The angular-based distance used in `VaEA` [23] and `LMEA` [40] is instead effective on spherical fronts since the distance on a sphere can be better measured by angles.

Hence, recent research has been devoted to designing adaptive environmental selection strategies depending on the geometry of the Pareto front. To this aim, fitting methods are used to model the set of non-dominated solutions produced during the evolutionary process. In particular, Martinez et al. [25], and Tian et al. [36] have applied non-linear fitting methods that can effectively model the shape formed by the non-dominated solution set. However, non-linear fitting methods like the Levenberg-Marquardt (LM) algorithm [36] incur a substantial overhead, which increases dramatically with the population size and the number of objectives.

GECCO '22, July 9–13, 2022, Boston, MA, USA

ACM ISBN 978-1-4503-9237-2/22/07.
https://doi.org/10.1145/3512290.3528732

Instead, AGE-MOEA [29] is a recently proposed MOEA that uses simple heuristics to model the shape of the Pareto front. In particular, it uses one single non-dominated point (the *central point*), as a reference throughout the search. Although simple, the single-point-based heuristic has been shown to be effective while incurring a negligible computational cost.

In this paper, we observe two areas of improvement in the overall AGE-MOEA framework. First, the accuracy of the front-modeling heuristic used in [29] strongly depends on the single reference point being used. This can lead to non-accurate modeling of the non-dominated front with the corresponding inaccurate computation of converge and diversity. Second, the distances used to compute the diversity among non-dominated solutions are generalizations of the Euclidean distance (p-norm) that do not capture the front's topological structure (i.e., the curvature).

This paper introduces two novel strategies (1) to model the non-dominated front and (2) to measure the distance among non-dominated solutions. In particular, we use the Newton-Raphson iterative method for root finding [38], which is much more robust and less sensitive to the choice of the non-dominated point used as reference (as empirically shown in Section 2.2). Second, we use *geodesic* [27] to measure the distance between non-dominated solutions on a curved manifold. In differential geometry, a geodesic is the shortest path between two points on a curved manifold, e.g., the great circle on a sphere. Hence, we rely on these "curved" paths on the front models to better measure solution diversity.

We have incorporated the Newton-Raphson iterative method and the geodesic distance into the AGE-MOEA framework. We named the resulting evolutionary algorithm AGE-MOEA-II. To assess the performance of the proposed MOEA, we conducted an empirical study with 17 test problems from two benchmarks, namely WFG [18] and SMOP [37]. These benchmarks have been widely used in literature since they contain problems with different search challenges and Pareto front shapes. Our study considers these test problems with a different number of objective functions M=3, 6, and 9. Experimental results indicate that AGE-MOEA-II achieves higher hypervolume (HV) values than its predecessor AGE-MOEA and outperforms other state-of-the-art MOEAs, namely NSGA-III, MOEA/D, LMEA, and VaEA.

The remainder of the paper is organized as follows. Section 2 introduces basic concepts on Pareto-front modeling techniques and many-objective problems. Section 3 introduces the novel front-modeling and techniques as well as the geodesic distance. Section 4 details the empirical study we have carried out to assess the performance of AGE-MOEA-II. Finally, Section 5 concludes the paper.

2 PRELIMINARIES

Multi-objective optimization problems are very common in real-world engineering problems [24]. Without loss of generality, a multi-objective optimization problem can be defined as follows:

$$\begin{aligned} & minF(x) = [f_1(x), f_2(x), \ldots, f_M(x)]^T \\ & \text{subject to } x \in \Omega \subseteq \mathbb{R}^k \end{aligned} \tag{1}$$

where $x = (x_1, \ldots, x_k)^T$ is a k-dimensional solution (or decision) vector from the feasible region $\Omega \subset \mathbb{R}^k$; $f : \Omega \rightarrow \mathbb{R}^M$ is an objective vector with M conflicting objective functions (or simply objectives).

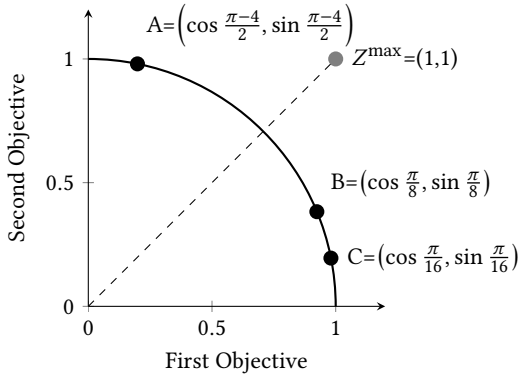


Figure 1: Example of a non-dominated front with p=2 shape and two objectives. Z^{max} denotes the nadir point.

Multi-objective problems with more than three objectives ($M > 3$) are often referred to as *many-objective* in the literature.

A solution x *dominates* another solution y ($x \prec y$), if $f_i(x) \leq f_i(y)\ \forall f_i \in F$ and there exists at least one objective $f_j \in F$ such that $f_j(x) < f_j(y)$. A *Pareto optimal* solution $x^* \in \Omega$ is a solution that is non-dominated by any other solution in Ω, i.e., $\nexists\ y \in \Omega$ such that $y \prec x^*$. The set of all Pareto optimal solutions is called *Pareto optimal solution set* (PS), while the corresponding objective vectors form the so-called *Pareto front* (PF).

2.1 Pareto front modelling

In recent years, researchers have proposed different front modeling techniques with the goal of defining diversity measures that are specific or adaptive w.r.t. the shape (curvature) of the Pareto front. The modeling is done using the following family of manifolds:

$$\left(a_1^p + \cdots + a_M^p\right)^{1/p} = 1 \tag{2}$$

where a_i are the coordinates (objective values) of the non-dominated solution A, and p is the parameter that control the front's curvature. Equation 2 is also referred to as $||\cdot||_p$ norm or simply L_p space. L_p is a generalization of the Euclidean (L_2) distance [34] and defines the so-called unit circle, as shown in Figure 1. In particular, points A, B, and C are all at the same Euclidean distance from the origin of the axes (O) that is equal to one (i.e., $d(A, O) = ||A - O||_2 = 1$). Hence, L_2 accurately models the Pareto front that forms a (hyper) sphere. If the Pareto front forms a flat hyper-plane, the best fitting norm is $p = 1$. Instead, for fronts that correspond to hyperbolic manifolds, the corresponding norm has $p < 1$ values. Hence, modeling the Pareto front requires finding the value of p that minimizes the fitting error, i.e., the difference between the actual front and the curvature obtained with the L_p norm. To this aim, multiple numerical methods can be used to solve this non-linear fitting problem.

GFM-MOEA by Tian et al. [36] uses a generalized version of the L_p family of manifolds:

$$\begin{cases} \sqrt[p]{c_1 \cdot a_{1,1}^p + \cdots + c_M \cdot a_{1,M}^p} = 1 \\ \cdots \\ \sqrt[p]{c_1 \cdot a_{l,1}^p + \cdots + c_M \cdot a_{l,M}^p} = 1 \end{cases} \tag{3}$$

Algorithm 1: Pseudo-code of AGE-MOEA [29]

```
Input: M: Number of objectives
       N: Population size
Result: Final population P
1  begin
2      P ⟵ RANDOM-POPULATION(N)
3      while not (stop_condition) do
4          Q ⟵ GENERATE-OFFSPRING(P)
5          𝔽 ⟵ FAST-NONDOMINATED-SORT(P ⋃ Q)
6          𝔽 ⟵ NORMALIZE(𝔽)
7          p ⟵ GET-GEOMETRY(𝔽_1, M)            /* Equation 4 */
8          d ⟵ 1                    /* First non-dominated rank */
9          while | P | + | 𝔽_d | ⩽ N do
10             SURVIVAL-SCORE(𝔽_d, d, p)
11             P ⟵ P ⋃ 𝔽_d
12             d ⟵ d + 1
13         SORT(𝔽_d)                          /* by survival scores */
14         P ⟵ P ⋃ 𝔽_d[1 : (N − |P|)]
15     return P
```

where $a_{i,j}$ denotes j-th objective value for the i-the non-dominated solution. Solving such a system of equations requires finding the constants p and $c_1, \ldots, c_M$ that minimize the fitting error. Given this set of non-linear equations, GFM-MOEA uses the Levenberg-Marquardt (LM) algorithm. The fitting process in [36] is particularly expensive since it has an overall complexity of $O(G'M^2(M+N))$, where G' is the number of iterations of the LM algorithm, N is the population size, while M is the number of objectives. Besides, the fitting method must be re-applied in every K generations based on a non-dominated set of solutions produced in the latest generation/iteration of the evolutionary process.

While GFM-MOEA uses the entire non-dominated front to find the best fitting L_p manifold, AGE-MOEA [29] uses one single point (called *central point*) to determine the curvature p of the approximated Pareto front. This reduces the cost of modeling the front to $O(M)$. This much lower computational cost allows to re-compute the p-value in each generation.

The pseudo-code of AGE-MOEA is reported in Algorithm 1 [29]. There are three key ingredients in AGE-MOEA [29]: (1) front normalization; (2) determining the curvature p of the first non-dominated front; (3) environmental selection based on *proximity* and *diversity*, both computed using the fitted p-value. We elaborate on these key ingredients in the following paragraphs.

Normalization (line 6 in Algorithm 1). First, the *fast non-dominated sorting algorithm* (from NSGA-II [12]) groups the population (parents and offspring) in sub-sequent non-dominated fronts. Then, all fronts are normalized by scaling the objective scores within the range $[0; 1]$ using the same procedure of NSGA-III [5]. The normalization is computed by (i) scaling the ideal point Z^{min} to the origin of the axes, and (ii) diving the objective values based on the hyperplane intercepting the extreme points in the front [29].

Front modeling (line 7 in Algorithm 1). Once the first front is normalized, its curvature is computed using the following formula:

$$p = \frac{log(M)}{log(M) - log\left(\sum_{i=1}^{M} C_i\right)} \tag{4}$$

where M is the number of objectives; C is the objective vector for the non-dominated solution closer to the central point within the non-dominated front.

Environmental selection (line 10 in Algorithm 1). The population for the next generation is selected by computing the *survival score*, which combines two sub-scores: *convergence* and *diversity*. The convergence score for each non-dominated solution A is computed using the distance between as L_p of the vector $||A||_p$, also called Minkowski distance [34]. The diversity for a non-dominated solution A is computed using the formula:

$$diversity(A, \mathbb{F}_1) = \min_{B \neq A \in \mathbb{F}_1} \|A - B\|_p \tag{5}$$

In other words, the diversity of the solution $A \in \mathbb{F}_1$ is computed as the minimum distance (L_p norm) of A to all other solutions in $\mathbb{F}_1$ using the estimated p norm.

2.2 Limitations of Existing Modeling Methods

The first limitation in the original AGE-MOEA framework regards the fitting accuracy of Equation 4. The accuracy strongly depends on the non-dominated solution used as the reference one. To better explain this limitation, let us consider the graphical example depicted in Figure 1. The plot shows three non-dominated points A, B, and C, forming a circular bi-dimensional front (manifold with p=2). If we apply Equation 4 for the different non-dominated points, we obtain the following results:

- For point A, p = 0.495074054167603
- For point B, p = 1.628072268133888
- For point C, p = 1.305032304384445

In the example of Figure 1, the exact solution is $p = 2$. Hence, the accuracy of computing/approximating the value p strongly depends on how close the chosen reference point is to the theoretical center of the front. Only if we apply Equation 4 to the theoretical middle point $C^* = (1/\sqrt{2}, 1/\sqrt{2})$, we obtain the correct value p=2.

Finally, the diversity is computed in AGE-MOEA using the $||\cdot||_p$ norm for the estimated geometry p. For example, for a flat front (p=1), the distance between two non-dominated solutions A and B is computed using the Manhattan distance, i.e., $d(A, B) = ||A - B||_1$. From a geometric point of view, the shortest distance between two points in a flat space ($p = 1$) is the length of the straight line between them computed with the Euclidean distance (or $||\cdot||_2$ norm). Similarly, the distance between two points A and B on a sphere (3-d manifold with p=2 geometry) is the great circle distance, which is the length of the chord of the great circle between the two points. For example, the great circle distance between points A and B in Figure 1 is the arc length $L = \theta \cdot r = \left(2 - \frac{2\pi}{8}\right) * 1 \approx 0.8219$, where θ is the angle delimited by the two points, and r is the ray of the circle. Recall that the circle is the geometry (shape) of the non-dominated front after normalization.

3 IMPROVED FRONT MODELLING MOEA

This paper introduces (i) a new accurate yet fast method to compute the curvature of the non-dominated front, and (2) a new diversity metrics for the non-dominated solutions based on Riemannian geometry. We called the resulting evolutionary algorithm AGE-MOEA-II. The key differences with its predecessor AGE-MOEA are highlighted in bold-face in Algorithm 2. We discuss them in the following sections.

3.1 Newton-Raphson Method

This paper proposes a more precise method to determine the curvature (L_p geometry) of the manifold formed by the non-dominated front. In particular, we propose to use the Newton-Raphson method, which is *fast* for approximating the roots (zeros) of real-valued functions of the form $f(x) = 0$ [19]. Let $f(x)$ be a continuous differential function; the Newton-Raphson method iteratively approximates the root of $f(x)$ using the formula:

$$x_{n+1} = x_n + \frac{f(x_n)}{f'(x_n)} \tag{6}$$

where x_{n+1} is the root approximation at the $n+1$ iteration, while x_n is the approximation produced in the previous iteration x_n. The iterative method starts with an initial user-provided point x_0 and terminates when the difference between two consecutive approximations is lower than a given threshold ϵ, i.e., when $|x_{n+1} - x_n| \leq \epsilon$.

In our context, we want to determine the value p that solves the following non-linear equation:

$$\sqrt[p]{\sum_{i=1}^{M} (a_i - z_i^{min})^p} = 1 \tag{7}$$

where a_i and z_i^{min} are the i-th coordinates of the non-dominated point $A = \{a_1, \ldots, a_M\}$ and the ideal point z^{min}. We consider the case where the ideal point z^{min} is the origin of the axes ($z_i^{min} = O, \forall i = 1 \ldots M$), which can be easily obtained by normalizing the non-dominated fronts within the interval $[0; 1]$. Hence, our problem consists of finding the root of the function (after simplification):

$$f(p) = log\left(\sum_{i=1}^{M} (a_i)^p\right) \tag{8}$$

The Newton-Raphson iterative formula for the function $f(p)$ in Equation 8 for a generic non-dominated point $A = \{a_1, \ldots, a_M\}$ is:

$$p_{n+1} = p_n + \frac{log\left[\sum_{i=1}^{M} (a_i)^{p_n}\right]}{\left[\sum_{i=1}^{M} (a_i)^{p_n} \cdot log\,(a_i)\right]} \tag{9}$$

with p_n and p_{n+1} being the approximated roots obtained at the iterations n and $n+1$, respectively. To initialize the Newton-Raphson method, we have chosen the seed $p_0 = 1$, which corresponds to a flat manifold (with no curvature). To make Equation 9 well-defined within the interval $p \in [0, +\infty)$, we consider $(a_i)^{p_n} \cdot log\,(a_i) = 0$ when $a_i = 0$ since it corresponds to the limit $\lim_{x \to 0} x \cdot log(x)$.

We have chosen the Newton-Raphson method over other numerical root-finding approaches (such as the bisection [31] or secant [2] methods) because of its fast convergence [17] and our function $f(p)$ is differentiable. In our preliminary analysis, we found that Equation 9 requires, on average, 3 iterations to produce an error of $|p_{n+1} - p_n| \leq 0.001$ for the benchmark problems we have considered in our empirical evaluation (See Section 4).

3.1.1 Graphical example. To better explain the advantages of the Newton-Raphson method over the heuristic used in [29], let us consider the graphical example depicted in Figure 1. The plot shows three non-dominated points A, B, and C forming a circular bi-dimensional front (manifold with p=2). If we use Equation 9, we obtain the following approximation of the value p when considering the different non-dominated points, setting the initial seed $p_0 = 1$ and with only four iterations:

- For point A, p_4 = 1.999983149170659
- For point B, p_4 = 1.999999999879300
- For point C, p_4 = 1.999979682971426

Instead, if we use the equation used in [29] for the point closer to the bisector of the first quadrant (e.g., point B), we obtain the value p = 1.628072268133887 (as shown in Section 2.2). Hence, the accuracy of the original formula used in AGE-MOEA depends on how close the reference point (i.e., the one used to approximate the front curvature) is to the bisector. Instead, Equation 9 achieves a very accurate approximation (with error <0.0001) in only four iterations, independently from which non-dominated point is actually considered as the reference one.

3.1.2 Complexity. The computational complexity of the Newton-Raphson iterative method is $O(M \times G)$, where M is the number of objectives while G is the number of iterations. In practice, the number of required iterations is very small (maximum 4-5 iterations) given the fast convergence of the method.

Algorithm 2: AGE-MOEA-II: An improved front modeling-based MOEA

Input: M: Number of objectives
N: Population size
Result: Final population P

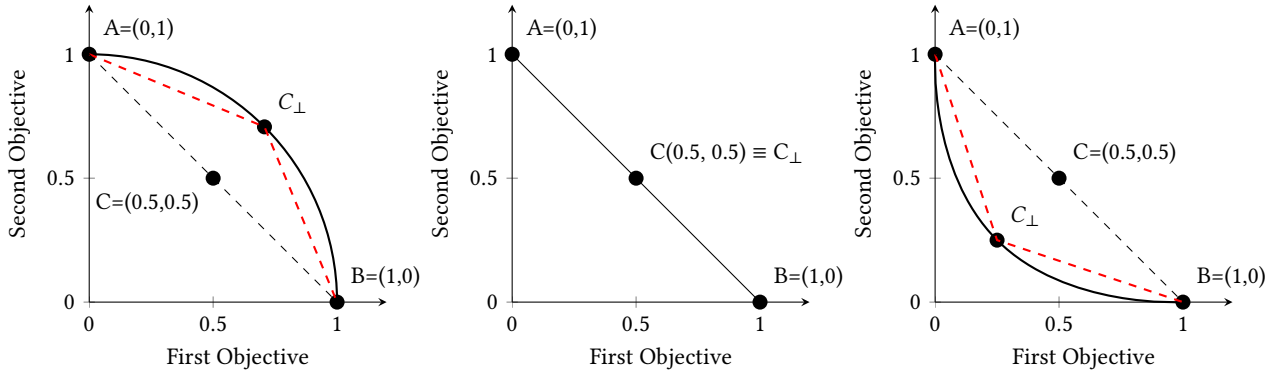
begin
  $P \longleftarrow$ RANDOM-POPULATION(N)
  while *not (stop_condition)* **do**
    $Q \longleftarrow$ GENERATE-OFFSPRING(P)
    $\mathbb{F} \longleftarrow$ FAST-NONDOMINATED-SORT($P \bigcup Q$)
    $\mathbb{F} \longleftarrow$ NORMALIZE($\mathbb{F}$)
    $p \longleftarrow$ **NEWTON-RAPHSON**($\mathbb{F}_1$) /* Eq. 9 */
    $d \longleftarrow 1$ /* First non-dominated rank */
    while $|P| + |\mathbb{F}_d| \leqslant N$ **do**
      $\mathbb{F}_\perp \longleftarrow$ **MANIFOLD-PROJECTION**($\mathbb{F}_d, p$)
      $D \longleftarrow$ **GEODESIC-DIV**($\mathbb{F}_\perp, p$) /* Eq. 15 */
      SURVIVAL-SCORE(D, $\mathbb{F}$, d, p)
      $P \longleftarrow P \bigcup \mathbb{F}_d$
      $d \longleftarrow d + 1$
    SORT($\mathbb{F}_d$) /* by survival scores */
    $P \longleftarrow P \bigcup \mathbb{F}_d[1 : (N - |P|)]$
  return P

3.2 Calculating Diversity with Geodesics

In differential geometry, the minimum distance between two points on a non-Euclidean topological space is computed busing geodesics, which generalize of the straight line between two points [22]. Geodesics are well-known geometry constructs for defining distances and metrics on Riemannian topologies [27]. Without loss of generality, the distance between two points A and B on a manifold is defined by the following integral:

$$d(A, B) = \inf_{\gamma} \int_{A}^{B} \left|\gamma'(t)\right| \, dt \tag{10}$$

where γ is a piecewise smooth curve (geodesic) joining A and B. In other words, the distance is the infimum of the lengths of all piecewise smooth paths between the two points.



(a) Distance approximation for $p = 1$ **(b) Distance approximation for $p = 2$** **(c) Distance approximation for $p = 0.5$**

Figure 2: Graphical examples on how approximating the distance between two points on fronts with different curvatures.

To simplify the computation of the distance between two points on a generic manifold, we propose using the approximation procedure described in the following subsection.

3.2.1 Geodesic Approximation. Let us consider a generic normalized set of non-dominated solutions $\mathbb{F}$ and the curvature of p computed using Equation 9. To approximate the distance between two generic points $A = \{a_1, \ldots, a_M\}$ and $B = \{b_1, \ldots, b_M\} \in \mathbb{F}$, we first compute the middle point C, i.e., the point in the middle of the straight line connecting the two points:

$$C = \left\{ \frac{a_1 + b_1}{2}, \ldots, \frac{a_M + b_M}{2} \right\} \tag{11}$$

Then, we compute the projection $C_\perp$ of the middle point C into the estimated unit-manifold (e.g., unit circle for p=2) using the following equation:

$$C_\perp = C \div \sqrt[p]{\sum_{i=1}^{M} (c_i)^p} \tag{12}$$

Finally, the geodesic distance between the two points A and B on the manifold with the estimated p is approximated as follow:

$$gd(A, B) = ||A - C_\perp||_2 + ||B - C_\perp||_2 \tag{13}$$

where $||A - C_\perp||_2$ is the Euclidean distance between A and $C_\perp$; and $||B - C_\perp||_2$ is the Euclidean distance between B and $C_\perp$.

3.2.2 Graphical example. . Figure 2 graphically shows how the geodesic distance is computed for fronts with three different curvatures, i.e., $p \in \{0.5, 1, 2\}$. Let us consider the case $p = 2$ shown in Figure 2a as an example. In this case, the shape/geometry of the front is the unit circle depicted in the figure. Let us assume we want to compute the geodesic distance between the two non-dominated points $A = (0, 1)$ and $B = (1, 0)$. The midpoint computed using Equation 11 is $C = (0.5, 0.5)$, while its projection on the unit circle is $C_\perp = (1/\sqrt{2}, 1/\sqrt{2})$ obtained by applying Equation 12. Then, the geodesic distance is approximated by Equation 13 as $d(A, B) \approx 0.7653 + 0.7653 \approx 1.5306$. This approximated value is very close to great-circle distance $L = \theta \cdot r = pi/2 \approx 1.5708$. While Equation 13 approximates the exact geodesic distance, it provides a better measure of the distance on the unit-circle than simply relying on the Euclidean distance A and B since $||A - B||_2 \approx 1.4142$. Notice that the distance $|| \cdot ||_2$ is used in the original AGE-MOEA for fronts with curvature p=2.

3.2.3 Diversity Measure. Let $\mathbb{F}$ be the set of non-dominated solutions after normalization, and let p be the curvature/geometry computed using Equation 9. First, we project all non-dominated points onto the estimated front manifold with curvature p:

$$A_\perp = A \div \sqrt[p]{\sum_{i=1}^{M} (a_i)^p}, \forall A \in \mathbb{F} \tag{14}$$

The diversity for each non-dominated point $A \in \mathbb{F}$ is computed using the geodesic distance in Equation 13 between its projected point $A_\perp$ and all other projected points of the front. Algorithm 2 denotes the projected front with D, where $D = \cup\{A_\perp\}, \forall A \in \mathbb{F}$. More formally, the geodesic diversity of each non-dominated point A is computed as:

$$geodesic\_div(A, \mathbb{F}_1) = \min_{B_\perp \neq A_\perp \in D} gd\,(A_\perp - B_\perp) \tag{15}$$

where gd is the geodesic distance computed using Equation 13. In other words, the diversity of the solution $A \in \mathbb{F}_1$ is computed as the minimum geodesic distance between its projection $A_\perp$ and all other projected points in D.

3.2.4 Complexity. The computational complexity of the geodesic distance (Equation 13) is $O(M \times N^2)$. In particular, for each pair A and B, it requires $O(M)$ operations to compute the Euclidean distance between A and the middle point $C_\perp$ and other $O(M)$ operations for the Euclidean distance between A and $C_\perp$.

4 EMPIRICAL STUDY

To assess the performance of AGE-MOEA-II, we performed a computational experiment to compare our MOEA with several state-of-the-art MOEAs on multiple benchmark problems.

4.1 Empirical Setting

4.1.1 Test Problems. We have selected two test benchmarks, namely WFG [18] and SMOP [37]. WFG is a well-known test benchmark

with 9 test problems with different properties, such as convex (WFG1, WFG2), degenerate (WFG3), concave (WFG5, WFG6, WFG7, and WFG8), and multimodal (WFG2, WFG9) fronts. SMOP has been recently proposed by Tian et al. [37] and contains eight large-scale sparse test problems with different shape and optimization challenges, such as deceptive objectives (SMOP2, SMOP3, SMOP8), sub-optima (SMOP1, SMOP4, SMOP6, and SMOP7), and epistasis (SMOP7 and SMOP8). We have chosen these two benchmarks since they cover different real-world scenarios [18, 37], and can be configured with a different number of objectives and decision variables. To assess the performance of the selected MOEAs, we considered the test problems with $m \in \{3, 6, 9\}$ objectives. Besides, we set the number of decision variables $D = 100$ for all test problems in both WFG and SMOP benchmarks.

4.1.2 Algorithms. Five state-of-the-art MOEAs are compared with AGE-MOEA-II, namely AGE-MOEA [29], NSGA-III [13], MOEA/D [28], VaEA [23], and LMEA [40]. AGE-MOEA [29] is an intuitive baseline since it is the direct predecessor of AGE-MOEA-II. The main differences between the two variants are related to how the non-dominated front is modeled and how the distance/diversity between the different solutions is computed. NSGA-III uses reference points and Pareto-ranking to compute diversity and convergence of the non-dominated fronts. MOEA/D [28] relies on a well-distributed set of weighted vectors and aggregation functions to produce a well-diversified set of non-dominated solutions. VaEA [23] uses an angle-based selection strategy that promotes only a few solutions within a given minimum angle distance. Finally, LMEA [40] is an MOEA designed for large-scale problems. It clusters decision variables into convergence-related and diversity-related ones. Then, the environmental selection is based on both the clustering method and the diversity of the generated solutions measured using an angle-based distance. We have selected these baselines since they use different strategies to balance convergence and diversity (e.g., reference point, Pareto-based, decomposition, angle-based distance, and clustering methods).

4.1.3 Performance metrics. We run each MOEA 30 times on each test problem to account for their stochastic nature. We stored the non-dominated front produced at the end of the search by each MOEA on each test problem. To assess the performance of the different MOEAs, we employed the hypervolume (HV) quality indicator [33]. HV takes values in $[0, 1]$, and it measures the size (portion) of the feasible region dominated by the set of non-dominated points produced by an MOEA (e.g., AGE-MOEA-II). Hence, a large HV value indicates better results. For calculating the HV scores, we consider the nadir point Z^{max} of the true Pareto front as the reference point. We have chosen the HV as the quality indicator since it simultaneously measures both convergence and diversity of the produced non-dominated fronts. To analyze the results, we report the median and the interquartile range (IQR) of the HV values produced by each MOEA across the 30 independent runs. Furthermore, we apply the Wilcoxon rank-sum test [10] to assess the significance of the difference (if any) between AGE-MOEA-II and the baselines (other MOEAs). For this test, we use the significance level $\alpha = 0.05$.

4.1.4 Paremeter settings. For all MOEAs, we used the same parameter settings for the shared parameters and genetic operators. Table 1 details the applied genetic operators as well as the corresponding parameter values [1, 9, 12, 13, 21]. Notice that for both mutation and crossover, we use the probability values and distributed indexes suggested in the literature [1, 12]. MOEA/D and NSGA-III require to set the reference points. To this aim, we have used Das and Dennis's systematic approach [11, 13] and set the number of reference points to the maximum value possible given the population size, that is 190, 273, and 330 for M=3, 6, and 9, respectively. For the stopping condition, we set the maximum number of solution evaluations (SE) proportional to the population size. In particular, we set SE=$N \times 200$, where N is the population size.

Table 1: Parameter values used for all MOEAS. M denotes the number of objectives.

Parameters	M=3	M=6	M=9
Population size	200	300	400
Number of Solution Evaluations	40,000	60,000	80,000
Number of iterations	200		
SBX probability [1]	$p_c = 1$		
SBX distributed index [1]	$\eta_c = 30$		
Polynomial mutation probability [12]	$p_m = 1/n$		
Mutation distributed index [12]	$\eta_m = 20$		

4.1.5 Implementation. We have implemented AGE-MOEA-II in Matlab within the PlatEMO framework [35]. We have selected this framework because it contains the original implementation of AGE-MOEA, and it implements all MOEAs used in our experiment as well as the benchmark problems. Besides, PlatEMO is public available on GitHub[1], and it can be easily extended with new MOEAs. The source code of AGE-MOEA-II is also publicly available at:

https://zenodo.org/record/6462859

All experiments were conducted on the same machine with the following characteristics: 6-Core Intel Core i7 processor running at 3.2GHz with 16GB RAM.

4.2 Empirical Results

Table 2 reports the median and the IQR values for the HV indicator achieved by the different MOEAs over 30 independent runs. The table highlights the results with ▼ and △ when the HV values achieved by AGE-MOEA-II are statistically better (larger) or worse (smaller) than the baseline MOEA according to the Wilcoxon rank-sum test [10]. In addition, Table 3 counts the number of test problems for which AGE-MOEA-II significantly outperforms (or is outperformed by) a baseline (e.g., AGE-MOEA).

From the tables, we can observe that AGE-MOEA-II achieves a significantly higher HV than NSGA-III in 15 out of 17 test problems when $M = 3$ and $M = 6$. The 15 test problems with a significant difference are the same independently of the number of objectives. When increasing the number of objectives to $M = 9$, AGE-MOEA-II achieves a significantly better HV value for an additional test problem, namely WFG3. It is worth noting that WFG3 has a degenerate true Pareto front with $p = 1$ curvature. Our results suggest that AGE-MOEA-II outperforms NSGA-III for degenerate fronts only for many objectives. Remarkable are the differences when considering a larger number of objectives. We observe 2-3 test problems with

[1] https://github.com/BIMK/PlatEMO

Table 2: HV values (median and IQR) yielded by the AGE-MOEA-II and the baselines on the SMOP and WFG benchmarks with M=3,6, and 9 objectives. Best values are highlighted in grey color.

Problem	M	NSGA-III	AGE-MOEA	MOEA/D	VaEA	LMEA	AGEMOEA-II
SMOP1	3	7.9383e-1 (1.04e-2) ▼	8.1416e-1 (1.14e-2) ▼	7.6983e-1 (2.66e-2) ▼	7.8485e-1 (8.47e-3) ▼	0.0000e+0 (1.48e-4) ▼	8.2104e-1 (8.81e-3)
SMOP2	3	4.8527e-1 (5.86e-2) ▼	6.2387e-1 (4.70e-2) ▼	3.3598e-1 (1.14e-1) ▼	5.1073e-1 (4.52e-2) ▼	0.0000e+0 (0.00e+0) ▼	6.7715e-1 (4.73e-2)
SMOP3	3	1.3283e-1 (1.03e-2) ▼	1.6283e-1 (1.52e-2) ▼	5.8745e-2 (2.64e-2) ▼	1.3495e-1 (7.01e-3) ▼	0.0000e+0 (0.00e+0) ▼	1.7365e-1 (5.93e-3)
SMOP4	3	9.1571e-1 (1.44e-2) ▼	9.6175e-1 (6.80e-3) ▼	8.9440e-1 (2.30e-2) ▼	9.3308e-1 (8.98e-3) ▼	3.1649e-1 (1.41e-2) ▼	9.7360e-1 (3.94e-3)
SMOP5	3	8.3630e-1 (2.01e-3) ▼	8.5338e-1 (1.29e-3) △	8.4850e-1 (3.12e-3) ▼	8.3946e-1 (1.20e-3) ▼	3.3764e-1 (2.03e-2) ▼	8.5254e-1 (1.36e-3)
SMOP6	3	9.6060e-1 (1.74e-3) ▼	9.7041e-1 (7.76e-4) ▼	9.5560e-1 (3.25e-3) ▼	9.5990e-1 (1.19e-3) ▼	7.6192e-1 (7.81e-3) ▼	9.7307e-1 (4.87e-4)
SMOP7	3	2.1646e-1 (5.14e-2) ▼	2.6638e-1 (2.22e-2) ▼	3.2481e-2 (2.26e-2) ▼	2.1337e-1 (3.19e-2) ▼	0.0000e+0 (0.00e+0) ▼	2.8582e-1 (3.43e-2)
SMOP8	3	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)
WFG1	3	3.8954e-1 (1.20e-2) ▼	4.4376e-1 (1.56e-2) ▼	4.1812e-1 (3.83e-2) ▼	3.8184e-1 (7.88e-3) ▼	0.0000e+0 (0.00e+0) ▼	4.6969e-1 (1.27e-2)
WFG2	3	8.5378e-1 (6.59e-3) ▼	8.7808e-1 (6.75e-3) ▼	7.4573e-1 (4.27e-2) ▼	8.5574e-1 (6.20e-3) ▼	5.0972e-1 (2.17e-2) ▼	8.8293e-1 (9.80e-3)
WFG3	3	3.2949e-1 (7.75e-3)	3.2793e-1 (4.03e-3)	2.1978e-1 (3.51e-2) ▼	3.0509e-1 (6.90e-3) ▼	9.6791e-2 (1.79e-3) ▼	3.3085e-1 (7.55e-3)
WFG4	3	4.8723e-1 (2.88e-3) ▼	5.1364e-1 (4.19e-3) ▼	4.4788e-1 (7.35e-3) ▼	4.9490e-1 (3.45e-3) ▼	2.7076e-1 (1.06e-2) ▼	5.1827e-1 (2.82e-3)
WFG5	3	4.7312e-1 (5.02e-3) ▼	5.0431e-1 (2.70e-3) ▼	4.2370e-1 (1.45e-2) ▼	4.7992e-1 (3.52e-3) ▼	1.9892e-1 (2.83e-3) ▼	5.0723e-1 (2.95e-3)
WFG6	3	4.5648e-1 (7.10e-3) ▼	4.9439e-1 (5.54e-3) ▼	4.1391e-1 (1.26e-2) ▼	4.5843e-1 (4.89e-3) ▼	1.4230e-1 (3.31e-3) ▼	4.9923e-1 (5.31e-3)
WFG7	3	4.9211e-1 (8.22e-3) ▼	5.2730e-1 (4.16e-3) ▼	3.1497e-1 (2.99e-2) ▼	4.8442e-1 (1.06e-2) ▼	3.2986e-1 (9.64e-2) ▼	5.3535e-1 (3.91e-3)
WFG8	3	4.5709e-1 (4.49e-3) ▼	4.8521e-1 (4.81e-3)	4.2197e-1 (1.45e-2) ▼	4.6605e-1 (5.52e-3) ▼	4.9750e-1 (7.76e-3) △	4.8645e-1 (6.75e-3)
WFG9	3	4.4233e-1 (1.14e-2) ▼	4.6916e-1 (1.48e-2) ▼	2.9674e-1 (2.97e-2) ▼	4.4047e-1 (1.03e-2) ▼	3.5208e-1 (1.14e-2) ▼	4.8294e-1 (9.85e-3)
SMOP1	6	9.0587e-1 (1.22e-2) ▼	9.7475e-1 (3.62e-3) ▼	9.5277e-1 (1.98e-2) ▼	8.3600e-1 (1.68e-2) ▼	3.7740e-2 (1.06e-2) ▼	9.8920e-1 (9.81e-4)
SMOP2	6	9.4595e-2 (1.73e-2) ▼	7.1181e-1 (6.13e-2) ▼	5.1517e-1 (1.26e-1) ▼	9.7716e-2 (1.78e-2) ▼	6.4543e-4 (1.22e-3) ▼	9.6290e-1 (8.46e-3)
SMOP3	6	3.7803e-2 (8.58e-3) ▼	4.0438e-1 (2.41e-2) ▼	6.8498e-2 (5.51e-2) ▼	2.5869e-2 (1.41e-2) ▼	0.0000e+0 (5.33e-6) ▼	5.4054e-1 (1.62e-2)
SMOP4	6	9.4333e-1 (2.54e-2) ▼	9.9851e-1 (4.15e-4) ▼	9.5481e-1 (1.14e-2) ▼	9.7842e-1 (1.46e-3) ▼	8.5187e-1 (2.91e-2) ▼	9.9991e-1 (1.60e-5)
SMOP5	6	9.6938e-1 (5.80e-3) ▼	9.9247e-1 (9.81e-4) ▼	9.0342e-1 (1.96e-2) ▼	9.7330e-1 (2.12e-3) ▼	8.4851e-1 (1.85e-2) ▼	9.9369e-1 (1.12e-3)
SMOP6	6	9.9446e-1 (2.78e-3) ▼	9.9989e-1 (2.10e-5) ▼	9.7485e-1 (5.59e-3) ▼	9.9890e-1 (2.13e-4) ▼	9.7433e-1 (8.39e-3) ▼	9.9989e-1 (1.20e-5)
SMOP7	6	4.7756e-3 (2.69e-3) ▼	2.9255e-1 (6.06e-2) ▼	4.3521e-2 (4.33e-2) ▼	5.0929e-4 (4.20e-4) ▼	0.0000e+0 (0.00e+0) ▼	5.2498e-1 (4.26e-2)
SMOP8	6	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)
WFG1	6	2.7449e-1 (2.23e-3) ▼	2.8373e-1 (4.04e-3) ▼	2.7277e-1 (2.19e-2) ▼	2.7240e-1 (1.51e-3) ▼	0.0000e+0 (0.00e+0) ▼	2.9030e-1 (3.68e-3)
WFG2	6	8.4769e-1 (2.59e-2) ▼	9.2756e-1 (6.01e-3) ▼	6.9661e-1 (3.28e-2) ▼	8.3469e-1 (1.15e-2) ▼	5.0834e-1 (1.98e-2) ▼	9.4392e-1 (8.92e-3)
WFG3	6	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)
WFG4	6	6.7253e-1 (1.15e-2) ▼	7.4670e-1 (5.48e-3) ▼	5.3248e-1 (6.44e-2) ▼	6.7395e-1 (9.97e-3) ▼	3.3950e-1 (1.61e-2) ▼	7.5855e-1 (5.54e-3)
WFG5	6	6.1381e-1 (1.02e-2) ▼	7.1530e-1 (5.40e-3) ▼	5.8198e-1 (3.58e-2) ▼	6.0171e-1 (1.20e-2) ▼	3.1960e-1 (7.52e-3) ▼	7.4264e-1 (6.38e-3)
WFG6	6	5.7755e-1 (1.24e-2) ▼	7.0568e-1 (1.00e-2) ▼	4.7854e-1 (6.33e-2) ▼	5.7803e-1 (8.70e-3) ▼	7.8580e-1 (5.30e-1) △	7.3071e-1 (7.15e-3)
WFG7	6	5.4299e-1 (1.71e-2) ▼	7.6026e-1 (1.30e-2) ▼	4.9288e-1 (4.48e-2) ▼	6.0440e-1 (1.58e-2) ▼	5.1953e-1 (1.72e-2) ▼	7.9293e-1 (6.03e-3)
WFG8	6	6.2718e-1 (1.11e-2) ▼	7.0843e-1 (6.29e-3) ▼	3.9970e-1 (6.94e-2) ▼	6.2668e-1 (1.49e-2) ▼	7.0121e-1 (1.07e-2) ▼	7.2105e-1 (7.20e-3)
WFG9	6	5.3602e-1 (1.88e-2) ▼	6.5562e-1 (1.40e-2) ▼	4.8193e-1 (1.23e-1) ▼	5.0751e-1 (2.24e-2) ▼	4.1141e-1 (2.22e-2) ▼	7.0852e-1 (1.43e-2)
SMOP1	9	9.1194e-1 (1.36e-2) ▼	9.9652e-1 (6.51e-4) ▼	7.6769e-1 (4.36e-2) ▼	8.2945e-1 (2.60e-2) ▼	6.7208e-2 (2.19e-2) ▼	9.9918e-1 (9.60e-5)
SMOP2	9	1.6797e-1 (1.55e-2) ▼	8.3050e-1 (3.26e-2) ▼	4.8073e-1 (4.09e-2) ▼	1.6800e-1 (2.51e-2) ▼	2.1433e-3 (2.96e-3) ▼	9.9585e-1 (1.26e-3)
SMOP3	9	2.5468e-2 (7.19e-3) ▼	5.1486e-1 (3.35e-2) ▼	7.5394e-2 (5.23e-2) ▼	2.5798e-2 (8.67e-3) ▼	0.0000e+0 (2.44e-6) ▼	6.3187e-1 (1.06e-2)
SMOP4	9	9.6621e-1 (1.23e-2) ▼	9.9997e-1 (9.00e-6) ▼	8.6900e-1 (7.73e-3) ▼	9.9844e-1 (1.72e-4) ▼	8.9264e-1 (1.93e-2) ▼	1.0000e+0 (1.00e-6)
SMOP5	9	9.8093e-1 (1.65e-2) ▼	9.9997e-1 (6.00e-6) △	8.8156e-1 (5.79e-3) ▼	9.9937e-1 (1.04e-4) ▼	8.9361e-1 (2.49e-2) ▼	9.9985e-1 (2.60e-5)
SMOP6	9	9.9263e-1 (5.52e-3) ▼	1.0000e+0 (1.00e-6)	9.2078e-1 (7.62e-3) ▼	9.9995e-1 (1.70e-5) ▼	9.8272e-1 (4.63e-3) ▼	1.0000e+0 (1.00e-6)
SMOP7	9	2.8711e-3 (3.03e-3) ▼	4.0556e-1 (7.24e-2) ▼	1.3145e-1 (9.49e-2) ▼	0.0000e+0 (0.00e+0) ▼	0.0000e+0 (0.00e+0) ▼	7.2868e-1 (5.93e-2)
SMOP8	9	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)	0.0000e+0 (0.00e+0)
WFG1	9	2.4892e-1 (5.08e-3) ▼	3.2931e-1 (7.91e-3) ▼	4.7066e-1 (6.14e-2) △	2.3761e-1 (2.84e-3) ▼	1.5701e-2 (1.80e-2) ▼	3.8569e-1 (1.58e-2)
WFG2	9	8.9753e-1 (2.67e-2) ▼	9.3605e-1 (1.13e-2) ▼	6.6769e-1 (3.70e-2) ▼	8.1996e-1 (1.58e-2) ▼	4.7675e-1 (1.22e-2) ▼	9.5272e-1 (7.45e-3)
WFG3	9	3.3299e-1 (7.53e-3) ▼	3.4988e-1 (3.84e-3)	2.4602e-1 (3.61e-2) ▼	3.2881e-1 (4.36e-3) ▼	1.0350e-1 (1.60e-3) ▼	3.4847e-1 (5.14e-3)
WFG4	9	6.8611e-1 (2.26e-2) ▼	7.8342e-1 (9.12e-3) ▼	3.1054e-1 (3.17e-2) ▼	7.0813e-1 (8.11e-3) ▼	3.4294e-1 (1.37e-2) ▼	8.0922e-1 (7.27e-3)
WFG5	9	5.8525e-1 (1.80e-2) ▼	7.4103e-1 (1.08e-2) ▼	4.3496e-1 (4.22e-2) ▼	6.1813e-1 (1.47e-2) ▼	3.2565e-1 (5.19e-3) ▼	7.9309e-1 (8.74e-3)
WFG6	9	5.4269e-1 (1.43e-2) ▼	7.3866e-1 (8.99e-3) ▼	1.8879e-1 (1.58e-2) ▼	5.7646e-1 (1.64e-2) ▼	8.4480e-1 (1.13e-2) △	7.7980e-1 (1.16e-2)
WFG7	9	5.7171e-1 (1.25e-2) ▼	8.0721e-1 (1.58e-2) ▼	2.3464e-1 (9.53e-2) ▼	6.6157e-1 (1.40e-2) ▼	5.6795e-1 (1.74e-2) ▼	8.5471e-1 (6.94e-3)
WFG8	9	5.9656e-1 (1.55e-2) ▼	7.3638e-1 (8.60e-3) ▼	1.6326e-1 (5.12e-2) ▼	6.3040e-1 (1.17e-2) ▼	7.4579e-1 (1.12e-2) ▼	7.5891e-1 (8.66e-3)
WFG9	9	5.2598e-1 (3.83e-2) ▼	6.9463e-1 (1.48e-2) ▼	2.5318e-1 (2.44e-1) ▼	5.5946e-1 (2.11e-2) ▼	4.6368e-1 (1.82e-2) ▼	7.6207e-1 (2.11e-2)

at least one order of magnitude difference in the median HV values achieved by the two MOEAs (e.g., SMOP7 with M=6). Table 2 indicates that AGE-MOEA-II outperforms its predecessor AGE-MOEA in the large majority of test problems. In particular, this applies in 13, 15, and 13 problems for $M = 3$, $M = 6$, and $M = 9$, respectively. There is only one test problem, namely SMPO5, for which AGE-MOEA produces significantly better HV values than AGE-MOEA-II. However, this happens only for $M = 3$ and $M = 9$ while there is no significant difference for $M = 6$. However, we notice that the difference in the median HV is negligible in practice, being 0.0008 on

Table 3: Number of test problems in which an algorithm A statistically outperforms (<) another algorithm B w.r.t. the HV metric according to the Wilcoxon test.

Comparison	M=3	M=6	M=9
AGE-MOEA-II < NSGA-III	15	15	16
NSGA-III < AGE-MOEA-II	-	-	-
AGE-MOEA-II < AGE-MOEA	13	15	13
AGE-MOEA < AGE-MOEA-II	1	-	1
AGE-MOEA-II < MOEA/D	16	15	15
MOEA/D < AGE-MOEA-II	1	-	1
AGE-MOEA-II < VaEA	16	15	16
VaEA < AGE-MOEA-II	-	-	-
AGE-MOEA-II < LMEA	15	14	15
LMEA < AGE-MOEA-II	1	1	1

average for $M = 3$ and $M = 6$. For WFG3, there is no significant difference between AGE-MOEA-II and AGE-MOEA, independently of the number of objectives M. This result is expected since the front for WFG3 is degenerate (e.g., one single line), for which the difference in the numerical methods applied in AGE-MOEA-II and its predecessor is minimal.

With regards to MOEA/D, Table 2 reveals that AGE-MOEA-II significantly outperforms the latter in most test problems independently of the number of objectives. In detail, AGE-MOEA-II yielded significantly higher HV values than MOEA/D in 16 out of 17 problems for M=3, and in 15 problems for both M=6 and M=9. For some problems, the difference is very remarkable, e.g., for SMOP7 with M=3, where there is one order of magnitude difference in the HV values produced by the two MOEAs. In particular, AGE-MOEA-II produced a median HV≈0.54 against an HV≈0.04 for MOEA/D. The true Pareto front for SMOP7 has a $p = 2$ curvature and multiple local optima.We generally observe large differences for other deceptive and multimodal problems such as SMOP3 and WFG8. Vice versa, there is only one test problem (namely WFG1 with M=9), for which the baseline outperforms AGE-MOEA-II.

From Tables 2-3, AGE-MOEA-II achieves significantly larger HV values than VaEA in 16 out of 17 test problems for M=3 and M=9, respectively. For M=6, the significance holds in 15 test problems. The largest difference can be observed for SMOP7 with M=9, for which the proposed MOEA obtains an average HV≈0.72 while VaEA achieved an average HV=0. Vice versa, in none of the test problems, the baseline could outperform AGE-MOEA-II.

Particularly relevant are the results of the comparison between AGE-MOEA-II and LMEA. The latter is an MOEA suitably designed for large-scale problems, e.g., the SMOP benchmark. As reported in Table 3, AGE-MOEA-II statistically outperformed LMEA in 15 out of 17 test problems for M=3 and M=9. Instead, for M=6, the statistical significance holds for 14 out of 17 test problems. There are various test problems (e.g., SMOP3 and SMOP7 with any M) for which LMEA produced a zero HV value while AGE-MOEA achieved non-zero values. That indicates that for certain problems (e.g., WFG1 with $M = 2$), LMEA struggles to generate non-dominated solutions that reach at least the nadir point of the true Pareto. For $M \in \{3, 6, 9\}$, there is only one test problem where LMEA significantly outperforms AGE-MOEA-II, but the test problem for which this happens varies when increasing the number of objectives.

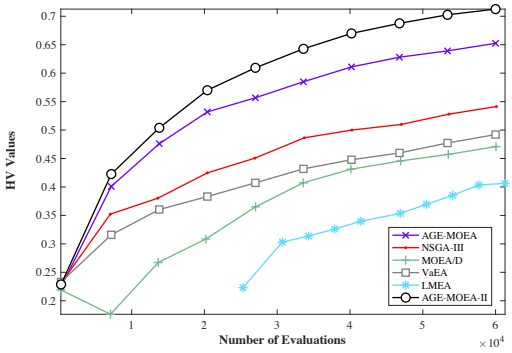


Figure 3: Median HV values achieved by the different MOEAs for WFG9 with M=6 objectives.

4.2.1 Graphical comparison. To graphically compare the MOEAs, Figure 3 depicts the median HV values achieved for WFG9 with M=6 and D=100, using the parameters described in Table 1. The median HV values achieved by AGE-MOEA-II are larger than those achieved by all other MOEAs when varying the number of fitness evaluations (FEs). The delta between AGE-MOEA-II and its predecessor AGE-MOEA increases with a larger number of FEs. Furthermore, the delta with the other MOEAs remains very large through the generations. Finally, AGE-MOEA yielded the second-best HV results. This confirms the better effectiveness/performance of MOEAs that use front modeling techniques.

5 CONCLUSION AND FUTURE WORK

We have introduced an accurate yet fast method to model the geometry/curvature of the non-dominated fronts. By leveraging the Newton-Raphson method, the novel technique estimates the actual geometry more accurately within a few iterations, allowing its usage with negligible overhead. Finally, we introduce a novel method to compute the diversity among the generated solutions using the geodesic distance, which is a generalization of the Euclidean distance for curved Riemannian manifolds. We have implemented these new methods within the AGE-MOEA framework, obtaining a new MOEA, which we called AGE-MOEA-II.

We conducted an empirical study to assess the performance of AGE-MOEA-II against its predecessor AGE-MOEA and other four state-of-the-art MOEAs, namely NSGA-III, MOEA/D, LMEA, and VaEA. We selected 17 test problems from two benchmarks, namely WFG and SMOP, that are characterized by different front shapes and optimization challenges. The achieved results show that AGE-MOEA-II significantly outperforms the other MOEAs w.r.t. the HV indicator, also when increasing the number of objectives from $M = 3$ to 9.

In our future work, we aim to assess the impact of using different normalization methods [5], different numerical methods to approximate the geodesic distance, as well as incorporating front modeling methods into other MOEA categories, such as decomposition-based [39] and clustering-based [6, 40] MOEAs. Furthermore, we plan to apply AGE-MOEA-II to real-world engineering problems, such as regression testing [3, 4, 26], test case generation [15, 30], and multi-objective predictive models [8].

REFERENCES

[1] Ram Bhushan Agrawal, K Deb, and RB Agrawal. 1995. Simulated binary crossover for continuous search space. *Complex systems* 9, 2 (1995), 115–148.

[2] Myron B Allen III and Eli L Isaacson. 2011. *Numerical analysis for applied science*. Vol. 35. John wiley & sons.

[3] Christian Birchler, Nicolas Ganz, Sajad Khatiri, Alessio Gambi, and Sebastiano Panichella. 2022. Cost-effective simulation-based test selection in self-driving cars software with SDC-Scissor. In *29th IEEE International Conference on Software Analysis, Evolution, and Reengineering, Honolulu, USA (online), 15-18 March 2022*. ZHAW Zürcher Hochschule für Angewandte Wissenschaften.

[4] Christian Birchler, Sajad Khatiri, Pouria Derakhshanfar, Sebastiano Panichella, and Annibale Panichella. 2021. Single and Multi-objective Test Cases Prioritization for Self-driving Cars in Virtual Environments. https://doi.org/10.48550/ARXIV.2107.09614

[5] Julian Blank, Kalyanmoy Deb, and Proteek Chandan Roy. 2019. Investigating the normalization procedure of NSGA-III. In *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 229–240.

[6] Anton Bouter, Ngoc Hoang Luong, Cees Witteveen, Tanja Alderliesten, and Peter AN Bosman. 2017. The multi-objective real-valued gene-pool optimal mixing evolutionary algorithm. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 537–544.

[7] Dimo Brockhoff, Tobias Wagner, and Heike Trautmann. 2012. On the properties of the R2 indicator. In *Proceedings of the 14th annual conference on Genetic and evolutionary computation*. 465–472.

[8] Gerardo Canfora, Andrea De Lucia, Massimiliano Di Penta, Rocco Oliveto, Annibale Panichella, and Sebastiano Panichella. 2013. Multi-objective cross-project defect prediction. In *2013 IEEE Sixth International Conference on Software Testing, Verification and Validation*. IEEE, 252–261.

[9] Ran Cheng, Miqing Li, Ye Tian, Xingyi Zhang, Shengxiang Yang, Yaochu Jin, and Xin Yao. 2017. A benchmark test suite for evolutionary many-objective optimization. *Complex & Intelligent Systems* 3, 1 (01 Mar 2017), 67–81. https://doi.org/10.1007/s40747-017-0039-7

[10] W. J. Conover. 1998. *Practical Nonparametric Statistics* (3rd edition ed.). Wiley.

[11] Indraneel Das and John E Dennis. 1998. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization* 8, 3 (1998), 631–657.

[12] Kalyanmoy Deb. 2001. *Multi-objective optimization using evolutionary algorithms*. Vol. 16. John Wiley & Sons.

[13] K. Deb and H. Jain. 2014. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints. *IEEE Transactions on Evolutionary Computation* 18, 4 (Aug 2014), 577–601. https://doi.org/10.1109/TEVC.2013.2281535

[14] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation* 6, 2 (2002), 182–197.

[15] Pouria Derakhshanfar, Xavier Devroey, Andy Zaidman, Arie Van Deursen, and Annibale Panichella. 2020. Good things come in threes: Improving search-based crash reproduction with helper objectives. In *2020 35th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 211–223.

[16] Giuseppe Di Domenico, Dror Weisman, Annibale Panichella, Dolev Roitman, and Ady Arie. 2022. Large-Scale Inverse Design of a Planar On-Chip Mode Sorter. *ACS Photonics* (2022).

[17] JC Ehiwario and SO Aghamie. 2014. Comparative study of bisection, Newton-Raphson and secant methods of root-finding problems. *IOSR Journal of Engineering* 4, 4 (2014), 01–07.

[18] Simon Huband, Philip Hingston, Luigi Barone, and Lyndon While. 2006. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation* 10, 5 (2006), 477–506.

[19] Eugene Isaacson and Herbert Bishop Keller. 2012. *Analysis of numerical methods*. Courier Corporation.

[20] Hisao Ishibuchi, Hiroyuki Masuda, Yuki Tanigaki, and Yusuke Nojima. 2015. Modified distance calculation in generational distance and inverted generational distance. In *International conference on evolutionary multi-criterion optimization*. Springer, 110–125.

[21] H. Jain and K. Deb. 2014. An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point Based Nondominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach. *IEEE Transactions on Evolutionary Computation* 18, 4 (Aug 2014), 602–622. https://doi.org/10.1109/TEVC.2013.2281534

[22] John M Lee. 2018. *Introduction to Riemannian manifolds*. Springer.

[23] Zhi-Zhong Liu, Yong Wang, and Pei-Qiu Huang. 2020. AnD: A many-objective evolutionary algorithm with angle-based selection and shift-based density estimation. *Information Sciences* 509 (2020), 400–419.

[24] R Timothy Marler and Jasbir S Arora. 2004. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization* 26, 6 (2004), 369–395.

[25] Saúl Zapotecas Martínez, Víctor A Sosa Hernández, Hernán Aguirre, Kiyoshi Tanaka, and Carlos A Coello Coello. 2014. Using a family of curves to approximate the Pareto front of a multi-objective optimization problem. In *International Conference on Parallel Problem Solving from Nature*. Springer, 682–691.

[26] Mitchell Olsthoorn and Annibale Panichella. 2021. Multi-objective test case selection through linkage learning-based crossover. In *International Symposium on Search Based Software Engineering*. Springer, 87–102.

[27] John Oprea. 2019. *Differential geometry and its applications*. Vol. 59. American Mathematical Soc.

[28] Suat Özdemir, A Attea Bara'a, and Önder A Khalil. 2013. Multi-objective evolutionary algorithm based on decomposition for energy efficient coverage in wireless sensor networks. *Wireless personal communications* 71, 1 (2013), 195–215.

[29] Annibale Panichella. 2019. An adaptive evolutionary algorithm based on non-euclidean geometry for many-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 595–603.

[30] Annibale Panichella, Fitsum Meshesha Kifetew, and Paolo Tonella. 2015. Reformulating branch coverage as a many-objective optimization problem. In *2015 IEEE 8th international conference on software testing, verification and validation (ICST)*. IEEE, 1–10.

[31] Alfio Quarteroni, Riccardo Sacco, and Fausto Saleri. 2010. *Numerical mathematics*. Vol. 37. Springer Science & Business Media.

[32] Günter Rudolph, Oliver Schütze, Christian Grimme, Christian Domínguez-Medina, and Heike Trautmann. 2016. Optimal averaged Hausdorff archives for bi-objective problems: theoretical and numerical results. *Computational Optimization and Applications* 64, 2 (2016), 589–618.

[33] Ke Shang, Hisao Ishibuchi, Linjun He, and Lie Meng Pang. 2020. A survey on the hypervolume indicator in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 25, 1 (2020), 1–20.

[34] Anthony C Thompson and Anthony C Thompson. 1996. *Minkowski geometry*. Vol. 63. Cambridge University Press.

[35] Ye Tian, Ran Cheng, Xingyi Zhang, and Yaochu Jin. 2017. PlatEMO: A MATLAB Platform for Evolutionary Multi-Objective Optimization. *IEEE Computational Intelligence Magazine* 12 (2017), 73–87. Issue 4.

[36] Ye Tian, Xingyi Zhang, Ran Cheng, Cheng He, and Yaochu Jin. 2018. Guiding Evolutionary Multio-bjective Optimization With Generic Front Modeling. *IEEE transactions on cybernetics* (2018).

[37] Ye Tian, Xingyi Zhang, Chao Wang, and Yaochu Jin. 2019. An evolutionary algorithm for large-scale sparse multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation* 24, 2 (2019), 380–393.

[38] Tjalling J Ypma. 1995. Historical development of the Newton–Raphson method. *SIAM review* 37, 4 (1995), 531–551.

[39] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation* 11, 6 (2007), 712–731.

[40] Xingyi Zhang, Ye Tian, Ran Cheng, and Yaochu Jin. 2016. A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization. *IEEE Transactions on Evolutionary Computation* 22, 1 (2016), 97–112.

[41] Eckart Zitzler and Simon Künzli. 2004. Indicator-based selection in multiobjective search. In *International conference on parallel problem solving from nature*. Springer, 832–842.