Comparing Decomposition-Based Evolutionary Algorithms for Multi and Many-Objective Optimization Problems☆

Comparando Algoritmos Evolutivos Baseados em Decomposição para Problemas de Otimização Multiobjetivo e com Muitos Objetivos

Marcela C. C. Peito, Dênis E. C. Vargas†, Elizabeth F. Wanner

Centro Federal de Educação Tecnológica de Minas Gerais - Belo Horizonte, MG, Brasil
†**Corresponding author:** denis.vargas@cefetmg.br

Abstract

Many real-world problems can be mathematically modeled as Multiobjective Optimization Problems (MOPs), as they involve multiple conflicting objective functions that must be minimized simultaneously. MOPs with more than 3 objective functions are called Many-objective Optimization Problems (MaOPs). MOPs are typically solved through Multiobjective Evolutionary Algorithms (MOEAs), which can obtain a set of non-dominated optimal solutions, known as a Pareto front, in a single run. The MOEA Based on Decomposition (MOEA/D) is one of the most efficient, dividing a MOP into several single-objective subproblems and optimizing them simultaneously. This study evaluated the performance of MOEA/D and four variants representing the state of the art in the literature (MOEA/DD, MOEA/D-DE, MOEA/D-DU, and MOEA/D-AWA) in MOPs and MaOPs. Computational experiments were conducted using benchmark MOPs and MaOPs from the DTLZ suite considering 3 and 5 objective functions. Additionally, a statistical analysis, including the Wilcoxon test, was performed to evaluate the results obtained in the IGD+ performance indicator. The Hypervolume performance indicator was also considered in the combined Pareto front, formed by all solutions obtained by each MOEA. The experiments revealed that MOEA/DD performed best in IGD+, and MOEA/D-AWA achieved the highest Hypervolume in the combined Pareto front, while MOEA/D-DE registered the worst result in this set of problems.

Keywords

Evolutionary Algorithm • Decomposition • Multiobjective Optimization

Resumo

Muitos problemas oriundos do mundo real podem ser modelados matematicamente como Problemas de Otimização Multiobjetivo (POMs), já que possuem diversas funções objetivo conflitantes entre si que devem ser minimizadas simultaneamente. POMs com mais de 3 funções objetivo recebem o nome de Problemas de Otimização com Muitos Objetivos (MaOPs, do inglês Many-objective Optimization Problems). Os POMs geralmente são resolvidos através de Algoritmos Evolutivos Multiobjetivos (MOEAs, do inglês Multiobjective Evolutionary Algorithms), os quais conseguem obter um conjunto de soluções ótimas não dominadas entre si, conhecidos como frente de Pareto, em uma única execução. O MOEA baseado em decomposição (MOEA/D) é um dos mais eficientes, o qual divide um POM em vários subproblemas monobjetivos otimizando-os ao mesmo tempo. Neste estudo foi realizada uma avaliação dos desempenhos do MOEA/D e quatro de suas variantes que representam o estado da arte da literatura (MOEA/DD,

☆This article is an extended version of the work presented at the Joint XXVI ENMC National Meeting on Computational Modeling, XIV ECTM Meeting on Science and Technology of Materials held in Nova Friburgo – Brazil, from October 25th to 27th, 2023

MOEA/D-DE, MOEA/D-DU e MOEA/D-AWA) em POMs e MaOPs. Foram conduzidos experimentos computacionais utilizando POMs e MaOPs benchmark do suite DTLZ considerando 3 e 5 funções objetivo. Além disso, foi realizada uma análise estatística que incluiu o teste de Wilcoxon para avaliar os resultados obtidos no indicador de desempenho IGD+. Também foi considerado o indicador de desempenho Hypervolume na frente de Pareto combinada, que é formada por todas as soluções obtidas por cada MOEA. Os experimentos revelaram que o MOEA/DD apresentou a melhor performance no IGD+ e o MOEA/D-AWA obteve o maior Hypervolume na frente de Pareto combinada, enquanto o MOEA/D-DE registrou o pior resultado nesse conjunto de problemas.

Palavras-chave

Algoritmo Evolutivo • Decomposição • Otimização Multiobjetivo

1 Introduction

Evolutionary Algorithms (EAs) are known for their ability to solve complex optimization problems, including Multi-Objective Optimization Problems (MOPs), which involve optimizing conflicting objective functions simultaneously. Solving a MOP means finding solutions that achieve the best possible trade-offs among the objective functions, known as the Pareto front. Multi-Objective Evolutionary Algorithms (MOEAs) are a popular method for solving MOPs, as they can efficiently approximate the Pareto front in a single run [1].

The Multi-Objective Evolutionary Algorithm Based on Decomposition (MOEA/D) is one of the most popular and efficient MOEAs in the literature. Proposed by Zhang and Li [2], MOEA/D decomposes the MOP into several single-objective optimization subproblems and solves them simultaneously using aggregation functions. Each individual in the population represents the current best solution for one of the subproblems, and each subproblem is solved using information from subproblems in its neighborhood. MOEA/D has been widely employed for efficiently solving complex MOPs in various domains.

Most current state-of-the-art MOEAs treat a MOP as a whole without involving decomposition. MOEA/D has some features that can enhance its performance over non-decomposition MOEAs. For example, assigning fitness and preserving diversity can be easily managed in MOEA/D, in addition to having a low computational complexity in each generation. Also, MOEA/D allows easy integration of objective normalization techniques, an excellent option for addressing objectives with disparate scales. Furthermore, scalar optimization methods can be easily employed in MOEA/D as each solution is intrinsically associated with a scalar optimization problem, contrasting with non-decomposition MOEAs, which face a challenge in effectively utilizing these methods.

Xu et al. [3] provide a comprehensive survey of publications related to MOEA/D and its variants, including challenges and future research directions. Since its proposition, MOEA/D has had multiple improvements and extensions, resulting in variants with new methods for solving different problems in various fields. These variants aim to overcome limitations in their original components and enhance their efficiency in several MOPs. Among the MOEA/D variants mentioned in [3] are MOEA/DD, MOEA/D-DE, MOEA/D-DU, and MOEA/D-AWA, which will be introduced below.

Li et al. [4] introduced the MOEA Based on Dominance and Decomposition (MOEA/DD), which combines dominance-based and decomposition-based approaches to balance convergence and diversity in the evolutionary process of Many-objective Optimization Problems (MaOPs, which are MOPs with more than three objective functions). Each weight vector in MOEA/DD defines a subproblem and designates a subregion that helps estimate a local population density. [4] compared the MOEA/DD's performance with four other state-of-the-art MOEAs on a set of unconstrained benchmark problems with up to 15 objectives. Empirical results fully demonstrate the superiority of MOEA/DD on all considered test instances.

Another variant is MOEA/D Based on Differential Evolution (MOEA/D-DE) [5], which uses differential evolution and a polynomial mutation operator to generate new solutions. Experimental results showed that MOEA/D-DE significantly outperformed the well-known Non-dominated Sorting Genetic Algorithm II (NSGA-II) [6], one of the most popular MOEAs, in addressing MOPs with complicated Pareto set shapes.

Yuan et al. [7] proposed the MOEA/D with a Distance-based Updating Strategy (MOEA/D-DU), which maintains the diversity of the population using the perpendicular distance of a solution to the weight vector in the objective space. The experimental results show that MOEA/D-DU is generally more effective than other MOEAs in balancing convergence and diversity, besides being very competitive for solving MaOPs.

Qi et al. [8] presented the MOEA/D with Adaptive Weight Adjustment (MOEA/D-AWA), which adopts a new initialization method and an adaptive weight adjustment strategy. The weight vectors are periodically adjusted to obtain better uniformity of solutions. An external population helps add new subproblems in sparse regions of the Pareto front. Experimental results indicated that MOEA/D-AWA outperformed four other MOEAs on ten widely used test MOPs, two complex MOPs, and two MaOPs regarding the Inverted Generational Distance (IGD) performance indicator [9], particularly in MOPs with complex Pareto fronts.

Although these MOEA/D variants stand out as representatives of the state of the art in decomposition-based MOEAs [3] and some works have compared them to other MOEAs, to the best of our knowledge, there is a gap in the literature that provides a comprehensive comparison among MOEA/D, MOEA/DD, MOEA/D-DE, MOEA/D-DU, and MOEA/D-AWA. Therefore, this study aims to evaluate the performance of these algorithms on MOPs and MaOPs through numerical experiments conducted on benchmark problems from the DTLZ suite [10]. The statistical analysis of the results was executed on the Inverted Generational Distance Plus (IGD+) performance indicator [11] associated with the non-parametric Wilcoxon test to verify the existence of statistically significant differences between the obtained results. The Hypervolume performance indicator was also considered in the combined Pareto front, formed by all solutions obtained by each MOEA.

The remainder of the work is organized as follows: Section 2 describes the theoretical foundation of the problems and algorithms, while Section 3 presents the numerical experiments. Discussions and an analysis of the obtained results are provided in Section 4, and finally, Section 5 presents the conclusions and prospects for future research.

2 Methods

2.1 MOPs

The MOPs treated in this work with m objective functions can be written as:

$$\begin{array}{ll} \min & F(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \dots, f_m(\boldsymbol{x})) \\ s.t. & x_i \in (l_i, u_i) \quad \forall i = 1, \dots, n \\ & \boldsymbol{x} = (x_1, \dots, x_n) \in \mathbb{R}^n \end{array} \tag{1}$$

Given $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, we say that $\boldsymbol{x} \prec \boldsymbol{y}$ (read as $\boldsymbol{x}$ dominates $\boldsymbol{y}$) if $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{y})$, $\forall i = 1, \dots, n$, and there is some j integer between 1 and n such that $f_j(\boldsymbol{x}) < f_j(\boldsymbol{y})$. If $\boldsymbol{x} \nprec \boldsymbol{y}$ and $\boldsymbol{y} \nprec \boldsymbol{x}$, $\boldsymbol{x}$ and $\boldsymbol{y}$ are said to be non-dominated by each other. The set of Pareto optimal solutions is formed by non-dominated solutions that are not dominated by any other. The image of this set in the objective space is called the Pareto Front. When $m \geq 4$, MOPs are called MaOPs due to their complexity, which increases as the number of objectives rises. As a result, these problems receive special attention, including an emphasis on developing MOEAs with techniques specifically designed to address their challenges. Further details on MOPs and MOEAs can be found in Ref. [1].

2.2 MOEA/D and Its Variants

MOEA/D works by decomposing a MOP (Eq. (1)) into several single-objective optimization subproblems and optimizing them simultaneously. Consider $\boldsymbol{\lambda}^1, \dots, \boldsymbol{\lambda}^N$ a set of weight vectors and $\boldsymbol{z}^* = (z_i, \dots, z_m)$ a reference point, where z_i is the best value found so far for the objective function f_i. Using the Tchebycheff aggregate function, the objective function of the jth problem can be defined as

$$g(\boldsymbol{x}|\boldsymbol{\lambda}^j, \boldsymbol{z}^*) = max\{\lambda_i^j |f_i(\boldsymbol{x}) - z_i^*|\} \tag{2}$$

in which $\boldsymbol{\lambda}^j = (\lambda_1^j, \dots, \lambda_m^j)$, where $\lambda_i \geq 0$ with $i = 1, \dots, m$ and $\sum_{i=1}^m \lambda_i = 1$.

For each $\boldsymbol{\lambda}^j$, among the other weight vectors, those closest are considered its neighborhood. This way, the neighborhood of the jth subproblem will be defined by the subproblems that have their weight vector in the neighborhood of $\boldsymbol{\lambda}^j$. Thus, a population is formed with the best solution found for each subproblem (Eq. (2)), which will be used for the rest of the algorithm (reproduction and updating of solutions). Figure 1 illustrates an example of the distribution of MOEA/D with 6 weight vectors. Algorithm 1 displays the pseudocode of MOEA/D and a complete description of it can be found in Ref. [2].

MOEA/DD suggests a unified paradigm that combines dominance- and decomposition-based approaches to exploit the merits of balancing the convergence and diversity of the evolutionary process. MOEA/DD uses the method proposed by [13] to generate a set of weight vectors sampled from a unit simplex. Each weight vector in MOEA/DD defines a subproblem and simultaneously estimates a population's local density.

MOEA/D-DE combines MOEA/D with Differential Evolution (DE) and polynomial mutation for the reproduction of new candidate solutions, which works as follows: for each $r_1 = i$, two indices r_2 and r_3 are randomly selected from the population. After this selection, a solution $\bar{y}$ is generated from x^{r_1}, x^{r_2}, and x^{r_3} through DE. Then, mutation is applied to $\bar{y}$ with probability p_m to produce a new solution y. Mathematically, it is expressed as:

$$\bar{y}_k = \begin{cases} x_k^{r_1} + F \times \left(x_k^{r_2} - x_k^{r_3} \right) & \text{with probability } CR, \\ x_k^{r_1}, & \text{with probability } 1 - CR \end{cases} \tag{3}$$

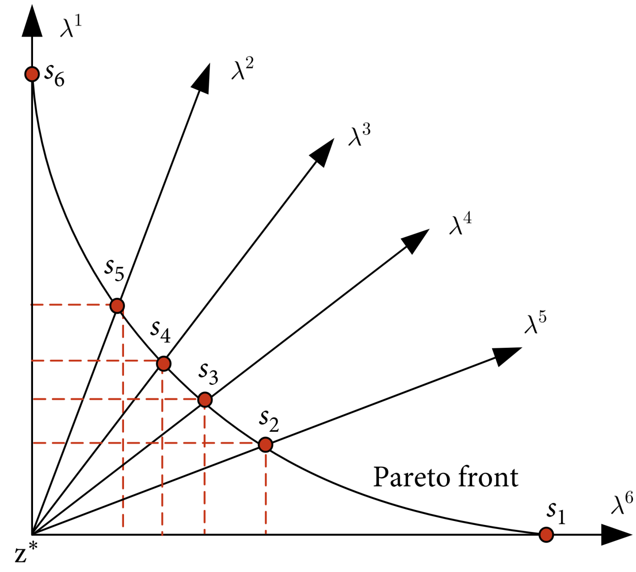


Figure 1: In MOEA/D, the aim is to minimize the distance from each point in the population to the ideal reference point z^*. Adapted from [12].

where CR and F are user-defined control parameters. Subsequently, polynomial mutation generates $y = (y_1, \dots, y_n)$ from $\bar{y}$ according to the following equation:

$$y_k = \begin{cases} \bar{y}_k + \sigma_k \times (b_k - a_k) & \text{with probability } p_m, \\ \bar{y}_k & \text{with probability } 1 - p_m \end{cases} \tag{4}$$

with

$$\sigma_k = \begin{cases} (2 \times r_4)^{\frac{1}{\eta+1}} - 1 & \text{if } r_4 < 0.5 \\ 1 - (2 - 2 \times r_4)^{\frac{1}{\eta+1}} & \text{otherwise} \end{cases} \tag{5}$$

where r_4 is a random number between 0 and 1. The distribution index η and the mutation rate p_m are user-defined control parameters. The values of a_k and b_k are the lower and upper bounds of the decision variable k, respectively.

While MOEA/D generally uses the Tchebycheff function as an aggregation function, MOEA/D-DU has as one of its main features a modified version of it. Let $\dot{\lambda}_j = \left(\lambda_{j,1}, \lambda_{j,2}, \dots, \lambda_{j,m}\right)^{\mathrm{T}}$, $j = 1, 2, \dots, N$, be a set of uniformly distributed weight vectors, and $\mathbf{z}^*$ be the ideal point, then the function for the j-th subproblem can be defined as:

$$\mathcal{F}_j(\mathbf{x}) = \max_{k=1}^{m} \left\{ \frac{1}{\lambda_{j,k}} \left| f_k(\mathbf{x}) - z_k^* \right| \right\} \tag{6}$$

where $\lambda_{j,k} \geq 0$ for every $k \in \{1, 2, \dots, m\}$ and $\sum_{k=1}^{m} \lambda_{j,k} = 1$.

This new aggregation function has two advantages over the original function. The first is that with uniformly distributed weight, vectors lead to directions of search evenly distributed in the objective space. Second, each weight vector corresponds to a unique solution located on the Pareto front. Due to these two advantages, the difficulty in preserving diversity is reduced.

Finally, MOEA/D-AWA aims to obtain an optimally uniformly distributed solution on the Pareto front of MOPs using MOEA/D and assigning appropriate weight vectors to scalar subproblems. To achieve this, MOEA/D-AWA uses a weight vector initialization method to generate a set of these vectors by applying the WS transformation to the original weight used in MOEA/D, defined as follows:

$$\lambda' = WS(\lambda) = \left(\frac{\frac{1}{\lambda_1}}{\sum_{i=1}^{m} \frac{1}{\lambda_i}}, \frac{\frac{1}{\lambda_2}}{\sum_{i=1}^{m} \frac{1}{\lambda_i}}, \cdots, \frac{\frac{1}{\lambda_m}}{\sum_{i=1}^{m} \frac{1}{\lambda_i}} \right). \tag{7}$$

The weight vectors generated by this initialization strategy will lead to a set of solution mapping vectors that are evenly distributed.

Input:
N: population size
$\lambda^1, \dots, \lambda^N$: weight vectors uniformly distributed
T: neighborhood size
G: maximum number of iterations
Output:
PE: external population
$PE = \emptyset$
for $i \neq j \leq N$ **do**
    Calculate the Euclidean distance between λ^i and λ^j
end for
for $i \leq N$ **do**
    Neighborhood of i, $B(i) = \{i_1, \dots, i_T\}$ where $\lambda^{i_1}, \dots, \lambda^{i_T}$ are the T weight vectors closest to λ^i
end for
Generate initial population $\{x^1, \dots, x^N\}$ randomly
Calculate the objective function $F(x^i) = (f_1(x^i), \dots, f_m(x^i))^T$ for each x^i in the initial population
Initialize $z = (z_1, \dots, z_m)^T$, where $z_i = \min\{f_i(x^1), \dots, f_i(x^N)\}$
while (number of iterations $< G$) **do**
    for $i \leq N$ **do**
        Select two indices k and l from $B(i)$
        Obtain a solution y from x^k and x^l using reproduction operators
        Apply a repair algorithm to y to produce y'
        for $j = 1, \dots, m$ **do**
            if $f_j(y') < z_j$ **then**
                $f_j(y') = z_j$
            end if
        end for
        for $j \in B(i)$ **do**
            if $g^{dec}(y'|\lambda^j, z) \leq g^{dec}(x^j|\lambda^j, z)$ **then**
                $x^j = y'$
                $F(x^j) = F(y')$
            end if
        end for
        Delete all vectors dominated by $F(y')$ from PE
        if no vector in PE dominates $F(y')$ **then**
            Add $F(y')$ to PE
        end if
    end for
end while
Return PE

Algorithm 1: MOEA/D.

The idea of MOEA/D-AWA is to apply a two-stage strategy to handle weight vector generation. In the first stage, a set of predetermined weight vectors is used until the population converges at a certain point. Then, a portion of weight vectors is adjusted according to the current Pareto optimal solutions based on geometric analysis. Specifically, some subproblems will be removed from the whole parts of the Pareto front, and some new subproblems will be created in other parts of it. Thus, the main differences between MOEA/D and MOEA/D-AWA are the weight vector initialization method and the update of weight vectors during the search procedure.

3 Numerical Experiments

3.1 DTLZ Suite Problems

Deb et al. [10] introduced the benchmark problem suite DTLZ, scalable to any number of decision variables and objectives. Among its features is the knowledge of the exact shape and location of the resulting Pareto optimal front. This makes it suitable for testing the MOEA's ability to control challenges in converging to the true Pareto optimal front, maintaining a widely distributed set of solutions. Detailed information about the DTLZ suite can be found in [10], and problems DTLZ (1-4) employed in this study are shown below:

3.1.1 DTLZ-1

$$
\begin{array}{ll}
\text{Minimize} & f_1(\mathbf{x}) = \frac{1}{2}x_1x_2\cdots x_{M-1}(1+g(\mathbf{x}_M)),\\
\text{Minimize} & f_2(\mathbf{x}) = \frac{1}{2}x_1x_2\cdots(1-x_{M-1})(1+g(\mathbf{x}_M)),\\
\vdots & \vdots\\
\text{Minimize} & f_{M-1}(\mathbf{x}) = \frac{1}{2}x_1(1-x_2)(1+g(\mathbf{x}_M)),\\
\text{Minimize} & f_M(\mathbf{x}) = \frac{1}{2}(1-x_1)(1+g(\mathbf{x}_M)),\\
\text{subject to} & 0 \le x_i \le 1, \quad \text{for } i = 1,2,\dots,n(1+g(\mathbf{x}_M)),\\
\text{with} & g(\mathbf{x}_M) = 100\left[|\mathbf{x}_M| + \sum_{x_i\in\mathbf{x}_M}(x_i-0.5)^2 - \cos(20\pi(x_i-0.5))\right]
\end{array}
\tag{8}
$$

3.1.2 DTLZ-2

$$
\begin{array}{ll}
\text{Min.} & f_1(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1\pi/2)\cdots\cos(x_{M-2}\pi/2)\cos(x_{M-1}\pi/2),\\
\text{Min.} & f_2(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1\pi/2)\cdots\cos(x_{M-2}\pi/2)\sin(x_{M-1}\pi/2),\\
\text{Min.} & f_3(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1\pi/2)\cdots\sin(x_{M-2}\pi/2),\\
\vdots & \vdots\\
\text{Min.} & f_M(\mathbf{x}) = (1+g(\mathbf{x}_M))\sin(x_1\pi/2),\\
\text{with} & g(\mathbf{x}_M) = \sum_{x_i\in\mathbf{x}_M}(x_i-0.5)^2,\\
 & 0 \le x_i \le 1, \quad \text{for } i = 1,2,\dots,n.
\end{array}
\tag{9}
$$

3.1.3 DTLZ-3

$$
\begin{array}{ll}
\text{Min.} & f_1(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1\pi/2)\cdots\cos(x_{M-2}\pi/2)\cos(x_{M-1}\pi/2),\\
\text{Min.} & f_2(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1\pi/2)\cdots\cos(x_{M-2}\pi/2)\sin(x_{M-1}\pi/2),\\
\text{Min.} & f_3(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1\pi/2)\cdots\sin(x_{M-2}\pi/2),\\
\vdots & \vdots\\
\text{Min.} & f_M(\mathbf{x}) = (1+g(\mathbf{x}_M))\sin(x_1\pi/2),\\
\text{with} & g(\mathbf{x}_M) = 100\left[|\mathbf{x}_M| + \sum_{x_i\in\mathbf{x}_M}(x_i-0.5)^2 - \cos(20\pi(x_i-0.5))\right],\\
 & 0 \le x_i \le 1, \quad \text{for } i = 1,2,\dots,n.
\end{array}
\tag{10}
$$

3.1.4 DTLZ-4

$$
\begin{array}{ll}
\text{Min.} & f_1(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1^\alpha\pi/2)\cdots\cos(x_{M-2}^\alpha\pi/2)\cos(x_{M-1}^\alpha\pi/2),\\
\text{Min.} & f_2(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1^\alpha\pi/2)\cdots\cos(x_{M-2}^\alpha\pi/2)\sin(x_{M-1}^\alpha\pi/2),\\
\text{Min.} & f_3(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(x_1^\alpha\pi/2)\cdots\sin(x_{M-2}^\alpha\pi/2),\\
\vdots & \vdots\\
\text{Min.} & f_M(\mathbf{x}) = (1+g(\mathbf{x}_M))\sin(x_1^\alpha\pi/2),\\
\text{with} & g(\mathbf{x}_M) = \sum_{x_i\in\mathbf{x}_M}(x_i-0.5)^2,\\
 & 0 \le x_i \le 1, \quad \text{for } i = 1,2,\dots,n.
\end{array}
\tag{11}
$$

3.2 Performance Indicators

The performance indicators Hypervolume and Inverted Generational Distance Plus (IGD+) are adopted here as mappings that assign scores to Pareto front approximations.

Given a point set $X \subset \mathbb{R}^d$ and a reference point $\boldsymbol{r} \in \mathbb{R}^d$, the Hypervolume indicator is

$$
H(X) = \lambda\left(\bigcup_{\boldsymbol{p}\in X}[\boldsymbol{p},\boldsymbol{r}]\right)
\tag{12}
$$

where $[\boldsymbol{p},\boldsymbol{r}] = \{\boldsymbol{q} \in \mathbb{R}^d \mid \boldsymbol{p} \prec \boldsymbol{q} \wedge \boldsymbol{q} \prec \boldsymbol{r}\}$ and $\lambda(\cdot)$ denotes the Lebesgue measure. Hypervolume was introduced as a tool for analyzing multiobjective optimization algorithms by Zitzler and Thiele [14]. It assesses the optimization process results by taking into account multiple aspects, such as the proximity of the solutions to the Pareto front, diversity, and spread.

Denoting the cardinality of a set Z by $|Z|$, the Inverted Generational Distance (IGD) indicator is defined as

$$\mathrm{IGD}(A) = \frac{1}{|Z|}\left(\sum_{j=1}^{|Z|} \hat{d}_j^p\right)^{1/p} \tag{13}$$

where $\hat{d}_j$ is the Euclidean distance from $\boldsymbol{z}_j$ to its nearest objective vector in A. The IGD Plus (IGD+) is the IGD indicator with the follow modified distance calculation:

$$d^+(\boldsymbol{z},\boldsymbol{a}) = \sqrt{d_1^{+2} + \cdots + d_m^{+2}} = \sqrt{(\max\{a_1 - z_i, 0\})^2 + \cdots + (\max\{a_m - z_m, 0\})^2}. \tag{14}$$

Note that the higher the Hypervolume value, better the MOEA performance. In IGD+, the lower the value, better the performance of the MOEA.

3.3 Results

In the computational experiments, it was considered DTLZ (1-4) (Equations 811). MOEA/D, MOEA/DD, MOEA/D-DE, MOEA/D-DU, and MOEA/D-AWA were executed 20 times each on MOPs DTLZ (1-4) with 3 objective functions, as well as on MaOPs DTLZ (1-4) with 5 objective functions. The experiments were conducted using PlatEMO, a Matlab framework containing optimization codes presented in [15]. Table 1 defines the populations, while Table 2 defines the generations. The means and standard deviations of the results obtained by the analyzed algorithms concerning IGD+, as well as the results of the non-parametric Wilcoxon test (with p-values ≤ 0.05), are shown in Table 3 to verify the existence of statistically significant differences between the results.

For a given MOP, the combined Pareto front of a MOEA is defined as the non-dominated solutions resulting from the union of Pareto sets obtained in each independent run. Figures 2 and 3 show the combined Pareto front of all MOEAs over the 20 independent runs and the corresponding Hypervolume percentages in relation to the biggest.

As MaOPs require effective methods for visualizing high-dimensional solution sets, 3 presents the combined Pareto front of MaOPs using a parallel coordinates plot. In this representation, N-dimensional data is illustrated through N equally spaced, parallel axes, symbolizing all objective functions. Each data point in this N-dimensional space, representing one of the Pareto optimal solutions, is delineated by a polyline intersecting each axis according to its corresponding value for that specific objective function. In Figure 3, the parallel coordinates have been normalized between 0 and 1 based on the minimum and maximum values of the objective function. Further information can be found in reference [16] for additional details on parallel coordinates.

Table 1: Population Size.

m	Population Size
3	91
5	210

Table 2: Number of Generations.

Problem	$m = 3$	$m = 5$
DTLZ-1	400	600
DTLZ-2	250	350
DTLZ-3	1,000	1,000
DTLZ-4	600	1,000

4 Results Analysis and Discussion

The MOEA/DD algorithm outperforms others in the IGD+ indicator among MaOPs. It has achieved the lowest values and shown statistically significant differences from other algorithms in almost all problems. The only exception occurred in DTLZ-2, where no statistically significant differences existed between the MOEA/DD and MOEA/D.

Table 3: Mean and Standard Deviation of IGD+ Values for DTLZ Problems. Values in bold are the best, while the symbol (+) indicates p-value ≤ 0.05 in the Wilcoxon test.

		MOEA/D	MOEA/DD	MOEA/D-DE	MOEA/D-DU	MOEA/D-AWA
		MOPs (3 Objective Functions)				
DTLZ-1	M	0.0031(+)	**0.0022**	0.0737(+)	0.0023(+)	0.0048(+)
	SD	0.0010	0.0007	0.1396	0.0005	0.0018
DTLZ-2	M	0.0050(+)	0.0051(+)	0.0307(+)	**0.0046**	0.0049
	SD	0.0000	0.0001	0.0011	0.0002	0.0010
DTLZ-3	M	0.0070(+)	0.0056	0.5176(+)	0.0056	**0.0055**
	SD	0.0019	0.0006	2.1497	0.0009	0.0010
DTLZ-4	M	0.0832(+)	0.0148(+)	0.0359(+)	**0.0037**	0.0148(+)
	SD	0.1376	0.0491	0.0103	0.0001	0.0480
		MaOPs (5 Objective Functions)				
DTLZ-1	M	0.0012(+)	**0.0007**	0.2210(+)	0.0008(+)	0.0021(+)
	SD	0.0007	0.0001	0.1950	0.0002	0.0005
DTLZ-2	M	0.0076	**0.0075**	0.0884(+)	0.0084(+)	0.0326(+)
	SD	0.0001	0.0000	0.0017	0.0002	0.0036
DTLZ-3	M	0.0056(+)	**0.0050**	1.4687(+)	0.0055(+)	0.0294(+)
	SD	0.0011	0.0004	2.1182	0.0005	0.0064
DTLZ-4	M	0.0704(+)	**0.0077**	0.1063(+)	0.0079(+)	0.0444(+)
	SD	0.0777	0.0000	0.0071	0.0001	0.0702

In MOPs with 3 objective functions, the performance of different MOEAs depends on the problem. For instance, in DTLZ-1, the MOEA/DD algorithm showed the best result with statistically significant differences compared to other MOEAs. In DTLZ (2 and 4), the MOEA/D-DU algorithm demonstrated superior performance with statistically significant differences compared to other MOEAs, except for MOEA/D-AWA in the DTLZ-2. In DTLZ-3, the MOEA/D-AWA algorithm emerged as the best performer, showing statistically significant differences only when compared to MOEA/D and MOEA/D-DE.

Regarding IGD+, MOEA/DD stands out as the top-performing among the considered problems. It demonstrated superior performance in MaOPs and DTLZ-1 with 3 objective functions. Moreover, it was competitive with the best-performing MOEA in DTLZ-3 with 3 objective functions, with no statistically significant difference.

On the other hand, MOEA/D-AWA obtained the highest Hypervolume in all evaluated problems for combined Pareto fronts, showing its effectiveness as the best MOEA when combining the solutions obtained in all 20 independent runs.

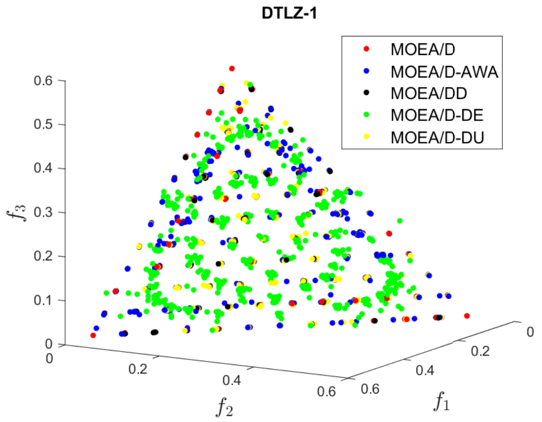
Conversely, MOEA/D-DE was identified as the least effective algorithm. It showcases the lowest IGD+ on almost all problems (except for DTLZ-4 with 3 objective functions, where MOEA/D obtains the worst value) and Hypervolume values (except for MaOP DTLZ-3, where all MOEAs achieve the same Hypervolume) and exhibits statistically significant differences from the best-performing algorithm in all MOPs and MaOPs.

5 Conclusion

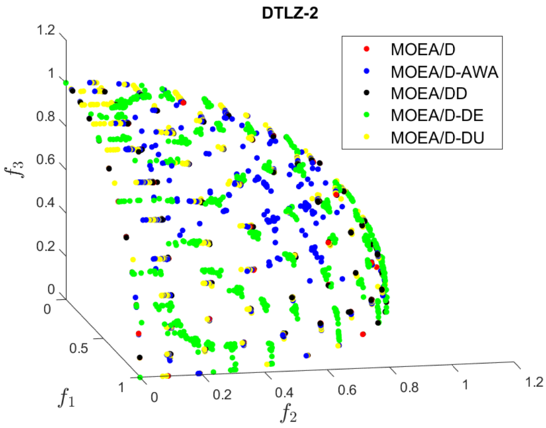
This study analyzed the performance of MOEA/D and some of its variants (MOEA/DD, MOEA/D-DE, MOEA/D-DU, and MOEA/D-AWA) in MOPs and MaOPs. Computational experiments were conducted on benchmark problems from the DTLZ family with 3 and 5 objective functions. Statistical analysis was carried out using the Wilcoxon test on the results obtained in the IGD+ performance indicator, and the Hypervolume performance indicator was considered in the combined Pareto front.

The experiments revealed that MOEA/DD performed best in IGD+, and MOEA/D-AWA achieved the highest Hypervolume in the combined Pareto front, while MOEA/D-DE registered the worst result in this set of problems.

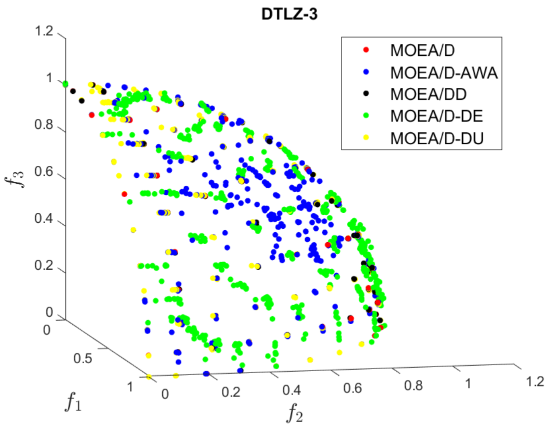
In future research, exploring hybridizations that involve MOEA/DD and MOEA/D-AWA with other algorithms, as exemplified in Ref. [17], can be worthy of investigation. That study proposed an algorithm named MOEA/D-IWOA, a hybridization of MOEA/D with the IWOA algorithm. Another potential option for hybridization is the Sine



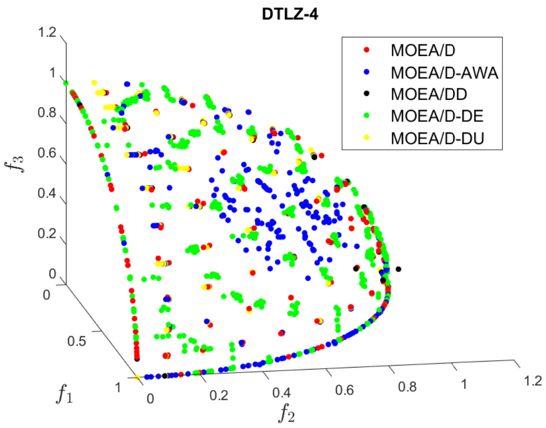
(a) 0.9891; **1**; 0.9903; 0.9626; 0.9966



(b) 0.9615; **1**; 0.9650; 0.9353; 0.9821



(c) 0.9598; **1**; 0.9614; 0.9325; 0.9676



(d) 0.9813; **1**; 0.9604; 0.9572; 0.9654

Figure 2: Combined Pareto front of all MOEAs concerning each MOP (DTLZ 1-4 with 3 objective functions). The corresponding Hypervolume percentages in comparison to the largest are provided in the caption, listed in the order of MOEA/D, MOEA/D-AWA, MOEA/DD, MOEA/D-DE, and MOEA/D-DU.

Cosine Algorithm (SCA), proposed in Ref. [18]. Gabis et al. [19] indicates SCA for tackling MaOPs and utilizing a decomposition-based MOEA may be a promising option.

Acknowledgements.
The authors thank the Federal Center for Technological Education of Minas Gerais (CEFET-MG) and FAPEMIG (APQ-00408-21) for their support.

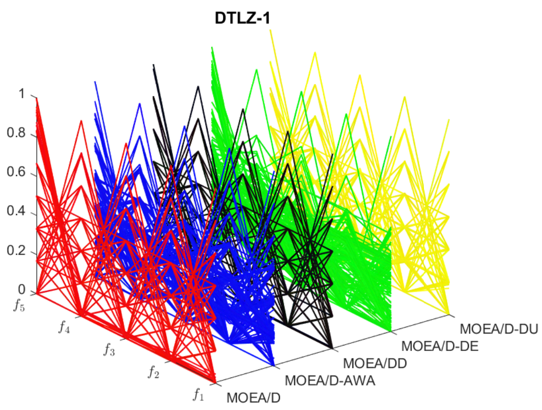
References

[1] K. Deb, *Multiobjective Evolutionary Algorithms*. John Wiley & Sons, 2001.

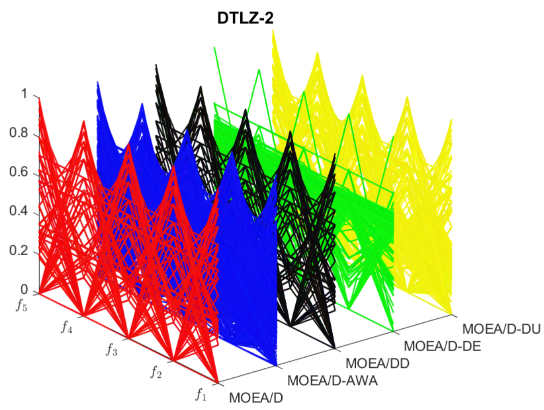
[2] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007. Available at: https://doi.org/10.1109/TEVC.2007.892759

[3] Q. Xu, Z. Xu, and T. Ma, "A survey of multiobjective evolutionary algorithms based on decomposition: variants, challenges and future directions," *IEEE Access*, vol. 8, pp. 41 588–41 614, 2020. Available at: https://doi.org/10.1109/ACCESS.2020.2973670

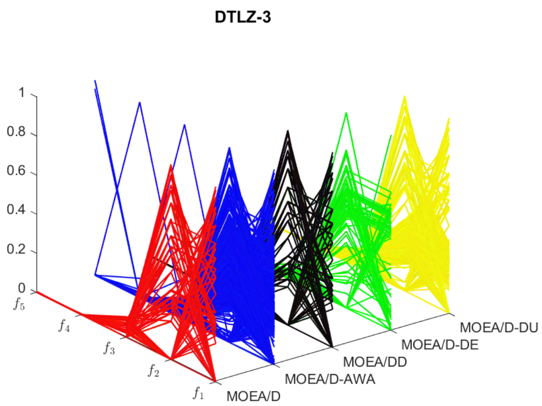
[4] K. Li, K. Deb, Q. Zhang, and S. Kwong, "An evolutionary many-objective optimization algorithm based on dominance and decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 694–716, 2015. Available at: https://doi.org/10.1109/TEVC.2014.2373386



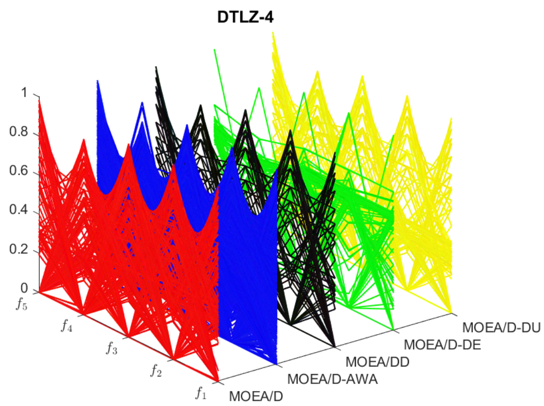
(a) 0.9974; **1**; 0.9974; 0.9544; 0.9980



(b) 0.9461; **1**; 0.9458; 0.7642; 0.9661



(c) **1**; **1**; **1**; **1**; **1**



(d) 0.9609; **1**; 0.9361; 0.7870; 0.9397

Figure 3: Parallel coordinates normalized between 0 and 1 of the combined Pareto front of all MOEAs concerning each MaOP (DTLZ 1-4 with 5 objective functions). The corresponding Hypervolume percentages in comparison to the largest are provided in the caption, listed in the order of MOEA/D, MOEA/D-AWA, MOEA/DD, MOEA/D-DE, and MOEA/D-DU.

[5] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 284–302, 2009. Available at: https://doi.org/10.1109/TEVC.2008.925798

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

[7] Y. Yuan, H. Xu, B. Wang, B. Zhang, and X. Yao, "Balancing convergence and diversity in decomposition-based many-objective optimizers," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 2, pp. 180–198, 2016. Available at: https://doi.org/10.1109/TEVC.2015.2443001

[8] Y. Qi, X. Ma, F. Liu, L. Jiao, J. Sun, and J. Wu, "MOEA/D with adaptive weight adjustment," *Evolutionary Computation*, vol. 22, no. 2, pp. 231–264, 2014. Available at: https://doi.org/10.1162/EVCO_a_00109

[9] M. R. Sierra and C. A. Coello Coello, "Improving PSO-based multi-objective optimization using crowding, mutation and e-dominance," in *Proceedings of the International Conference on Evolutionary Multi-criterion Optimization*. Springer, 2005, pp. 505–519. Available at: https://doi.org/10.1007/978-3-540-31880-4_35

[10] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable multi-objective optimization test problems," in *Proceedings of the 2002 Congress on Evolutionary Computation (CEC)*, vol. 1. IEEE, 2002, pp. 825–830. Available at: https://doi.org/10.1109/CEC.2002.1007032

[11] H. Ishibuchi, H. Masuda, and Y. Nojima, "Sensitivity of performance evaluation results by inverted generational distance to reference points," in *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC)*, 2016, pp. 1107–1114. Available at: https://doi.org/10.1109/CEC.2016.7743912

[12] G.-Z. Fu, T. Yu, W. Li, Q. Deng, and B. Yang, "A decomposition-based multiobjective optimization evolutionary algorithm with adaptive weight generation strategy," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–12, 2021. Available at: https://doi.org/10.1155/2021/2764558

[13] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems," *SIAM J. on Optimization*, vol. 8, no. 3, pp. 631–657, 1998. Available at: http://dx.doi.org/10.1137/S1052623496307510

[14] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999. Available at: https://doi.org/10.1109/4235.797969

[15] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization," *IEEE Computational Intelligence Magazine*, vol. 12, no. 4, pp. 73–87, 2017. Available at: https://doi.org/10.1109/MCI.2017.2742868

[16] M. Li, L. Zhen, and X. Yao, "How to read many-objective solution sets in parallel coordinates," *IEEE Computational Intelligence Magazine*, vol. 12, no. 4, pp. 88–100, 2017. Available at: https://doi.org/10.1109/MCI.2017.2742869

[17] A. O. Martins, M. C. C. Peito, D. E. C. Vargas, and E. F. Wanner, "Development of a bio-inspired hybrid decomposition algorithm based on whale and differential evolution strategies for multiobjective optimization," *VETOR*, vol. 33, no. 1, pp. 13–24, 2023. Available at: https://doi.org/10.14295/vetor.v33i1.15567

[18] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-based Systems*, vol. 96, pp. 120–133, 2016. Available at: https://doi.org/10.1016/j.knosys.2015.12.022

[19] A. B. Gabis, Y. Meraihi, S. Mirjalili, and A. Ramdane-Cherif, "A comprehensive survey of sine cosine algorithm: variants and applications," *Artificial Intelligence Review*, vol. 54, no. 7, pp. 5469–5540, 2021. Available at: https://doi.org/10.1007/s10462-021-10026-y