Annals of Operations Research. Special Issue manuscript No.
(will be inserted by the editor)

Multiobjective Variable Mesh Optimization

Yamisleydi Salgueiro · Jorge L. Toro · Rafael Bello · Rafael Falcon

Received: date / Accepted: date

Abstract In this article we introduce a new multiobjective optimizer based on a recently proposed metaheuristic algorithm named Variable Mesh Optimization (VMO). Our proposal (multiobjective VMO, MOVMO) combines typical concepts from the multiobjective optimization arena such as Pareto dominance, density estimation and external archive storage. MOVMO also features a crossover operator between local and global optima as well as dynamic population replacement. We evaluated MOVMO using a suite of four well-known benchmark function families, and against seven state-of-the-art optimizers: NSGA-II, SPEA2, MOCell, AbYSS, SMPSO, MOEA/D and MOEA/D.DRA. The statistically validated results across the *additive epsilon*, *spread* and *hypervolume* quality indicators confirm that MOVMO is indeed a competitive and effective method for multiobjective optimization of numerical spaces.

Keywords Multi-objective Optimization · Evolutionary Computation · Variable Mesh Optimization · Meta-heuristic Optimization

1 Introduction

Saving resources, time and obtaining products with the highest quality are common objectives to attain in the industrial and business environments. The same can

Yamisleydi Salgueiro
Informatics Department, University of Las Tunas, Cuba
E-mail: yamisalgueiro@gmail.com

Jorge L. Toro
Informatics Department, University of Las Tunas, Cuba
E-mail: jorgitoltp@gmail.com

Rafael Bello
Computer Science Department, Central University of Las Villas, Cuba
E-mail: rbellop@uclv.edu.cu

Rafael Falcon
Electrical Engineering and Computer Science, University of Ottawa, Canada
E-mail: rfalcon@uottawa.ca

be extrapolated to daily life decisions. The decision making processes is usually governed by the simultaneous consideration of several conflictive goals. This kind of problems are known as *multiobjective optimization problems* (MOP), where there is often no single optimal solution but rather a set of alternatives with different advantages/limitations trade-offs. In the multiobjective optimization literature, these are referred to as *Pareto optimal solutions* or *non-dominated solutions* [1].

Obtaining the complete *Pareto optimal set* (PS), i.e., the set of Pareto optimal solutions, can be a computationally expensive endeavor and, for vast search spaces, the cost could turn out prohibitive. Due to their parallel nature, *population-based metaheuristic algorithms*, in particular *evolutionary algorithms* (EAs), are able to approximate the entire Pareto optimal set of a MOP in a single run [23]. Additionally, population-based EAs have also been successfully applied in *single-objective optimization problems* (SOPs) given their proved ability to explore the search space in an inherently concurrent fashion, which results in a higher computational efficiency. Therefore, it becomes a quite natural proposition to develop multiobjective versions of these population-based optimizers, as they could simultaneously identify several non-dominated solutions in a single iteration and characterize the trade-off among their objectives [2].

Variable Mesh Optimization (VMO) is a population-based metaheuristic algorithm created by Puris et al [19] in 2012. It showed competitive results compared with improved versions of Genetic Algorithms (GA), Particle Swarm Optimization (PSO) and Differential Evolution (DE) [19]. More recently, VMO has been augmented with niching methods to embrace the multimodal optimization realm [11] [13] [12] and promising results have been obtained. Therefore, formulating a multiobjective VMO implementation is the next step towards the scientific maturity and development of this metaheuristic optimization scheme.

This research work makes the following contributions: (1) we formalize a multiobjective optimization extension to VMO, termed as *MOVMO*, and dissect its algorithmic building blocks; (2) we conduct an experimental analysis between MOVMO and seven state-of-the-art multiobjective optimizers, namely NSGA-II, SPEA2, MOCell, AbYSS, SMPSO, MOEA/D and MOEA/D.DRA; (3) we demonstrate via statistical validation the benefits of the proposed MOVMO technique.

The remaining of this paper is organized as follows: Section 2 describes some basic concepts pertaining to the domain of multiobjective optimization (MOO). Sections 3 and 4 unveil the main elements of the canonical VMO formulation and our multiobjective extension, respectively. The experimental framework is presented in Section 5 while Section 6 is devoted to the statistical analysis of the empirical results. Conclusions are outlined in Section 7.

2 Basic Concepts

A multiobjective optimization problem can be mathematically defined as:

$$\begin{array}{l} \text{minimize } \; F(x) = (f_1(x), \ldots, f_m(x))^T \\ \text{subject to } x \in \Omega \end{array} \tag{1}$$

where Ω is the (non-empty) decision space and $x \in \Omega$ is the decision vector. $F(x)$ consists of $m \geq 2$ conflicting objective functions $f_i : \Omega \to \mathbb{R}, i = 1, \ldots, m$ where $\mathbb{R}^m$ is the objective space. Notice that maximizing f_i implies minimizing -f_i.

In single-objective optimization problems it is easy to determine the best solution in a population, as this is the solution with the best objective function value. Nevertheless in MOPs this process is not trivial, as there is no longer a unique best solution but instead a set of equally attractive solutions. In such a set, some solutions improve one or more objectives yet exhibit not-so-great values in the rest of the objectives. These solutions can be formalized using the following definitions:

Definition 1 A vector $u = (u_1, \ldots, u_m)^T$ *strongly dominates* another vector $v = (v_1, \ldots, v_m)^T$, denoted as $u \prec v$, iff $\forall i \in \{1, \ldots, m\}, u_i < v_i$.

Definition 2 A vector $u = (u_1, \ldots, u_m)^T$ *weakly dominates* another vector $v = (v_1, \ldots, v_m)^T$, denoted as $u \preceq v$, iff $\forall i \in \{1, \ldots, m\}, u_i \leq v_i$ and $\exists j \in \{1, \ldots, m\}$ such that $u_j < v_j$.

Definition 3 A feasible solution $x^* \in \Omega$ of Equation (1) is called a *Pareto optimal solution*, iff $\nexists y \in \Omega$ such that $y \preceq x^*$. The set of all the Pareto optimal solutions is called the *Pareto set* (PS), denoted as: $PS = \{x^* \in \Omega \mid \nexists y \in \Omega, y \preceq x^*\}$.

Definition 4 The image of the Pareto set in the objective space is called the *Pareto front* (PF): $PF = \{F(x) \mid x \in PS\}$.

Population-based multiobjective optimization methods usually do not guarantee to identify the optimal solution but instead, they aim at locating a good Pareto set that approximates the true Pareto set as much as possible. We are therefore interested in the algorithmic technique that provides the best approximation to the true Pareto set for a given problem. The comparison among the Pareto sets yielded by different multiobjective optimizers is conducted on the basis of *quality indicators* that act as quantitative manifestations of performance indices. In this way, the strengths and weaknesses of the competing approaches are exposed across different quality indicators.

Definition 5 An m-ary *quality indicator* I is a function $I : \Psi^m \to \mathbb{R}$, which maps m sets $A_1, A_2 \ldots, A_m \in \Psi$ to a real value $\mathbb{R}$.

A quality indicator maps each Pareto set approximation to a real number; the underlying idea is to quantify the differences in quality among multiple approximation sets [25]. Quality indicators are often reflective of the decision maker's preferences.

In this study, we will evaluate the different MOO optimizers according to three widely used quality indicators: *Epsilon*, *Spread*, and *Hypervolume*. The epsilon indicator family was introduced in [8]. Given a computed approximation front A, this indicator is a measure of the smallest distance that one would need to translate every solution in A so that it dominates the Pareto optimal front of this problem. *Spread*-based indicators measure the distribution of the individuals in the Pareto front. The spread technique used in this article was proposed in [4] and gauges the non-uniformities in the distribution. Similarly to the epsilon indicator, higher values corresponds to worse behavior. *Hypervolume* indicator, which was introduced in [26], calculates the volume of the region (in the objective function space) covered by members of a non-dominated set of solutions w.r.t. a reference point.

3 Variable Mesh Optimization

Variable Mesh Optimization (VMO) [19], is a population-based metaheuristic technique in which the set of solutions (or population) represents a mesh in a multidimensional space. This mesh is composed of P nodes $(n_1, \ldots, n_p)$, each denoting a solution to the underlying optimization problem in an M-dimensional search space. Each node n_i is encoded as a vector of M real-valued numbers $n_i = (v_1^i, \ldots, v_M^i);\ i \in \{1, \ldots, P\}$.

The algorithmic parameters are:

1. the population size P
2. the maximum number of nodes T generated during the mesh expansion process
3. the size k of each mesh node's neighborhood
4. the maximum number of objective function evaluations C (used as the stop criterion)

The search process undertaken by VMO can be summarized in three main phases. First, the mesh is *initialized* (either randomly or following an inexpensive heuristic method) with P nodes along the search space. Afterwards, the mesh goes through an *expansion* procedure characterized by the generation of other mesh nodes towards the local optima, the global optimum and the frontier of the explored space. At this point the mesh has grown from the initial P nodes to $P < T \leq 3P$ nodes at the most. The next phase is the mesh *contraction*, during which poor solutions as well as those clustered around local optima are removed from the mesh. The expansion and contraction phases alternate in each iteration until the stop condition is met. Below is the VMO algorithmic workflow.

- **Step 1.** [**Mesh initialization**] Generate P nodes for the initial mesh and select among them the global best n_g.
- **Step 2.** [**Mesh expansion.1**] For each mesh node n_i, find its closest k nodes (in the decision space), then select the best neighbor n_i^* in the objective space. If n_i is not the local best, then generate a new node n_z toward the local best by Equation (2). With this step, Z new nodes are created, where $Z < P$.
- **Step 3.** [**Mesh expansion.2**] For each mesh node n_i but the global best n_g, generate a new node n_x toward the global best by Equation (2). With this step, X new nodes are created, where $X \leq P - 1$.

$$n_z = F(n_i, n_i^*, Pr(n_i, n_i^*)) \tag{2}$$

where Pr is called the *proximity factor* and represents the relationship between the objective function value (fitness) of the current node and that of its local/global optimum. This proximity factor is calculated by Equation (3):

$$Pr(n_i, n_i^*) = \frac{1}{1 + |\text{fitness}(n_i) - \text{fitness}(n_i^*)|} \tag{3}$$

The function F generates the coordinate v_j^z of the new node n_z along the j-th dimension and can be described by Equation (4):

$$v_j^z = \begin{cases} \overline{m_j} & \text{if } \left|\overline{m_j} - v_j^{i^*}\right| > \xi_j \wedge \\ & \text{rand1()} \le Pr(n_i, n_i^*) \\ v_j^{i^*} + \text{rand2()} & \text{if } \left|\overline{m_j} - v_j^{i^*}\right| \le \xi_j \\ \text{rand3()} & \text{otherwise} \end{cases} \quad (4)$$

where $\overline{m_j}$ is the midpoint between v_j^i and $v_j^{i^*}$, rand1() $\sim U(0,1)$, rand2() $\sim U(-\xi_j, \xi_j)$ and rand3() $\sim U(v_j^i, \overline{m_j})$. The notation $U(a,b)$ denotes a uniformly distributed random number in the interval $[a,b]$ and ξ_j defines the minimum allowed distance for each dimension. These values decrease monotonically during the run of the algorithm, as depicted by Equation (5):

$$\xi_j = \begin{cases} \frac{\max_j - \min_j}{4} & \text{if } c < 0.15C \\ \frac{\max_j - \min_j}{8} & \text{if } 0.15C \le c < 0.30C \\ \frac{\max_j - \min_j}{16} & \text{if } 0.30C \le c < 0.60C \\ \frac{\max_j - \min_j}{50} & \text{if } 0.60C \le c < 0.80C \\ \frac{\max_j - \min_j}{100} & \text{if } c \ge 0.80C \end{cases} \quad (5)$$

where C and c are the maximum and current number of objective function evaluations, respectively, and $\max_j$, $\min_j$ denote the optimization bounds of the j-th dimension, $j \in \{1, \ldots, M\}$

- **Step 4.** [**Mesh expansion.3**] Generate nodes from those in the mesh frontier as described in [19].
- **Step 5.** [**Mesh contraction.1**] Sort nodes according to their fitness values.
- **Step 6.** [**Mesh contraction.2**] Apply the adaptive clearing operator, i.e., remove from the mesh those nodes that are too close to each other in the decision space.
- **Step 7.** [**Mesh contraction.3**] Select the P best nodes to build the initial mesh for the next iteration. If needed, randomly generate new nodes to complete the initial mesh with P nodes for the next iteration.

4 Multi-Objective Variable Mesh Optimization

In this section we describe our proposed VMO extension termed *Multiobjective Variable Mesh Optimization* (MOVMO). The new algorithm largely follows the canonical VMO search strategy outlined in Section 3, including the dynamic population replacement. Because MOVMO is aimed at tackling multiobjective optimization problems, several traditional algorithmic underpinnings in this field are incorporated to its design, namely *Pareto dominance*, *objective space density estimation* (via crowding distance) and *elitism* in the form of an external archive storing the set of discovered non-dominated solutions.

Algorithm 1 displays MOVMO's general workflow. The input parameters P, k and C are just like in VMO (refer to Section 3). Yet the T parameter (maximum mesh size after the expansion procedure) in VMO is no longer necessary in this multiobjective formulation. A new parameter S, the maximum size of the leaders (Pareto) archive is introduced instead.

The method starts by generating the initial mesh $\mathcal{M}_0$ and initializing the leaders archive L with all the non-dominated solutions in $\mathcal{M}_0$. This archive will only store non-dominated solutions, (the Pareto optimal approximation found so far).

Algorithm 1: Multiobjective Variable Mesh Optimization (MOVMO)

Input: mesh size P, neighborhood size k, max. obj. func. eval. C, max. archive size S
Output: An approximation L of the true Pareto set L^*
1 Generate the initial mesh $\mathcal{M} = \{n_i\}$ and calculate the fitness vector F_i $\forall i \in \{1, \ldots, P\}$
2 Initialize the leaders archive L with each mesh node n_i by Algorithm 2
3 $c \leftarrow 1$
4 **while** $c \leq C$ **do**
5   **foreach** *node* n_i *in the current mesh* $\mathcal{M}$ **do**
6         $n_i^* \leftarrow$ the best among the k neighbors of n_i
7   **if** $n_i^* \prec n_i$ *(see Definition 1)* **then**
8             $n_l \leftarrow$ generate a new node by Equations (2) and (6)
9   **else**
10             $n_l \leftarrow n_i$
11         $n_g \leftarrow$ apply binary tournament to select a global leader from L
12         $n_x \leftarrow$ crossover(n_l, n_g)
13   evaluateFitness(n_x)
14   add n_x to the leaders archive L (see Algorithm 2)
15   **if** $n_x \preceq n_i$ *(Definition 2)* **then**
16   Replace n_i with n_x in the current population
17         $c \leftarrow c + 1$
18 **return** L

$$Pr(n_i, n_i^*) = \frac{1}{1 + \sqrt{\sum_{j=1}^{m} \left(f_j(n_i) - f_j(n_i^*)\right)^2}} \tag{6}$$

For each node n_i of the current mesh $\mathcal{M}$, the following steps are carried out:

1. The best node n_i^* among n_i's k nearest neighbors in the decision variable space is selected (Line 6) according to the dominance criterion described in Section 2. If two or more of these k neighbors are mutually non-dominated, the closest to n_i is picked.
2. If the local optimum dominates n_i, a new node n_l is generated in that direction by Equations (2) and (6). (Line 8); otherwise n_i is the local optimum itself (Line 9). Notice that Equation (6) measures the Euclidean distance between the two mesh nodes n_i and n_i^* in the objective space and crafts a probability accordingly.
3. A global leader n_g from the archive L is selected through binary tournament (Line 11). Two non-dominated solutions from L are arbitrarily picked and the one with largest crowding distance in L wins the tournament.
4. This elite solution n_g is crossed over with the local optimum n_l from the preceding step (Line 12). The best of the two offspring (in terms of dominance, crowding distance to break ties) is retained.
5. After that, the resulting offspring node n_x is evaluated and added to the leaders archive L (Line 14).

6. Finally, if n_i is weakly dominated by n_x, it will be replaced with n_x in the current mesh (Line 15).

MOVMO returns the leaders archive L as the approximation of the Pareto optimal set found.

Algorithm 2 describes the addition of a new mesh node n_x to the bounded leader archive L. First, all nodes in L that are dominated by the incoming solution are deleted from the archive prior to n_x's addition. If the archive reached its maximum size, we drop the node with the lowest crowding distance. This ensures that a well-spread set of non-dominated solutions is maintained in L.

Algorithm 2: Add the n_x solution to the leader archive L

foreach n_j *of* L **do**
    if $n_x \prec n_j$ *(Definition 1)* **then**
        $L \leftarrow L - \{n_j\}$; `/* remove` n_j `from the archive */`
    else if $n_x = n_j \mid\mid n_j \preceq n_x$ *(Definition 2)* **then**
        exit ; `/* discard` n_x `*/`
$L \leftarrow L \cup \{n_x\}$; `/* add` n_x `to the archive */`
if $L.size() > L.maxSize()$ **then**
    recompute crowding distances in L
    $L \leftarrow L - \{L.\text{worstByCrowdingDistance}\}$; `/* remove most crowded solution */`

The proposed changes to the classical VMO workflow featured by MOVMO are aimed at efficiently tackling multiobjective optimization problems. For instance, MOVMO maintains a leader archive with a set of well-distributed non-dominated solutions (global leaders) instead of the global best recorded by VMO. Another example is the substitution of the T-node augmented mesh generation in VMO with a dynamical node replacement mechanism. The rationale for this decision was motivated by the large computational effort needed for merging and sorting two sets (the current mesh of size P and the expanded T-node mesh) in a multiobjective environment. Moreover, the crowding distance measure applied to the non-dominated solutions in L plays the role of VMO's adaptive clearing operator used in decluttering the heavily explored portions of the decision space.

5 Experimental Framework

Over the years, several studies [8] [5] [24] have suggested how to design an experimental framework for the validation of new multiobjective optimization techniques. In this research work, we confine ourselves to the guidelines put forth by K. Deb in [1].

5.1 Test Functions

The purpose of test functions is to reproduce some of the complications that a new MOO algorithm could encounter in real-world problems. The underlying assumption is that should the new algorithm behave well with these functions, it is reasonable to expect that it will yield similar results when exploring a real-world problem whose optimization landscape resembles that of the test function(s).

We have chosen four function suites for benchmarking purposes: ZDT [27], DTLZ [3], WFG[7] and LZ09 [9]. They include convex, non-convex, multimodal, non-uniformly spaced, multiple Pareto fronts, among others complex fitness landscapes.

- The **ZDT** family contains six functions; ZDT1 and ZDT4 are convex meanwhile ZDT3 features a disconnected Pareto front. ZDT6 tests for an optimizer's ability to deal with multimodality. The remaining ZDT2 and ZDT6 are non-convex. In all cases the problems' dimension vary from 10 to 30, and all functions are bi-objectives.
- The **DTLZ** family contains seven functions; in its three-objective version, DTLZ1 and DTLZ3 are multimodal as well as DTLZ7, which also exhibits a disconnected Pareto front. The remaining DTLZ2, DTLZ4, DTLZ5 and DTLZ6 are non-convex. In all cases, the problems' dimension range from 7 to 22.
- The **WFG** family [7] consists of nine scalable, multiobjective test problems (WFG1 - WFG9). WFG1 is convex, WFG2 is convex and disconnected. WFG3 is linear. WFG4 to WFG9 are non-convex while WFG4 and WFG9 are multimodal. In all cases the problems' dimension is 6.
- The **LZ09** [9] is a more recent family and consists of nine test functions, all of them are bi-objective except F6, which has three objectives. LZ09 is a general class of continuous MOO test instances with arbitrary prescribed Pareto set shapes. F9 has a non-convex Pareto front. In all cases the problems' dimension vary from 10 to 30.

5.2 Comparison with State-of-the-Art Algorithms

In this section we briefly describe the seven state-of-the-art MOO methods that have been selected as MOVMO competitors. They are very popular algorithms in this arena and have enjoyed well-deserved recognition due to their high performance and efficiency in solving entangled problem instances of different sorts. These techniques belong to the Genetic Algorithms, Scatter Search, Particle Swarm Optimization and Decomposition-based Multiobjective Optimizer families.

- **MOEA/D** was initially proposed in [22] as a decomposition-based multiobjective optimizer. MOEA/D breaks down a MOP into a set of single-objective problems (SOPs) with neighborhood relationship and approximates the Pareto set by solving these SOPs. In out experiments, we used the MOEA/D version put forth in [9].
- **MOEA/D.DRA** (MOEA/D with Dynamic Resource Allocation) [21] is a version of the previous algorithm that defines and computes a utility value for each subproblem. Computational efforts are distributed to these subproblems based on their utilities.

- The **NSGA-II** algorithm [4] was proposed by Deb et al in 2002 as an improvement over an existing genetic-algorithm-based technique for MOPs. It uses two solutions sets: one for the current population and the other one for the offspring population. Selection, crossover, and mutations operators are employed in each generation to create an offspring population of the same size as the current population. Then, both populations are merged. Fast non-dominated sorting from the resulting joint population and the subsequent crowding-distance-based comparison procedure ensure elitism and diversity in the population that survives to the next generation. Originally, NSGA-II does not include a leader archive for storing the best discovered solutions. Instead, the last population contains the final approximation to the Pareto optimal set.
- **SPEA2** [26] was proposed by Zitzler et al. in 2001. This genetic-algorithm-based technique relies on two solutions sets: one for the current population and another one for an archive. Selection, crossover, and mutations operators are employed to fill an archive; then, the non-dominated individuals of both the original population and the archive are copied into a new population. If the number of non-dominated individuals exceeds the population size, a truncation operator based on density estimation is applied.
- **MOCell** [15] is a cellular genetic algorithm designed by Nebro et al in 2009. It includes an external archive to store the non-dominated solutions found so far. This archive makes use of NSGA-II's crowding distance to maintain diversity in the population. During the algorithm execution, a number of solutions are moved back into the population from the archive after each generation, thus replacing randomly selected population members.
- **AbYSS** [16] is an adaptation of the Scatter Search metaheuristic to the multiobjective domain put forth by Nebro et al in 2008. This algorithm uses an external archive similar to the one employed by MOCell. The algorithm borrows operators from the evolutionary realm, including polynomial mutation and simulated binary crossover into the improvement and solution combination methods, respectively.
- **SMPSO** [14] is a multiobjective PSO algorithm which utilizes polynomial mutation as a turbulence factor and an external archive to store the non-dominated solutions found during the search. It also includes a velocity constriction equation, instead of using the upper and lower bounds of each dimension to clamp the velocity of the particles.

As can be seen, the algorithms chosen for the experimental analysis cover a wide range of bio-inspired MOO schemes. The first five of them use a density estimator whereas the last three and MoCell lean upon a neighborhood structure for the evolution.

5.3 Performance Metrics and Parametric Setup

The performance metrics selected for the empirical analysis are based on the following quality indicators: *additive epsilon* (I_{ϵ}^{+}) [8], *spread* (I_{SP}) [4] and *hypervolume* (I_{HV}) [26]. The first two indicators respectively measure the convergence to the true Pareto front and the spread of the resulting Pareto front while the last one gauges both aspects.

MOVMO has been implemented using jMetal version 4.5 [6], a Java-based framework that includes implementations of all previously described state-of-the-art algorithms, test suites (ZDT, DTLZ, etc.) and performance metrics.

In order to ensure a fair comparison among all competing techniques, the following parametric configuration was adopted: population size of 20 for AbYSS and 100 for the rest; the maximum leaders archive size was set to 100 for all algorithms. NSGA-II, SPEA2 and MOCell are based on genetic algorithms, hence they use the same crossover, mutation, and selection operators. SBX is the crossover strategy of choice with distribution index $n_c = 20$ and $p_c = 0.9$ as crossover rate (probability). For the polynomial mutation operator, we considered a distribution index of $n_m = 20$ and mutation rate $p_m = 1/D$, where D is the number of decision variables (solution dimensionality). Finally, binary tournament acts as the selection operator. In MOEA/D and MOEA/D.DRA, the probability that parent solutions are selected from the neighborhood was fixed at 0.9. For MOEA/D the number of weight vectors in the neighborhood of each weight vector and the maximal number of solutions replaced by each child solution were set to $T = 20$ and $n_r = 2$ respectively. Meanwhile, MOEA/D.DRA use $T = 10\%$ of the population size and n_r=1% of the population size. The differential-evolution-based implementations of MOEA/D and MOEA/D.DRA employ a crossover operator with probability $CR = 1.0$ and amplification factor of the difference vector equal to $F = 0.5$. All experiments have been carried out with 25,000 objective function evaluations as the stop criterion.[1]

6 Empirical Results and Statistical Analysis

In this section we statistically analyze MOVMO's performance compared to that of NSGA-II, SPEA2, AbYSS, MOCell, SMPSO, MOEA/D and MOEA/D.DRA on the bi-objective (2Obj) and three-objective (3Obj) configurations of the four test suites described in Section 5.1. For each algorithm, the median $\tilde{x}$ and inter-quartile range *IQR* of each performance metric over 30 independent runs are reported as measures of central tendency and statistical dispersion, respectively. The best and second best values of a performance metric are colored in the tables with dark gray and light gray, respectively. The base number corresponds to the median and the subscript represents the IQR. Recall that lower values for the *additive epsilon* and *spread* indicators are preferred whereas higher *hypervolume* values denote superior performance.

To determine the statistical significance of the obtained results, we will lean on the Wilcoxon rank-sum test [5]. This nonparametric procedure is often employed in hypothesis testing to detect significant differences between two samples (in this case, the performance values of two algorithms) [5]. We have used the R function `wilcox.test(a,b)` for that purpose. In all cases, we are interested in identifying significant differences at the 95% confidence level. This means we will reject the null hypothesis whenever the obtained p-value is smaller than 5%.

[1] The parameter values for each benchmarking algorithm have been drawn from their original articles.

6.1 2Obj Results

6.1.1 2Obj Additive Epsilon Results

Table 1 lists the median and IQR values for the I_ϵ^+ indicator of the approximation sets computed by all algorithms. Notice that MOVMO achieved the best (lowest) values in nine out of twenty-nine test functions and six second best values. SMPSO is the algorithm with second best overall performance; it attains six first places.

Table 1: I_ϵ^+ median and IQR values for the 2Obj configuration.

	MOVMO	MOEAD.DRA	MOEAD	NSGAII	SPEA2	AbYSS	MOCell	SMPSO
ZDT1	$5.44E-03_{2.2E-04}$	$1.12E-01_{6.5E-02}$	$2.34E-02_{7.5E-03}$	$1.32E-02_{2.4E-03}$	$8.97E-03_{1.2E-03}$	$7.37E-03_{1.2E-03}$	$6.43E-03_{4.4E-04}$	$5.59E-03_{2.1E-04}$
ZDT2	$5.40E-03_{2.4E-04}$	$1.88E-01_{7.7E-02}$	$4.19E-02_{3.1E-02}$	$1.31E-02_{2.2E-03}$	$8.94E-03_{1.6E-03}$	$7.68E-03_{1.9E-03}$	$5.74E-03_{4.0E-04}$	$5.50E-03_{3.1E-04}$
ZDT3	$5.19E-03_{3.7E-04}$	$1.88E-01_{1.4E-02}$	$1.27E-01_{5.9E-02}$	$8.68E-03_{2.9E-03}$	$9.81E-03_{2.3E-03}$	$5.79E-03_{3.1E-01}$	$6.77E-03_{9.0E-04}$	$5.81E-03_{6.5E-04}$
ZDT4	$4.42E-01_{3.1E-01}$	$1.15e+00_{1.1e+00}$	$4.10E-01_{4.4E-01}$	$1.42E-02_{5.8E-03}$	$2.02E-02_{8.2E-02}$	$1.28E-02_{5.9E-03}$	$9.17E-03_{2.0E-03}$	$6.14E-03_{5.4E-04}$
ZDT6	$4.51E-03_{2.5E-04}$	$6.64E-03_{3.4E-03}$	$5.02E-03_{1.9E-04}$	$1.51E-02_{2.0E-03}$	$2.61E-02_{4.7E-03}$	$5.46E-03_{4.5E-04}$	$8.06E-03_{6.7E-04}$	$4.90E-03_{3.7E-04}$
DTLZ1	$3.22E-03_{2.5E-01}$	$3.90E-01_{8.3E-01}$	$3.81E-03_{2.4E-03}$	$7.52E-03_{2.5E-03}$	$6.28E-03_{4.7E-03}$	$5.58E-03_{2.8E-03}$	$4.33E-03_{9.8E-04}$	$3.12E-03_{2.4E-04}$
DTLZ2	$5.14E-03_{1.7E-04}$	$7.14E-03_{9.9E-05}$	$7.22E-03_{9.3E-05}$	$1.26E-02_{3.1E-03}$	$8.14E-03_{8.4E-04}$	$5.61E-03_{3.2E-04}$	$5.56E-03_{2.2E-03}$	$5.52E-03_{4.8E-04}$
DTLZ3	$8.36e+00_{8.5e+00}$	$1.78e+01_{4.8e+01}$	$6.61E-01_{3.4e+00}$	$1.66e+00_{1.4e+00}$	$1.70e+00_{1.3e+00}$	$1.64e+00_{2.2e+00}$	$8.02E-01_{1.4e+00}$	$5.97E-03_{7.0E-01}$
DTLZ4	$5.31E-03_{9.9E-01}$	$7.63E-03_{2.2E-04}$	$7.60E-03_{2.9E-04}$	$1.27E-02_{4.3E-03}$	$8.16E-03_{9.9E-01}$	$5.67E-03_{3.0E-04}$	$1.00e+00_{9.9E-01}$	$5.59E-03_{1.7E-04}$
DTLZ5	$5.04E-03_{2.1E-04}$	$7.16E-03_{7.1E-05}$	$7.17E-03_{9.9E-05}$	$1.07E-02_{2.2E-03}$	$7.20E-03_{1.0E-03}$	$5.49E-03_{5.7E-04}$	$4.98E-03_{3.2E-04}$	$5.16E-03_{2.0E-04}$
DTLZ6	$5.11E-03_{6.9E-04}$	$7.10E-03_{2.9E-05}$	$7.10E-03_{7.0E-06}$	$3.68E-02_{3.1E-02}$	$2.84E-01_{4.8E-02}$	$7.98E-02_{4.4E-02}$	$4.16E-02_{2.7E-02}$	$5.10E-03_{5.6E-04}$
DTLZ7	$2.31e+00_{2.5E-05}$	$2.32e+00_{1.7E-02}$	$2.33e+00_{1.3e+00}$	$2.31e+00_{1.6E-04}$	$2.31e+00_{4.4E-04}$	$2.31e+00_{7.0E-07}$	$2.31e+00_{1.3e+00}$	$2.31e+00_{7.7E-05}$
WFG1	$6.17E-02_{1.9E-02}$	$7.79E-01_{4.8E-02}$	$6.83E-01_{1.3E-01}$	$6.02E-01_{4.0E-01}$	$9.57E-01_{2.2E-01}$	$8.98E-01_{5.9E-01}$	$1.09e+00_{2.5E-01}$	$1.15e+00_{3.7E-02}$
WFG2	$3.63E-01_{7.0E-01}$	$2.90E-02_{5.6E-03}$	$2.88E-02_{2.7E-03}$	$2.03E-02_{7.0E-01}$	$2.10E-02_{7.0E-01}$	$7.13E-01_{2.9E-03}$	$7.11E-01_{8.7E-04}$	$1.42E-02_{1.6E-03}$
WFG3	$2.00e+00_{3.9E-03}$	$2.00e+00_{6.6E-04}$	$2.00e+00_{2.3E-04}$	$2.00e+00_{8.2E-04}$	$2.00e+00_{2.4E-03}$	$2.00e+00_{3.2E-03}$	$2.00e+00_{1.2E-03}$	$2.00e+00_{9.0E-04}$
WFG4	$1.52E-02_{1.4E-03}$	$6.17E-02_{9.0E-03}$	$5.25E-02_{6.3E-03}$	$3.34E-02_{6.5E-03}$	$2.51E-02_{4.5E-03}$	$1.47E-02_{5.1E-04}$	$1.54E-02_{9.2E-04}$	$5.30E-02_{4.3E-03}$
WFG5	$6.34E-02_{4.5E-04}$	$7.30E-02_{8.0E-04}$	$7.28E-02_{5.9E-04}$	$8.41E-02_{8.3E-03}$	$7.23E-02_{3.6E-03}$	$6.39E-02_{6.4E-04}$	$6.34E-02_{6.8E-04}$	$6.35E-02_{1.1E-03}$
WFG6	$3.58E-02_{3.3E-02}$	$2.48E-02_{4.5E-03}$	$2.43E-02_{5.6E-04}$	$3.96E-02_{1.3E-02}$	$3.09E-02_{1.7E-02}$	$5.74E-02_{6.1E-02}$	$3.55E-02_{6.3E-02}$	$1.73E-02_{2.1E-03}$
WFG7	$1.40E-02_{7.4E-04}$	$2.48E-02_{9.0E-04}$	$2.53E-02_{5.5E-04}$	$3.61E-02_{1.2E-02}$	$2.56E-02_{4.6E-03}$	$1.50E-02_{6.0E-04}$	$1.49E-02_{7.8E-04}$	$1.80E-02_{1.5E-03}$
WFG8	$5.12E-01_{2.0E-01}$	$4.80E-01_{2.2E-01}$	$2.95E-01_{2.4E-01}$	$4.84E-01_{2.3E-01}$	$5.12E-01_{2.3E-01}$	$5.14E-01_{1.2E-02}$	$5.10E-01_{1.9E-01}$	$3.94E-01_{3.6E-02}$
WFG9	$1.82E-02_{4.1E-03}$	$3.49E-02_{2.9E-03}$	$3.40E-02_{1.6E-03}$	$3.82E-02_{5.1E-03}$	$2.91E-02_{3.5E-03}$	$2.14E-02_{5.7E-03}$	$1.99E-02_{4.9E-03}$	$2.82E-02_{2.8E-03}$
LZ09.F1	$1.80E-02_{7.0E-03}$	$1.19E-02_{5.5E-03}$	$7.93E-03_{1.2E-03}$	$1.81E-02_{2.8E-03}$	$2.70E-02_{3.0E-02}$	$2.01E-02_{4.4E-03}$	$3.37E-02_{3.6E-02}$	$8.73E-03_{1.1E-03}$
LZ09.F2	$2.33E-01_{1.0E-01}$	$1.88E-01_{5.7E-02}$	$1.78E-01_{7.3E-02}$	$1.90E-01_{1.0E-01}$	$1.97E-01_{4.6E-02}$	$3.01E-01_{1.3E-01}$	$2.90E-01_{1.2E-01}$	$2.59E-01_{3.7E-02}$
LZ09.F3	$1.61E-01_{1.0E-01}$	$1.20E-01_{1.2E-01}$	$2.43E-01_{1.8E-01}$	$1.30E-01_{9.4E-03}$	$1.91E-01_{6.3E-02}$	$1.76E-01_{1.6E-01}$	$1.93E-01_{1.0E-01}$	$1.75E-01_{2.4E-02}$
LZ09.F4	$1.66E-01_{4.5E-02}$	$9.37E-02_{2.6E-02}$	$1.95E-01_{3.6E-02}$	$1.65E-01_{3.9E-02}$	$1.91E-01_{1.9E-02}$	$1.84E-01_{2.7E-02}$	$1.93E-01_{3.2E-02}$	$1.44E-01_{2.5E-02}$
LZ09.F5	$1.29E-01_{3.4E-02}$	$1.24E-01_{8.1E-02}$	$1.25E-01_{6.7E-02}$	$1.20E-01_{1.7E-02}$	$1.37E-01_{1.7E-02}$	$1.79E-01_{5.0E-02}$	$1.41E-01_{7.9E-02}$	$1.45E-01_{2.2E-02}$
LZ09.F7	$5.05E-01_{2.6E-01}$	$4.02E-01_{2.1E-01}$	$2.94E-01_{1.0E-01}$	$3.22E-01_{1.8E-01}$	$4.04E-01_{2.1E-01}$	$5.60E-01_{2.5E-01}$	$4.77E-01_{1.5E-01}$	$5.19E-01_{8.1E-02}$
LZ09.F8	$4.32E-01_{1.8E-01}$	$3.82E-01_{1.3E-01}$	$3.68E-01_{1.3E-01}$	$2.99E-01_{9.7E-02}$	$3.42E-01_{1.3E-01}$	$4.74E-01_{1.6E-01}$	$5.32E-01_{1.9E-01}$	$5.73E-01_{5.4E-02}$
LZ09.F9	$2.94E-01_{1.1E-01}$	$2.03E-01_{1.1E-01}$	$1.87E-01_{1.0E-01}$	$2.91E-01_{1.1E-01}$	$2.72E-01_{1.2E-01}$	$3.03E-01_{1.0E-01}$	$3.26E-01_{6.8E-02}$	$2.27E-01_{5.3E-02}$

The Wilcoxon test results confirm that MOVMO yielded better performance at 95% significance level on the following functions: ZDT1, ZDT2, ZDT3, ZDT6, DTLZ2 and WFG7. For WFG9, MOCell and MOVMO's performance differences are not statistically significant. Furthermore, MOVMO reports the worst performance on DTLZ3 and WFG3 with respect to its peers. Although the MOVMO results obtained on DTLZ1, DTLZ5, DTLZ6, WFG4 and WFG5 ranked second place, the Wilcoxon test indicates that they are not statistically significant. Conversely, despite the fact that MOVMO never climbed to the first or second place in any function of the new LZ09 family, the Wilcoxon test results reveal that the performance differences between MOVMO's I_ϵ^+ values and the second place in LZ09.F5 exhibited by MOEA/D.DRA are not statistically significant at the 95% level.

Friedman test results on the I_ϵ^+ performance metric are reported in Table 2. MOVMO is the second best-ranked algorithm following SMPSO whereas SPEA2 and AbYSS do not generally converge well to the true Pareto front of the functions under consideration. The fact that the top rank is above three indicates that there is no clear winner across all four benchmark function families but that some algorithms are better than the rest in coping with certain intricacies in a function's landscape.

Table 2: Friedman test. Average ranks of the algorithms in expected I_{ϵ}^{+} with two objectives (distributed according to χ^2 with 7 degrees of freedom: 17.471264367816207).

Algorithm	Average Rank
SMPSO	3.207
MOVMO	3.759
NSGAII	4.310
MOEAD	4.517
MOEAD.DRA	4.759
MOCell	5.069
SPEA2	5.103
AbYSS	5.276

6.1.2 2Obj Spread Results

Table 3 depicts the median and *IQR* values for the I_{SP} indicator. In this case MOVMO achieves the best (lowest) values in twelve out of twenty-nine functions and second in another nine functions. AbYSS with ten first places and eight second ones and SMPSO with five first and five second places were the next best-performing schemes. The Wilcoxon test confirms that the MOVMO spread values obtained on ZDT1, ZDT3, ZDT6, DTLZ6, WFG1, WFG9 and LZ09.F3 are statistically significant at the 95% level, unlike those in ZDT2 (vs. SMPSO), WFG3 and WFG6 (vs. MOCell), LZ09.F4 (vs. AbYSS) and LZ09.F5 (vs. AbYSS and MOCell). Once again, MOVMO exhibits a poor performance on ZDT4 compared with all algorithms but MOEA/D and MOEA/D.DRA. However, the observed performance gain of MOCell (2nd place in WFG2) over MOVMO is not judged statistically significant at the aforesaid level by the Wilcoxon test.

Table 3: I_{SP} median and IQR values for the 2Obj configuration.

	MOVMO	MOEAD.DRA	MOEAD	NSGAII	SPEA2	AbYSS	MOCell	SMPSO
ZDT1	$6.41E-02_{1.2E-02}$	$6.54E-01_{9.7E-02}$	$3.57E-01_{4.7E-02}$	$3.63E-01_{3.9E-02}$	$1.48E-01_{1.9E-02}$	$1.00E-01_{1.6E-02}$	$7.97E-02_{1.6E-02}$	$7.47E-02_{1.7E-02}$
ZDT2	$6.92E-02_{1.2E-02}$	$9.15E-01_{2.7E-01}$	$3.00E-01_{1.5E-01}$	$3.82E-01_{4.3E-02}$	$1.53E-01_{1.8E-02}$	$1.09E-01_{2.0E-02}$	$8.77E-02_{1.8E-02}$	$7.43E-02_{1.6E-02}$
ZDT3	$7.02E-01_{1.7E-03}$	$1.08e+00_{6.9E-02}$	$9.99E-01_{4.1E-02}$	$7.50E-01_{1.8E-02}$	$7.06E-01_{9.2E-03}$	$7.05E-01_{1.7E-02}$	$7.08E-01_{7.0E-03}$	$7.09E-01_{7.3E-03}$
ZDT4	$5.92E-01_{2.7E-01}$	$1.43e+00_{4.4E-02}$	$1.00e+00_{1.6E-01}$	$3.96E-01_{6.8E-02}$	$2.31E-01_{1.6E-01}$	$1.31E-01_{4.4E-02}$	$1.31E-01_{3.8E-02}$	$9.28E-02_{1.6E-02}$
ZDT6	$6.01E-02_{1.6E-02}$	$1.87E-01_{1.8E-01}$	$1.52E-01_{3.4E-03}$	$3.64E-01_{4.1E-02}$	$2.30E-01_{3.4E-02}$	$8.44E-02_{1.6E-02}$	$1.06E-01_{1.3E-02}$	$1.13E-01_{9.9E-01}$
DTLZ1	$9.07E-02_{3.9E-01}$	$1.15e+00_{6.3E-01}$	$3.58E-02_{1.3E-01}$	$3.77E-01_{6.0E-02}$	$1.76E-01_{1.0E-01}$	$1.27E-01_{7.3E-02}$	$1.16E-01_{8.0E-02}$	$7.07E-02_{1.7E-02}$
DTLZ2	$1.08E-01_{2.6E-02}$	$2.36E-01_{2.9E-02}$	$1.84E-01_{1.8E-03}$	$3.83E-01_{5.2E-02}$	$1.46E-01_{1.4E-02}$	$1.03E-01_{2.1E-02}$	$1.19E-01_{2.4E-02}$	$1.24E-01_{2.1E-02}$
DTLZ3	$6.17E-01_{3.9E-02}$	$1.28e+00_{1.5E-01}$	$9.44E-01_{7.3E-01}$	$9.56E-01_{1.0E-01}$	$9.77E-01_{1.7E-01}$	$8.77E-01_{4.7E-01}$	$1.06e+00_{4.5E-01}$	$1.43E-01_{1.2e+00}$
DTLZ4	$1.28E-01_{8.9E-01}$	$3.16E-01_{5.4E-02}$	$1.91E-01_{1.7E-02}$	$3.93E-01_{7.2E-02}$	$1.49E-01_{8.6E-01}$	$1.13E-01_{2.2E-02}$	$1.00e+00_{8.8E-01}$	$1.16E-01_{1.7E-02}$
DTLZ5	$1.12E-01_{2.3E-02}$	$2.46E-01_{3.0E-02}$	$1.88E-01_{3.0E-03}$	$3.74E-01_{3.3E-02}$	$1.50E-01_{1.4E-02}$	$1.09E-01_{2.2E-02}$	$1.19E-01_{2.0E-02}$	$1.30E-01_{1.7E-02}$
DTLZ6	$9.80E-02_{1.2E-02}$	$1.95E-01_{2.0E-02}$	$1.86E-01_{4.3E-04}$	$8.50E-01_{3.6E-01}$	$8.24E-01_{1.5E-01}$	$2.20E-01_{5.0E-02}$	$1.81E-01_{4.0E-01}$	$1.12E-01_{1.6E-02}$
DTLZ7	$7.77E-01_{4.3E-04}$	$9.31E-01_{3.5E-02}$	$8.96E-01_{3.1E-02}$	$8.24E-01_{1.5E-02}$	$7.88E-01_{4.4E-03}$	$7.77E-01_{2.7E-04}$	$7.78E-01_{1.2E-01}$	$7.77E-01_{9.2E-04}$
WFG1	$3.04E-01_{2.4E-02}$	$1.03e+00_{2.6E-01}$	$1.05e+00_{2.8E-01}$	$7.23E-01_{6.3E-02}$	$6.52E-01_{4.3E-02}$	$6.88E-01_{5.1E-02}$	$6.45E-01_{7.0E-02}$	$1.00e+00_{3.8E-02}$
WFG2	$7.57E-01_{1.1E-02}$	$1.11e+00_{8.4E-03}$	$1.10e+00_{7.2E-03}$	$7.87E-01_{1.5E-02}$	$7.61E-01_{8.6E-03}$	$7.47E-01_{8.9E-03}$	$7.51E-01_{3.5E-03}$	$8.02E-01_{2.3E-02}$
WFG3	$3.68E-01_{1.3E-02}$	$5.63E-01_{2.0E-03}$	$5.63E-01_{7.8E-04}$	$5.88E-01_{2.9E-02}$	$4.37E-01_{1.2E-02}$	$3.77E-01_{8.2E-03}$	$3.71E-01_{9.3E-03}$	$3.79E-01_{7.5E-03}$
WFG4	$1.31E-01_{1.9E-02}$	$6.56E-01_{1.3E-01}$	$5.05E-01_{5.0E-02}$	$3.89E-01_{3.1E-02}$	$2.69E-01_{2.5E-02}$	$1.26E-01_{2.1E-02}$	$1.34E-01_{2.4E-02}$	$4.55E-01_{6.6E-02}$
WFG5	$1.31E-01_{1.7E-02}$	$4.63E-01_{8.8E-03}$	$4.61E-01_{4.8E-03}$	$4.11E-01_{3.7E-02}$	$2.87E-01_{2.6E-02}$	$1.31E-01_{2.1E-02}$	$1.41E-01_{2.0E-02}$	$1.39E-01_{2.1E-02}$
WFG6	$1.19E-01_{3.8E-02}$	$4.37E-01_{3.7E-02}$	$4.11E-01_{1.2E-03}$	$3.86E-01_{5.1E-02}$	$2.50E-01_{3.1E-02}$	$1.40E-01_{3.8E-02}$	$1.35E-01_{4.5E-02}$	$1.50E-01_{2.0E-02}$
WFG7	$1.12E-01_{1.9E-02}$	$4.17E-01_{1.9E-02}$	$4.11E-01_{5.2E-03}$	$3.76E-01_{5.7E-02}$	$2.43E-01_{2.3E-02}$	$1.05E-01_{1.9E-02}$	$1.23E-01_{2.1E-02}$	$1.56E-01_{1.7E-02}$
WFG8	$5.65E-01_{6.6E-02}$	$7.11E-01_{9.1E-02}$	$6.33E-01_{6.0E-02}$	$6.41E-01_{3.4E-02}$	$5.96E-01_{6.1E-02}$	$5.87E-01_{9.0E-02}$	$5.56E-01_{5.5E-02}$	$7.28E-01_{4.2E-02}$
WFG9	$1.37E-01_{1.8E-02}$	$5.16E-01_{2.6E-02}$	$4.84E-01_{1.9E-02}$	$3.89E-01_{4.1E-02}$	$2.93E-01_{2.4E-02}$	$1.50E-01_{2.2E-02}$	$1.51E-01_{2.3E-02}$	$2.05E-01_{1.7E-02}$
LZ09.F1	$4.98E-01_{2.7E-01}$	$3.35E-01_{5.8E-02}$	$2.91E-01_{9.8E-03}$	$4.93E-01_{8.9E-02}$	$4.87E-01_{3.0E-01}$	$2.58E-01_{4.8E-02}$	$4.24E-01_{2.5E-01}$	$1.34E-01_{1.9E-02}$
LZ09.F2	$1.40e+00_{1.3E-01}$	$1.23e+00_{1.8E-01}$	$9.65E-01_{1.4E-01}$	$1.43e+00_{1.2E-01}$	$1.50e+00_{1.1E-01}$	$1.43e+00_{2.6E-01}$	$1.46e+00_{2.5E-01}$	$8.39E-01_{1.1E-01}$
LZ09.F3	$5.64E-01_{9.4E-02}$	$8.96E-01_{2.6E-01}$	$6.91E-01_{1.2E-01}$	$7.06E-01_{1.1E-01}$	$7.15E-01_{1.4E-01}$	$6.36E-01_{1.9E-01}$	$6.43E-01_{1.4E-01}$	$7.97E-01_{1.1E-01}$
LZ09.F4	$4.21E-01_{7.4E-02}$	$9.12E-01_{1.9E-01}$	$9.61E-01_{2.6E-01}$	$5.81E-01_{5.7E-02}$	$5.66E-01_{8.7E-02}$	$4.46E-01_{6.8E-02}$	$5.10E-01_{9.9E-02}$	$6.87E-01_{1.6E-01}$
LZ09.F5	$4.99E-01_{9.6E-02}$	$7.69E-01_{1.2E-01}$	$6.31E-01_{1.1E-01}$	$6.49E-01_{7.7E-02}$	$5.81E-01_{8.5E-02}$	$5.13E-01_{9.9E-02}$	$5.26E-01_{9.9E-02}$	$6.78E-01_{1.2E-01}$
LZ09.F7	$1.05e+00_{3.0E-01}$	$1.42e+00_{1.3E-01}$	$1.23e+00_{1.2E-01}$	$1.37e+00_{1.6E-01}$	$1.40e+00_{2.0E-01}$	$1.02e+00_{3.0E-01}$	$1.29e+00_{4.4E-01}$	$1.40e+00_{2.4E-01}$
LZ09.F8	$1.39e+00_{2.0E-01}$	$1.53e+00_{1.2E-01}$	$1.28e+00_{9.0E-02}$	$1.31e+00_{8.1E-02}$	$1.30e+00_{1.2E-01}$	$1.00e+00_{8.1E-02}$	$1.36e+00_{2.1E-01}$	$1.44e+00_{5.0E-01}$
LZ09.F9	$1.52e+00_{2.3E-01}$	$1.15e+00_{1.5E-01}$	$1.01e+00_{1.3E-01}$	$1.61e+00_{2.9E-01}$	$1.58e+00_{2.2E-01}$	$1.55e+00_{1.3E-01}$	$1.60e+00_{1.5E-01}$	$8.85E-01_{1.5E-01}$

Friedman test results on the *spread* performance metric are reported in Table 4. They obey a χ^2 distribution with seven degrees of freedom at a 1% significance level (p-value< 0.01).

Table 4: Friedman test. Average ranks of the algorithms in expected I_{SP} with two objectives (distributed according to χ^2 with 7 degrees of freedom: 91.42528735632169).

Algorithm	Average Rank
MOVMO	2.345
AbYSS	2.621
MOCell	3.690
SMPSO	3.966
SPEA2	5.000
MOEAD	5.207
NSGAII	6.172
MOEAD.DRA	7.000

The *spread* performance indicator gauges the distribution of solutions along the Pareto front; therefore in the context of the experiments carried out, MOVMO is the algorithm with better distribution of the non-dominated solutions, followed by the AbYSS and MOCell methods. It is interesting to note that the more recently proposed decomposition-based schemes (MOEA/D and MOEA/D.DRA) do not fare well along this front.

6.1.3 2Obj Hypervolume Results

Table 5 unfolds the median and *IQR* values corresponding to the *hypervolume* indicator. MOVMO exhibits the best (highest) values on the ZDT1, ZDT2, ZDT3, ZDT6, and all the functions in the DTLZ family except DTLZ3. It also climbs to the top on WFG1 and WFG7. However, the hypervolume values between MOVMO and AbYSS (2nd place) on DTLZ7 are not statistically significant. The same can be stated for the perceived advantages of SPEA2 and NSGA-II (1st and 2nd place on WFG2), MOCell (2nd place in WFG4), SMPSO and MOCell (1st and 2nd place in WFG5), NSGA-II (2nd place in LZ09.F2), MOEA/D (2nd place in LZ09.F3) and MOEA/D, MOEA/D.DRA (1st and 2nd place in LZ09.F9) over MOVMO. For the rest of the cases, the observed performance differences are statistically significant at the 95% level, as confirmed by the Wilcoxon test.

The values reached by MOVMO on the I_{HV} indicator (that measures both convergence and diversity) have corroborated the results obtained by the two previous indicators. The hypervolume results portrayed in Table 5 point out to the fact that MOVMO reaches the best (highest) values on thirteen out of twenty-nine possible benchmark functions, eleven of which are deemed statistically significant at the 95% level. MOEAD/D had the second best behavior with seven first places and seven second best values. SMPSO was the third best-performing technique.

Friedman test results on the I_{HV} performance metric are reported in Table 6.

6.1.4 2Obj Discussion

Table 7 summarizes the set of pairwise comparisons between MOVMO and the rest of the algorithms across the three quality indicators for the 2Obj configuration. MOVMO's performance was compared to that of the seven state-of-the-art optimizers throughout the twenty-nine benchmark functions ($29 \times 7 = 203$ pairwise tests). The '+' symbol indicates that MOVMO's result was statistically significant compared to the other algorithm, '-' that the result was statistically significant in

Table 5: I_{HV} median and IQR values for the 2Obj configuration.

	MOVMO	MOEAD.DRA	MOEAD	NSGAII	SPEA2	AbYSS	MOCell	SMPSO
ZDT1	$6.62E-01_{4.1E-05}$	$5.24E-01_{9.0E-02}$	$6.39E-01_{9.5E-03}$	$6.59E-01_{4.2E-04}$	$6.60E-01_{3.3E-04}$	$6.61E-01_{4.2E-04}$	$6.61E-01_{3.9E-04}$	$6.62E-01_{7.6E-05}$
ZDT2	$3.29E-01_{8.7E-05}$	$1.96E-01_{6.4E-02}$	$3.10E-01_{9.3E-03}$	$3.26E-01_{4.3E-04}$	$3.26E-01_{9.2E-04}$	$3.28E-01_{3.5E-04}$	$3.28E-01_{3.3E-04}$	$3.29E-01_{9.2E-05}$
ZDT3	$5.16E-01_{2.5E-05}$	$3.82E-01_{4.1E-02}$	$4.56E-01_{3.7E-02}$	$5.15E-01_{1.7E-04}$	$5.14E-01_{3.5E-04}$	$5.16E-01_{3.5E-03}$	$5.14E-01_{3.6E-04}$	$5.15E-01_{3.5E-04}$
ZDT4	$4.52E-01_{1.7E-01}$	$0.00e+00_{2.8E-02}$	$2.86E-01_{3.7E-01}$	$6.56E-01_{3.6E-03}$	$6.52E-01_{9.0E-03}$	$6.52E-01_{8.6E-03}$	$6.58E-01_{2.4E-03}$	$6.61E-01_{1.9E-04}$
ZDT6	$4.01E-01_{6.3E-05}$	$4.01E-01_{9.7E-05}$	$4.01E-01_{4.3E-06}$	$3.89E-01_{3.0E-03}$	$3.79E-01_{4.1E-03}$	$4.00E-01_{1.1E-04}$	$3.94E-01_{1.1E-03}$	$4.01E-01_{7.4E-05}$
DTLZ1	$4.95E-01_{4.9E-01}$	$0.00e+00_{4.5E-01}$	$4.93E-01_{3.8E-03}$	$4.89E-01_{6.2E-03}$	$4.88E-01_{1.1E-02}$	$4.88E-01_{8.4E-03}$	$4.91E-01_{3.1E-03}$	$4.94E-01_{1.8E-04}$
DTLZ2	$2.11E-01_{2.8E-05}$	$2.10E-01_{1.3E-04}$	$2.10E-01_{1.0E-04}$	$2.09E-01_{2.3E-04}$	$2.10E-01_{1.4E-04}$	$2.11E-01_{6.2E-05}$	$2.11E-01_{4.8E-05}$	$2.10E-01_{1.1E-04}$
DTLZ3	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$2.77E-02_{1.9E-01}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{1.4E-01}$	$2.10E-01_{1.2E-01}$
DTLZ4	$2.11E-01_{2.1E-01}$	$2.09E-01_{2.2E-04}$	$2.09E-01_{1.4E-04}$	$2.09E-01_{5.3E-04}$	$2.10E-01_{2.1E-01}$	$2.11E-01_{6.2E-05}$	$0.00e+00_{2.1E-01}$	$2.10E-01_{1.9E-04}$
DTLZ5	$2.12E-01_{3.7E-05}$	$2.11E-01_{1.2E-04}$	$2.11E-01_{6.1E-05}$	$2.11E-01_{2.5E-04}$	$2.12E-01_{1.5E-04}$	$2.12E-01_{3.7E-05}$	$2.12E-01_{5.7E-05}$	$2.12E-01_{1.5E-04}$
DTLZ6	$2.12E-01_{3.1E-05}$	$2.12E-01_{2.8E-05}$	$2.12E-01_{3.2E-07}$	$1.80E-01_{4.2E-02}$	$1.56E-02_{1.6E-02}$	$1.24E-01_{4.1E-02}$	$1.62E-01_{3.2E-02}$	$2.12E-01_{7.7E-05}$
DTLZ7	$3.09E-01_{7.4E-06}$	$3.03E-01_{7.4E-03}$	$3.04E-01_{2.1E-01}$	$3.09E-01_{3.2E-05}$	$3.09E-01_{9.2E-05}$	$3.09E-01_{6.2E-06}$	$3.09E-01_{2.1E-01}$	$3.09E-01_{2.7E-05}$
WFG1	$6.31E-01_{1.5E-03}$	$2.49E-01_{1.5E-02}$	$2.89E-01_{7.2E-02}$	$5.12E-01_{1.0E-01}$	$4.33E-01_{1.3E-01}$	$2.33E-01_{1.3E-01}$	$3.32E-01_{1.7E-01}$	$1.16E-01_{8.8E-03}$
WFG2	$5.61E-01_{3.1E-03}$	$5.62E-01_{3.6E-04}$	$5.62E-01_{3.5E-04}$	$5.63E-01_{2.6E-03}$	$5.63E-01_{2.5E-03}$	$5.60E-01_{2.0E-03}$	$5.61E-01_{6.3E-04}$	$5.61E-01_{1.1E-03}$
WFG3	$4.41E-01_{1.3E-03}$	$4.41E-01_{1.7E-04}$	$4.41E-01_{5.8E-05}$	$4.41E-01_{4.7E-04}$	$4.41E-01_{4.4E-04}$	$4.41E-01_{1.2E-03}$	$4.41E-01_{3.7E-04}$	$4.41E-01_{1.9E-04}$
WFG4	$2.18E-01_{7.9E-04}$	$2.09E-01_{4.7E-03}$	$2.11E-01_{1.5E-03}$	$2.17E-01_{4.0E-04}$	$2.18E-01_{3.5E-04}$	$2.19E-01_{2.1E-04}$	$2.19E-01_{2.3E-04}$	$2.02E-01_{2.6E-03}$
WFG5	$1.96E-01_{5.6E-05}$	$1.95E-01_{6.1E-05}$	$1.95E-01_{5.2E-05}$	$1.95E-01_{3.4E-04}$	$1.96E-01_{1.5E-04}$	$1.96E-01_{7.5E-05}$	$1.96E-01_{6.7E-05}$	$1.96E-01_{6.7E-05}$
WFG6	$1.96E-01_{2.0E-02}$	$2.08E-01_{3.9E-04}$	$2.09E-01_{1.5E-04}$	$2.05E-01_{1.0E-02}$	$2.04E-01_{1.0E-02}$	$1.83E-01_{3.6E-02}$	$1.96E-01_{3.8E-02}$	$2.09E-01_{5.3E-04}$
WFG7	$2.11E-01_{8.2E-05}$	$2.09E-01_{2.1E-04}$	$2.09E-01_{1.3E-04}$	$2.09E-01_{5.6E-04}$	$2.10E-01_{1.9E-04}$	$2.11E-01_{2.1E-04}$	$2.10E-01_{1.6E-04}$	$2.09E-01_{2.3E-04}$
WFG8	$1.46E-01_{3.4E-03}$	$1.52E-01_{1.3E-03}$	$1.50E-01_{8.9E-04}$	$1.47E-01_{2.5E-03}$	$1.47E-01_{2.3E-03}$	$1.44E-01_{3.0E-03}$	$1.47E-01_{2.0E-03}$	$1.47E-01_{1.2E-03}$
WFG9	$2.38E-01_{3.4E-03}$	$2.35E-01_{1.3E-03}$	$2.35E-01_{7.6E-04}$	$2.37E-01_{2.1E-03}$	$2.38E-01_{2.0E-03}$	$2.38E-01_{3.8E-03}$	$2.38E-01_{3.8E-03}$	$2.35E-01_{6.6E-04}$
LZ09.F1	$6.53E-01_{1.9E-03}$	$6.60E-01_{9.4E-04}$	$6.61E-01_{1.3E-04}$	$6.52E-01_{9.6E-04}$	$6.54E-01_{2.9E-03}$	$6.52E-01_{2.4E-03}$	$6.52E-01_{3.8E-03}$	$6.58E-01_{9.7E-04}$
LZ09.F2	$5.18E-01_{7.2E-02}$	$5.23E-01_{5.3E-02}$	$5.38E-01_{5.2E-02}$	$5.29E-01_{8.1E-02}$	$5.28E-01_{2.3E-02}$	$4.93E-01_{1.0E-01}$	$4.50E-01_{1.0E-01}$	$3.74E-01_{7.8E-02}$
LZ09.F3	$5.95E-01_{1.7E-02}$	$6.13E-01_{1.5E-02}$	$6.00E-01_{3.5E-02}$	$5.98E-01_{7.9E-03}$	$5.80E-01_{8.6E-03}$	$5.81E-01_{3.1E-02}$	$5.82E-01_{4.0E-02}$	$5.75E-01_{1.2E-02}$
LZ09.F4	$6.04E-01_{3.7E-03}$	$6.29E-01_{7.4E-03}$	$6.15E-01_{1.3E-02}$	$6.09E-01_{3.7E-03}$	$5.98E-01_{5.0E-03}$	$5.95E-01_{8.8E-03}$	$5.95E-01_{3.6E-03}$	$5.84E-01_{9.2E-03}$
LZ09.F5	$6.07E-01_{5.3E-03}$	$6.15E-01_{9.6E-03}$	$6.18E-01_{9.2E-03}$	$6.11E-01_{6.4E-03}$	$6.02E-01_{8.1E-03}$	$5.96E-01_{2.0E-02}$	$5.99E-01_{2.5E-02}$	$5.96E-01_{7.9E-03}$
LZ09.F7	$3.73E-01_{8.3E-02}$	$3.87E-01_{1.9E-01}$	$4.99E-01_{9.9E-02}$	$4.69E-01_{5.4E-02}$	$4.31E-01_{7.4E-02}$	$3.50E-01_{8.7E-02}$	$3.77E-01_{2.8E-02}$	$1.64E-01_{1.4E-01}$
LZ09.F8	$3.51E-01_{9.1E-02}$	$3.12E-01_{1.3E-01}$	$3.77E-01_{1.2E-01}$	$4.30E-01_{3.7E-02}$	$4.21E-01_{4.3E-02}$	$3.41E-01_{7.9E-02}$	$3.39E-01_{9.5E-02}$	$4.40E-02_{6.2E-02}$
LZ09.F9	$1.58E-01_{6.7E-02}$	$1.93E-01_{7.0E-02}$	$1.95E-01_{9.7E-02}$	$1.58E-01_{9.0E-02}$	$1.72E-01_{6.2E-02}$	$1.47E-01_{7.4E-02}$	$1.50E-01_{7.5E-02}$	$1.26E-01_{3.6E-02}$

Table 6: Friedman test. Average ranks of the algorithms in expected I_{HV} with two objectives (distributed according to χ^2 with 7 degrees of freedom: 8.137931034482813).

Algorithm	Average Rank
MOVMO	3.828
MOEAD	3.966
SPEA2	4.310
SMPSO	4.414
NSGAII	4.483
MOCell	4.724
AbYSS	5.0
MOEAD.DRA	5.276

favor of the competing algorithm and '=' that no significant differences were uncovered in the comparison.

Table 7: MOVMO pairwise comparisons - Wilcoxon test summary for the 2Obj configuration.

	Epsilon	Spread	Hypervolume	Total
+	99	136	114	349
-	44	21	36	101
=	60	46	53	159

It is clear from Table 7 that MOVMO outperformed its competitors across each quality indicator at least twice as much as it was beaten by them. This superiority is more strongly evidenced in the Spread metric (where the '+'/'-' ratio soars above six times), although the additive epsilon and hypervolume summaries are quite encouraging too. Overall, MOVMO performed very well on the ZDT, DTLZ and WFG families whereas there is still room for improvement on the LZ09 family.

6.2 3Obj Results

6.2.1 3Obj Additive Epsilon Results

Table 8 lists the median and *IQR* values for the I_ϵ^+ indicator of the approximation sets computed by all algorithms in three objectives. Notice that MOVMO achieved the lowest (best) values in three out of seventeen test functions and six second best values. SPEA2 is the algorithm with best overall performance; it attains eleven first places. The Wilcoxon test disclosed that MOVMO yielded statistically significant performance at the 95% level on DTLZ5 and WFG1 but not on DTLZ6 over SMPSO. MOVMO reports poor performance on DTLZ3 and LZ09.F6 with respect to its peers. Nevertheless, the apparent performance gain of AbYSS (2nd place in DTLZ4), MOEA/D (2nd place in WFG3) and MOCell (2nd place in WFG9) over MOVMO is not statistically backed up by the Wilcoxon test.

Table 8: I_ϵ^+ median and IQR values for the 3Obj configuration.

	MOVMO	MOEAD.DRA	MOEAD	NSGAII	SPEA2	AbYSS	MOCell	SMPSO
LZ09.F6	$3.54E-01_{1.5E-01}$	$2.55E-01_{3.1E-02}$	$2.53E-01_{9.4E-03}$	$3.06E-01_{6.1E-02}$	$2.70E-01_{3.9E-02}$	$4.28E-01_{2.5E-01}$	$3.72E-01_{3.5E-01}$	$3.88E-01_{1.6E-01}$
DTLZ1	$6.31E-02_{2.2E-01}$	$2.36E-01_{3.8E-01}$	$5.10E-02_{1.0E-02}$	$6.56E-02_{3.1E-02}$	$4.28E-02_{7.7E-03}$	$9.88E-02_{2.4E-01}$	$3.44E-01_{4.4E-01}$	$5.76E-02_{7.9E-03}$
DTLZ2	$1.21E-01_{2.8E-02}$	$1.36E-01_{8.3E-03}$	$1.34E-01_{7.7E-03}$	$1.26E-01_{2.3E-02}$	$7.74E-02_{7.9E-03}$	$1.32E-01_{2.2E-02}$	$1.26E-01_{2.5E-02}$	$1.38E-01_{2.4E-02}$
DTLZ3	$6.44e+00_{2.8e+00}$	$6.13e+00_{2.6e+01}$	$2.41E-01_{2.3e+00}$	$5.45e+00_{5.0e+00}$	$4.20e+00_{4.0e+00}$	$4.15e+00_{3.8e+00}$	$9.35e+00_{9.7e+00}$	$1.41E-01_{4.5E-01}$
DTLZ4	$1.09E-01_{2.3E-02}$	$1.20E-01_{1.3E-01}$	$1.28E-01_{1.3E-01}$	$1.12E-01_{2.5E-02}$	$7.41E-02_{5.6E-01}$	$1.01E-01_{2.2E-02}$	$1.13E-01_{3.2E-02}$	$1.31E-01_{3.1E-02}$
DTLZ5	$4.33E-03_{1.7E-04}$	$2.64E-02_{5.8E-04}$	$2.61E-02_{9.3E-04}$	$1.06E-02_{1.7E-03}$	$7.28E-03_{1.7E-03}$	$4.76E-03_{6.1E-04}$	$4.57E-03_{9.1E-04}$	$4.66E-03_{1.9E-04}$
DTLZ6	$4.31E-03_{7.6E-04}$	$2.66E-02_{7.2E-05}$	$2.66E-02_{8.8E-05}$	$8.65E-01_{6.4E-02}$	$7.92E-01_{9.6E-02}$	$6.45E-01_{1.1E-01}$	$1.68e+00_{1.9E-01}$	$4.40E-03_{4.5E-04}$
DTLZ7	$1.18E-01_{7.5E-02}$	$5.47E-01_{1.7E-01}$	$3.65E-01_{2.1E-01}$	$1.24E-01_{3.4E-02}$	$1.02E-01_{1.4E-02}$	$1.27e+00_{2.4e+00}$	$1.48E-01_{1.2e+00}$	$1.66E-01_{3.9E-02}$
WFG1	$4.14E-01_{4.2E-02}$	$1.31e+00_{1.0E-01}$	$1.34e+00_{1.5E-01}$	$5.41E-01_{4.6E-02}$	$6.36E-01_{9.0E-02}$	$4.45E-01_{1.4E-01}$	$7.19E-01_{1.4E-01}$	$1.74e+00_{7.9E-02}$
WFG2	$3.52E-01_{5.5E-02}$	$2.52E-01_{1.9E-02}$	$2.53E-01_{1.9E-02}$	$3.17E-01_{4.1E-02}$	$2.33E-01_{2.8E-02}$	$3.33E-01_{5.3E-02}$	$3.01E-01_{4.3E-02}$	$3.21E-01_{4.2E-02}$
WFG3	$1.54E-01_{5.2E-02}$	$1.34E-01_{1.6E-02}$	$1.43E-01_{1.8E-02}$	$1.94E-01_{7.4E-02}$	$1.61E-01_{3.4E-02}$	$1.51E-01_{6.2E-02}$	$1.67E-01_{6.4E-02}$	$2.88E-01_{5.7E-02}$
WFG4	$3.85E-01_{9.3E-02}$	$7.54E-01_{9.0E-02}$	$7.45E-01_{7.4E-02}$	$4.07E-01_{5.4E-02}$	$3.18E-01_{4.5E-02}$	$3.92E-01_{5.3E-02}$	$4.23E-01_{9.5E-02}$	$4.68E-01_{9.8E-02}$
WFG5	$4.33E-01_{5.9E-02}$	$7.80E-01_{3.5E-02}$	$7.72E-01_{5.6E-02}$	$4.39E-01_{6.2E-02}$	$3.39E-01_{4.4E-02}$	$4.51E-01_{9.0E-02}$	$4.34E-01_{7.8E-02}$	$4.67E-01_{8.7E-02}$
WFG6	$4.82E-01_{1.1E-01}$	$7.48E-01_{3.0E-02}$	$7.36E-01_{3.4E-02}$	$4.53E-01_{9.1E-02}$	$3.23E-01_{3.0E-02}$	$4.52E-01_{7.2E-02}$	$4.35E-01_{7.7E-02}$	$4.46E-01_{5.4E-02}$
WFG7	$3.85E-01_{9.9E-02}$	$7.34E-01_{4.1E-02}$	$7.41E-01_{3.4E-02}$	$4.31E-01_{8.9E-02}$	$3.40E-01_{5.2E-02}$	$4.37E-01_{1.0E-01}$	$4.17E-01_{9.5E-02}$	$4.89E-01_{6.2E-02}$
WFG8	$7.22E-01_{9.4E-02}$	$7.78E-01_{2.6E-02}$	$7.77E-01_{2.7E-02}$	$7.54E-01_{1.3E-01}$	$5.35E-01_{1.7E-01}$	$7.26E-01_{6.5E-02}$	$7.63E-01_{5.9E-02}$	$7.64E-01_{7.7E-02}$
WFG9	$4.50E-01_{5.8E-02}$	$6.75E-01_{4.3E-02}$	$6.63E-01_{3.9E-02}$	$4.65E-01_{8.1E-02}$	$3.32E-01_{4.2E-02}$	$4.61E-01_{7.8E-02}$	$4.36E-01_{9.0E-02}$	$4.44E-01_{7.7E-02}$

The Friedman test results on the I_ϵ^+ performance metric are reported in Table 9. MOVMO is the second best-ranked algorithm following SPEA2. NSGA-II rose to the third place despite the fact that it didn't individually rank first or second in any function. The two decomposition-based optimizers tail the list.

Table 9: Friedman test. Average ranks of the algorithms in expected I_ϵ^+ with three objectives (distributed according to χ^2 with 7 degrees of freedom: 18.49019607843132).

Algorithm	Average Rank
SPEA2	2.824
MOVMO	3.412
NSGAII	4.412
AbYSS	4.647
SMPSO	4.706
MOCell	5.0
MOEAD	5.176
MOEAD.DRA	5.824

6.2.2 3Obj Hypervolume Results

The hypervolume results portrayed in Table 10 show that MOVMO reaches the best (highest) values on five out of seventeen benchmark functions, and ranked second place on four of them. SPEA2 had the best behavior with nine first places and two second bests.

Table 10: I_{HV} median and IQR values for the 3Obj configuration.

	MOVMO	MOEAD.DRA	MOEAD	NSGAII	SPEA2	AbYSS	MOCell	SMPSO
LZ09.F6	$1.69E-01_{6.4E-02}$	$2.80E-01_{1.3E-02}$	$2.87E-01_{1.5E-02}$	$1.59E-01_{6.3E-02}$	$2.53E-01_{1.5E-02}$	$1.19E-01_{7.3E-02}$	$1.47E-01_{5.6E-02}$	$1.71E-01_{5.0E-02}$
DTLZ1	$7.51E-01_{6.9E-01}$	$2.11E-01_{7.4E-01}$	$7.46E-01_{2.5E-02}$	$7.38E-01_{7.2E-02}$	$7.66E-01_{1.5E-02}$	$7.05E-01_{7.0E-01}$	$4.26E-02_{4.9E-01}$	$7.38E-01_{1.2E-02}$
DTLZ2	$3.84E-01_{5.5E-03}$	$3.65E-01_{3.6E-03}$	$3.70E-01_{2.6E-03}$	$3.74E-01_{7.2E-03}$	$4.04E-01_{2.3E-03}$	$3.79E-01_{8.3E-03}$	$3.74E-01_{7.6E-03}$	$3.47E-01_{6.0E-03}$
DTLZ3	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$1.82E-01_{3.4E-01}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$3.51E-01_{9.4E-02}$
DTLZ4	$3.81E-01_{7.5E-03}$	$3.61E-01_{7.0E-02}$	$3.60E-01_{7.1E-02}$	$3.76E-01_{8.6E-03}$	$3.98E-01_{1.9E-01}$	$3.87E-01_{6.3E-03}$	$3.77E-01_{7.5E-03}$	$3.59E-01_{9.8E-03}$
DTLZ5	$9.40E-02_{3.1E-05}$	$8.93E-02_{4.3E-04}$	$9.00E-02_{1.0E-04}$	$9.28E-02_{2.6E-04}$	$9.32E-02_{2.0E-04}$	$9.40E-02_{5.7E-05}$	$9.40E-02_{4.1E-05}$	$9.38E-02_{1.2E-04}$
DTLZ6	$9.49E-02_{7.7E-05}$	$9.14E-02_{1.0E-04}$	$9.13E-02_{1.3E-05}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$9.49E-02_{5.4E-05}$
DTLZ7	$2.91E-01_{4.4E-03}$	$8.81E-02_{7.6E-02}$	$1.76E-01_{5.9E-02}$	$2.78E-01_{7.0E-03}$	$2.90E-01_{2.9E-03}$	$2.59E-01_{7.1E-02}$	$2.71E-01_{2.2E-02}$	$2.73E-01_{1.2E-02}$
WFG1	$8.94E-01_{1.8E-02}$	$2.45E-01_{1.8E-02}$	$2.36E-01_{2.7E-02}$	$7.80E-01_{4.6E-02}$	$6.81E-01_{6.5E-02}$	$8.82E-01_{1.5E-01}$	$6.73E-01_{1.2E-01}$	$9.23E-02_{2.0E-02}$
WFG2	$8.96E-01_{8.3E-03}$	$8.81E-01_{5.0E-03}$	$8.80E-01_{5.9E-03}$	$8.98E-01_{6.0E-03}$	$9.11E-01_{2.5E-03}$	$8.98E-01_{6.2E-03}$	$8.99E-01_{4.8E-03}$	$8.83E-01_{4.6E-03}$
WFG3	$3.15E-01_{5.3E-03}$	$3.07E-01_{2.3E-03}$	$3.07E-01_{2.5E-03}$	$3.11E-01_{2.6E-03}$	$2.93E-01_{5.5E-03}$	$3.17E-01_{5.3E-03}$	$3.14E-01_{3.0E-03}$	$2.92E-01_{6.7E-03}$
WFG4	$3.85E-01_{5.5E-03}$	$3.22E-01_{9.7E-03}$	$3.23E-01_{9.4E-03}$	$3.68E-01_{8.3E-03}$	$3.92E-01_{2.3E-03}$	$3.87E-01_{6.9E-03}$	$3.72E-01_{4.6E-03}$	$3.23E-01_{9.9E-03}$
WFG5	$3.52E-01_{6.8E-03}$	$3.06E-01_{2.6E-03}$	$3.05E-01_{3.6E-03}$	$3.43E-01_{8.2E-03}$	$3.68E-01_{5.7E-03}$	$3.51E-01_{5.5E-03}$	$3.50E-01_{4.4E-03}$	$3.39E-01_{8.5E-03}$
WFG6	$3.65E-01_{2.1E-02}$	$3.40E-01_{4.8E-03}$	$3.40E-01_{6.2E-03}$	$3.63E-01_{1.6E-02}$	$3.96E-01_{1.3E-02}$	$3.51E-01_{4.2E-02}$	$3.71E-01_{1.5E-02}$	$3.46E-01_{9.9E-03}$
WFG7	$3.83E-01_{6.0E-03}$	$3.32E-01_{6.2E-03}$	$3.31E-01_{3.9E-03}$	$3.58E-01_{1.1E-02}$	$3.82E-01_{5.6E-03}$	$3.77E-01_{7.6E-03}$	$3.67E-01_{9.2E-03}$	$3.35E-01_{8.2E-03}$
WFG8	$2.49E-01_{5.6E-03}$	$2.25E-01_{7.9E-03}$	$2.28E-01_{6.9E-03}$	$2.44E-01_{6.6E-03}$	$2.73E-01_{1.1E-02}$	$2.50E-01_{9.1E-03}$	$2.48E-01_{6.3E-03}$	$2.29E-01_{9.5E-03}$
WFG9	$3.59E-01_{1.2E-02}$	$3.30E-01_{5.2E-03}$	$3.30E-01_{3.4E-03}$	$3.55E-01_{9.4E-03}$	$3.77E-01_{4.2E-03}$	$3.55E-01_{1.0E-02}$	$3.61E-01_{7.7E-03}$	$3.54E-01_{9.2E-03}$

MOVMO achieves a statistically significant performance difference at the 95% level over its rivals on DTLZ5, DTLZ6 and WFG1 but not over SPEA2 on DTLZ7 or WFG7. Moreover, the Wilcoxon test suggests that the slightly superior hypervolume values produced by AbYSS in WFG3 (1st place), WFG4 (2nd place), WFG8 (2nd place) and by MOCell in WFG6 (2nd place) and WFG9 (2nd place) are not not statistically significant to those by MOVMO. Finally, MOVMO does not fare well with LZ09.F6 but it is not the worst there either.

The Friedman test results on the I_{HV} performance metric for the 3Obj configuration are reported in Table 11.

Table 11: Friedman test. Average ranks of the algorithms in expected I_{HV} with three objectives (distributed according to χ^2 with 7 degrees of freedom: 28.338235294117517).

Algorithm	Ranking
SPEA2	2.941
MOVMO	2.971
AbYSS	4.088
NSGAII	4.235
MOCell	4.588
SMPSO	5.176
MOEAD	5.882
MOEAD.DRA	6.118

6.2.3 3Obj Discussion

Table 12 sums up the set of pairwise comparisons between MOVMO and the rest of the algorithms in terms of the additive epsilon and hypervolume quality indicators for the 3Obj configuration. MOVMO's performance was compared to that of the seven state-of-the-art optimizers throughout seventeen benchmark functions (17 $\times$ 7 = 119 pairwise tests).

Table 12: MOVMO pairwise comparisons - Wilcoxon test summary for the 3Obj configuration.

	Epsilon	Hypervolume	Total
+	57	83	140
-	28	17	45
=	34	19	53

Table 12 reveals that MOVMO outdid its competitors for the two quality indicators between two and five times as much as it was defeated by them. We may see this pattern more stressed in regards to the hypervolume metric. This is somewhat expected as MOVMO had shown great promise regarding the spread of its approximation sets and the hypervolume metric aims at capturing that aspect too.

6.3 Final discussion

The *no free lunch* theorem explicitly states that what an algorithm gains in performance on one class of problems it necessarily pays for on the remaining problems [20]. According to the empirical results shown in this article, our MOVMO algorithm in general reaches a good performance on the ZDT, DTLZ and WFG test suites in two and three-objective configurations. These functions tested MOVMO's performance on problems with convex, concave and disconnected Pareto Fronts.

On the other hand, MOVMO did not behave well on the LZ09 problems. This test suite is designed to evaluate algorithms performance in problems with complicated Pareto Sets and continuous, multi-modal and non-linear Pareto Fronts. In the very article where this suite was proposed, the authors recognize that the SBX crossover operator did not perform well in these test problems. This is mainly because, at an early stage of the search process, SBX loses diversity, which is needed for exploring the search space effectively. MOVMO implements SBX as crossover operator; it shows good performance on the ZDT, DTLZ and WFG test suites without the assistance of other genetic operators such as mutation. However, the performance of this operator deteriorates considerably when it is applied to functions with *epistasis* [17] among variables, i.e., nonlinearities in fitness functions due to changes in the values of interacting variables. This is because SBX always reduces the covariance of the offspring distribution close to zero. We hence concluded that this operator may not be suitable for dealing with the LZ09 test instances, which exhibit *epistasis* in all its functions.

To overcome this drawback, we recommend changing the type of crossover operator (Line 12 Algorithm 1) to one that works better with LZ09 functions. For instance, the authors in [9] report that the classical Differential Evolution (DE) crossover operator proposed by Price et. al. in [18] rendered encouraging results in presence of the LZ09 functions.

With this idea in mind, we conducted several experiments where the original SBX crossover operator in MOVMO was changed to a DE-related one. This implementation is hereafter called MOVMO-DE. During the experiments we tested several variants of DE crossover operators, parent selection strategies as well as the inclusion or not of a mutation operator. We used the same experimental framework outlined in Section 5 and the following parameter settings: $CR = 1.0$ as crossover probability and $F = 0.5$ as amplification factor of the differential vector. For the polynomial mutation operator, we considered a distribution index of $n_m = 20$ and mutation rate $p_m = 1/D$, where D is the number of decision variables (solution dimensionality). For a detailed explanation about the difference among several DE crossover operators and the role of each parent selection strategy during the recombination, we refer the reader to [10].

The best results shown by MOVMO-DE on the LZ09 test suite were obtained with the *best/1/bin* DE crossover operator and the inclusion of the polynomial mutation operator after DE crossover. The most successful parent selection strategy was applying binary tournament to the solutions from the archive set and using the local best solution n_i^* (Line 6 Algorithm 1) and the locally produced solution n_l (Line 9 Algorithm 1) to calculate the mutation differential.

Table 13 summarizes the set of pairwise comparisons between MOVMO-DE and MOVMO throughout the set of LZ09 benchmark functions for the 2Obj configuration. It is clear from Table 13 that MOVMO-DE outperformed its competitor across each quality indicator, especially on the spread metric.

Table 13: MOVMO-DE vs. MOVMO pairwise comparison - Wilcoxon test summary for the LZ09 test suite.

	Epsilon	Spread	Hypervolume	Total
+	6	7	5	18
-	2	0	2	4
=	0	1	1	2

Table 14 depicts the ranks achieved by MOVMO and MOVMO-DE on the Friedman test for the LZ09 functions over the I_ϵ^+, I_{SP} and I_{HV} quality indicators. In this case MOVMO-DE improved Friedman test results across all quality indicators.

Table 14: Ranks obtained by MOVMO and MOVMO-DE according to the Friedman test for the LZ09 test functions along the I_ϵ^+, I_{SP} and I_{HV} quality indicators.

	Epsilon	Spread	Hypervolume
MOVMO	6^{th}	2^{nd}	5^{th}
MOVMO-DE	3^{rd}	1^{st}	4^{th}

The experimental results indeed confirm that, after changing the crossover operator, MOVMO-DE achieved a better performance on the LZ09 test suite and, consequently, on those functions with entangled Pareto sets. It is important to highlight that MOVMO-DE did not fare very well on the ZDT, DTLZ and WFG test functions in comparison to the original MOVMO implementation. This suggests the need to dynamically adapt the set of genetic operators during the search process; however, such endeavor is beyond the scope of the current study.

7 Conclusions

In this paper we have introduced MOVMO, a new multiobjective metaheuristic optimizer based on variable mesh optimization. Our method combines typical concepts from the MOO realm such as Pareto dominance, density estimation and an external archive to ensure elitism; it also performs crossover between local and global optima and features a dynamic population replacement. MOVMO was empirically compared against seven other state-of-the-art multiobjective optimizers on twenty-nine bi-objective and seventeen three-objective functions coming from four popular families. The experimental evidence suggests that approximations of the Pareto optimal front obtained via MOVMO are competitive and in many cases outperform those produced by other techniques. Future work will concentrate on testing MOVMO's behavior in presence of higher-dimensional functions as well as its application to a real-world problem.

Acknowledgements This work was funded by the eureka SD project (agreement number 2013-2591), that is supported by the Erasmus Mundus programme of the European Union.

References

[1] Deb, K.: Multi-objective Optimization. In: E.K. Burke, G. Kendall (eds.) Search Methodologies. Introductory tutorials in optimization and decision support techniques, chap. 10, pp. 273–316. Springer Science and Business Media (2014)
[2] Deb, K.: Introduction to Evolutionary Multiobjective Optimization. In: J. Branke, K. Deb, K. Miettinen, R. Sowiski (eds.) Multiobjective Optimization, LNCS, vol. 5252, chap. 3, pp. 59–96. Springer Berlin Heidelberg (2008)
[3] Deb, K., Thiele, L., Laumanns, M., Zitzler, E., Abraham, A., Jain, L., Goldberg, R.: Scalable test problems for evolutionary multiobjective optimization. Evolutionary Multiobjective Optimization pp. 105–145 (2005)
[4] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)
[5] Derrac, J., Garca, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation 1(1), 3–18 (2011)
[6] Durillo, J.J., Nebro, A.J.: JMetal: A Java framework for multi-objective optimization. Advances in Engineering Software 42(10), 760–771 (2011)
[7] Huband, S., Barone, L., While, L., Hingston, P.: A Scalable Multi-objective Test Problem Toolkit. In: C.A. Coello, A.H. Aguirre, E. Zitzler (eds.) Evolutionary Multi-Criterion Optimization, LNCS, vol. 3410, pp. 280–295. Springer Berlin Heidelberg (2005)
[8] Knowles, J., Thiele, L., Zitzler, E.: A tutorial on the performance assessment of stochastic multiobjective optimizers. Technical Report 214 (2006)
[9] Li, H., Zhang, Q.: Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. IEEE Transactions on Evolutionary Computation 13(2), 284–302 (2009)
[10] Mezura-Montes, E., Reyes-Sierra, M., Coello, C.: Multi-objective Optimization Using Differential Evolution: A Survey of the State-of-the-Art. In: U. Chakraborty (ed.) Advances in Differential Evolution, Studies in Computational Intelligence, vol. 143, pp. 173–196. Springer Berlin Heidelberg (2008)
[11] Molina, D., Puris, A., Bello, R., Herrera, F.: Variable Mesh Optimization for the 2013 CEC Special Session Niching Methods for Multimodal Optimization. In: IEEE Congress on Evolutionary Computation (CEC '13), pp. 87–94 (2013)
[12] Navarro, R., Falcon, R., Murata, T., Hae, K.C.: A Generic Niching Framework for Variable Mesh Optimization. In: 2015 IEEE Congress on Evolutionary Computation (CEC '15), pp. 1994–2001. Sendai, Japan (2015)

[13] Molina, D., Puris, A., Bello, R., Herrera, F.: Variable Mesh Optimization for the 2013 CEC Special Session Niching Methods for Multimodal Optimization. In: IEEE Congress on Evolutionary Computation (CEC '13), pp. 87–94 (2013)
[14] Nebro, A.J., Durillo, J.J., Garcia-Nieto, J., Coello Coello, C.A., Luna, F., Alba, E.: SMPSO: A new PSO-based metaheuristic for multi-objective optimization. In: IEEE Symposium on Computational intelligence in Multi-criteria Decision-Making, 2009. mcdm '09, pp. 66–73 (2009)
[15] Nebro, A.J., Durillo, J.J., Luna, F., Dorronsoro, B., Alba, E.: MOCell: A cellular genetic algorithm for multiobjective optimization. International Journal of Intelligent Systems 24(7), 726–746 (2009)
[16] Nebro, A.J., Luna, F., Alba, E., Dorronsoro, B., Durillo, J.J., Beham, A.: AbYSS: Adapting scatter search to multiobjective optimization. IEEE Transactions on Evolutionary Computation 12(4), 439–457 (2008)
[17] Ono, I., Kita, H., Kobayashi, S.: A Real-coded Genetic Algorithm using the Unimodal Normal Distribution Crossover. In: A. Ghosh, S. Tsutsui (eds.) Advances in Evolutionary Computing, chap. 8, pp. 213–237. Springer Berlin Heidelberg (2003)
[18] Price, K.V., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization. Springer-Verlag, New York, USA (2005)
[19] Puris, A., Bello, R., Molina, D., Herrera, F.: Variable mesh optimization for continuous optimization problems. Soft Computing 16(3), 511–525 (2012)
[20] Wolper, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation 1(1), 67–82 (1996)
[21] Zhang, Q., Liu, W., Li, H.: The performance of a new version of MOEA/D on CEC'09 unconstrained MOP test instances. In: IEEE Congress on Evolutionary Computation (CEC '09), pp. 203–208 (2009)
[22] Zhang, Q.Z.Q., Li, H.L.H.: MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Transactions on Evolutionary Computation 11(6), 712–731 (2007)
[23] Zhou, A., Qu, B.Y., Li, H., Zhao, S.Z., Suganthan, P.N., Zhang, Q.: Multiobjective evolutionary algorithms: A survey of the state of the art. Swarm and Evolutionary Computation 1(1), 32–49 (2011)
[24] Zitzler, E., Knowles, J.D., Thiele, L.: Quality assessment of pareto set approximations. In: J. Branke, K. Deb, K. Miettinen, R. Sowiski (eds.) Multiobjective optimization: interactive and evolutionary approaches, LNCS, vol. 5252, chap. 14, pp. 373–404. Springer Berlin Heidelberg (2008)
[25] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., da Fonseca, V.G.: Performance assessment of multiobjective optimizers: an analysis and review. IEEE Transactions on Evolutionary Computation 7(2), 117–132 (2003)
[26] Zitzler Eckart, L.M.T.L.: Spea2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems, pp. 95–100 (2001)
[27] Zitzler, E., Deb, K., Thiele, L.: Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. Evolutionary Computation 8(2), 173–195 (2000)