# MH-MOEA: A New Multi-Objective Evolutionary Algorithm based on the Maximin Fitness Function and the Hypervolume Indicator

Adriana Menchaca-Mendez and Carlos A. Coello Coello

CINVESTAV-IPN (Evolutionary Computation Group)
Departamento de Computación
México D.F. 07300, MÉXICO
adriana.menchacamendez@gmail.com, ccoello@cs.cinvestav.mx

**Abstract.** In this paper, we propose an approach that combines a modified version of the maximin fitness function and the hypervolume indicator for selecting individuals into a Multi-Objective Evolutionary Algorithm (MOEA). Our proposed selection mechanism is incorporated into a MOEA which adopts the crossover and mutation operators of the Nondominated Sorting Genetic Algorithm-II (NSGA-II), giving rise to the so-called "Maximin-Hypervolume Multi-Objective Evolutionary Algorithm (MH-MOEA)". Our proposed MH-MOEA is validated using standard test problems taken from the specialized literature, using from three to six objectives. Our results are compared with respect to those produced by MC-MOEA (which is based on the maximin fitness function and a clustering technique), MOEA/D using Penalty Boundary Intersection (PBI), which is based on decomposition and iSMS-EMOA (which is based on the hypervolume indicator). Our preliminary results indicate that our proposed MH-MOEA is a good alternative to solve multi-objective optimization problems having both low dimensionality and high dimensionality in objective function space.

## 1 Introduction

In the real world, there are many optimization problems which involve multiple objective functions (normally in conflict with each other) that need to be satisfied at the same time. They are called *multi-objective optimization problems (MOPs)*. In MOPs, the notion of optimality refers to the best possible trade-offs among all the objectives. Consequently, there are several optimal solutions (the so-called *Pareto optimal set* whose image is called the *Pareto front*). The use of evolutionary algorithms for solving MOPs has become very popular, giving rise to the so-called Multi-Objective Evolutionary Algorithms (MOEAs). We can classify MOEAs, based on their selection mechanism, into two groups: (i) those that incorporate the concept of Pareto optimality, and (ii) those that do not use Pareto dominance to select individuals. Since Pareto-based MOEAs have several limitations (from which the main one is that their behavior quickly degrades as

we increase the number of objectives), MOEAs of type (ii) have gained increasing popularity in the last few years.

We are interested in the *maximin fitness function* (MFF) and the *hypervolume indicator* ($I_H$). Both are of type (ii). MFF has some interesting properties and its complexity is linear with respect to the number of objective functions. $I_H$ is the only unary indicator which is known to be "Pareto compliant" [18]. The main disadvantage of MOEAs based on $I_H$ is their high computational cost. In this paper, we propose a hybrid of MFF and $I_H$ for selecting individuals into a MOEA. The motivation behind this proposal is to alleviate the disadvantages of MFF which does not select well-distributed individuals [13, 15]. Our conjecture is that it is possible to improve the approximation of the Pareto optimal set obtained by a MOEA based on MFF, if we can improve the diversity of the population at each generation. Therefore, we propose to use $I_H$ to correct the possible errors produced when selecting with MFF. Finally, we incorporate our new selection mechanism into a MOEA that uses the crossover and mutation operators of NSGA-II to create new individuals. Our proposed MOEA is called *"Maximin-Hypervolume Multi-Objective Evolutionary Algorithm (MH-MOEA)"*.

The remainder of this paper is organized as follows. Section 2 states the problem of our interest. The maximin fitness function is described in Section 3. Section 4 describes the hypervolume indicator. Our proposal is discussed in Section 5. Our experimental validation and the results obtained are shown in Section 6. Finally, we provide our conclusions and some possible paths for future work in Section 7.

## 2 Problem Statement

We are interested in the general *multi-objective optimization problem (MOP)*, which is defined as follows: Find $\boldsymbol{x}^* = [x_1^*, x_2^*, \ldots, x_n^*]^T$ which optimizes

$$\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x})]^T \tag{1}$$

such that $\boldsymbol{x}^* \in \Omega$, where $\Omega \subset R^n$ defines the feasible region of the problem. Assuming minimization problems, we have the following definitions.

**Definition 1** *We say that a vector* $\boldsymbol{x} = [x_1, \ldots, x_n]^T$ *dominates vector* $\boldsymbol{y} = [y_1, \ldots, y_n]^T$*, denoted by* $\boldsymbol{x} \prec \boldsymbol{y}$*, if and only if* $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{y})$ *for all* $i \in \{1, ..., k\}$ *and there exists an* $i \in \{1, \ldots, k\}$ *such that* $f_i(\boldsymbol{x}) < f_i(\boldsymbol{y})$.

**Definition 2** *We say that a vector* $\boldsymbol{x} = [x_1, \ldots, x_n]^T$ *weakly dominates vector* $\boldsymbol{y} = [y_1, \ldots, y_n]^T$*, denoted by* $\boldsymbol{x} \preceq \boldsymbol{y}$*, if* $\boldsymbol{x}$ *is not worse than* $\boldsymbol{y}$ *in all objectives.*

**Definition 3** *A point* $\boldsymbol{x}^* \in \Omega$ *is Pareto optimal if there does not exist any* $\boldsymbol{x} \in \Omega$ *such that* $\boldsymbol{x} \prec \boldsymbol{x}^*$.

**Definition 4** *For a given MOP,* $\boldsymbol{f}(\boldsymbol{x})$*, the Pareto optimal set is defined as:* $\mathcal{P}^* = \{\boldsymbol{x} \in \Omega | \neg\exists \boldsymbol{y} \in \Omega : \boldsymbol{f}(\boldsymbol{y}) \prec \boldsymbol{f}(\boldsymbol{x})\}$.

**Definition 5** *Let* $\boldsymbol{f}(\boldsymbol{x})$ *be a given MOP and* $\mathcal{P}^*$ *the Pareto optimal set. Then, the Pareto Front is defined as:* $\mathcal{PF}^* = \{\boldsymbol{f}(\boldsymbol{x}) \mid \boldsymbol{x} \in \mathcal{P}^*\}$.

## 3 Maximin Fitness Function

The maximin fitness function (MFF) was proposed by Balling [3]. Balling and Wilson proposed the modified MFF [5] which works as follows. Let's consider a MOP with $K$ objective functions. Let $f_k^i$ be the normalized value of the $k^{th}$ objective for the $i^{th}$ individual in a particular generation. Then, the modified MFF of individual $i$ is defined as:

$$fitness^i = max_{j \neq i, j \in \mathcal{ND}}(min_k(f_k^i - f_k^j)) \tag{2}$$

where $\mathcal{ND}$ is the set of non-dominated individuals. The $min$ is taken over all the objective functions, and the $max$ is taken over all non-dominated individuals in the population, except for the same individual $i$. From eq. (2), we can say the following: any individual whose maximin fitness is greater than zero is a dominated individual, any individual whose maximin fitness is less than zero is a non-dominated individual and, any individual whose maximin fitness is equal to zero is a weakly nondominated individual. Also, MFF penalizes clustering of non-dominated individuals and the maximin fitness of dominated individuals is a metric of the distance to the non-dominated front.

MFF and its modified version have been incorporated into evolutionary algorithms such as genetic algorithms [5, 4], particle swarm [11, 12] and ant colony [10] optimizers. However, in those papers, only low dimensionality MOPs were considered and no extra diversity mechanism was adopted based on the idea that MFF penalizes clustering. In recent years, two important disadvantages of MFF were identified in [13]. The main disadvantage of MFF is related to the following question: **Is it better to prefer weakly nondominated individuals to dominated individuals?** The answer provided in [13] was that it is not good to prefer weakly nondominated individuals (even if they are weakly nondominated by any dominated individual). The authors showed in [13, 15] that if we use a MOEA based on MFF to solve a MOP in which one objective function is easier to solve than the others, it is likely that the MOEA only obtains weakly Pareto points or that its convergence slows down. In order to address this problem, the following constraint was proposed in [13]: *Any individual that we want to select must not be similar (in any objective function) to another (selected) individual.*

The second disadvantage of MFF has to do with the poor diversity obtained in objective function space when we use it to select individuals. In [13, 15], the authors proposed to combine either MFF or its modified version with a clustering technique in order to improve diversity.

## 4 Hypervolume Indicator

The hypervolume indicator ($I_H$) was originally proposed by Zitzler and Thiele in [17]. If $\Lambda$ denotes the Lebesgue measure, $I_H$ is defined as:

$$I_H(\mathcal{A}, \boldsymbol{y}_{ref}) = \Lambda\left(\bigcup_{\boldsymbol{y} \in \mathcal{A}} \{\boldsymbol{y}' \mid \boldsymbol{y} \prec \boldsymbol{y}' \prec \boldsymbol{y}_{ref}\}\right) \tag{3}$$

where $\boldsymbol{y}_{ref} \in \mathbb{R}^k$ denotes a reference point that should be dominated by all the Pareto optimal points. The contribution to $I_H$ of a solution $\boldsymbol{x}$ is defined as:

$$C_H(\boldsymbol{x}, \mathcal{A}) = I_H(\mathcal{A}, \boldsymbol{y}_{ref}) - I_H(\mathcal{A} \setminus \boldsymbol{x}, \boldsymbol{y}_{ref}) \tag{4}$$

where $\boldsymbol{x} \in \mathcal{A}$. Then, the contribution of $\boldsymbol{x}$ is the space that is only covered by $\boldsymbol{x}$.

Perhaps, the most popular MOEA based on $I_H$ is the S metric selection Evolutionary Multi-Objective Algorithm (SMS-EMOA) [8]. SMS-EMOA generates only one solution by iteration using the crossover and mutation operators of the NSGA-II. After that, it applies Pareto ranking. When the last front has more than one solution, SMS-EMOA uses the contribution to $I_H$ to decide which solution will be removed. Therefore, when all individuals are non-dominated, SMS-EMOA needs to calculate the contribution to $I_H$ of all individuals in the population and the contribution of the new individual. This is not good because we know that calculating these contributions is computationally expensive for more than three objective functions. There are other MOEAs based on $I_H$. However, most of them use the same competition scheme. Recently, a new selection scheme based on $I_H$ and its locality property [2, 1] was proposed in [14]. It works as follows: Let's assume that at each iteration of a MOEA, only one solution $\boldsymbol{x}_{new}$ is created and the current population is $\mathcal{P}$. Then, we choose the nearest neighbor ($\boldsymbol{x}_{near}$) of $\boldsymbol{x}_{new}$ in $\mathcal{P}$ and we also choose (randomly) another solution, $\boldsymbol{x}_{rand}$, such that $\boldsymbol{x}_{rand} \in \mathcal{P}$ and $\boldsymbol{x}_{rand} \neq \boldsymbol{x}_{near}$. After that, $\boldsymbol{x}_{rand}$, $\boldsymbol{x}_{new}$ and $\boldsymbol{x}_{near}$) will compete to survive. The solution with the worst contribution to $I_H$ is eliminated.

## 5 Our Proposed Approach

We propose here a selection mechanism based on the modified MFF and $I_H$. The idea is to use the modified MFF as our main selection mechanism and $I_H$ to correct its possible errors. Unlike SMS-EMOA [8] or iSMS-EMOA [14], in which only one individual is created per iteration, our mechanism is designed to work with population schemes. This is possible for two reasons: The maximin fitness of each individual determines the order in which each individual competes to survive using $I_H$ and in the competition scheme proposed in [14] each individual only competes with two other individuals of the population (its nearest neighbor and a randomly selected individual). Then, the combinatorial problem no longer exists.

Our selection mechanism works as follows: If we want to select $S$ individuals from a population $\mathcal{P}$, we assign first a fitness value to each individual using the modified MFF (see eq. (2)). Then, we proceed to select the individuals according to their fitness, verifying similarity between selected individuals (see Algorithm 1, lines 5 to 11). If we consider all individuals in the population and we do not select $S$ individuals, we select the remaining individuals considering only the maximin fitness (see Algorithm 1, lines 13 to 20). If we already selected the $S$ individuals but there are still non-dominated individuals in $\mathcal{P}$ who have not participated in the selection process, then, we proceed to use the contribution to $I_H$ as follows: Let $\mathcal{S}$ be the set of current selected individuals. Then, for each

nondominated individual $\mathcal{P}[i]$ who has not participated in the selection process, we obtain the index of its nearest neighbor in $\mathcal{S}$ (we call it $NN$) and we choose a random index $RI$ such that $RI \in \{1, \cdots, ||\mathcal{S}||\}$ and $RI \neq NN$. Finally, we calculate the contribution to $I_H$ of $\mathcal{P}[i]$, $\mathcal{S}[NN]$ and $\mathcal{S}[RI]$. If $\mathcal{P}[i]$ has a better contribution than $\mathcal{S}[NN]$ or $\mathcal{S}[RI]$, then $\mathcal{P}[i]$ replaces the individual with the worst contribution ($\mathcal{S}[NN]$ or $\mathcal{S}[RI]$). See Algorithm 1, lines 22 to 34). The process to verify similarity between individuals is shown in Algorithm 2, where $min_dif$ is the minimum difference allowed between solutions with respect to all objective functions and $K$ is the number of objective functions.

In order to evaluate our new selection mechanism, we incorporate it into a MOEA that uses the crossover and mutation operators of NSGA-II to create new individuals. The proposed MOEA is called **"Maximin-Hypervolume Multi-Objective Evolutionary Algorithm (MH-MOEA)"** and it works as follows: If the size of the population is $P$, then we create $P$ new individuals. We use a binary tournament to select the parents. At each tournament, two individuals are randomly selected and the one with the higher maximin fitness value is chosen. After that, we combine the population of parents and offspring to obtain a population of size $2P$. Then, we use our proposed selection mechanism to choose the $P$ individuals that will take part of the following generation. This process is repeated for a certain (pre-defined) number of generations.

## 6 Experimental Results

We compared our proposed MH-MOEA with respect to MC-MOEA [15] (the version in which the modified MFF is used all the time), MOEA/D [16] (using PBI to decompose the MOP[1]) and iSMS-EMOA [14].[2] For MOEA/D, we generated the convex weights using the technique proposed in [6] and after that, we applied clustering ($k$-means) to obtain a specific number of weights. It is worth noticing that all of these MOEAs use the same operators to create new individuals, which allows a fair comparison of the selection operators.

For our experiments, we adopted seven problems from the Deb-Thiele-Laumanns-Zitzler (DTLZ) test suite [7]. We used $k = 5$ for DTLZ1, DTLZ3 and DTLZ6 and $k = 10$ for the remaining test problems. Also, we adopted seven problems from the Walking-Fish Group (WFG) toolkit [9], with $k_factor = 2$ and $l_factor = 10$. For each test problem, we performed 30 independent runs. For all algorithms, we adopted the parameters suggested by the authors of NSGA-II: $p_c = 0.9$ (crossover probability), $p_m = 1/n$ (mutation probability), where $n$ is the number of decision variables. Both for the crossover and mutation operators, we adopted $\eta_c = 15$ and $\eta_m = 20$, respectively. In the case of MC-MOEA and our MH-MOEA, we used $min_dif = 0.0001$. We performed a maximum of

[1] We decided to use the PBI approach because the resultant optimal solutions in the PBI should have a more uniform distribution than those obtained by the Tchebycheff approach [16].

[2] The source code of the all algorithms used here can be provided by the first author upon request.

**Algorithm 1:** Maximin-Hypervolume Selection

```
Input  : 𝒫 (Population), S (number of individuals to choose S < ‖𝒫‖).
Output: 𝒮 (Selected individuals).
/*Assign fitness to each individual in the population, using the modified maximin
  fitness function                                                              */
AssignFitness(𝒫);
numNonDom ← Number of nondominated solutions in 𝒫;
/*Sorting with respect to the maximin fitness                                   */
Sort(𝒫);
s ← 1, i ← 1, 𝒮 ← ∅;
/*Fill up the new population with the best copies according to the maximin fitness,
  verifying that there is not a similar one                                     */
while s ≤ S AND i ≤ ‖𝒫‖ do
    if 𝒫[i] is not similar to any individual in 𝒮 then
        /*Select individual i                                                   */
        𝒮 ← 𝒮 ∪ 𝒫[i];
        s ← s + 1;
    end
    i ← i + 1;
end
if s ≤ S then
    /*Choose the remaining individuals considering only the maximin fitness     */
    i ← 1;
    while s ≤ S do
        if 𝒫[i] has not been selected then
            𝒮 ← 𝒮 ∪ 𝒫[i];
            s ← s + 1;
        end
        i ← i + 1;
    end
else
    /*Improve the diversity according to the contribution to I_H               */
    while i < numNonDom do
        if 𝒫[i] is not similar to any individual in 𝒮 then
            NN ← Index of nearest neighbor to 𝒫[i] in 𝒮;
            RI ← Index of a randomly selected individual in 𝒮 such that NN ≠ RI;
            /*Calculate the contributions to the hypervolume                    */
            C_NN ← C_H(𝒮[NN], 𝒮);
            C_RI ← C_H(𝒮[RI], 𝒮);
            C_i ← C_H(𝒫[i], 𝒮);
            /*Remove the individual with the worst contribution                 */
            worst ← Index of the individual with the worst contribution (NN, RI or i);
            if worst = NN or worst = RI then
                Replace 𝒮[worst] with 𝒫[i];
            end
        end
    end
end
return 𝒮;
```

**Algorithm 2:** Verify similarity

```
Input  : x (individual), 𝒮 (population).
Output: Returns 1, if the individual x is similar to any individual in the population 𝒮;
        otherwise, returns 0.
for i ← 1 to ‖𝒮‖ do
    for k ← 1 to K do
        if |x.f[k] − 𝒮[i].f[k]| < min_dif then
            return 1;
        end
    end
end
return 0;
```

50,000 fitness function evaluations (we used a population size of 100 individuals and we iterated for 500 generations). Only in DTLZ3 we performed 100,000 evaluations (we used a population size of 100 individuals and we iterated for 1000 generations).

### 6.1 Performance Indicators

We adopted only $I_H$ to validate our results because it rewards both convergence towards the Pareto front as well as the maximum spread of the solutions obtained. Also, $I_H$ is Pareto compliant. To calculate the hypervolume indicator, we used the following reference points: $y_{ref} = [y_1, \cdots, y_M]$ such that $y_i = 0.7$ for DTLZ1, $y_{ref} = [y_1, \cdots, y_M]$ such that $y_i = 1.1$ for DTLZ(2-6), $y_{ref} = [y_1, \cdots, y_M]$ such that $y_M = 6.1$ and $y_{i \neq M} = 1.1$ for DTLZ7. In the case of the WFG test problems, we generated the reference point using a value slightly higher than the highest value found for each objective function taking into account all the outputs of the algorithms.

### 6.2 Discussion of Results

In Table 1, we present the results obtained with respect to $I_H$ as well as the statistical analysis applied to the experiments using Wilcoxon's rank sum. In (a), we can see that our MH-MOEA obtained better results than MC-MOEA in thirteen problems. It is important to see that we can reject the null hypothesis "medians are equal" in all cases, and then, we can say that in these problems our MH-MOEA outperformed MC-MOEA. Only in DTLZ1, MC-MOEA outperformed our MH-MOEA. In (b), we compare our MH-MOEA with respect to MOEA/D and we can see that it outperformed MOEA/D in thirteen cases and, only in DTLZ1, MOEA/D outperformed our MH-MOEA. Finally, in (c), we can see that iSMS-EMOA outperformed our MH-MOEA in six problems, our MH-MOEA outperformed iSMS-EMOA in four problems and, in four problems, we can observe that the null hypothesis cannot be rejected, which means that both algorithms have a similar behavior. Also, in (d), (e) and (f), we show an scalability analysis with respect to the number of objectives for some of the problems adopted, using four, five and six objective functions. In these Tables, we can see that our MH-MOEA continues to work well when we increase the number of objectives. For example, we can say that our MH-MOEA outperformed MC-MOEA and MOEA/D because it obtained better results with respect to $I_H$, in all problems, and only in two cases, the null hypothesis cannot be rejected: In DTLZ3 with six objectives, our MH-MOEA has a behavior similar to MC-MOEA and in DTLZ7 with six objective functions, our MH-MOEA has a similar behavior to MOEA/D. With respect to iSMS-EMOA, our MH-MOEA outperformed iSMS-EMOA in two problems, it is outperformed by iSMS-EMOA in four problems and they have a similar behavior in three problems. Finally, in Table 2, we present plots of the average running time required by each algorithm to find the approximation of the Pareto optimal set in the problems adopted for the scalability analysis and we can note that MOEA/D and MC-MOEA are the

fastest algorithms. However, in Table 1, we saw that they are outperformed by our proposed MH-MOA. An interesting thing is that our MH-MOEA requires less running time than iSMS-EMOA and, as we saw in Table 1, it obtains competitive results with respect to iSMS-EMOA.

| $f$ | mc-moea $I_H$ | mh-moea $I_H$ | $P(H)$ |
|---|---|---|---|
| DTLZ1(3) | **0.311634 (0.001536)** | 0.301778 (0.059500) | 0.000 (1) |
| DTLZ2(3) | 0.696078 (0.008462) | **0.757779 (0.000125)** | 0.000 (1) |
| DTLZ3(3) | 0.666299 (0.022591) | **0.732405 (0.073179)** | 0.000 (1) |
| DTLZ4(3) | 0.703924 (0.008896) | **0.757701 (0.000329)** | 0.000 (1) |
| DTLZ5(3) | 0.424854 (0.005358) | **0.439303 (0.000034)** | 0.000 (1) |
| DTLZ6(3) | 0.381283 (0.026286) | **0.406017 (0.020026)** | 0.002 (1) |
| DTLZ7(3) | 1.851081 (0.202273) | **1.939227 (0.218993)** | 0.000 (1) |
| WFG1(3) | 17.622577 (1.379125) | **21.038451 (0.602411)** | 0.000 (1) |
| WFG2(3) | 0.115221 (0.009956) | **0.118381 (0.012508)** | 0.016 (1) |
| WFG3(3) | 0.407785 (0.006808) | **0.465227 (0.001959)** | 0.000 (1) |
| WFG4(3) | 20.858572 (0.557784) | **33.806224 (0.339882)** | 0.000 (1) |
| WFG5(3) | 7.997923 (0.203347) | **9.877391 (0.011298)** | 0.000 (1) |
| WFG6(3) | 0.920378 (0.012003) | **1.024659 (0.003593)** | 0.000 (1) |
| WFG7(3) | 17.286919 (0.727687) | **22.748887 (0.086435)** | 0.000 (1) |

(a)

| $f$ | moead $I_H$ | mh-moea $I_H$ | $P(H)$ |
|---|---|---|---|
| | **0.303053 (0.000450)** | 0.301778 (0.059500) | 0.000 (1) |
| | 0.708105 (0.000205) | **0.757779 (0.000125)** | 0.000 (1) |
| | 0.702575 (0.004807) | **0.732405 (0.073179)** | 0.000 (1) |
| | 0.708295 (0.000133) | **0.757701 (0.000329)** | 0.000 (1) |
| | 0.416468 (0.000529) | **0.439303 (0.000034)** | 0.000 (1) |
| | 0.352563 (0.026612) | **0.406017 (0.020026)** | 0.000 (1) |
| | 1.607354 (0.202304) | **1.939227 (0.218993)** | 0.000 (1) |
| | 16.211830 (0.312977) | **21.038451 (0.602411)** | 0.000 (1) |
| | 0.088530 (0.006124) | **0.118381 (0.012508)** | 0.000 (1) |
| | 0.388538 (0.014871) | **0.465227 (0.001959)** | 0.000 (1) |
| | 22.853573 (0.541417) | **33.806224 (0.339882)** | 0.000 (1) |
| | 8.501790 (0.147219) | **9.877391 (0.011298)** | 0.000 (1) |
| | 0.845541 (0.009395) | **1.024659 (0.003593)** | 0.000 (1) |
| | 16.227431 (1.859549) | **22.748887 (0.086435)** | 0.000 (1) |

(b)

| $f$ | isms-emoa $I_H$ | mh-moea $I_H$ | $P(H)$ |
|---|---|---|---|
| | 0.206981 (0.099188) | **0.301778 (0.059500)** | 0.000 (1) |
| | 0.757459 (0.000122) | **0.757779 (0.000125)** | 0.000 (1) |
| | 0.049505 (0.122515) | **0.732405 (0.073179)** | 0.000 (1) |
| | **0.757900 (0.000071)** | 0.757701 (0.000329) | 0.000 (1) |
| | **0.439351 (0.000017)** | 0.439303 (0.000034) | 0.000 (1) |
| | **0.416488 (0.019702)** | 0.406017 (0.020026) | 0.017 (1) |
| | 1.923408 (0.229443) | **1.939227 (0.218993)** | 0.620 (0) |
| | **21.211584 (0.164660)** | 21.038451 (0.602411) | 0.579 (0) |
| | **0.124559 (0.009472)** | 0.118381 (0.012508) | 0.000 (1) |
| | **0.466669 (0.000707)** | 0.465227 (0.001959) | 0.003 (1) |
| | 29.382365 (0.077628) | **33.806224 (0.339882)** | 0.000 (1) |
| | 9.875098 (0.013724) | **9.877391 (0.011298)** | 0.662 (0) |
| | 1.023947 (0.002415) | **1.024659 (0.003593)** | 0.264 (0) |
| | **23.897145 (0.106150)** | 22.748887 (0.086435) | 0.000 (1) |

(c)

| $f$ | mc-moea $I_H$ | mh-moea $I_H$ | $P(H)$ |
|---|---|---|---|
| DTLZ3(4) | 0.857388 (0.036140) | **1.027826 (0.032578)** | 0.000 (1) |
| DTLZ5(4) | 0.205774 (0.028565) | **0.436266 (0.000698)** | 0.000 (1) |
| DTLZ7(4) | 0.554462 (0.066262) | **0.758183 (0.103790)** | 0.000 (1) |
| DTLZ3(5) | 0.999126 (0.042205) | **1.117145 (0.321966)** | 0.000 (1) |
| DTLZ5(5) | 0.164502 (0.026344) | **0.444185 (0.001890)** | 0.000 (1) |
| DTLZ7(5) | 0.064793 (0.013324) | **0.153578 (0.056167)** | 0.000 (1) |
| DTLZ3(6) | 1.060825 (0.096036) | **1.098540 (0.492904)** | 0.125 (0) |
| DTLZ5(6) | 0.138851 (0.044069) | **0.453138 (0.004788)** | 0.000 (1) |
| DTLZ7(6) | 0.003920 (0.001524) | **0.020082 (0.011365)** | 0.000 (1) |

(d)

| $f$ | moead $I_H$ | mh-moea $I_H$ | $P(H)$ |
|---|---|---|---|
| | 0.849726 (0.008275) | **1.027826 (0.032578)** | 0.000 (1) |
| | 0.395004 (0.003938) | **0.436266 (0.000698)** | 0.000 (1) |
| | 0.510402 (0.136447) | **0.758183 (0.103790)** | 0.000 (1) |
| | 0.907270 (0.016862) | **1.117145 (0.321966)** | 0.000 (1) |
| | 0.384474 (0.004732) | **0.444185 (0.001890)** | 0.000 (1) |
| | 0.090559 (0.025208) | **0.153578 (0.056167)** | 0.000 (1) |
| | 0.836798 (0.127543) | **1.098540 (0.492904)** | 0.001 (1) |
| | 0.386218 (0.005819) | **0.453138 (0.004788)** | 0.000 (1) |
| | 0.017423 (0.001653) | **0.020082 (0.011365)** | 1.000 (0) |

(e)

| $f$ | isms-emoa $I_H$ | mh-moea $I_H$ | $P(H)$ |
|---|---|---|---|
| | 1.013137 (0.102415) | **1.027826 (0.032578)** | 0.000 (1) |
| | **0.437185 (0.000308)** | 0.436266 (0.000698) | 0.000 (1) |
| | 0.739654 (0.162259) | **0.758183 (0.103790)** | 0.145 (0) |
| | **1.288363 (0.026772)** | 1.117145 (0.321966) | 0.000 (1) |
| | **0.446454 (0.000582)** | 0.444185 (0.001890) | 0.000 (1) |
| | **0.164838 (0.056642)** | 0.153578 (0.056167) | 0.233 (0) |
| | 1.178007 (0.604886) | **1.426338 (0.222236)** | 0.000 (1) |
| | **0.461971 (0.001111)** | 0.453138 (0.004788) | 0.000 (1) |
| | **0.027512 (0.008047)** | 0.020082 (0.011365) | 0.206 (0) |

(f)

Table 1: Comparison of results in the DTLZ and WFG test problems using $I_H$. We show average values over 30 independent runs. The values in parentheses correspond to the standard deviations. The third column of each table shows the results of the statistical analysis applied to our experiments using Wilcoxons rank sum. $P$ is the probability of observing a given result (the null hypothesis is true). Small values of $P$ cast doubt on the validity of the null hypothesis. $H = 0$ indicates that the null hypothesis ("medians are equal") cannot be rejected at the 5% level. $H = 1$ indicates that the null hypothesis can be rejected at the 5% level.

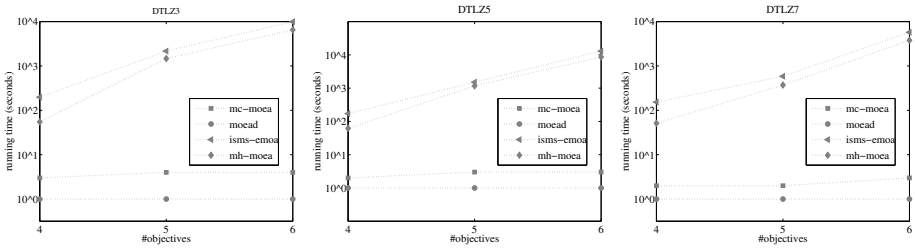


Table 2: Time required by MC-MOEA, MOEA/D, iSMS-EMOA and our proposed MH-MOEA for the test problems adopted in the scalability analysis. $s$ = seconds. All algorithms were compiled using the GNU C compiler and they were executed on a computer with a 2.66GHz processor and 4GB in RAM. We can see that the worst algorithm, regarding running time, is ISMS-EMOA in all three MOPs (DTLZ3, DTLZ5 and DTLZ7 with 4, 5 and 6 objective functions). While the best algorithm is MOEA/D. Also, we can see that our MH-MOEA outperforms iSMS-EMOA in all cases.

## 7 Conclusions and Future Work

We have proposed a new selection mechanism based on the modified maximin fitness function (MFF) and the hypervolume indicator ($I_H$). Unlike other selection mechanisms based on $I_H$, such as the one adopted in the SMS-EMOA algorithm or its improved version (iSMS-EMOA), our selection mechanism works with populations. Our idea is to use the modified MFF as our main selection mechanism and $I_H$ to correct the possible errors in the selection process. Our preliminary results indicate that our MH-MOEA is able to outperform MOEAs such as MC-MOEA and MOEA/D, both with few and many objectives. Also, MH-MOEA is competitive with respect to iSMS-EMOA, outperforming it in some cases, while requiring a lower computational time.

As part of our future work, we are interested in studying mechanisms to approximate the contribution to $I_H$ and to use one of them instead of adopting the exact calculation. Since we only use $I_H$ to correct possible errors generated when we select with the maximin fitness, we expect to retain, as much as possible, the quality in our solutions when we approximate the contribution to $I_H$.

## References


1. Anne Auger, Johannes Bader, and Dimo Brockhoff. Theoretically Investigating Optimal $\mu$-Distributions for the Hypervolume Indicator: First Results for Three Objectives. In Robert Schaefer, Carlos Cotta, Joanna Kołodziej, and Günter Rudolph, editors, *Parallel Problem Solving from Nature–PPSN XI, 11th International Conference, Proceedings, Part I*, pages 586–596. Springer, Lecture Notes in Computer Science Vol. 6238, Kraków, Poland, September 2010.
2. Anne Auger, Johannes Bader, Dimo Brockhoff, and Eckart Zitzler. Theory of the Hypervolume Indicator: Optimal $\{\mu\}$-Distributions and the Choice Of The Reference Point. In *FOGA '09: Proceedings of the tenth ACM SIGEVO workshop on Foundations of genetic algorithms*, pages 87–102, Orlando, Florida, USA, January 2009. ACM.
3. Richard Balling. Pareto sets in decision-based design. *Journal of Engineering Valuation and Cost Analysis*, 3:189–198, 2000.
4. Richard Balling. The Maximin Fitness Function; Multiobjective City and Regional Planning. In Carlos M. Fonseca, Peter J. Fleming, Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele, editors, *Evolutionary Multi-Criterion Optimization. Second International Conference, EMO 2003*, pages 1–15, Faro, Portugal, April 2003. Springer. Lecture Notes in Computer Science. Volume 2632.
5. Richard Balling and Scott Wilson. The Maximin Fitness Function for Multiobjective Evolutionary Computation: Application to City Planning. In Lee Spector, Erik D. Goodman, Annie Wu, W.B. Langdon, Hans-Michael Voigt, Mitsuo Gen, Sandip Sen, Marco Dorigo, Shahram Pezeshk, Max H. Garzon, and Edmund Burke, editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001)*, pages 1079–1084, San Francisco, California, 2001. Morgan Kaufmann Publishers.
6. Indraneel 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*, 8(3):631–657, March 1998.

7. Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. Scalable Test Problems for Evolutionary Multiobjective Optimization. In Ajith Abraham, Lakhmi Jain, and Robert Goldberg, editors, *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications*, pages 105–145. Springer, USA, 2005.
8. Michael Emmerich, Nicola Beume, and Boris Naujoks. An EMO Algorithm Using the Hypervolume Measure as Selection Criterion. In Carlos A. Coello Coello, Arturo Hernández Aguirre, and Eckart Zitzler, editors, *Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005*, pages 62–76, Guanajuato, México, March 2005. Springer. Lecture Notes in Computer Science Vol. 3410.
9. Simon Huband, Phil Hingston, Luigi Barone, and Lyndon While. A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. *IEEE Transactions on Evolutionary Computation*, 10(5):477–506, October 2006.
10. Hongga Li, Xiaoxia Huang, and Quan Feng. Optimizing expressway maintenance planning by coupling ant algorithm and geography information system transportation in hubei province, china. In *Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International*, pages 2977 –2979, july 2011.
11. Xiaodong Li. Better Spread and Convergence: Particle Swarm Multiobjective Optimization Using the Maximin Fitness Function. In Kalyanmoy Deb et al., editor, *Genetic and Evolutionary Computation–GECCO 2004. Proceedings of the Genetic and Evolutionary Computation Conference. Part I*, pages 117–128, Seattle, Washington, USA, June 2004. Springer-Verlag, Lecture Notes in Computer Science Vol. 3102.
12. Xiaodong Li, Jürgen Branke, and Michael Kirley. On Performance Metrics and Particle Swarm Methods for Dynamic Multiobjective Optimization Problems. In *2007 IEEE Congress on Evolutionary Computation (CEC'2007)*, pages 576–583, Singapore, September 2007. IEEE Press.
13. Adriana Menchaca-Mendez and Carlos A. Coello Coello. Solving Multi-Objective Optimization Problems using Differential Evolution and a Maximin Selection Criterion. In *2012 IEEE Congress on Evolutionary Computation (CEC'2012)*, pages 3143–3150, Brisbane, Australia, June 10-15 2012. IEEE Press.
14. Adriana Menchaca-Mendez and Carlos A. Coello Coello. A New Selection Mechanism Based on Hypervolume and its Locality Property. In *2013 IEEE Congress on Evolutionary Computation (CEC'2013)*, pages 924–931, Cancún, México, 20-23 June 2013. IEEE Press.
15. Adriana Menchaca-Mendez and Carlos A. Coello Coello. Selection Operators based on Maximin Fitness Function for Multi-Objective Evolutionary Algorithms. In *Evolutionary Multi-Criterion Optimization, 7th International Conference, EMO 2013*, Sheffield, UK, March 2013. Springer. Lecture Notes in Computer Science Vol. 6576.
16. Qingfu Zhang and Hui Li. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, December 2007.
17. Eckart Zitzler and Lothar Thiele. Multiobjective Optimization Using Evolutionary Algorithms—A Comparative Study. In A. E. Eiben, editor, *Parallel Problem Solving from Nature V*, pages 292–301, Amsterdam, September 1998. Springer-Verlag.
18. Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane Grunert da Fonseca. Performance Assessment of Multiobjective Optimizers: An Analysis and Review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, April 2003.