A More Efficient Selection Scheme in iSMS-EMOA

Adriana Menchaca-Mendez[2], Elizabeth Montero[1](✉), María-Cristina Riff[1], and Carlos A. Coello Coello[2]

[1] Department of Computer Science, Universidad Técnica Federico Santa María, Valparaíso, Chile
{Elizabeth.Montero,Maria-Cristina.Riff}@inf.utfsm.cl
[2] CINVESTAV-IPN, Departamento de Computación, Mexico, DF, Mexico
adriana.menchacamendez@gmail.com, ccoello@cs.cinvestav.mx

Abstract. In this paper, we study iSMS-EMOA, a recently proposed approach that improves the well-known *S metric selection Evolutionary Multi-Objective Algorithm* (SMS-EMOA). These two indicator-based multi-objective evolutionary algorithms rely on hypervolume contributions to select individuals. Here, we propose to define a probability of using a randomly selected individual within the iSMS-EMOA's selection scheme. In order to calibrate the value of such probability, we use the EVOCA tuner. Our preliminary results indicate that we are able to save up to 33% of computations of the contribution to hypervolume with respect to the original iSMS-EMOA, without any significant quality degradation in the solutions obtained. In fact, in some cases, the approach proposed here was even able to improve the quality of the solutions obtained by the original iSMS-EMOA.

Keywords: Multi-objective evolutionary algorithms · Tuning · Hypervolume contribution

1 Introduction

Many optimization problems involve the simultaneous optimization of several objectives. They are known as *multi-objective optimization problems (MOPs)* and in them, the notion of optimality refers to the best possible trade-offs among the objectives. Consequently, there is no single optimal solution but a set of solutions (the so-called *Pareto optimal set* whose image is called the *Pareto front*). The use of *Multi-Objective Evolutionary Algorithms (MOEAs)* to solve MOPs has become increasingly popular. In recent years, MOEAs based on the hypervolume indicator (I_H) have become relatively popular. This is due to two main reasons: first, the use of Pareto-based selection has several limitations[1]. And, second, I_H has interesting mathematical properties. For example, it is

This work is supported by the collaboration project Conacyt-Conicyt 2010-199. The last author acknowledges support from project B330.261.

[1] The number of non-dominated solutions grows exponentially as we increase the number of objective functions, and this rapidly dilutes the selection pressure of a MOEA [4].

A.L.C. Bazzan and K. Pichara (Eds.): IBERAMIA 2014, LNAI 8864, pp. 371–380, 2014.
DOI: 10.1007/978-3-319-12027-0_30

the only unary indicator which is known to be "Pareto compliant" [10]. I_H was originally proposed by Zitzler and Thiele in [9], and it is defined as the size of the space covered by the Pareto optimal solutions. I_H rewards convergence towards the Pareto front as well as the maximum spread of the solutions obtained. Fleischer proved in [5] that, given a finite search space and a reference point, *maximizing the hypervolume indicator is equivalent to finding the Pareto optimal set*. However, I_H has one important disadvantage: its high computational cost. The "S metric selection Evolutionary Multi-Objective Algorithm (SMS-EMOA)" [3] is currently, the most popular MOEA based on I_H and it works as follows: it creates only one individual by iteration. After that, it applies Pareto ranking. If the last front has more than one individual, SMS-EMOA deletes the individual with the worst contribution to I_H. SMS-EMOA is impractical when we want to solve MOPs with many objectives because if all individuals are non-dominated, it needs to compute the contribution to I_H of all individuals and we know that this task is computational expensive (the calculation of the minimal contribution to I_H is an **NP-hard** [1] problem). Recently, in [7] authors proposed a selection scheme based on I_H and its locality property giving rise to an improved version of SMS-EMOA called iSMS-EMOA. With this scheme, the new individual only competes with two other individuals of the population: its nearest neighbor and a randomly selected individual. This scheme allows a significant reduction in the running time. However, in [7], it was noted that the use of the randomly selected individual is not necessary in all iterations and it was left as future work to identify the cases in which it is required. In this paper, we propose to define a probability of use of the randomly selected individual which is automatically adjusted using the EVOCA tuner [8] with the two following aims: to maximize I_H and to minimize the running time (reducing the number of computations of the contribution to I_H). This is clearly a MOP, but with a clear order of preference: we aim to reduce the number of computations of the contribution to I_H without affecting the quality of the solutions. Thus, we decided to solve it using the ϵ-constraint method. Let $\mathcal{A}$ be the approximation of the Pareto optimal set obtained by the iSMS-EMOA algorithm and p_{rsi} be the probability of use of the randomly selected individual. First, we calibrate p_{rsi}, maximizing $I_H(\mathcal{A})$. After that, we calibrate p_{rsi}, minimizing the running time required by the iSMS-EMOA algorithm in order to obtain $\mathcal{A}$, having as a constraint: $I_H(\mathcal{A}) > maxI_H - \epsilon$, where $maxI_H$ is the maximum hypervolume found in the previous step and ϵ is a tolerance. We will show how this scheme produces savings of up to 33% of computations of the contribution to I_H (with respect to the original iSMS-EMOA) without losing quality in the solutions obtained. In fact, we will see how, in some cases, we can even improve the quality of $\mathcal{A}$ with respect to I_H when using our proposed approach.

The remainder of this paper is organized as follows: Section 2 states the problem of our interest and provides some basic definitions. The original iSMS-EMOA is described in Section 3. Our proposal is discussed in Section 4 and it is validated in Section 5. Finally, we provide our conclusions and some possible paths for future work in Section 6.

2 Basic Definitions and Problem Statement

We are interested in the general 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 $\Omega \subset \mathbb{R}^n$ 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. *For a given MOP,* $\boldsymbol{f}(\boldsymbol{x})$*, the Pareto optimal set is defined as:* $\mathcal{P}^* = \{\boldsymbol{x} \in \Omega | \neg\exists \boldsymbol{y} \in \Omega : \boldsymbol{y} \prec \boldsymbol{x}\}$*.*

Definition 3. *Let* $\boldsymbol{f}(\boldsymbol{x})$ *be a given MOP and* $\mathcal{P}^*$ *the Pareto optimal set. Then, the Pareto Front is defined as:* $\mathcal{PF}^* = \{\boldsymbol{f}(\boldsymbol{x}) \mid \boldsymbol{x} \in \mathcal{P}^*\}$*.*

Definition 4. *If* Λ *denotes the Lebesgue measure, the hypervolume indicator* (I_H) *is defined as:*

$$I_H(\mathcal{A}, \boldsymbol{y}_{ref}) = \Lambda \left(\bigcup_{y \in \mathcal{A}} \{\boldsymbol{y}' \mid \boldsymbol{y} < \boldsymbol{y}' < \boldsymbol{y}_{ref}\} \right) \tag{2}$$

where $\boldsymbol{y}_{ref} \in \mathbb{R}^k$ *denotes a reference point that should be dominated by all the Pareto optimal points.*

Definition 5. *The contribution to* I_H *of a solution* $\boldsymbol{x}$ *is defined as:*

$$C_H(\boldsymbol{x}, \mathcal{A}) = I_H(\mathcal{A}, \boldsymbol{y}_{ref}) - I_H(\mathcal{A} \setminus \boldsymbol{x}, \boldsymbol{y}_{ref}) \tag{3}$$

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

3 iSMS-EMOA

The Improved S Metric Selection Evolutionary Multi-Objective Algorithm (iSMS-EMOA) [7] works as follows: First, it creates an initial population. After that, only one individual is created at each iteration using the operators of the NSGA-II (crossover and mutation). Let $\boldsymbol{x}_{new}$ be the new individual and $\mathcal{A}$ be the current population. We calculate the Euclidean distance of $\boldsymbol{x}_{new}$ to each solution in $\mathcal{A}$ and, we choose the nearest solution $\boldsymbol{x}_{near}$. These two solutions ($\boldsymbol{x}_{new}$ and $\boldsymbol{x}_{near}$) compete to survive. The core idea is to move a solution within its neighborhood with the aim of improving its contribution to I_H (locality property). It is important to consider the case in which $\boldsymbol{x}_{new}$ is located in an unexplored region. In this case, it is not a good idea to remove $\boldsymbol{x}_{new}$ or $\boldsymbol{x}_{near}$. To address this problem, the authors proposed to choose (randomly) another solution, $\boldsymbol{x}_{rand}$, such that $\boldsymbol{x}_{rand} \in \mathcal{A}$ and $\boldsymbol{x}_{rand} \neq \boldsymbol{x}_{near}$. This is considering that the probability of choosing a solution in a crowded region is high and the probability of choosing a solution in an unexplored region is low. Then, $\boldsymbol{x}_{rand}$, $\boldsymbol{x}_{new}$ and $\boldsymbol{x}_{near}$ will compete to survive.

4 Our Proposed Approach

We propose here to use a probability which enables us to decide when to incorporate the randomly selected individual into iSMS-EMOA. The new algorithm is called "improved S metric selection Evolutionary Multi-Objective Algorithm II (iSMS-EMOA II)", see Algorithm 1. The only difference between iSMS-EMOA and iSMS-EMOA II is that now, we flip a coin to decide if we use the randomly selected individual at each iteration, see Algorithm 1, line 6. Setting the value of p_{rsi} is not trivial: large values will lead to a waste of computational effort for calculating hypervolume contribution of solutions that won't be eliminated. On the other side, small values of p_{rsi} can decrease the diversification ability of the algorithm, reducing its capacity to generate solutions in specific zones of Pareto front.

For calibrating p_{rsi}, we used the EVOCA [8] tuner. This is an evolutionary algorithm that works with a population of parameter calibrations. The population size is computed considering the number of parameters and their domain sizes. The key idea is to include all the values allowed for each parameter, in an independent way, on the first population. EVOCA uses two transformation operators. First, it adopts a crossover operator (wheel-crossover) that constructs one calibration from the whole population. The child calibration generated replaces the worst calibration on the current population. Second, it adopts a mutation operator which is a hill climbing first improvement procedure that takes a copy of the child generated by the crossover operator and tries to improve it by modifying one of its parameter values. In case of a numerical parameter, it will try to randomly take a new value from the parameter interval, regarding it as a continuous range. The calibration generated by applying mutation replaces the second worst calibration on the current population, when a better individual is found. Algorithm 2 shows the EVOCA structure. We have considered two scenarios to calibrate p_{rsi}: first, we maximize the hypervolume of the approximation of the Pareto optimal set obtained by iSMS-EMOA II for a given MOP. And, second, we minimize the number of computations of the contribution to I_H required by iSMS-EMOA II to obtain the approximation of the Pareto optimal set of that MOP, avoiding to affect the value of the hypervolume obtained before.

4.1 Scenario 1: Maximizing the Hypervolume Indicator

In this part, we calibrate the probability p_{rsi}, solving the following problem:

$$\max I_H(\mathcal{A}) \tag{4}$$

where $\mathcal{A}$ is the approximation of the Pareto optimal set obtained by iSMS-EMOA II for a given MOP.

Setting EVOCA for iSMS-EMOA II in Scenario 1. For applying EVOCA in this scenario, we need to define the following criteria:

Algorithm 1. iSMS-EMOA II

Input : MOP to be solved.
Output: The approximation of the Pareto optimal set ($\mathcal{A}$).
Generate a random initial population ($\mathcal{A}$);
while *Stopping criterion is not met* **do**
  Select randomly two individuals from $\mathcal{A}$ ($\boldsymbol{x}_1$ and $\boldsymbol{x}_2$);
  Obtain an offspring ($\boldsymbol{x}_{new}$) from $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, applying the operators of NSGA-II (crossover and mutation);
  $near \leftarrow$ Index of the nearest neighbor to $\boldsymbol{x}_{new}$ in $\mathcal{A}$;
  if $random(0,1) < p_{rsi}$ **then**
    $rand \leftarrow$ Integer random number between 1 and $|\mathcal{A}|$ (such that $near \neq rand$);
    Calculate the contribution to I_H of $\boldsymbol{x}_{new}$, $\boldsymbol{x}_{near}$ and $\boldsymbol{x}_{rand}$;
    if $C_H(\boldsymbol{x}_{new}, \mathcal{A})$ *is better than* $C_H(\boldsymbol{x}_{near}, \mathcal{A})$ *or* $C_H(\boldsymbol{x}_{rand}, \mathcal{A})$ **then**
      if $C_H(\boldsymbol{x}_{near}, \mathcal{A}) < C_H(\boldsymbol{x}_{rand}, \mathcal{A})$ **then**
        Replace $\boldsymbol{x}_{near}$ with $\boldsymbol{x}_{new}$;
      else
        Replace $\boldsymbol{x}_{rand}$ with $\boldsymbol{x}_{new}$;
      end
    end
  else
    Compute the contribution to I_H of $\boldsymbol{x}_{new}$ and $\boldsymbol{x}_{near}$;
    if $C_H(\boldsymbol{x}_{new}, \mathcal{A}) > C_H(\boldsymbol{x}_{near}, \mathcal{A})$ **then**
      Replace $\boldsymbol{x}_{near}$ with $\boldsymbol{x}_{new}$;
    end
  end
end
return $\mathcal{A}$;

Algorithm 2. EVOCA

Input : Definition of parameters for target algorithm $\mathcal{M}$
Output: Set of best performing parameter calibrations for $\mathcal{M}$
Generate initial population ($\mathcal{P}$);
while *Stopping criterion is not met* **do**
  child $\leftarrow$ wheel-crossover(P);
  Evaluate *child* using R random seeds ;
  Replace worst calibration in $\mathcal{P}$ by *child* ;
  mutated_child $\leftarrow$ mutation(*child*) ;
  Evaluate *mutated_child* using R random seeds;
  if *mutated_child is* ***better*** *than child* **then**
    Replace the second worst calibration in $\mathcal{P}$ by *mutated_child*;
  end
end
return $\mathcal{P}$;

- When do we consider a parameter calibration to be better than another one? In this case, EVOCA takes into account one criterion to evaluate each parameter calibration for iSMS-EMOA II. One calibration c' is considered better than another c, in the case in which the use of c' in iSMS-EMOA II provides a higher hypervolume than the use of c.
- A parameter precision level for the initial population: Here, the initial precision must be defined for parameter p_{rsi}. It is important to remark that EVOCA is able to increase this precision during the calibration process using the mutation operator, which selects values from an interval. Thus, this precision is only considered to generate the initial EVOCA's population.
- Which is the result of the calibration? The best calibration that belongs to the final EVOCA's population is the one with the best hypervolume value.

4.2 Scenario 2: Minimizing the Number of Calculations of the Contribution to I_H

In this part, we are interested in minimizing the number of calculations of the contribution to I_H required by iSMS-EMOA II without losing too much quality in the solutions. For this, we calibrate p_{rsi}, solving the following problem:

$$\min(\text{Time required to obtain } \mathcal{A}) \text{ such that } I_H(\mathcal{A}) > maxI_H - \epsilon \quad (5)$$

where $\mathcal{A}$ is the approximation of the Pareto optimal set obtained by iSMS-EMOA II for a given MOP; $maxI_H$ is the maximum hypervolume obtained when we solve eq. (4) and ϵ is a tolerance.

Setting EVOCA for iSMS-EMOA II in Scenario 2. For applying EVOCA in this scenario, we need to define the following criteria:

- When do we consider a parameter calibration to be better than another one? In this case, EVOCA takes into account two criteria to evaluate each parameter calibration for iSMS-EMOA II. One calibration c' is considered to be better than another one c, using two objectives: when the use of c' allows iSMS-EMOA II to achieve both, that the hypervolume value is higher than the tolerance level ϵ and that a lower running time than when using c is achieved.
- A parameter precision level for the initial population: Here, the initial precision must be defined for parameter p_{rsi}.
- Which is the result of the calibration? The best calibration that belongs to the final EVOCA's population is the one with the best hypervolume value.

We note that the tolerance value is used to define a minimum quality of the calibrations, in terms of hypervolume respect to the quality obtained with the iSMS-EMOA II when solving eq. 4. For our experiments, we considered a tolerance of 1%.

5 Experimental Results

To measure the performance of iSMS-EMOA II, we compare it with respect to the original iSMS-EMOA[2]. For our experiments, we used four problems with 3, 4 and 5 objective functions taken from the Deb-Thiele-Laumanns-Zitzler (DTLZ) test suite [2]. We used $k = 5$ for DTLZ1 and $k = 10$ for the remaining test problems. Also, we used two problems with 3, 4 and 5 objective functions, taken from the WFG toolkit [6], with $k\_factor = 2$ and $l\_factor = 10$. We chose these problems because each of them has a Pareto front with distinct features; and also, they are scalables with respect to the number of objective functions. For each test problem, we performed 30 independent runs. For both 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. We performed a maximum of 50,000 fitness function evaluations (we used a population size of 100 individuals and we iterated for 500 generations). 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, iSMS-EMOA and iSMS-EMOA II, have as their aim to maximize the hypervolume and, therefore, it makes sense to use this indicator to assess their performance. To calculate I_H, we used the following reference points: $y_{ref} = [y_1, \cdots, y_M]$ such that $y_i = 0.7$ for DTLZ1, $y_i = 1.1$ for DTLZ2 and DTLZ5, $y_M = 6.1$ and $y_{i \neq M} = 1.1$ for DTLZ7. In the case of the WFG test problems, we generated the reference point using the highest value found for each objective function taking into account all the outputs of both algorithms.

5.1 Results in Scenario 1

In Table 1(a), we can observe that if the randomly generated individual is always selected (original iSMS-EMOA), we get better results in most cases. In fact, in most problems, EVOCA calibrates p_{rsi} with high values, e.g., it sets $p_{rsi} = 1.0$ for DTLZ1 with three objective functions and DTLZ1, DTLZ2, DTLZ5 and WFG1 with four objective functions. This means that in these problems EVOCA suggests to use all the time the randomly selected individual, as in the original iSMS-EMOA, to maximize I_H. For this reason, in these problems, iSMS-EMOA II does not save computations of the contribution to I_H. However, an interesting aspect is that in some problems this randomly selected individual is not necessary. In such cases, iSMS-EMOA results can be improved by selecting the randomly generated individual with a low probability. For example, in DTLZ7 and WFG4 with three objective functions, a probability $p_{rsi} = 0.127$ and $p_{rsi} = 0.1$, were calibrated respectively, which allowed us to save up to 30% of computations of the contribution to I_H. In the case of DTLZ7, we can note that iSMS-EMOA II significantly outperformed iSMS-EMOA, because it obtained better results, and the null

[2] iSMS-EMOA is compared to the original SMS-EMOA in [7], but such comparison was omitted here due to space limitations.

Table 1. We show average values over 30 independent runs. Values in parentheses correspond to the standard deviations. $P(H)$ shows the results of statistical analysis applied to our experiments using Wilcoxons rank sum considering I_H. P is the probability of observing the given result (the null hypothesis is true). Small values of P cast doubt on the validity of the null hypothesis. $H = 1$ indicates that the null hypothesis can be rejected at the 5% level. Both iSMS-EMOA and iSMS-EMOA II were compiled using the GNU C compiler and they were executed on a computer with a 2.66GHz processor and 4GB in RAM. (a) shows the results for scenario 1 and in (b) shows the results for scenario 2.

		isms-emoa	isms-emoa-ii	isms-emoa			isms-emoa-ii			
f	p_{rsi}	I_H	I_H	Eval C_H	Savings	time	Eval C_H	Savings	time	$P(H)$
DTLZ1(3)	1.0	0.316985 (0.000066)	**0.316998 (0.000046)**	150000 (0.00)	-0.00%	≈ 7s	150000 (0.00)	-0.00%	≈ 7s	0.450(0)
DTLZ2(3)	0.7	**0.757890 (0.000100)**	0.757863 (0.000075)	150000 (0.00)	-0.00%	≈ 7s	**135022 (95.82)**	**-9.99%**	**≈6s**	0.251(0)
DTLZ5(3)	0.8	**0.439350 (0.000017)**	0.439342 (0.000020)	150000 (0.00)	-0.00%	≈ 8s	**140009 (96.04)**	**-6.66%**	**≈7s**	0.062(0)
DTLZ7(3)	0.127	1.908824 (0.200002)	**2.019564 (0.000926)**	150000 (0.00)	-0.00%	≈ 7s	**106353 (68.35)**	**-29.10%**	**≈5s**	0.002(1)
DTLZ1(4)	1.0	**0.234451 (0.000018)**	0.234446 (0.000019)	150000 (0.00)	-0.00%	≈ 76s	150000 (0.00)	-0.00%	≈ 76s	0.314(0)
DTLZ2(4)	1.0	1.044211 (0.000159)	**1.044281 (0.000159)**	150000 (0.00)	-0.00%	≈ 80s	150000 (0.00)	-0.00%	≈ 80s	0.183(0)
DTLZ5(4)	1.0	**0.437073 (0.000308)**	0.437056 (0.000318)	150000 (0.00)	-0.00%	≈ 70s	150000 (0.00)	-0.00%	≈ 70s	0.801(0)
DTLZ7(4)	0.3	0.678273 (0.198672)	**0.797852 (0.001903)**	150000 (0.00)	-0.00%	≈ 49s	**114974 (86.30)**	**-23.35%**	**≈39s**	0.379(0)
DTLZ1(5)	0.8	0.166731 (0.000011)	**0.166733 (0.000008)**	150000 (0.00)	-0.00%	≈ 1254s	**139258 (1773.08)**	**-7.16%**	**≈1110s**	0.378(0)
DTLZ2(5)	0.8	**1.295672 (0.000166)**	1.295508 (0.000201)	150000 (0.00)	-0.00%	≈ 1413s	**133891 (6474.89)**	**-10.74%**	**≈1167s**	0.004(1)
DTLZ5(5)	0.8	**0.446086 (0.000612)**	0.445896 (0.000756)	150000 (0.00)	-0.00%	≈ 1411s	**139122 (3351.63)**	**-7.25%**	**≈1258s**	0.355(0)
DTLZ7(5)	0.4	0.158271 (0.058981)	**0.187287 (0.031902)**	150000 (0.00)	-0.00%	≈ 518s	**120001 (125.76)**	**-20.00%**	**≈363s**	0.325(0)
WFG1(3)	0.8	**21.205641 (0.177134)**	21.174222 (0.358121)	150000 (0.00)	-0.00%	≈ 8s	**140003 (88.75)**	**-6.66%**	**≈7s**	0.773(0)
WFG4(3)	0.1	**29.346993 (0.095911)**	29.328283 (0.081647)	150000 (0.00)	-0.00%	≈ 8s	**104995 (74.26)**	**-30.00%**	**≈6s**	0.363(0)
WFG1(4)	1.0	**88.573606 (0.541177)**	88.502834 (0.505897)	150000 (0.00)	-0.00%	≈ 95s	150000 (0.00)	-0.00%	≈ 95s	0.652(0)
WFG4(4)	0.7	301.253225 (1.155539)	**301.415556 (1.056422)**	150000 (0.00)	-0.00%	≈ 79s	**135003 (94.41)**	**-10.00%**	**≈71s**	0.695(0)
WFG1(5)	0.8	114.187823 (0.723549)	**114.290414 (0.646247)**	150000 (0.00)	-0.00%	≈ 1411s	**119253 (7811.77)**	**-20.50%**	**≈1268s**	0.970(0)
WFG4(5)	0.529	3465.333620 (17.840209)	**3466.819998 (13.297479)**	150000 (0.00)	-0.00%	≈ 1305s	**126362 (431.92)**	**-15.76%**	**≈1164s**	0.784(0)

(a)

		isms-emoa	isms-emoa-ii	isms-emoa			isms-emoa-ii			
f	p_{rsi}	I_H	I_H	Eval C_H	Savings	time	Eval C_H	Savings	time	$P(H)$
DTLZ1(3)	0.222	**0.316985 (0.000066)**	0.296946 (0.053188)	150000 (0.00)	-0.00%	≈ 7s	**111095 (85.53)**	**-25.94%**	**≈6s**	0.000(1)
DTLZ2(3)	0.193	**0.757890 (0.000100)**	0.757819 (0.000099)	150000 (0.00)	-0.00%	≈ 7s	**109632 (99.38)**	**-26.91%**	**≈6s**	0.011(1)
DTLZ5(3)	0.148	**0.439350 (0.000017)**	0.439266 (0.000029)	150000 (0.00)	-0.00%	≈ 8s	**107403 (96.41)**	**-28.40%**	**≈5s**	0.000(1)
DTLZ7(3)	0.1	1.908824 (0.200002)	**2.019672 (0.000753)**	150000 (0.00)	-0.00%	≈ 7s	**104981 (49.98)**	**-30.01%**	**≈5s**	0.000(1)
DTLZ1(4)	0.144	**0.234451 (0.000018)**	0.229403 (0.015504)	150000 (0.00)	-0.00%	≈ 76s	**107183 (97.41)**	**-28.54%**	**≈37s**	0.000(1)
DTLZ2(4)	0.075	**1.044211 (0.000159)**	1.043808 (0.000237)	150000 (0.00)	-0.00%	≈ 80s	**103732 (53.76)**	**-30.85%**	**≈51s**	0.000(1)
DTLZ5(4)	0.184	**0.437073 (0.000308)**	0.436148 (0.000376)	150000 (0.00)	-0.00%	≈ 70s	**109197 (79.72)**	**-27.20%**	**≈51s**	0.000(1)
DTLZ7(4)	0.107	0.678273 (0.198672)	**0.797284 (0.002003)**	150000 (0.00)	-0.00%	≈ 49s	**105345 (75.00)**	**-29.77%**	**≈35s**	0.830(0)
DTLZ1(5)	0.24	**0.166731 (0.000011)**	0.166415 (0.000599)	150000 (0.00)	-0.00%	≈ 1254s	**112014 (110.94)**	**-25.32%**	**≈657s**	0.000(1)
DTLZ2(5)	0.088	**1.295672 (0.000166)**	1.294868 (0.000348)	150000 (0.00)	-0.00%	≈ 1413s	**104408 (66.34)**	**-30.39%**	**≈947s**	0.000(1)
DTLZ5(5)	0.088	**0.446086 (0.000612)**	0.441584 (0.001664)	150000 (0.00)	-0.00%	≈ 1411s	**104400 (55.24)**	**-30.40%**	**≈941s**	0.000(1)
DTLZ7(5)	0.1	0.158271 (0.058981)	**0.188343 (0.025764)**	150000 (0.00)	-0.00%	≈ 518s	**104995 (84.41)**	**-30.00%**	**≈324s**	0.059(0)
WFG1(3)	0.4	**21.205641 (0.177134)**	20.718664 (0.794370)	150000 (0.00)	-0.00%	≈ 8s	**120003 (123.57)**	**-20.00%**	**≈7s**	0.005(1)
WFG4(3)	0.1	**29.346993 (0.095911)**	29.332057 (0.085921)	150000 (0.00)	-0.00%	≈ 8s	**109207 (79.77)**	**-27.20%**	**≈6s**	0.684(0)
WFG1(4)	0.149	**88.573606 (0.541177)**	87.677943 (1.065401)	150000 (0.00)	-0.00%	≈ 95s	**107450 (66.60)**	**-28.37%**	**≈73s**	0.000(1)
WFG4(4)	0	**301.253225 (1.155539)**	300.667942 (1.016813)	150000 (0.00)	-0.00%	≈ 79s	**99999 (0.00)**	**-33.33%**	**≈48s**	0.013(1)
WFG1(5)	0.124	114.187823 (0.723549)	**114.272000 (0.955455)**	150000 (0.00)	-0.00%	≈ 1411s	**106165 (156.71)**	**-29.22%**	**≈935s**	0.290(0)
WFG4(5)	0.127	**3465.333620 (17.840209)**	3457.761494 (14.631125)	150000 (0.00)	-0.00%	≈ 1305s	**106336 (87.78)**	**-29.11%**	**≈898s**	0.185(0)

(b)

hypothesis "medians are equal" in the statistical analysis (see column $P(H)$) can be rejected. In the remaining problems, the "null hypothesis" cannot be rejected, and then, both algorithms have a similar behavior. However, it is important to note that iSMS-EMOA II saved computations of the contribution to I_H in many problems without losing quality in their solutions.

5.2 Results in Scenario 2

In Table 1(b), we can observe that iSMS-EMOA II was able to save from 20% to 33% of computations of the contribution to I_H in all test problems and as the number of objective functions increases, a bigger impact in the running time can be observed (e.g., in DTLZ1 with five objective functions iSMS-EMOA-II decreases the running time in 9.9 minutes). Regarding the quality of the solutions, we can note that in five test problems both algorithms have a similar behavior because the null hypothesis "medians are equal" cannot be rejected. In one test problem, iSMS-EMOA II outperformed the original iSMS-EMOA and it saved 30% of computations of the contribution to I_H. In twelve cases, the original iSMS-EMOA outperformed iSMS-EMOA II. However, in this scenario the main objective is to minimize the computations of the contribution to I_H without losing more than an epsilon (ϵ) of quality in the solutions regarding I_H.

6 Conclusions and Future Work

We have proposed to define a probability of use for the randomly selected individual adopted by iSMS-EMOA, with the aim of saving calculations of the contribution to I_H. To set this probability, we used the ϵ-constraint method and the EVOCA tuner. Our preliminary results show that savings of up to 33% of computations of the contribution to I_H can be obtained. From the point of view of the tuner algorithm, it was able to successfully deal with two different objectives in the process of selecting good performing calibrations. This indicates the suitability of this tuner for calibrating an indicator-based multi-objective evolutionary algorithm and motivates the incorporation of this approach on other MOEAs that use indicator-based selection or decomposition schemes.

References

1. Bringmann, K., Friedrich, T.: Approximating the least hypervolume contributor: NP-hard in general, but fast in practice. Theoretical Computer Science **425**, 104–116 (2012)
2. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable Test Problems for Evolutionary Multiobjective Optimization. In: Abraham, A., Jain, L., Goldberg, R. (eds.) Evolutionary Multiobjective Optimization. Theoretical Advances and Applications, pp. 105–145. Springer, USA (2005)
3. Emmerich, M., Beume, N., Naujoks, B.: An EMO Algorithm Using the Hypervolume Measure as Selection Criterion. In: Coello, C.A.C., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 62–76. Springer, Heidelberg (2005)

4. Farina, M., Amato, P.: On the Optimal Solution Definition for Many-criteria Optimization Problems. In: Proceedings of the NAFIPS-FLINT International Conference 2002, Piscataway, New Jersey, pp. 233–238. IEEE Service Center (June 2002)
5. Fleischer, M.: The Measure of Pareto Optima Applications to Multi-objective Metaheuristics. In: Fonseca, C.M., Fleming, P.J., Zitzler, E., Deb, K., Thiele, L. (eds.) EMO 2003. LNCS, vol. 2632, pp. 519–533. Springer, Heidelberg (2003)
6. Huband, S., Hingston, P., Barone, L., While, L.: A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. IEEE Transaction on Evolutionary Computation **10**(5), 477–506 (2006)
7. Menchaca-Mendez, A., Coello, C.A.C.: A New Selection Mechanism Based on Hypervolume and its Locality Property. In: 2013 IEEE Congress on Evolutionary Computation (CEC 2013), Cancún, México, June 20-23, pp. 924–931. IEEE Press (2013)
8. Riff, M.-C., Montero, E.: A new algorithm for reducing metaheuristic design effort. In: IEEE Congress on Evolutionary Computation (CEC 2013), Cancún, México, pp. 3283–3290 (June 2013)
9. Zitzler, E., Thiele, L.: Multiobjective Optimization Using Evolutionary Algorithms - A Comparative Case Study. In: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (eds.) PPSN 1998. LNCS, vol. 1498, pp. 292–301. Springer, Heidelberg (1998)
10. 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)