Parallel SMS-EMOA for Many-Objective Optimization Problems

Raquel Hernández Gómez
Computer Science Department
CINVESTAV-IPN
Mexico City, México
rhernandez@computacion.cs.cinvestav.mx

Carlos A. Coello Coello
Computer Science Department
CINVESTAV-IPN
Mexico City, México
ccoello@cs.cinvestav.mx

ABSTRACT

In the last decade, there has been a growing interest in multi-objective evolutionary algorithms that use performance indicators to guide the search. A simple and effective one is the $\mathcal{S}$-Metric Selection Evolutionary Multi-Objective Algorithm (SMS-EMOA), which is based on the hypervolume indicator. Even though the maximization of the hypervolume is equivalent to achieving Pareto optimality, its computational cost increases exponentially with the number of objectives, which severely limits its applicability to many-objective optimization problems. In this paper, we present a parallel version of SMS-EMOA, where the execution time is reduced through the asynchronous island model with micro-populations, and diversity is preserved by external archives that are pruned to a fixed size employing a recently created technique based on a Parallel-Coordinates graph. The proposed approach, called $\mathcal{S}$-PAMICRO (PArallel MICRo Optimizer based on the $\mathcal{S}$ metric), is compared with an state-of-the-art algorithm (HypE) on the WFG test problems. Preliminary experimental results show that $\mathcal{S}$-PAMICRO is a promising alternative that can solve many-objective optimization problems at an affordable computational cost.

1. INTRODUCTION

We are interested in solving Multi-objective Optimization Problems (MOPs), which have the following form:

$$\text{Minimize} \quad \vec{F}(\vec{x}) := (f_1(\vec{x}), f_2(\vec{x}), \ldots, f_m(\vec{x})) \qquad (1)$$

$$\text{subject to} \quad \vec{x} \in \mathcal{S}, \qquad (2)$$

where $\vec{x}$ is the *vector of decision variables*, $\mathcal{S} \subset \mathbb{R}^n$ is the *feasible region set* and $\vec{F}(\vec{x})$ is the vector of m (≥ 2) *objective functions* ($f_i : \mathbb{R}^n \to \mathbb{R}$). The aim is to seek from among the set of all values which satisfy the constraint functions defined in equation (2) the particular set $\vec{x^*}$ which yields the optimum values of all the objective functions.

Multi-objective Evolutionary Algorithms (MOEAs) are stochastic, population-based, search techniques which are well-suited for solving a wide variety of complex MOPs. In the last decades, several MOEAs have been proposed [4], with the vast majority relying on two concepts: *Pareto dominance*[1] as their primary selection mechanism, followed by a *density estimator.*[2] The former favors non-dominated solutions over dominated ones, whereas the latter induces a total order of incomparable solutions, preserving *diversity*[3] at the same time.

One of the main concerns is that several of these MOEAs face difficulties to reach the *Pareto optimal front*[4] when dealing with many-objective optimization problems ($m \geq 4$) [12]. This is due to the fact that most or all solutions in the population quickly become non-dominated with respect to the rest, and the best individuals are identified only by the density estimator. Thus, in some cases good locally non-dominated solutions in terms of convergence might be discarded at the expense of keeping good solutions in terms of diversity, in spite of the fact that they may be distant from the Pareto optimal front [1]. To address this issue, a new trend is the incorporation of *performance indicators*[5] into the selection mechanism of a MOEA [2, 5, 8]. The *hypervolume indicator* [15] is, with no doubt, a natural choice, since it is the only unary indicator that is known to be Pareto compliant. Also, it has been proven that maximizing the hypervolume is equivalent to reaching the Pareto optimal set [6]. However, its main drawback is its computational cost, which increases exponentially with the number of objectives [3], making it prohibitive for many-objective optimization problems.

In this work, we focus on the $\mathcal{S}$-Metric Selection Evolutionary Multi-Objective Algorithm (SMS-EMOA) [5], due to its simplicity and superiority over other algorithms [14]. This optimizer is a steady state genetic algorithm that ranks individuals according to Pareto dominance and uses the hypervolume as its density estimator. The computational complexity of SMS-EMOA is of order $\mathcal{O}(|P|^m)$, where $|P|$ denotes the population size. Parallelizing SMS-EMOA arises

GECCO'16 Companion, July 20-24, 2016, Denver, CO, USA

DOI: http://dx.doi.org/10.1145/2908961.2931663

---

[1] A solution $\vec{x} \in \mathcal{S}$ dominates a solution $\vec{y} \in \mathcal{S}$ ($\vec{x} \prec \vec{y}$), if and only if $\forall i \in \{1, \ldots, m\}$, $f_i(\vec{x}) \leq f_i(\vec{y})$ and $\exists j \in \{1, \ldots, m\}$, $f_j(\vec{x}) < f_j(\vec{y})$.

[2] A density estimator measures the similarity degree among individuals in the population.

[3] *Diversity* refers to achieving a uniform distribution of solutions covering all regions of the objective space.

[4] $POF := \{\vec{F}(\vec{x}) \in \mathbb{R}^m : \vec{x} \in \mathcal{S}, \nexists \vec{y} \in \mathcal{S}, \vec{y} \prec \vec{x}\}$.

[5] A performance indicator, defined as $I : \mathbb{R}^m \to \mathbb{R}$, measures the quality of an approximation set (the final population of a MOEA).

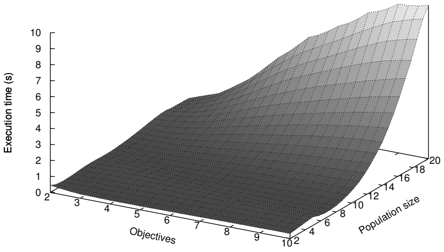

Figure 1: Average execution time of SMS-EMOA varying objective functions and population size.

as a possible way to reduce its computational cost, where at least two strategies are possible [13]: (1) parallelization of the computations, in which the operations applied to an individual are performed in parallel, and (2) parallelization of the population, in which the population is partitioned and each part evolves in semi-isolation (individuals can be exchanged between subpopulations). Klinkenberg et al. [10] and Lopez et al. [11] have studied the first approach. In [10], a variation of SMS-EMOA parallelized the evaluations of individuals using a surrogate model, whose purpose was to approximate the function values. In [11], the exact hypervolume contributions of SMS-EMOA were parallelized through the use of Graphics Processing Units (GPUs). To the best of our knowledge, our work is the first attempt to incorporate the second sort of approach (parallelization of the population) into SMS-EMOA.

In order to grasp the variability of the execution times of SMS-EMOA, we sampled several points on DTLZ1 [4], varying the number of objective functions and the population size on a PC Intel(R) Core(TM) i7 CPU 950 @ 3.07 GHz × 8 with 3.8 GB of RAM, using the same parameters in all experiments [5]. The average resulting surface of 30 runs is shown in Figure 1. An interesting observation is that, regardless of the number of objectives, time was almost negligible when using small populations (less than 12 individuals). This fact is considered in our proposal, where we improve diversity using the parallel asynchronous island model [4] and external archives for each micro-population. Furthermore, these external archives are kept to a constant size by a recently proposed density estimator based on the visualization technique of Parallel Coordinates [7], which is scalable in the objective space.

The remainder of this paper is organized as follows. Section 2 is devoted to the description of our proposed parallel MOEA. In Section 3 we present our experimental results using the Walking-Fish-Group (WFG) [4] benchmark. Finally, Section 4 provides our preliminary conclusions and some possible potential lines of future research.

2. OUR PROPOSED APPROACH

The PArallel MICRo Optimizer based on the $\mathcal{S}$ metric ($\mathcal{S}$-PAMICRO) draws ideas from the island model, where the overall population is split into l micro-populations, called *islands*, containing less than 12 individuals each. Every island evolves independently a serial SMS-EMOA with an external archive of size $l|P|$. In this approach, the islands are connected in a logical unidirectional ring, exchanging *nmig* solutions occasionally[6] in an asynchronous fashion. The goal of $\mathcal{S}$-PAMICRO is to reduce the execution time of SMS-EMOA, hopefully also improving the quality of solutions in high dimensional spaces.

Algorithm 1 Outline of an island in $\mathcal{S}$-PAMICRO

```
Input: MOP, stopping criterion, island identification $i$, number of islands $l$, number of migrants $nmig$, and frequency of migration $fmig$.
Output: Final sub-population $A$
 1: $A \leftarrow \emptyset$
 2: $n \leftarrow l|P|$ {archive size limit}
 3: Initialize micro-population $P$ at random
 4: while the stopping criterion is not satisfied do
 5:     $P \leftarrow$ SMS-EMOA(MOP, $fmig$, $P$) {execute during $fmig$ evaluations of the objective vector}
 6:     $R \leftarrow$ Check the arrival of migrants from $(l+i-1) \pmod{l}$ island
 7:     $A \leftarrow A \cup P \cup R$
 8:     if $|A| > n$ then
 9:         $A \leftarrow$ Pruning($A$,$n$)
10:     $S \leftarrow$Uniform_Random_Selection($A$)
11:     Send copies of $S$ to the $(i+1) \pmod{l}$ island
12:     $P \leftarrow$ Elitist_Ranking_Replacement($P \cup R$)
13: return $A$
```

In Algorithm 1, we present the pseudocode of an island in $\mathcal{S}$-PAMICRO. First, the external archive A and its maximum size are specified. Next, the micro-population P is initialized at random. In line 5, SMS-EMOA is executed during *fmig* function evaluations. Then, an island receives, without blocking, the immigrants R from the source island, according to the adopted topology. In line 7, the external archive is updated, adding the current micro-population as well as the immigrants. In lines 8 and 9, the external archive is truncated if it exceeds its limits, using the technique described in the next paragraph. In the following two lines, the candidates to be migrated are selected by using the policy of uniform-random migration [4], in which *nmig* random individuals are selected from the archive and a copy of them is sent to the destination island. In line 12, the micro-population is updated, replacing some individuals with the immigrants. Here, we employed elitist-ranking replacement [4], where immigrants are combined with the current population, and then they are ranked using Pareto dominance, and the worst solutions are removed. This elitist mechanism preserves the currently best solutions for the next iteration, assuring proximity to the Pareto optimal front. At the end, the final sub-populations of all islands $i \in \{0, 1, \ldots, l-1\}$ are collected and adjusted to the size $l|P|$, using the same pruning technique.

Our pruning technique is explained in Algorithm 2. First the population is ranked using the well-known non-dominated sorting procedure [4]. In line 2, the population is normalized in the objective space by means of two reference points: z^{min}, composed of the best objective values found so far, and z^{max}, formed with those vectors parallel to the axes with the lowest L_2 norm. Next, individuals are removed from the worst current front. If the size of this front is less or equal than the number of individuals to be removed, then the whole front is discarded (lines 4-6). Otherwise, the

---

[6] This is known as *migration*.

most densely populated members are eliminated from the current front (lines 8-11). The density estimator, originally proposed by the authors of this paper in [7], is based on a visualization technique, called Parallel Coordinates.[7] The core idea is to create a *digital image*[8] containing the Parallel Coordinates of each distinct pair of objective functions. These $m(m-1)/2$ digital images are attached next to each other and only normalized individuals are considered. Such images are represented as a 2D matrix, whose dimension depends on the number of objectives (m), the population size $|P|$ and a resolution parameter (γ). An element of this matrix identifies the level of overlapping line segments and those individuals covering a wide area of the image have a lower density estimator. Interested readers are referred to [7] for more details.

Algorithm 2 Pruning

```
Input: Population $P$, desired size $n$
Output: Reduced population $P$
 1: $\{F_1, \ldots, F_k\} \leftarrow$ Rank population $P$ in fronts according
    to Pareto dominance.
 2: Normalize population $P$
 3: while $|P| > n$ do
 4:   if $|F_k| \leq |P| - n$ then
 5:     $r \leftarrow F_k$
 6:     $k \leftarrow k - 1$
 7:   else
 8:     $D \leftarrow$ Calculate pop. density of $P$
 9:     $r \leftarrow \arg\max_{\vec{p} \in F_k} D[p]$
10:     $F_k \leftarrow F_k \setminus \{r\}$
11:   $P \leftarrow P \setminus \{r\}$
12: return $P$
```

$\mathcal{S}$-PAMICRO was developed in the *EMO Project*, our framework for Evolutionary Multi-Objective Optimization. This software is implemented in C language and MPICH[9].

3. EXPERIMENTAL RESULTS

In order to illustrate the efficiency of $\mathcal{S}$-PAMICRO in many-objective optimization problems, we focus on WFG1 [4] (the complete study is available at [9]), which is a well-known difficult, separable and unimodal test problem with mixed geometry. The decision variables (n) of this MOP were set to $\{24, 24, 47, 105\}$ and the position-related parameter (k) were set to $\{4, 4, 8, 18\}$ for $\{2, 3, 5, 10\}$ objectives, respectively. We compared the results of our proposed algorithm with respect to SMS-EMOA, $\mathcal{S}$-PAMICRO without external archives (pSMS-EMOA), and the Hypervolume Estimation Algorithm (HypE) [2]. This latter MOEA ranks the population by means of Pareto dominance and its secondary selection criterion is based on the estimation of the hypervolume contributions using Monte Carlo sampling (for 2 and 3 objectives, the exact value is computed).

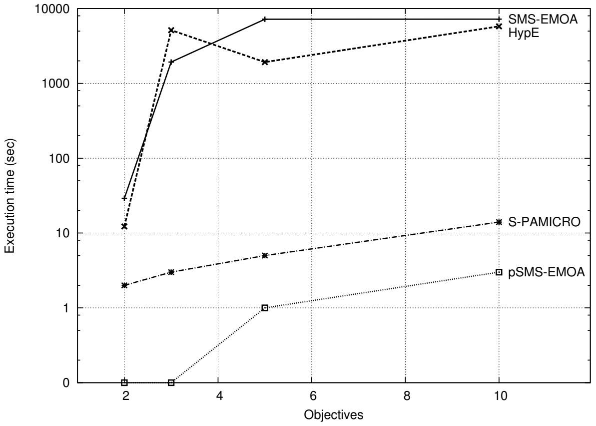

Figure 2: Average execution time on WFG1.

The total number of function evaluations was set to 40,000, 50,000 and 80,000 for 2D, 3-5D and 10D, respectively; limiting the execution time to no more than two hours for each run. In the sequential algorithms, the population size of SMS-EMOA and HypE was set to 100, 120, 198 and 275 for 2D, 3D, 5D and 10D, respectively. In the case of pSMS-EMOA and $\mathcal{S}$-PAMICRO the micro-populations were set to 10 for two and three objectives, and to 11 for five and ten dimensions. The number of islands (processors) were 10, 12, 18 and 25 for 2D, 3D, 5D and 10D, respectively.

Experiments were carried on a Cluster of 10 PCs Intel(R) Core(TM) i7 CPU 950 @ 3.07 GHz × 8 with 3.8 GB of RAM. The frequency of migration, $fmig$, was set to 80 function evaluations and the number of migrants $nmig$ was set to 2. All the MOEAs were implemented in the EMO Project, using real-numbers encoding. For fair comparisons, the parameters were similar in the sequential and parallel cases. The crossover and mutation rate were set to 0.9 and $1/n$, respectively. For HypE, the number of sampling points was fixed to 20,000. The resolution parameter of $\mathcal{S}$-PAMICRO (γ) was set to 3 for 2D and 2 for higher dimensions [7].

We performed 30 independent runs for all scenarios. For comparing results, we adopted the hypervolume indicator, bounded by the reference point $(2.2, 4.2, 6.2, \ldots)$. We applied the Wilcoxon rank sum test (one-tailed) to the mean hypervolume indicator values, in order to determine wheter if $\mathcal{S}$-PAMICRO performed better than the other MOEAs at the significance level of 5%.

The resulting execution time, using a logarithmic scale for the y-axis, is shown in Figure 2. As it can be observed, $\mathcal{S}$-PAMICRO spent considerably less time than SMS-EMOA and HypE. Even in low dimensionality, our algorithm could reduce run time a little bit. Moreover, the overhead of handling the external archive in $\mathcal{S}$-PAMICRO is relatively low.

But much more important are our results with respect to the quality of the solutions. In Table 1, we present the hypervolume indicator values of all the experiments. The best values are shown in gray scale. An arrow pointing upwards ($\uparrow$) means that our algorithm outperformed in a significantly better way, the other MOEAs compared. Conversely, an arrow pointing downwards ($\downarrow$) means that our algorithm was significantly beaten. For 2, 5 and 10 objectives $\mathcal{S}$-PAMICRO obtained the best results, outperforming SMS-EMOA, HypE and pSMS-EMOA. While in 3D, our proposal surpassed pSMS-EMOA and SMS-EMOA.

---

[7] This graph is built in the 2-dimensional plane, where m copies of the real line $\mathbb{R}$ are placed perpendicular to the x-axis and a point in $\mathbb{R}^m$ is represented by a series of connected line segments with vertices on the parallel axes.

[8] The term *digital image* refers to a two-dimensional light intensity function $g(a, b)$ where a and b denote spatial coordinates and the value of g at any point (a, b) is proportional to the gray level of the image at that point; where a, b, and g take discrete values.

[9] https://www.mpich.org

Table 1: Median and standard deviation of the hypervolume indicator on the WFG1.

m	HypE	SMS-EMOA	pSMS-EMOA	$\mathcal{S}$-PAMICRO
10	4.19e+09 ↑ (1.81e+8)	1.88e+09 ↑ (2.62e+8)	5.28e+09 ↑ (5.76e+7)	5.87e+09 (2.33e+8)
5	2.82e+03 ↑ (1.17e+2)	3.18e+03 ↑ (7.20e+1)	3.91e+03 ↑ (4.83e+1)	5.16e+03 (3.88e+2)
3	5.66e+01 ↓ (1.62e+0)	5.28e+01 ↑ (2.50e+0)	4.23e+01 ↑ (3.08e+0)	5.56e+01 (3.71e+0)
2	5.17e+00 ↑ (4.11e-1)	4.45e+00 ↑ (3.63e-1)	3.66e+00 ↑ (2.59e-1)	6.61e+00 (9.65e-1)

In summary, we observed that $\mathcal{S}$-PAMICRO could achieve much better results than SMS-EMOA and HypE in high dimensionality, spending much less computational time. For this reason, we believe that our proposed approach is a promising alternative for solving many-objective optimization problems.

4. CONCLUSIONS AND FUTURE WORK

This paper presented a parallel version of the $\mathcal{S}$-Metric Selection Evolutionary Multi-Objective Algorithm (SMS-EMOA). The new approach, called PArallel MICRo Optimizer based on the $\mathcal{S}$ metric ($\mathcal{S}$-PAMICRO), draws ideas from the asynchronous island model with relatively small populations. Diversity is preserved through external archives that are pruned to a limit size, using a recently proposed technique that is based on automatic image analysis. We compared our proposal with respect to HypE (Hypervolume Estimation Algorithm), and with respect to the serial version of SMS-EMOA and another parallel version of it. We observed that $\mathcal{S}$-PAMICRO is a viable alternative for solving many-objective optimization problems at an affordable computational time. Further studies are required using more benchmarks. We are also interested in studying the effects of the additional parameters related to the migration operator.

5. REFERENCES

[1] S. F. Adra and P. J. Fleming. Diversity Management in Evolutionary Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation*, 15(2):183–195, April 2011.

[2] J. Bader and E. Zitzler. HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization. *Evolutionary Computation*, 19(1):45–76, Spring, 2011.

[3] K. Bringmann and T. Friedrich. Don't be Greedy when Calculating Hypervolume Contributions. In *FOGA '09: Proceedings of the tenth ACM SIGEVO workshop on Foundations of genetic algorithms*, pages 103–112, Orlando, Florida, USA, January 2009. ACM.

[4] C. A. Coello Coello, G. B. Lamont, and D. A. Van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, New York, second edition, September 2007. ISBN 978-0-387-33254-3.

[5] M. Emmerich, N. Beume, and B. Naujoks. An Emo Algorithm using the Hypervolume Measure as Selection Criterion. In C. Coello Coello, A. Hernández Aguirre, and E. Zitzler, editors, *Evolutionary Multi-Criterion Optimization*, volume 3410 of *Lecture Notes in Computer Science*, pages 62–76. Springer Berlin Heidelberg, 2005.

[6] M. Fleischer. The Measure of Pareto Optima Applications to Multi-objective Metaheuristics. In C. Fonseca, P. Fleming, E. Zitzler, L. Thiele, and K. Deb, editors, *Evolutionary Multi-Criterion Optimization*, volume 2632 of *Lecture Notes in Computer Science*, pages 519–533. Springer Berlin Heidelberg, 2003.

[7] R. Hernández Gómez and C. A. Coello Coello. A Multi-Objective Evolutionary Algorithm based on Parallel Coordinates. GECCO '16, New York, NY, USA. ACM. To be appear in the Proceedings of the 2016 Annual Conference on Genetic and Evolutionary Computation, July 2016.

[8] R. Hernández Gómez and C. A. Coello Coello. Improved Metaheuristic Based on the R2 Indicator for Many-Objective Optimization. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, GECCO '15, pages 679–686, New York, NY, USA, 2015. ACM.

[9] R. Hernández Gómez, C. A. Coello Coello, and E. Alba. A Parallel Version of SMS-EMOA for Many-Objective Optimization Problems. Technical Report EVOCINV-01-2016, Department of Computer Science, CINVESTAV-IPN, Mexico City, México, May 2016. http://delta.cs.cinvestav.mx/~ccoello/2016.html

[10] J.-W. Klinkenberg, M. T. M. Emmerich, A. H. Deutz, O. M. Shir, and T. Bäck. *Multiple Criteria Decision Making for Sustainable Energy and Transportation Systems: Proceedings of the 19th International Conference on Multiple Criteria Decision Making, Auckland, New Zealand, 7th - 12th January 2008*, chapter A Reduced-Cost SMS-EMOA Using Kriging, Self-Adaptation, and Parallelization, pages 301–311. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[11] E. M. Lopez, L. M. Antonio, and C. A. Coello Coello. *Evolutionary Multi-Criterion Optimization: 8th International Conference, EMO 2015, Guimarães, Portugal, March 29 –April 1, 2015. Proceedings, Part II*, chapter A GPU-Based Algorithm for a Faster Hypervolume Contribution Computation, pages 80–94. Springer International Publishing, Cham, 2015.

[12] C. Lücken, B. Barán, and C. Brizuela. A Survey on Multi-Objective Evolutionary Algorithms for Many-Objective Problems. *Computational Optimization and Applications*, 58(3):707–756, 2014.

[13] F. Luna and E. Alba. Parallel multiobjective evolutionary algorithms. In J. Kacprzyk and W. Pedrycz, editors, *Springer Handbook of Computational Intelligence*, pages 1017–1031. Springer Berlin Heidelberg, 2015.

[14] T. Wagner, N. Beume, and B. Naujoks. *Evolutionary Multi-Criterion Optimization: 4th International Conference, EMO 2007, Matsushima, Japan, March 5-8, 2007. Proceedings*, chapter Pareto-, Aggregation-, and Indicator-Based Methods in Many-Objective Optimization, pages 742–756. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[15] E. Zitzler. *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, November 1999.