A Multi-objective Particle Swarm Optimizer Based on Decomposition

Saul Zapotecas Martinez
CINVESTAV-IPN (EVOCINV)
Departamento de Computación
México, D.F. 07360, MEXICO
saul.zapotecas@gmail.com

Carlos A. Coello Coello
CINVESTAV-IPN (EVOCINV)
Departamento de Computación
México, D.F. 07360, MEXICO
ccoello@cs.cinvestav.mx

ABSTRACT

The simplicity and success of particle swarm optimization (PSO) algorithms, has motivated researchers to extend the use of these techniques to the multi-objective optimization field. This paper presents a multi-objective particle swarm optimization (MOPSO) algorithm based on a decomposition approach, which is intended for solving continuous and unconstrained multi-objective optimization problems (MOPs). The proposed decomposition-based multi-objective particle swarm optimizer (dMOPSO), updates the position of each particle using a set of solutions considered as the global best according to the decomposition approach. dMOPSO is mainly characterized by the use of a memory reinitialization process which aims to provide diversity to the swarm. Our proposed approach is compared with respect to two decomposition-based multi-objective evolutionary algorithms (MOEAs) which are representative of the state-of-the-art in the area. Our results indicate that our proposed approach is competitive and it outperforms the two MOEAs with respect to which it was compared in most of the test problems adopted.

Categories and Subject Descriptors
I.2.8 [Computing Methodologies]: Artificial Intelligence—Problem Solving, Control Methods, and Search.

General Terms
Algorithms, Theory.

Keywords
Multi-objective optimization, particle swarm optimization, decomposition approach

1. INTRODUCTION

Particle swarm optimization (PSO) [6] has been found to be a very successful bio-inspired metaheuristic for dealing with unconstrained and continuous optimization problems. This has motivated the interest of researchers in extending PSO to the solution of multi-objective optimization problems (for a survey on multi-objective particle swarm optimizers, see [18]).

Recently, a new multi-objective evolutionary algorithm based on a decomposition approach (MOEA/D) [21] has been proposed. MOEA/D decomposes a MOP into several single-objective optimization problems. In this way, a set of approximate solutions to the Pareto optimal set is reached by minimizing each subproblem instead of using Pareto ranking. This has given rise to a new generation of MOEAs.

Regarding multi-objective particle swarm optimizers (MOPSOs), to the authors’ best knowledge, there exist only two of them based on a decomposition approach (see [16, 15]). However, we believe that the potential of this sort of approach, when combined with PSO, has not been fully exploited, which has motivated the work reported in this paper.

Peng and Zhang [16] proposed the multi-objective particle swarm optimizer based on decomposition (MOPSO/D). This approach uses the framework adopted by MOEA/D but replaces the genetic operators (crossover and mutation) by the inertia flight equations used in traditional PSO. MOPSO/D uses a turbulence (or mutation) operator and adopts an archiving strategy (which is based on $\varepsilon$-dominance [13]) to store the nondominated solutions found during the search.

More recently, Moubayed et al. [15] proposed a novel smart multi-objective particle swarm optimizer using decomposition (SDMOPSO). This algorithm is also based on MOEA/D, and adopts an external archive based on $\varepsilon$-dominance for maintaining diversity in the swarm. An interesting aspect of this approach is that it uses Pareto dominance for updating the personal best position of each particle. In SDMOPSO, the global best set of each particle is defined by all the solutions located within a certain neighborhood, and this set is updated analogously to the mechanism adopted by a MOEA/D variant proposed in [22].

In this paper, we introduce a new decomposition-based multi-objective particle swarm optimizer (dMOPSO) for continuous and unconstrained MOPs. The proposed approach does not require either the use of turbulence or the Pareto optimality concept for approximating solutions towards the Pareto optimal set. Instead, we adopt both a mechanism for selecting global best solutions based on the nature of the decomposition approach and a mechanism to reinitialize the particles based on their age. dMOPSO avoids the use of
an external archive, which results in a lower computational time than that of state-of-the-art MOPSOs.

The remainder of the paper is organized as follows. In Section 2, we provide the basic background required for understanding the rest of the paper. In Section 3, we explain in detail our proposed approach. In Section 4, we show the results of our comparative study. Finally, in Section 5 we provide our conclusions and some possible paths for future work.

2. BASIC BACKGROUND

2.1 Multi-Objective Optimization

A continuous and unconstrained multi-objective optimization problem can be stated \(^1\) as follows:

\[
\min_{x \in \Omega} \{F(x)\}
\]

where \(\Omega\) define the decision variable space and \(F\) is defined as the vector of the objective functions:

\[
F : \Omega \rightarrow \mathbb{R}^k, \quad F(x) = (f_1(x), \ldots, f_k(x))^T
\]

where \(f_i : \mathbb{R}^n \rightarrow \mathbb{R}\) is a continuous and unconstrained function.

In multi-objective optimization, a set of trade-off solutions are normally aimed for, because to minimize a function \(f_i\) implies to deteriorate another one. To describe the concept of optimality in which we are interested, the following definitions are provided.

**Definition 1.** Let \(x, y \in \Omega\), we say that \(x\) dominates \(y\) (denoted by \(x < y\)) if and only if \(f_i(x) \leq f_i(y)\) and \(F(x) \neq F(y)\).

**Definition 2.** Let \(x^* \in \Omega\), we say that \(x^*\) is a Pareto optimal solution, if there is no other solution \(y \in \Omega\) such that \(y < x^*\).

**Definition 3.** The Pareto Optimal Set \(\mathcal{P}S\) is defined by:

\[
\mathcal{P}S = \{x \in \Omega | x \text{ is Pareto optimal solution}\}
\]

and its image \((\mathcal{P}F = \{F(x) | x \in \mathcal{P}S\})\) is called the Pareto Optimal front.

We are interested in generating as many (different) elements of the Pareto optimal set as possible, while maintaining a distribution of solutions as uniform as possible along the Pareto front.

2.2 Decomposition Approach

Although Pareto ranking has been the most common approach adopted by MOEAs in the last few years, scalarization functions have been available in the literature for a much longer time (see for example [7, 14, 20]). These methods use a weighted vector (as a search direction) to define a scalar function. In this way and under certain assumptions (e.g., the minimum is unique, the weighting coefficients are positive, etc.), an optimal Pareto solution is achieved by minimizing such a scalarization function. Among these methods, probably the two most widely used are the Tchebycheff and the Weighted Sum approaches. However, as it has been previously discussed in [3, 21], the approaches based on boundary intersection possess certain advantages over those based on either Tchebycheff or the Weighted Sum. Next, we introduce a decomposition approach based on the boundary intersection, which is the approach adopted in this work.

\(^1\)Without loss of generality, we assume minimization

2.2.1 Penalty Boundary Intersection Approach

The Penalty Boundary Intersection (PBI)\(^2\) approach proposed by Zhang and Li [21], uses a weighted vector \(w\) and a penalty value \(\theta\) for minimizing both the distance to the utopia vector \(d_1\) and the direction error to the weighted vector \(d_2\) from the solution \(F(x)\) (see Fig. 1). Therefore, the optimization problem can be stated as:

\[
\text{minimize: } g(x|w, z^*) = d_1 + \theta d_2
\]

where

\[
d_1 = \frac{||F(x) - z^*||^T w}{||w||}
\]

and

\[
d_2 = \frac{||F(x) - z^* - d_1||}{||w||}
\]

as the Tchebycheff approach, \(x \in \mathbb{R}^n\) and \(z^* = \min\{f_i(x) | x \in \Omega\}\).

In this way, both the Tchebycheff and the PBI approaches can generate a good representation of the Pareto front by defining a well-distributed set of weighted vectors.

2.3 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a metaheuristic originally proposed by Kennedy and Eberhart [11] for dealing with continuous and unconstrained nonlinear optimization problems. PSO simulates the movements of a flock of birds which aim to find food. In PSO, a particle is represented by a solution \(x_i \in \mathbb{R}^n\). A population can be defined by one or several swarms of particles. Each swarm is evolved by one or several swarms of particles. Each swarm is evolved by updating both the velocity \(v_i^{t+1}\) and the position of each particle \(x_i^{t+1}\) according to the following equations:

\[
v_i^{t+1} = wv_i^t + c_1r_1(x_{pb,i} - x_i^t) + c_2r_2(x_{gb,i} - x_i^t)
\]

and the new particle’s position is updated according to the equation:

\[
x_i^{t+1} = x_i^t + v_i^{t+1}
\]

where \(w \geq 0\) represents the inertia factor, \(c_1, c_2 \geq 0\) are the constraints on the velocity, \(r_1, r_2\) are two random variables having a uniform distribution in the range \((0, 1)\), \(v_i, x_{pb,i}\) and \(x_{gb,i}\) represent the velocity, the personal best and the global best position for the \(i^{th}\) particle, respectively.

\(^2\)based on the well-known Normal Boundary Intersection (NBI) method [3]
3. THE DECOMPOSITION-BASED MOPSO

3.1 General Framework

Our proposed dMOPSO, employs a decomposition-based framework similar to the one adopted by MOEA/D. Therefore, a well-distributed set of weighted vectors $W$ has to be previously defined. Here, we use the same method as in [21], however, other methods can be used, see for example [2]. The boundary-intersection-based techniques possess certain advantages over Tchebycheff-based techniques (which have been previously discussed in [3] and [21]). However, the PBI approach is adopted in our dMOPSO. Because of that, the notion of elitism used in Tchebycheff or another decomposition approach could also be easily coupled to our approach.

At the beginning of the algorithm, a set of $N$ particles $P = \{x_1, \ldots, x_N\}$ is randomly initialized. Each particle possesses a flight velocity $v_i$ and an age $a_i$, both of which are initially set to zero. Along the flight cycles, each particle tries to minimize one of the subproblems defined by the weighted vector $w_i$. Therefore, each particle undertakes its flight towards a better position in order to minimize a single subproblem $g(x_i | w_i, z^*)$.

The personal best $x_{pb,i}$ of the $i^{th}$ particle, represents the best position of the particle to the $i^{th}$ subproblem. Since at the beginning a particle does not have a previous movement, the best personal position is initialized with the same position as the particle, i.e., $x_{pb,i} = x_i$.

At each cycle, the flight historical record of each particle is used to find the best solutions to each subproblem. Therefore, the set of global best $G_{best}$ is defined in a natural way. This set will contain the solutions that minimize each subproblem, and it is updated in each cycle according to the Algorithm 1. Thus, the notion of elitism used in evolutionary multi-objective optimization is implicitly employed. However, in this case, a decomposition approach is used instead of the more traditional Pareto optimality.

Once the global best set has been defined, the velocity and the position of each particle are updated according to equations (3) and (4), respectively. Since the proposed approach tries to minimize a set of subproblems (whose solutions at the end of the flight cycles should be very close to the Pareto optimal set), and the Algorithm 1 introduces a high selection pressure that should contribute to this, we assume that all solutions in $G_{best}$ are equally good (i.e., we assume that all the subproblems were satisfactorily solved). Thus, the velocity of each particle is computed using as their global best a solution which is randomly taken from $G_{best}$.

The age of each particle, promotes the diversity along the flight cycles and indicates when a particle is not providing good information in its flight experience. If a particle does not improve its personal position in a flight cycle, then the particle increases (by one) its age. On the other hand, if a particle exceeds a certain (pre-defined) age threshold, the particle (including its velocity, its age and its best personal) is reinitialized. dMOPSO employs a reinitialization mechanism based on a parametric probability density function, which involves the selected global best $x_{gb,i}$ and the personal best $x_{pb,i}$ of the current particle $x_i^t$. With that, dMOPSO aims to perform smart reinitialization movements from the personal best towards the global best solutions using a Gaussian distribution, as is shown in Fig. 2. Therefore, the $j^{th}$ component of the new particle is reset according to the following equation 3:

$$x_i^{t+1}(j) = N\left(\frac{x_{pb,i}(j) - x_{gb,i}(j)}{2}, |x_{pb,i}(j) - x_{gb,i}(j)|\right)$$

where $N(m, \sigma)$ represents a random number using a normal distribution with mean $m$ and sigma $\sigma$.

In PSO, it is very common that some particles go beyond the boundaries of the decision variables. In order to deal with this problem, we used one of the nine methods studied in [1]. Here, we adopted the deterministic back to repair solutions that are generated outside the allowable bounds. Therefore, the $j^{th}$ bound of the particle’s position $x_i^{t+1}$ and of its velocity $v_i^{t+1}$ are re-established as follows:

$$x_i^{t+1}(j) = \begin{cases} x_{ub}(j), & \text{if } x_i^{t+1}(j) < x_{lb}(j) \\ x_i^{t+1}(j), & \text{if } x_i^{t+1}(j) > x_{ub}(j) \end{cases}$$
$$v_i^{t+1}(j) = -\gamma v_i^{t+1}(j)$$

where $x_{lb}(j)$ and $x_{ub}(j)$ are the lower and upper bounds in the $j^{th}$ component of the allowable decision variable values, respectively. In the same way than other MOPSOs [18, 15], dMOPSO adopts $\gamma = 1$.

Summarizing, the dMOPSO algorithm can be stated as follows:

**Step 1) Initialization**

**Step 1.1** $t = 0$ // the number of flight cycles

**Step 1.2** Generate a well-distributed set of $N$ weighted vectors $W = \{w_1, \ldots, w_N\}$.

**Step 1.3** Generate a swarm $P^t = \{x_1, \ldots, x_N\}$ of $N$ random particles.

**Step 1.4** Initially, the velocity $v_i$ and the age $a_i$ of each particle $x_i$ is set in zero, i.e. $v_i^0 = a_i^0 = 0$ (for $i = 1, \ldots, N$).

**Step 1.5** Define the personal best: $x_{pb,i} = x_i$ (for $i = 1, \ldots, N$).

---

3This equation is a generalization of the one given in [10].
4. EXPERIMENTAL RESULTS

4.1 Test Problems

In order to assess the performance of the proposed approach, we compare its results with respect to those obtained by a MOEA/D [21] and SDMOPSO [15]. We adopted nine test problems whose Pareto fronts have different characteristics including convexity, concavity, disconnections and multi-frontality. The first adopted MOP was proposed by Fonseca [8]. The two-objective test suite of Zitzler-Deb-Thiele (ZDT) [23] (except for ZDT5, which is a binary problem) is also adopted. For three-objective problems, we adopted three problems taken from the Deb-Thiele-Laumanns-Zitzler (DTLZ) test suite [5] (DTLZ2, DTLZ6 and DTLZ7).

We used 30 decision variables for ZDT1, ZDT2 and ZDT3. DDT4 and DDT6 were tested using 10 decision variables. For DTLZ2 and DTLZ6, 12 variables were adopted. DTLZ7 was tested with 22 decision variables. Finally, Fonseca’s problem was tested using only three decision variables.

4.2 Performance Measures

4.2.1 Hypervolume

The Hypervolume (Hv) performance measure was proposed by Zitzler [25]. This performance measure is Pareto compliant [26] and quantifies the approximation of nondominated solutions to the Pareto optimal front. The hypervolume corresponds to the non-overlapped volume of all the Pareto front. Instead, it relies on the penalty from the PBI approach, which provides diversity to such solutions along the flight circuits. However, the use of other decomposition approaches could deteriorate the performance of our dMOPSO, especially when dealing with more complex MOPs. This is because the PBI approach forces particles to follow a single direction, since moving away from such direction will be penalized. dMOPSO does not use any archiving strategy and, instead, it relies on the fact that at the end of flight cycles the subproblems will be minimized and, thereby, nondominated solutions will be reached. Nevertheless, the use of archiving strategies (see e.g. [12, 13, 24]) could be used to cover those drawbacks that weighted-vector-based methods possess (e.g. when dealing with MOPs having either disconnected Pareto fronts or many objective functions). They could be employed to store well-distributed nondominated solutions, while the set of global best solutions could be used as guides in the optimization process.

Table 1: Parameters for dMOPSO, SDMOPSO and MOEA/D

<table>
<thead>
<tr>
<th>Parameter</th>
<th>dMOPSO</th>
<th>SDMOPSO</th>
<th>MOEA/D</th>
</tr>
</thead>
<tbody>
<tr>
<td>N_gen</td>
<td>100/300</td>
<td>100/300</td>
<td>100/300</td>
</tr>
<tr>
<td>Npop</td>
<td>150</td>
<td>150</td>
<td>150</td>
</tr>
<tr>
<td>Narc</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r惹</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>T1</td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>T2</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>T3</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>T4</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>T5</td>
<td>12/n</td>
<td>12/n</td>
<td>12/n</td>
</tr>
<tr>
<td>θ</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>
Table 2: Results of the $\mathcal{H}_V$ performance measure for dMOPSO, SDMOPSO and MOEA/D

<table>
<thead>
<tr>
<th>MOP</th>
<th>dMOPSO (average $\sigma$)</th>
<th>SDMOPSO (average $\sigma$)</th>
<th>MOEA/D (average $\sigma$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FONSECA</td>
<td>0.543629 (0.00197)</td>
<td>0.541803 (0.00174)</td>
<td>0.383232 (0.00104)</td>
</tr>
<tr>
<td>ZDT1</td>
<td>0.806928 (0.00460)</td>
<td>0.102469 (0.006452)</td>
<td>0.809970 (0.003063)</td>
</tr>
<tr>
<td>ZDT2</td>
<td>0.536473 (0.000425)</td>
<td>0.267007 (0.0035759)</td>
<td>0.316948 (0.0019159)</td>
</tr>
<tr>
<td>ZDT3</td>
<td>0.517748 (0.0092921)</td>
<td>0.114288 (0.0090791)</td>
<td>1.236748 (0.044609)</td>
</tr>
<tr>
<td>ZDT4</td>
<td>0.562445 (0.0009990)</td>
<td>0.099000 (0.0065025)</td>
<td>0.747990 (0.0065285)</td>
</tr>
<tr>
<td>ZDT6</td>
<td>0.504515 (0.0008649)</td>
<td>0.092415 (0.0084559)</td>
<td>0.437862 (0.0009460)</td>
</tr>
<tr>
<td>DT LZ52</td>
<td>0.741219 (0.0012310)</td>
<td>0.713463 (0.0050123)</td>
<td>0.771041 (0.003869)</td>
</tr>
<tr>
<td>DT LZ6</td>
<td>0.426532 (0.000009)</td>
<td>0.332548 (0.149801)</td>
<td>0.420153 (0.000998)</td>
</tr>
<tr>
<td>DT LZ7</td>
<td>0.149133 (0.007166)</td>
<td>1.115480 (0.068608)</td>
<td>1.375030 (0.145559)</td>
</tr>
</tbody>
</table>

Table 3: Results of the $S_P$ performance measure for dMOPSO, SDMOPSO and MOEA/D

<table>
<thead>
<tr>
<th>MOP</th>
<th>dMOPSO (average $\sigma$)</th>
<th>SDMOPSO (average $\sigma$)</th>
<th>MOEA/D (average $\sigma$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FONSECA</td>
<td>0.041188 (0.000104)</td>
<td>0.044997 (0.000739)</td>
<td>0.003055 (0.000674)</td>
</tr>
<tr>
<td>ZDT1</td>
<td>0.004822 (0.000157)</td>
<td>0.006129 (0.0037486)</td>
<td>0.002690 (0.002690)</td>
</tr>
<tr>
<td>ZDT2</td>
<td>0.004231 (0.000178)</td>
<td>0.006587 (0.002528)</td>
<td>0.008435 (0.002757)</td>
</tr>
<tr>
<td>ZDT3</td>
<td>0.016659 (0.000752)</td>
<td>0.022242 (0.014656)</td>
<td>0.019219 (0.003859)</td>
</tr>
<tr>
<td>ZDT4</td>
<td>0.006004 (0.005617)</td>
<td>0.020826 (0.048561)</td>
<td>0.009254 (0.03852)</td>
</tr>
<tr>
<td>DT LZ52</td>
<td>0.24236 (0.001323)</td>
<td>0.30867 (0.002005)</td>
<td>0.238992 (0.00179)</td>
</tr>
<tr>
<td>DT LZ6</td>
<td>0.038563 (0.000366)</td>
<td>0.381048 (0.194066)</td>
<td>0.036316 (0.046853)</td>
</tr>
<tr>
<td>DT LZ7</td>
<td>0.103966 (0.001991)</td>
<td>0.040463 (0.010550)</td>
<td>0.100617 (0.021034)</td>
</tr>
</tbody>
</table>

Table 4: Results of the $C$ performance measure for dMOPSO, SDMOPSO and MOEA/D

<table>
<thead>
<tr>
<th>MOP</th>
<th>dMOPSO (average $\sigma$)</th>
<th>SDMOPSO (average $\sigma$)</th>
<th>MOEA/D (average $\sigma$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FON</td>
<td>0.220667 (0.015383)</td>
<td>0.23333 (0.016059)</td>
<td>0.026233 (0.018883)</td>
</tr>
<tr>
<td>ZDT1</td>
<td>0.028328 (0.009794)</td>
<td>0.000000 (0.00000)</td>
<td>0.895667 (0.00000)</td>
</tr>
<tr>
<td>ZDT2</td>
<td>0.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
<td>0.496667 (0.00000)</td>
</tr>
<tr>
<td>ZDT3</td>
<td>0.103268 (0.00749)</td>
<td>0.001000 (0.0084559)</td>
<td>0.815567 (0.001913)</td>
</tr>
<tr>
<td>ZDT4</td>
<td>0.256550 (0.42550)</td>
<td>0.000000 (0.015041)</td>
<td>0.229000 (0.00000)</td>
</tr>
<tr>
<td>ZDT6</td>
<td>1.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
</tr>
<tr>
<td>DT LZ52</td>
<td>0.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
</tr>
<tr>
<td>DT LZ6</td>
<td>0.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
</tr>
<tr>
<td>DT LZ7</td>
<td>0.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
<td>0.000000 (0.00000)</td>
</tr>
</tbody>
</table>

4.3 Experimental Settings

For each MOP, 30 independent runs were performed with each algorithm. The parameters used in each algorithm are summarized in Table 1, where $N_{op}$ represents the population size (100 for the bi-objective and 300 for the three-objective problems). $N_{gen}$ represents the number of generations (which was set in 150). Therefore, we performed 15,000 (for the bi-objective problems) and 45,000 (for the three-objective problems) fitness function evaluations. For dMOPSO, $T_a$ represents the age threshold. For SDMOPSO, $N_{rec}$ represents the maximum number of elements in the external archive, $\epsilon$ is the parameter used for $\epsilon$-dominance, $n_r$ is the number of solutions which are replaced in the neighborhood, and $T_n$ defines the neighborhood size. For MOEA/D, $T_n$, $\eta_r$, $\eta_m$, $P_r$ and $P_m$ represent the neighborhood size, crossover index, mutation index, crossover rate and mutation rate, respectively. Finally, the parameter $\theta$, represents the penalty value used in the PBI approach. For dMOPSO and SDMOPSO, the constraints on the velocity ($c_1$, $c_2$) and the inertia factor ($w$) were dynamically defined (as in other
Figure 3: dMOPSO, SDMOPSO and MOEA/D in the bi-objective optimization problems
MOPSOs, see e.g. [17, 15]). Additionally, we used the values proposed by the authors of SDMOPSO [15], which take uniformly distributed values, such that: $c_1, c_2 \in (1.2, 2.0)$ and $w \in (0.1, 0.5)$.

For each MOP, the algorithms were evaluated using the three performance measures previously indicated (Hypervolume, Spacing and Coverage of Two Sets). The results are summarized in Tables 2 to 4. Regarding the hypervolume, the reference vectors adopted were: $r = (1.1, \ldots, 1.1)^T$ for FONSECA, DTLZ2, DTLZ6 and ZDT test problems, while for DTLZ7 the reference vector $r = (1.0, 1.0, 6.1)^T$ was used.

Each table displays both the average and the standard deviation ($\sigma$) of each performance measure, for each of the test problems adopted. The best results are shown in boldface for each performance measure and test problem adopted.

4.4 Discussion of Results

As shown in Table 2, our proposed approach (dMOPSO) outperformed both SDMOPSO and MOEA/D in most of the test problems with respect to the Hypervolume ($H_v$). This indicates that our algorithm produced a better approximation along the Pareto front. The exception was DTLZ2, where MOEA/D obtained better results. However, the differences in the values obtained by our approach are not significant.

Regarding the Spacing ($Sp$) performance measure, our proposed dMOPSO obtained better results with respect to those produced by SDMOPSO and MOEA/D in most test problems. The exceptions were ZDT2, DTLZ7 and FONSECA. However, a better distribution of solutions is relevant only when there is a good approximation to the Pareto front. For these specific problems (FONSECA, DTLZ2 and DTLZ7), our proposed approach achieved better convergence than both SDMOPSO and MOEA/D.

Finally, according to the Coverage of Two Sets ($C$) performance measure, dMOPSO outperformed the results obtained by SDMOPSO in all the test problems adopted. That means that our proposed approach obtained more solutions that dominate those generated by SDMOPSO. However, for the FONSECA, DTLZ2 and DTLZ7 test problems MOEA/D obtained the better results. However, in the case of FONSECA, it can be clearly seen in Fig. 3 that MOEA/D only covered a portion of the true Pareto front, whereas dMOPSO covered it entirely. So, in this case, MOEA/D generated more nondominated solutions than dMOPSO, but they were not properly distributed. Something similar happens in DTLZ7, being DTLZ2 the only case in which it is clear that MOEA/D outperformed our dMOPSO, although the results (both numerical and graphical values) regarding convergence and spread are not significantly different.

Figures 3 and 4 show the final Pareto fronts obtained by dMOPSO, SDMOPSO and MOEA/D algorithms in the adopted test problems. These plots show the final set of nondominated solutions found by each algorithm and correspond to the run with the value nearest to average of the $H_v$ metric for each problem.
5. CONCLUSIONS AND FUTURE WORK

We have presented a multi-objective particle swarm optimizer based on decomposition. Our proposed algorithm was able to outperform both MOEA/D and SDMOPSO in most of the test problems adopted, with respect to three performance measures (hypervolume, spacing and coverage of two sets). Our proposed approach does not use an external archive to store nondominated solutions and adopts instead, a mechanism to select the globally best solutions.

The use of PBI greatly benefits the diversity of solutions in the minimization process of our approach. The penalty parameter $\theta$ imposes a search direction defined by each weighted vector. However, a study of the variation of this parameter along the flight circuits is a promising path for future research. Evidently, it could relax the search process, and could lead us to achieve faster convergence towards the Pareto optimal set.

As part of our future work, we plan to exploit the properties of the decomposition approach when coupling it to other metaheuristics (e.g., scatter search [9] or artificial immune systems [4]) in order to design a more robust strategy which can effectively deal with more complicated multi-objective problems.

Acknowledgments

The first author acknowledges support from CONACyT project no. 103570.

6. REFERENCES