Universal Swarm Optimizer for Multi-Objective Functions

Luis A. Márquez-Vega and Luis M. Torres-Treviño
Universidad Autónoma de Nuevo León,
Facultad de Ingeniería Mecánica y Eléctrica,
Pedro de Alba s/n, C.P. 66455, San Nicolás de los Garza, Nuevo León, México.
luis.marquezvg@gmail.com
luis.torres.ciidit@gmail.com

Abstract. This paper presents the Universal Swarm Optimizer for Multi-Objective Functions (USO), which is inspired in the zone-based model proposed by Iain D. Couzin that represents in a more realistic way the behavior of biological species as fish schools and bird flocks. The algorithm is validated using 10 multi-objective benchmark problems and a comparison with the Multi-Objective Particle Swarm Optimization (MOPSO) is presented. The obtained results suggest that the proposed algorithm is very competitive and presents interesting characteristics which could be used to solve a wide range of optimization problems.

Keywords: Multi-Objective Optimization · Zone-based model · Swarm Intelligence.

1 Introduction

One of the objectives in engineering is to solve real-life problems, these problems have the particularity of being multi-objective, this means that a solution for the optimization of two or more objectives at the same time must be found, the difficulty of this type of problems lies in the fact that the optimization of the objectives is not closed to a single solution, as with single-objective problems, in this case may be a set of solutions that optimize the objectives. Currently there are many algorithms developed in order to give solution of this type of problems [5], among which the best known are Strenght-Pareto Evolutionary Algorithm (SPEA), Pareto Archived Evolution Strategy (PAES), Pareto-frontier Differential Evolution (PDE), Non-dominated Sorting Genetic Algorithm version 2 (NSGA-II), Multi-Objective Particle Swarm Optimization (MOPSO), Multi-Objective Evolutionary Algorithm based on Desomposition (MOEA/D).

In the literature is observed that all of the algorithms previously mentioned adequately optimize multi-objective problems, but the theorem called No Free Lunch proposed by Wolpert and Macready [6] in 1997 proves that there is no optimization algorithm which give solution to all problems efficiently. Starting from this point is that the algorithm presented in this paper is proposed.
This algorithm is inspired in a model proposed by Iain D. Couzin [2] which simulate the behavior of individuals based on three main characteristics of biological species; repulsion, alignment and attractive tendencies towards other individuals based upon the position and orientation of individuals relative to one another, with this model it is possible to configure the formation and the behavior of a swarm in order to solve a wide range of optimization problems.

The main idea of the proposed algorithm is to represent any swarm behavior in order to find, in a same algorithm, very different ways of giving an efficient solution to any optimization problem that may arise.

2 Methodology

In this section the basic concepts of Multi-Objective Optimization, the Multi-Objective Particle Swarm Optimization (MOPSO), the Universal Swarm Optimization for Multi-Objective functions (USO) and performance metrics are described.

2.1 Basic concepts

As mentioned before, the optimal solution of a Multi-Objective problem is not closed to a single value, in this type of problems, the solutions can not be compared by relational operators. In this case, to decide if a solution is better than another is necessary determine if one solution dominates the other, this means that all the objective values \( F(x) \) are better or equal than the objective values of other solution \( F(y) \), this is called Pareto dominance. The following definitions mentioned in [5] are formulated as a minimization problem:

**Definition 1. Pareto Dominance:**
Suppose that \( x, y \in \mathbb{R}^n \), is said that \( x \succ y \) (\( x \) dominates \( y \)) iff:
\[
\exists x, y \in \mathbb{R}^n \mid F(x) \leq F(y)
\] (1)

The definition of Pareto optimality is as follows:

**Definition 2. Pareto Optimality:**
A solution \( x_i^* \) is called a non-dominated solution iff:
\[
\exists x_i^* \mid x_i^* \succ x_j \forall x_i, x_j \in \mathbb{R}^n \ i \neq j
\] (2)

A set including all the non-dominated solutions of a problem is called Solution Space (\( SS \)) and it is defined as follows:

**Definition 3. Solution space:**
The set of all Pareto-optimal solutions is called Solution Space as follows:
\[
SS = \{x_1^*, x_2^*, ..., x_D^*\}
\] (3)

The Pareto front \( P_f \) is a set which contains the corresponding objectives values of Solution Space. The definition of the Pareto front is as follows:

**Definition 4. Pareto front:**
The set which contains the objective values of the solution space:
\[
P_f := \{F(x_i^*) \mid x_i^* \in SS\}
\] (4)
2.2 Multiple-Objective Particle Swarm Optimization (MOPSO)

The Multi-Objective Particle Swarm Optimization (MOPSO) was presented by Carlos C. Coello as an extension of the Particle Swarm Optimization (PSO) in order to solve Multi-Objective problems. The algorithm presented in [1] is briefly explained as follows:

**Main Algorithm** The algorithm of MOPSO is the following:

The position $X_i(t)$ and velocity $V_i(t)$ of each particle is initialized and evaluated, the non-dominated particles are saved in the repository $Rep$. Then the particles are located in the search space by generating hypercubes and the actual position is saved as the best position of the particle in $Ib_i$.

In every iteration the following must be done:

- The speed of the particle is computed using the following expression:

$$V_i(t+1) = w \cdot V_i(t) + \beta_1 \cdot u() \cdot (Ib_i - X_i(t)) + \beta_2 \cdot u() \cdot (Rep(h) - X_i(t))$$

  where $w$ is the inertia weight, $\beta_1$ is the personal learning coefficient, $\beta_2$ is the global learning coefficient, $u()$ is a random number between $[0,1]$ and $Rep(h)$ is the position of the leader taken from the repository, the index $h$ is a randomly particle from a hypercube which was selected by Roulette Wheel Selection, hypercubes with more particles has a lower fitness value.

- The new position of the particle is obtained by:

$$X_i(t+1) = X_i(t) + V_i(t+1)$$

Every new solution is evaluated in order to update the repository $Rep$ and the best position of each particle.

**External Repository** The external repository (or archive) has the objective of store the non-dominated particles along the iterations. The archive controller has the function of decide which particles maintain in the archive, mantaining only non-dominated particles. The adaptive grid is used to well distribute the non-dominated particles in the search space and is useful to decide which particle delete when the archive is full.

**Use of a Mutation Operator** A mutation operator is necessary in the MOPSO due to the very high convergence speed of the PSO, if the algorithm has very high convergence speed may converge to a false Pareto front.

2.3 Universal Swarm Optimizer for Multi-Objective functions (USO)

As said before, the Universal Swarm Optimizer for Multi-Objective functions (USO) is inspired in the zone-based model, which is very useful because provide control on the formation and behavior of swarms. The mathematical formulation of the model is explained in [4] and is as follows:
Zone-based model  The mathematical formulation of the model is based on give to each particle the notion of repulsion, orientation and attraction towards other particles in the swarm.

The swarm is composed by \( N \) particles with positions \( X_i(t) \) and unit directions \( \hat{v}_i(t) \). Each particle travel through the search space at constant speed, and every iteration the particles determine new desired direction of travel by detecting particles within two zones. The first zone is dominated “zone of repulsion”, is represented by a circle of radius \( \rho_r \), in the center of this circle lies the particle \( i \). The particle repel away from others within their zone of repulsion. The second zone is dominated “zone of orientation and attraction”, is represented by an annulus of inner radius \( \rho_r \) and outer radius \( \rho_p \), this zone has a blind area behind the particle, for which others particles are undetectable. The particles align with and are attracted towards particles within their zone of orientation and attraction.

For a given particle \( i \), the set of particles in the zone of repulsion is represented by \( Z^R_i \), and the set of particles in the zone of orientation and attraction is represented by \( Z^P_i \).

If a particle \( i \) finds others particles within its zone of repulsion, its desired direction of travel is given by the following equation:

\[
v_i(t + 1) = - \sum_{j \in Z^R_i} \frac{X_j(t) - X_i(t)}{|X_j(t) - X_i(t)|}
\]  

If in the zone of repulsion the particle \( i \) does not find other particles, then it aligns with and feels an attraction towards others particles that are in its zone of orientation and attraction. In this case, its desired direction of travel is given by the following equation:

\[
v_i(t + 1) = \omega_a \frac{a_i(t)}{|a_i(t)|} + \omega_o \frac{o_i(t)}{|o_i(t)|}
\]  

where \( \omega_a \) is the weight of attraction and \( \omega_o \) is the weight of orientation, \( a_i(t) \) and \( o_i(t) \) are given in the equations (9) and (10) respectively:

\[
a_i(t) = \sum_{j \in Z^P_i} \frac{X_j(t) - X_i(t)}{|X_j(t) - X_i(t)|}
\]  

\[
o_i(t) = \sum_{j \in Z^P_i} \hat{v}_j(t)
\]  

The desired direction of travel of a particle \( i \) must be normalized as the following equation:

\[
\hat{v}_i(t + 1) = \frac{v_i(t + 1)}{|v_i(t + 1)|}
\]  

If \( v_i(t + 1) = 0 \), then the particle \( i \) maintains its previous direction of travel as its desired direction of travel.
Finally, each particle’s position is updated as the following expression:

\[ X_i(t+1) = X_i(t) + s \cdot \hat{v}_i(t+1) \tag{12} \]

where \( s \) is a constant value of speed of travel.

With the previous formulations is clearly that, the formation and the behavior of the swarm can be easily manipulated by changing the radius of the zone of repulsion \( \rho_r \) and the outer radius of the zone of orientation and attraction \( \rho_p \), as well, manipulating the weights of attraction \( \omega_a \) and orientation \( \omega_o \) different behaviors of the swarm are obtained, because the behavior of each particle is modified.

**Main Algorithm** The algorithm of USO uses the previous mathematical formulations and also uses the archive controller and the adaptive grid proposed by Carlos C. Coello [1] in order to solve multi-objective problems. The main algorithm of USO is as follows:

1. Initialize the position \( X_i(t) \) and direction \( v_i(t) \) of each particle;
2. Evaluate \( X(t) \) in order to obtain its objective values;
3. Add non-dominated particles in the repository \( \text{Rep} \);
4. Use objective values of the particles as coordinates in the search space;
5. while cycle \( \leq \) TotalCycles do
   1. for \( i = 1, \text{TotalParticles} \) do
      1. if Particle finds other particles in its zone of repulsion then
          1. Calculate desired direction as the equation (7);
      2. else if Particle finds other particles in its zone of attraction then
          1. Calculate desired direction as the equation (8);
      3. else
          1. Actual direction becomes its desired direction
      end
      1. The desired direction is normalized as equation (11);
      1. The new position the particle is obtained using the equation (13);
      1. Maintain the new position of the particle inside the search space;
      1. Evaluate the new position of the particle;
   end
   1. Verify non-dominated particles and update them in \( \text{Rep} \);
end

\[ X_i(t+1) = X_i(t) + s \cdot \hat{v}_i(t+1) + \beta_2 \cdot u() \cdot (\text{Rep}(h) - X_i(t)) \tag{13} \]

where \( s \) is the constant speed of travel, \( \beta_2 \) is the global learning coefficient, \( u() \) is a random number between \([0,1]\) and \( \text{Rep}(h) \) is the position of the leader taken from the repository, the index \( h \) is selected as proposed in MOPSO.

Note that, another term was added in the equation (13) with respect to the equation (12), this is done with the purpose of giving the ability to the swarm to follow a leader and find better solutions in the search space.
2.4 Performance metrics

For measure the performance of the algorithms, the following metrics has been selected:

In order to measure the convergence the Inverted Generational Distance (IGD) was selected and for measure the coverage theSpacing (SP) and Maximum Spread (MS) were selected.

The expression of IGD is in [5] as follows:

$$IGD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n}$$  \hspace{1cm} (14)

where $n$ is the number of true Pareto optimal solutions and $d_i$ indicates the Euclidean distance between the $i$-th true Pareto optimal solution and its closest non-dominated solution obtained. The lower values of the IGD are preferable.

The corresponding expressions of SP and MS are in [3] as the equations (15) and (17) respectively:

$$SP = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (\bar{d} - d_i)^2}$$ \hspace{1cm} (15)

$$d_i = \min_{j \in NDS \land j \neq i} \sum_{k=1}^{K} \left| f^i_k - f^j_k \right|$$ \hspace{1cm} (16)

where $d_i$ is shown in the equation (16), NDS is the set of non-dominated solutions obtained, $\bar{d}$ is the mean of all $d_i$, $n$ is the size of the NDS and $f^i_k$ is the function value of the $k$-th objective function for solution $i$. The lower values of the SP are preferable.

$$MS = \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left[ \max_{i \in NDS} f^i_k - \min_{i \in NDS} f^i_k \right]^2}$$ \hspace{1cm} (17)

where NDS is the set of non-dominated solutions obtained, $K$ is the number of objectives, $f^i_k$ is the function value of the $k$-th objective function for solution $i$, $F^i_{k\text{max}}$ and $F^i_{k\text{min}}$ are the maximum and minimum value of the $k$-th objective in the true Pareto front respectively. The values of the MS closer to 1 are preferable.

The best run was taken for the qualitative results and the computational time are also obtained in each unconstrained problem.

3 Experimental setup

For the experiments is used as a benchmark 10 unconstrained problems proposed in CEC 2009 [7], because as said in [5] these test problems are considered as the most challenging test problems in the literature. Table 1 shows the mathematical formulation of each unconstrained problem:
Table 1. Mathematical formulation of test problems

<table>
<thead>
<tr>
<th>Name</th>
<th>Mathematical formulation</th>
</tr>
</thead>
</table>
| UP1  | \( f_1 = x_1 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} |x_j - \sin(6\pi x_j + \frac{\pi}{6})|^2 \)  
   | \( f_2 = 1 - \frac{1}{\sqrt{n}} \sum_{j=1}^{n} |x_j - \sin(6\pi x_j + \frac{\pi}{6})|^2 \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \) |
| UP2  | \( f_1 = x_1 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} y_j^2 \)  
   | \( f_2 = 1 - \frac{1}{\sqrt{n}} \sum_{j=1}^{n} y_j^2 \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - 0.3 \cos(2\pi x_j + \frac{4\pi}{3}) + 0.6 \sin(6\pi x_j + \frac{2\pi}{3}) \)  
   | \( y_j = x_j - 0.3 \cos(2\pi x_j + \frac{4\pi}{3}) + 0.6 \sin(6\pi x_j + \frac{2\pi}{3}) \) |
| UP3  | \( f_1 = x_1 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} y_j^2 \)  
   | \( f_2 = 1 - \frac{1}{\sqrt{n}} \sum_{j=1}^{n} y_j^2 \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - 0.3 \cos(2\pi x_j + \frac{4\pi}{3}) + 0.6 \sin(6\pi x_j + \frac{2\pi}{3}) \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - 0.3 \cos(2\pi x_j + \frac{4\pi}{3}) + 0.6 \sin(6\pi x_j + \frac{2\pi}{3}) \) |
| UP4  | \( f_1 = x_1 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} h(y_j) \)  
   | \( f_2 = 1 - x_1 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} h(y_j) \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - \sin(6\pi x_j + \frac{\pi}{6}) \)  
   | \( h(y_j) = \frac{1}{\sin(2\pi x_j + \frac{\pi}{6})} \) |
| UP5  | \( f_1 = x_1 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} h(y_j) \)  
   | \( f_2 = 1 - x_1 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} h(y_j) \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - \sin(6\pi x_j + \frac{\pi}{6}) \)  
   | \( h(y_j) = \frac{1}{\sin(2\pi x_j + \frac{\pi}{6})} \) |
| UP6  | \( f_1 = x_1 + \max\{0, 2(\frac{n}{\sqrt{n}} + c)\} \)  
   | \( f_2 = 1 - x_1 + \max\{0, 2(\frac{n}{\sqrt{n}} + c)\} \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - \sin(6\pi x_j + \frac{\pi}{6}) \)  
   | \( h(j) = 2^j - \cos(4\pi j) + 1 \) |
| UP7  | \( f_1 = \frac{x_1}{\sqrt{n}} \sum_{j=1}^{n} x_j \)  
   | \( f_2 = 1 - \frac{x_1}{\sqrt{n}} \sum_{j=1}^{n} x_j \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - \sin(6\pi x_j + \frac{\pi}{6}) \)  
   | \( h(j) = 2^j - \cos(4\pi j) + 1 \) |
| UP8  | \( f_1 = \cos(0.5\pi x_1) \cos(0.5\pi x_2) \)  
   | \( f_2 = \cos(0.5\pi x_1) \sin(0.5\pi x_2) \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - \sin(6\pi x_j + \frac{\pi}{6}) \)  
   | \( h(j) = 2^j - \cos(4\pi j) + 1 \) |
| UP9  | \( f_1 = 0.5 \max\{0, 1 + c\}(1 - 4(2x_1 - 1)^2) + 2x_2 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} (x_j - 2x_2 \sin(2\pi x_j + \frac{\pi}{6}))^2 \)  
   | \( f_2 = 0.5 \max\{0, 1 + c\}(1 - 4(2x_1 - 1)^2) + 2x_2 + \frac{1}{\sqrt{n}} \sum_{j=1}^{n} (x_j - 2x_2 \sin(2\pi x_j + \frac{\pi}{6}))^2 \)  
   | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
   | \( y_j = x_j - \sin(6\pi x_j + \frac{\pi}{6}) \)  
   | \( h(j) = 2^j - \cos(4\pi j) + 1 \) |
| UP10 | \( f_1 = \cos(0.5\pi x_1) \cos(0.5\pi x_2) \)  
      | \( f_2 = \cos(0.5\pi x_1) \sin(0.5\pi x_2) \)  
      | \( J_1 = \{ j | j \text{ is odd and } 2 \leq j \leq n \} \) and \( J_2 = \{ j | j \text{ is even and } 2 \leq j \leq n \} \)  
      | \( y_j = x_j - \sin(6\pi x_j + \frac{\pi}{6}) \)  
      | \( h(j) = 2^j - \cos(4\pi j) + 1 \) |
The following considerations were taken in the test of each unconstrained problem:

- Total of particles: 100.
- Dimension of every particle: 10.
- Size of the repository: 100.
- Total of iterations: 250.
- Total of runs: 10.

The following parameters for MOPSO are chosen:

- \( w = 0.5 \): Inertial weight.
- \( \beta_1 = 2 \): Personal learning coefficient.
- \( \beta_2 = 2 \): Global learning coefficient.
- \( n\text{Grid} = 10 \): Number of grids per dimension.
- \( \alpha = 0.1 \): Inflation rate.
- \( \beta = 2 \): Leader selection pressure.
- \( \gamma = 2 \): Deletion selection pressure.
- \( \mu = 0.1 \): Mutation rate.

The following parameters for USO are chosen:

- \( s = 0.5 \): Speed.
- \( \beta_2 = 2 \): Global learning coefficient.
- \( n\text{Grid} = 10 \): Number of grids per dimension.
- \( \alpha = 0.1 \): Inflation rate.
- \( \beta = 2 \): Leader selection pressure.
- \( \gamma = 2 \): Deletion selection pressure.

Despite the previous parameters, the values for the radius of the zone of repulsion \( \rho_r \), the outer radius of the zone of orientation and attraction \( \rho_p \) and the weights of attraction \( \omega_a \) and orientation \( \omega_o \) were separated in three different sets, in order to observe which formation of the swarm produces better performance in each unconstrained problem. Table 2 shows the different sets of these parameters that were proposed.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Balanced swarm</th>
<th>Joined swarm</th>
<th>Dispersed swarm</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho_r )</td>
<td>0.2125</td>
<td>0.01</td>
<td>0.3000</td>
</tr>
<tr>
<td>( \rho_p )</td>
<td>0.3250</td>
<td>0.75</td>
<td>0.3005</td>
</tr>
<tr>
<td>( \omega_a )</td>
<td>0.4750</td>
<td>0.85</td>
<td>0.5000</td>
</tr>
<tr>
<td>( \omega_o )</td>
<td>0.5250</td>
<td>0.15</td>
<td>0.5000</td>
</tr>
</tbody>
</table>

These sets of parameters produces different behaviors on the swarms and report better results in different unconstrained problems, as follows:
– Balanced swarm: UP1, UP2 and UP3.
– Joined swarm: UP4, UP7, UP8, UP9 and UP10.

For example, in the joined swarm the size of the zone of repulsion is very small and the size of the zone of orientation and attraction is very big, and the weight of attraction has more relevance than the weight of orientation. In the dispersed swarm the zone of repulsion is much bigger than the zone of orientation and attraction, and the weights of attraction and orientation are the same.

4 Results

The qualitative results of the experiments are presented in the Figure (1) and in the Tables 3 and 4 the statistical results of the performance metrics are presented.
Table 3. Results of performance metrics on UP1 to UP5

<table>
<thead>
<tr>
<th>Problem</th>
<th></th>
<th>Unconstrained Problem 1</th>
<th></th>
<th>Unconstrained Problem 2</th>
<th></th>
<th>Unconstrained Problem 3</th>
<th></th>
<th>Unconstrained Problem 4</th>
<th></th>
<th>Unconstrained Problem 5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>USO</td>
<td>IGD</td>
<td>SP</td>
<td>MS</td>
<td>IGD</td>
<td>SP</td>
<td>MS</td>
<td>IGD</td>
<td>SP</td>
<td>MS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Metric</td>
<td></td>
<td>allest</td>
<td>SP</td>
<td>MS</td>
<td>IGD</td>
<td>SP</td>
<td>MS</td>
<td>IGD</td>
<td>SP</td>
<td>MS</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Median</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Worst</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Best</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: The table presents the results of performance metrics on various unconstrained problems using different algorithms. The metrics include IGD, SP, MS, and time. The values are averaged across multiple runs, with additional statistics such as median, standard deviation, and best and worst cases.
Table 4. Results of performance metrics on UP6 to UP10

<table>
<thead>
<tr>
<th>Problem</th>
<th>Unconstrained Problem 6</th>
<th>Unconstrained Problem 7</th>
<th>Unconstrained Problem 8</th>
<th>Unconstrained Problem 9</th>
<th>Unconstrained Problem 10</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Algorithm</strong></td>
<td><strong>USO</strong></td>
<td><strong>MOPSO</strong></td>
<td><strong>USO</strong></td>
<td><strong>MOPSO</strong></td>
<td><strong>USO</strong></td>
</tr>
<tr>
<td><strong>Metric</strong></td>
<td><strong>IGD</strong></td>
<td><strong>SP</strong></td>
<td><strong>MS</strong></td>
<td><strong>Time</strong></td>
<td><strong>IGD</strong></td>
</tr>
<tr>
<td><strong>Problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>0.043753</td>
<td>0.084577</td>
<td>1.397498</td>
<td>0.566111</td>
<td>0.042290</td>
</tr>
<tr>
<td><strong>Median</strong></td>
<td>0.044915</td>
<td>0.062612</td>
<td>1.310586</td>
<td>0.565475</td>
<td>0.041177</td>
</tr>
<tr>
<td><strong>Std.Dev.</strong></td>
<td>0.006680</td>
<td>0.072269</td>
<td>0.679772</td>
<td>0.022192</td>
<td>0.003311</td>
</tr>
<tr>
<td><strong>Worst</strong></td>
<td>0.054447</td>
<td>0.290043</td>
<td>2.968560</td>
<td>0.600620</td>
<td>0.047360</td>
</tr>
<tr>
<td><strong>Best</strong></td>
<td>0.031073</td>
<td>0.030365</td>
<td>1.018302</td>
<td>0.529400</td>
<td>0.037744</td>
</tr>
<tr>
<td><strong>Problem</strong></td>
<td>Unconstrained Problem 7</td>
<td>Unconstrained Problem 8</td>
<td>Unconstrained Problem 9</td>
<td>Unconstrained Problem 10</td>
<td></td>
</tr>
<tr>
<td><strong>Algorithm</strong></td>
<td><strong>USO</strong></td>
<td><strong>MOPSO</strong></td>
<td><strong>USO</strong></td>
<td><strong>MOPSO</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Metric</strong></td>
<td><strong>IGD</strong></td>
<td><strong>SP</strong></td>
<td><strong>MS</strong></td>
<td><strong>Time</strong></td>
<td><strong>IGD</strong></td>
</tr>
<tr>
<td><strong>Problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>0.003922</td>
<td>0.015096</td>
<td>1.184840</td>
<td>0.850778</td>
<td>0.004228</td>
</tr>
<tr>
<td><strong>Median</strong></td>
<td>0.003781</td>
<td>0.014109</td>
<td>1.129905</td>
<td>0.834725</td>
<td>0.004038</td>
</tr>
<tr>
<td><strong>Std.Dev.</strong></td>
<td>0.001045</td>
<td>0.004521</td>
<td>0.127901</td>
<td>0.045985</td>
<td>0.000786</td>
</tr>
<tr>
<td><strong>Worst</strong></td>
<td>0.005895</td>
<td>0.023888</td>
<td>1.408987</td>
<td>0.924439</td>
<td>0.005553</td>
</tr>
<tr>
<td><strong>Best</strong></td>
<td>0.002624</td>
<td>0.009003</td>
<td>1.052942</td>
<td>0.793543</td>
<td>0.002934</td>
</tr>
<tr>
<td><strong>Problem</strong></td>
<td>Unconstrained Problem 8</td>
<td>Unconstrained Problem 9</td>
<td>Unconstrained Problem 10</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Algorithm</strong></td>
<td><strong>USO</strong></td>
<td><strong>MOPSO</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Metric</strong></td>
<td><strong>IGD</strong></td>
<td><strong>SP</strong></td>
<td><strong>MS</strong></td>
<td><strong>Time</strong></td>
<td><strong>IGD</strong></td>
</tr>
<tr>
<td><strong>Problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>0.001376</td>
<td>0.643688</td>
<td>2.857913</td>
<td>0.816707</td>
<td>0.001403</td>
</tr>
<tr>
<td><strong>Median</strong></td>
<td>0.001327</td>
<td>0.653234</td>
<td>2.796426</td>
<td>0.821308</td>
<td>0.001355</td>
</tr>
<tr>
<td><strong>Std.Dev.</strong></td>
<td>0.000213</td>
<td>0.004521</td>
<td>0.127901</td>
<td>0.045985</td>
<td>0.000786</td>
</tr>
<tr>
<td><strong>Worst</strong></td>
<td>0.005895</td>
<td>0.023888</td>
<td>1.408987</td>
<td>0.924439</td>
<td>0.005553</td>
</tr>
<tr>
<td><strong>Best</strong></td>
<td>0.002624</td>
<td>0.009003</td>
<td>1.052942</td>
<td>0.793543</td>
<td>0.002934</td>
</tr>
<tr>
<td><strong>Problem</strong></td>
<td>Unconstrained Problem 9</td>
<td>Unconstrained Problem 10</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Algorithm</strong></td>
<td><strong>USO</strong></td>
<td><strong>MOPSO</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Metric</strong></td>
<td><strong>IGD</strong></td>
<td><strong>SP</strong></td>
<td><strong>MS</strong></td>
<td><strong>Time</strong></td>
<td><strong>IGD</strong></td>
</tr>
<tr>
<td><strong>Problem</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>0.000338</td>
<td>0.780253</td>
<td>3.347833</td>
<td>0.795985</td>
<td>0.000383</td>
</tr>
<tr>
<td><strong>Median</strong></td>
<td>0.000356</td>
<td>0.910155</td>
<td>3.339535</td>
<td>0.792619</td>
<td>0.000360</td>
</tr>
<tr>
<td><strong>Std.Dev.</strong></td>
<td>0.000098</td>
<td>0.306858</td>
<td>0.831101</td>
<td>0.005004</td>
<td>0.000157</td>
</tr>
<tr>
<td><strong>Worst</strong></td>
<td>0.000499</td>
<td>1.247666</td>
<td>4.321281</td>
<td>0.801297</td>
<td>0.000620</td>
</tr>
<tr>
<td><strong>Best</strong></td>
<td>0.000170</td>
<td>0.342260</td>
<td>1.761628</td>
<td>0.782678</td>
<td>0.000058</td>
</tr>
</tbody>
</table>
5 Conclusion

As can be seen in the qualitative and statistical results, the proposed algorithm represents very good competence against the MOPSO algorithm. In the most problems the USO algorithm produces in general better statistical results than the MOPSO. Only in UP4 and UP5, the proposed algorithm presents worse results in the average of IGD and MS at the same time, despite this USO presents the best found result of IGD and MS in UP5. One of the biggest challenges of this algorithm is the computational time required, which is much bigger than the computational time of MOPSO and this reduces relevance to the good results obtained in the others statistical performance metrics.

The strongest feature of this algorithm is the easy design of swarms with desired qualities, others characteristics must be studied more deeply, for example the optimal number of particles in the swarm because bigger swarms produce crowded places in which the vision of the particle is reduced.

More studies about this algorithm must be done in order to find more characteristics which help to improve the performance of the algorithm and reduce the computational time required. Experiments with more dimension per individual and more comparisons with others multi-objective algorithms must be done in order to have a wider view of the real scope of this algorithm.

References