A Multi-Objective Particle Swarm Optimizer Based on Diversity

Dennis R. C. Silva and Carmelo J. A. Bastos-Filho
University of Pernambuco
Recife, Brazil
Email: carmelofilho@ieee.org

***Abstract*—This paper presents a novel multi-objective optimization algorithm based on Particle Swarm Optimization (MOPSO-DFR), which uses a global density estimator mechanism called Diversity Factor (DF) to select the cognitive and the social leaders. MOPSO-DFR also uses DF to update and to prune the external archive, whenever it is necessary. We used well known metrics to evaluate the results generated by our proposal in seven widely used benchmark functions. We also compared our approach to other four multi-objective optimization algorithms called MOPSO-CDR, SMPSO, NSGA-II and SPEA-2. The results showed that MOPSO-DFR outperforms the other approaches in most cases.**

Keywords*—*swarm intelligence; particle swarm optimization; multi-objective optimization

I. Introduction

Multi-objective optimization refers to the simultaneous optimization of two or more conflicting objective functions. For Multi-Objective Optimization Problems (MOPs), it is expected to obtain a set of trade-off solutions in a single run of an optimization algorithm. Besides, MOP with multiple decision variables are often difficult to tackle. Because of this, many approaches have been recently proposed for solving MOP in a faster and more efficient way. Meta-heuristics have been successfully applied to solve MOPs in the last years and most of the recent interesting approaches are based on evolutionary computation or swarm intelligence.

Particle Swarm Optimization (PSO) is one of the most used swarm intelligence algorithms. PSO was proposed by Kennedy and Eberhart in 1995 [1] and it was inspired by the behavior of flocks of birds. In general, PSO is used for solving single objective optimization problems in hyper-dimensional spaces with continuous variables. Due to the simplicity and fast convergence, some approaches based on PSO have been proposed to tackle MOPs. The first Multi-Objective Particle Swarm Optimizer (MOPSO) was proposed in 2002 by Coello Coello *et al* [2]. Since then, many other MOPSO approaches have been proposed. All of them propose to change the policy to select the cognitive and social leaders, that are used to update the velocity of the particles, and/or to define a criterion to update the External Archive (*EA*), which is used to store the non-dominated solutions obtained along the search process. Santana *et. al* [3] presented the MOPSO-CDR and successfully showed that one can use *Crowding distance (CD)* to select the leaders and to update the *EA*. They also showed that MOPSO-CDR outperforms some previous approaches, such as *m*-DNPSO [4] and CSS-MOPSO [5]. Nebro *et. al* [6] also proposed an interesting speed-constrained approach which is also useful for many-objective optimization.

Although there are many proposals presented in the literature, the policies to select the leaders and to update the *EA* are based on measures that evaluate local features within the *EA*, such as *CD* [3]. Recently, Zhan *et al* [7] proposed a global measure to assess the diversity of the whole swarm. We propose here to use this global measure, renamed in this paper as *Diversity factor (DF)*, to evaluate the diversity of the solutions within the EA in order to properly select the leaders and to update the *EA*.

This paper is organized as follow. In Section II, we present some basic concepts on Particle Swarm Optimization and Multi-Objective Optimization. In Section III, we briefly describe some approaches for tackling MOPs. We present our novel MOPSO approach based on the *DF* in Section IV. In Sections V, we present the simulation setup and some results in well known benchmark functions, including a comparison with MOPSO-CDR, SMPSO and two widely used multi-objective evolutionary computation optimizers (NSGA-II and SPEA-2). In Section VI, we give our conclusions and present some future works.

II. Basic Concepts

In this section we present basic concepts regarding Particle Swarm Optimization and Multi-objective optimization.

A. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a stochastic, bio-inspired, population-based global optimization technique [1]. The population is called swarm and the individuals are called particles. Each particle moves within the search space with an adaptive velocity looking for promising regions. Each particle i has four main attributes: the current position in the d-dimensional space $\vec{x}_i = (x_{i1}, x_{i2}, ..., x_{id})$, the best position found so far in the search process $\vec{p}_i = (p_{i1}, p_{i2}, ..., p_{id})$, the best position found by its neighborhood so far $\vec{n}_i = (n_{i1}, n_{i2}, ..., n_{id})$ and the velocity $\vec{v}_i = (v_{i1}, v_{i2}, ..., v_{id})$. The velocity and the position of every particle are updated iteratively according to the following equations:

$$\vec{v_i}(t+1) = \vec{v_i}(t) + c_1 \cdot r_1 \cdot (\vec{p}_{best} - \vec{x_i}) + c_2 \cdot r_2 \cdot (\vec{n}_{best} - \vec{x_i}), \quad (1)$$

$$\vec{x_i}(t+1) = \vec{x_i}(t) + \vec{v_i}(t+1), \quad (2)$$

 ISBN: 978-1-61208-269-1

where i is the label of the particle, c_1 and c_2 are the cognitive and the social acceleration coefficients, respectively. r_1 and r_2 are two random numbers generated by an uniform distribution in the interval $[0, 1]$.

B. Multi-Objective Optimization

A minimization MOP can be stated as:

$$\text{minimize}\ \vec{f}(\vec{x}) := [f_1(\vec{x}), f_2(\vec{x}), ..., f_k(\vec{x})] \tag{3}$$

subject to:

$$g_i(\vec{x}) <= 0 i = 1, 2, ..., m, \tag{4}$$

$$h_i(\vec{x}) = 0 j = 1, 2, ..., p, \tag{5}$$

where $\vec{x} = (x_1, x_2, ..., x_n) \in \mathbb{R}$ is the vector on the decision search space; and $g_i(\vec{x})$ and $h_j(\vec{x})$ are the constraint functions of the problem.

The best solutions that solves a MOP are called non-dominated solutions. The concept of dominance is given by: given two vectors $\vec{x}$ and $\vec{y}$, $\vec{x}$ dominates $\vec{y}$ (denoted by $\vec{x} \prec \vec{y}$) if $\vec{x}$ is better than $\vec{y}$ in at least one objective and $\vec{x}$ is not worse than $\vec{y}$ in any objective. $\vec{x}$ is not dominated if does not exist another solution $\vec{x}_i$ in the current population such that $\vec{x}_i \prec \vec{x}$. The set of non-dominated solutions in the objective space found by a particular algorithm trying to solve a MOP is known as Pareto Front.

In order to measure the quality of the Pareto Front obtained by the algorithms, several metrics have been proposed. The following metrics will be used in this paper.

1) Coverage Set (C): is used in order to evaluate the convergence reached by the algorithm [8]. Equation (6) presents how to calculate C using two different Pareto Fronts A and B:

$$C(A, B) = \frac{|\{b \in B; \exists a \in A : a \succ b\}|}{|B|}, \tag{6}$$

where $|B|$ represents the amount of solutions belonging to the Pareto Front B.

If the value $C(A, B) = 1$, all solutions of B are dominated by the solutions of A. On the other hand, if $C(A, B) = 0$, none of the solutions of B are dominated by A.

2) Spacing (S): is used in order to evaluate the distribution of the non-dominated solutions within the Pareto Front. S is calculated according to Equation (7) [9].

$$S = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (\bar{d} - d_i)^2}, \tag{7}$$

where $d_i = min_j(|f_1^i(\vec{x}) - f_1^j(\vec{x})| + |f_2^i(\vec{x}) - f_2^j(\vec{x})|), i, j = 1, ..., n$, $\bar{d}$ represents the average distance betwen all adjacent solutions and n is the number of non-dominated solutions. $S = 0$ means that all non-dominated are equidistant.

3) Maximum Spread (MS): measures the Euclidian distance between the two farthest solutions within the Pareto Front. MS is calculated using Equation (8) [10]:

$$MS = \sqrt{\sum_{m=1}^{M} (max_{i=1}^{n} f_m^i - min_{i=1}^{n} f_m^i)^2}, \tag{8}$$

where n is the label of the non-dominated solutions and M is the number of objectives of the problem. High values for MS means that the Pareto Front covers a significant area of the objective space.

4) Hypervolume (HV): is defined by the hypervolume in the objective space covered by the Pareto Front [8]. It is calculated by summing the area formed by the union of all hypercubes, where each hypercube is generated by one of the non-dominated solutions and a reference point in the objective space.

III. RELATED WORK

This section aims to present a brief review of some related work.

A. APSO - Adaptive PSO

APSO was proposed by Zhan *et al* [7] in 2009. APSO is a variation of the PSO that self-adapts the acceleration coefficients and the inertia factor depending on the diversity of the swarm at the current iteration. They proposed to define an evolutionary state for the swarm based on Evolutionary Factor. There are four possible states: Convergence, Escape, Exploration and Exploitation. We used the Evolutionary Factor to define the *Diversity Factor (DF)*, which is used in our proposal (MOPSO-DFR).

B. MOPSO-CDR: Multi-objective PSO Using Crowding Distance and Roulette Wheel

MOPSO-CDR was proposed by Santana *et al* in 2009 [3]. In MOPSO-CDR, particles select the social leaders by using a Roulette Wheel based on CD. The strategy used to update the cognitive leader also uses the dominance criterion and CD when the current position and the current social leader are incomparable. A similar mechanism is used to prune the EA. Non-dominated solutions that present higher CD values, *i.e.* particles in less crowded regions, have more chances to be selected as social leaders.

C. SMPSO: Speed-constrained Multi-objective PSO

SMPSO [6] incorporates a constriction mechanism in order to limit the maximum velocity of particles and enhance the search capability of the algorithm. SMPSO also have an EA, but does not use roulette wheel to select the social leaders. The cognitive leader is just updated if it dominates the current position. Moreover, SMPSO uses a speed-constriction approach proposed originally by Clerk and Kennedy [11] and bounds the accumulated velocity [6].

D. NSGA-II - Nondominated Sorting Genetic Algorithm II

NSGA-II is a very well known and widely used evolutionary algorithm for tackling multi-objective problems proposed by Deb *et al* in 2002 [12]. The most important feature of NSGA-II is that it uses a fast non-dominated sorting mechanism and CD for comparing the quality of the solutions. NSGA-II uses dominance ranking to classify the population into a number of layers, since NSGA-II does not use an EA. The truncation of the population for the next generation uses CD to define which individuals are better within the same layer.

E. SPEA-2 - Strength Pareto Evolutionary Algorithm

SPEA-2 is a multi-objective evolutionary algorithm proposed by Zitzler *et al* in 2001 [13]. SPEA2 uses elitism and EA. The non-dominated solutions in the EA are ranked according to a strength rule based on dominance. It presents a good performance since it differentiates the quality of the solution within the EA.

IV. MOPSO-DFR: Multi-objective PSO using Diversity Factor and Roulette Wheel

In this section we introduce our contribution, which aims to obtain a better convergence in MOPs. We name our proposal as Multi-objective Particle Swarm Optimization using Diversity Factor and Roulette Wheel (MOPSO-DFR).

We propose here a diversity estimator, called *Diversity Factor (DF)*. DF is used to select the cognitive and social leaders, and to update and prune the EA as well. DF is based on the Evolutionary Factor used in the APSO algorithm [7]. In order to calculate the DF, one needs to calculate the average distance from the particle i to the other particles within the Pareto Front using Equation (9).

$$d_i = \frac{1}{N-1} \sum_{j=1, j\neq i}^{N} \sqrt{\sum_{k=1}^{M} (x_i^k - x_j^k)^2}, \tag{9}$$

where N is the number of the particles of the Pareto Front and M is the number of objectives of the problem.

After this, one can calculate DF for each particle within the EA by using Equation (10).

$$DF_i = \frac{d_i - d_{min}}{d_{max} - d_{min}}, \tag{10}$$

where DF_i is the DF of the particle i, d_i is the average distance of particle i to the other particles, d_{max} and d_{min} are, respectively, the maximum and minimum average distances. One can observe that DF is a global estimator of diversity within the Pareto Front.

A. Cognitive Leader Selection

The cognitive leader selection process is crucial for the convergence and efficiency of the algorithm. We propose here a similar strategy to the one used in MOPSO-CDR. In our approach we use DF, instead of CD. We update $\vec{P}_{best}$ if the current position of the particle dominates $\vec{P}_{best}$. If they are non-dominated, the selection is performed using the EA. The two nearest solutions from the current position and $\vec{P}_{best}$ are found in the EA. After this, we check which one presents the higher DF. If the nearest solution from the current position has the higher DF value, $\vec{P}_{best}$ is updated to the current position, otherwise $\vec{P}_{best}$ remains.

B. Social Leader Selection

The social leader selection mechanism affects the convergence capability and the distribution of the solutions along the Pareto Front. The MOPSO-DFR algorithm selects the social leader from the EA using a roulette wheel with the DF of each particle as the sorting criterion.

C. External Archive Pruning

In each iteration we include all new non-dominated solutions in the EA and remove the solutions that became dominated. It is common to limit the number of solutions in the EA. In order to avoid to exceed the maximum number of solutions in the EA, we sort the solutions according to the DF value and discard the worst non-dominated.

D. Novel Update Velocity Equation

We proposed to include a fourth term in the equation used in the MOPSO-CDR [3] to update the velocity of the particles. The modified equation is given by:

$$\begin{aligned} \vec{v_i}(t+1) &= \vec{v_i}(t) + c_1 \cdot r_1 \cdot (\vec{p}_{best} - \vec{x_i}) \\ &+ c_2 \cdot r_2 \cdot (\vec{n}_{best_{DF}} - \vec{x_i}) + c_3 \cdot r_3 \cdot (\vec{n}_{best_{CD}} - \vec{x_i}), \end{aligned} \tag{11}$$

where $\vec{n}_{best_{DF}}$ is selected based on DF and $\vec{n}_{best_{CD}}$ is selected based on CD as in [3]. c_3 is the acceleration coefficient associated to this new leader, r_3 is a random number generated in the interval $[0, 1]$.

V. Simulation setup, Results and Discussion

A. Simulation setup

For MOPSO-DFR and MOPSO-CDR, we used a constant mutation rate equal to 0.5 and the inertia factor linearly decreasing from 0.4 to 0. For the SMPSO, the acceleration coefficients are randomly chosen in the interval $[1.5, 2.5]$ and the mutation rate is equal to 0.166. For NSGA-II and SPEA-2, we used the crossover rate and the mutation rate equal to 1.0 and 0.05, respectively. We used 100 particles (or individuals) for all algorithms and the maximum number of non-dominated solutions in the EA equal to 100. We run $300,000$ fitness function evaluations.

We used a widely used set of benchmark functions called DTLZ, which was proposed by Deb *et al* [14] in 2005. We used the following functions: DTLZ-1, DTLZ-2, DTLZ-3, DTLZ-4, DTLZ-5, DTLZ-6 and DTLZ-7.

All tables presented in the paper shows the average value and the (standard deviation) after 30 trials. The best results are bolded to facilitate the visualization.

B. Parametric analysis

Since we are proposing to include a fourth term in the equation used to update the velocity, it is necessary to perform a parametric analysis. Thus, we executed simulations using several different sets of configurations for the acceleration coefficients. These configurations for the acceleration coefficients are shown in Table I. One can observe that we almost do not vary the values of c_1. We decided to not vary it significantly since the major change in the algorithm regards on the social leaders selection. For the values of c_2 and c_3, we considered the combination of fractions of the original value of c_2 used for the MOPSO-CDR.

TABLE I. ACCELERATION COEFFICIENT CONFIGURATIONS ASSESSED FOR THE MOPSO-DFR.

	c_1	c_2	c_3
MOPSO-DFR-A	1,49445	1,49445	1,49445
MOPSO-DFR-B	1,49445	1,49445	0,0
MOPSO-DFR-C	0,9963	0,9963	0,9963
MOPSO-DFR-D	1,49445	0,74722	0,74722
MOPSO-DFR-E	1,49445	0,96966	0,48483
MOPSO-DFR-F	1,49445	0,48483	0,96966
MOPSO-DFR-G	1,49445	1,93932	0,96966
MOPSO-DFR-H	1,49445	0,96966	1,93932

Table II shows the simulation results for all MOPSO-DFR configurations for the DTLZ-1 problem. One can observe that the Coverage for MOPSO-DFR-A outperformed the other configurations (the results are slightly better when compared to MOPSO-DFR-G and MOPSO-DFR-H). This means that the Pareto Front generated by configuration MOPSO-DFR-A dominates the Pareto Fronts obtained by the other configurations.

We also analyzed one case without the CD-based social leader (MOPSO-DFR-B), *i.e.* $c_3 = 0$. MOPSO-DFR-B obtained good results, but MOPSO-DFR-A outperformed it. Therefore, it indicates that it might be useful to use both CD and DF in the equation to update the velocity of the particles.

One can also observe that MOPSO-DFR-F achieved a better S, but MOPSO-DFR-A achieved better results than MOPSO-DFR-F for MS and HV. MOPSO-DFR-G achieved a better HV, but MOPSO-DFR-A achieved better results than MOPSO-DFR-G for S and MS.

We observed for DTLZ-2 and DTLZ-3 functions a similar behavior when compared to DTLZ-1. Thus, we selected the MOPSO-DFR-A configuration for further simulations. From this point, we will call MOPSO-DFR-A as MOPSO-DFR.

C. Comparison with other Multi-objective algorithms

This subsection aims to compare MOPSO-DFR to previous proposed multi-objective optimizers, MOPSO-CDR, SMPSO, NSGA-II and SPEA-2. Table III, IV, V, VI, VII, VIII and IX show the results for DTLZ-1, DTLZ-2, DTLZ-3, DTLZ-4, DTLZ-5, DTLZ-6 and DTLZ-7, respectively, in terms of C, S, MS and HV.

One can observe that our proposal obtained better results in terms of C for most cases. For the DTLZ-3, the MOPSO-CDR achieved slightly better results in terms of coverage when compared to MOPSO-DFR, but one must observe that the difference is small when compared to the standard deviation. For the DTLZ-6, the MOPSO-CDR achieved the best results in terms of coverage when compared to MOPSO-DFR. Again for the DTLZ-6, the NSGA-II achieved slightly better results in terms of coverage when compared to MOPSO-DFR, but one must observe that the difference is small when compared to the standard deviation.

It is also important to notice that the MOPSO-DFR did not achieve the best results for the S, but it achieved the best results for the MS in most of the cases. This indicates that our approach is reaching more extreme solutions and converging for the optimum Pareto simultaneously when compared to the other algorithms.

TABLE II. COMPARISON OF DIFFERENT VERSIONS OF MOPSO-DFR FOR DTLZ-1 WITH 300,000 FITNESS EVALUATIONS.

Alg.	S	MS	HV	C(DFR-A,*)	C(*,DFR-A)
MOPSO DFR-A	7.326 (2.860)	**121.205 (10.559)**	0.411 (0.104)	- -	- -
MOPSO DFR-B	4.153 (4.115)	63.030 (45.028)	0.425 (0.134)	**0.791 (0.104)**	0.070 (0.123)
MOPSO DFR-C	6.314 (5.411)	79.263 (46.465)	0.512 (0.149)	**0.811 (0.266)**	0.092 (0.151)
MOPSO DFR-D	4.351 (4.406)	64.6481 (45.310)	0.556 (0.121)	**0.898 (0.190)**	0.042 (0.091)
MOPSO DFR-E	4.879 (3.675)	75.744 (45.903)	0.596 (0.107)	**0.845 (0.250)**	0.074 (0.150)
MOPSO DFR-F	**4.147 (5.329)**	59.542 (42.085)	0.514 (0.124)	**0.897 (0.213)**	0.045 (0.122)
MOPSO DFR-G	8.852 (4.313)	117.67 (12.874)	**0.341 (0.134)**	**0.406 (0.224)**	**0.321** (0.234)
MOPSO DFR-H	8.828 (4.959)	119.67 (14.010)	0.361 (0.121)	**0.378 (0.266)**	**0.370 (0.247)**

TABLE III. SIMULATION RESULTS FOR DTLZ-1 WITH 300.000 FITNESS FUNCTION EVALUATIONS.

Alg.	S	MS	HV	C(DFR,*)	C(*,DFR)
MOPSO DFR	7,326 (2,859)	**121,205 (10,559)**	**0,411 (0,105)**	- -	- -
MOPSO CDR	6,962 (4,717)	91,745 (35,343)	0,458 (0,139)	**0,753 (0,248)**	0,090 (0,151)
SMPSO	**0,002 (0,001)**	2,0 (0,002)	0,494 (9,1E-05)	**1,0 (0,0)**	0,0 (0,0)
NSGA-II	0,007 (0,001)	2,004 (0,004)	0,494 (3E-04)	**1,0 (0,0)**	0,0 (0,0)
SPEA-2	0,113 (0,493)	3,526 (6,870)	0,533 (0,119)	**1,0 (0,0)**	0,0 (0,0)

VI. CONCLUSION AND FUTURE WORK

This paper proposed a novel multi-objective particle swarm-based optimizer based on a measure of diversity of the whole swarm. The selection of the cognitive and social leaders is performed using a new measure called Diversity Factor (DF). DF is also used to update and prune the External Archive (EA), which is used to store the non-dominated solutions found during the search process so far.

TABLE IV. SIMULATION RESULTS FOR DTLZ-2 WITH 300.000 FITNESS FUNCTION EVALUATIONS.

Alg.	S	MS	HV	C(DFR,*)	C(*,DFR)
MOPSO DFR	0.041 (0.009)	**1.017 (0.005)**	**0.187 (0.006)**	- -	- -
MOPSO CDR	**0.001 (1E-04)**	1.0 (4E-08)	0.211 (1E-05)	**0.961 (0.037)**	0.0 (0.0)
SMPSO	0.002 (2E-04)	1.0 (4E-05)	0.210 (6E-05)	**0.960 (0.036)**	0.0 (0.0)
NSGA-II	0.007 (6E-04)	1.001 (0.005)	0.211 (0.008)	**0.953 (0.048)**	0.0 (0.0)
SPEA-2	0.003 (3E-04)	1.001 (0.001)	0.212 (0.002)	**0.953 (0.049)**	0.0 (0.0)

TABLE V. SIMULATION RESULTS FOR DTLZ-3 WITH 300.000 FITNESS FUNCTION EVALUATIONS.

Alg.	S	MS	HV	C(DFR,*)	C(*,DFR)
MOPSO DFR	42.289 (12.110)	187.997 (10.405)	**0.127 (0.044)**	- -	- -
MOPSO CDR	33.916 (22.403)	**208.675 (19.646)**	0.174 (0.067)	0.276 (0.294)	**0.451 (0.299)**
SMPSO	3.793 (5.758)	18.498 (26.729)	0.335 (0.279)	**1.0 (0.0)**	0.0 (0.0)
NSGA-II	**0.028 (0.043)**	1.489 (0.608)	0.263 (0.112)	**1.0 (0.0)**	0.0 (0.0)
SPEA-2	0.069 (0.158)	1.473 (0.590)	0.298 (0.168)	**1.0 (0.0)**	0.0 (0.0)

TABLE VI. SIMULATION RESULTS FOR DTLZ-4 WITH 300.000 FITNESS FUNCTION EVALUATIONS.

Alg.	S	MS	HV	C(DFR,*)	C(*,DFR)
MOPSO DFR	0.116 (0.031)	**1.020 (0.007)**	**0.131 (0.018)**	- -	- -
MOPSO CDR	**0.002 (3E-04)**	1.0 (2E-07)	0.210 (2E-04)	**0.804 (0.140)**	0.0 (0.0)
SMPSO	0.0018 (4E-04)	1.0 (3E-04)	0.211 (5E-04)	**0.808 (0.136)**	0.0 (0.0)
NSGA-II	0.0064 (0.002)	0.801 (0.40)	0.169 (0.085)	**0.846 (0.140)**	0.002 (0.010)
SPEA-2	0.0032 (0.001)	0.733 (0.442)	0.170 (0.104)	**0.833 (0.155)**	0.0 (0.0)

TABLE VII. SIMULATION RESULTS FOR DTLZ-5 WITH 300.000 FITNESS FUNCTION EVALUATIONS.

Alg.	S	MS	HV	C(DFR,*)	C(*,DFR)
MOPSO DFR	0.034 (0.008)	**1.026 (0.005)**	**0.186 (0.006)**	- -	- -
MOPSO CDR	**0.001 (2E-04)**	1.008 (3E-08)	0.211 (2E-05)	**0.962 (0.051)**	0.0 (0.0)
SMPSO	0.002 (3E-04)	1.008 (2E-05)	0.210 (7E-05)	**0.961 (0.053)**	0.0 (0.0)
NSGA-II	0.007 (6E-04)	1.009 (0.003)	0.211 (0.004)	**0.953 (0.064)**	0.0 (0.0)
SPEA-2	0.003 (3E-04)	1.009 (0.005)	0.213 (0.007)	**0.954 (0.065)**	0.0 (0.0)

TABLE VIII. SIMULATION RESULTS FOR DTLZ-6 WITH 300.000 FITNESS FUNCTION EVALUATIONS.

Alg.	S	MS	HV	C(DFR,*)	C(*,DFR)
MOPSO DFR	0.029 (0.008)	1.012 (0.009)	0.191 (0.005)	- -	- -
MOPSO CDR	0.196 (0.097)	**1.638 (0.232)**	**0.180 (0.056)**	0.0 (0.0)	**0.993 (0.037)**
SMPSO	**0.001 (2E-04)**	1.008 (4E-16)	0.211 (3E-05)	**0.843 (0.064)**	0.002 (0.005)
NSGA-II	0.011 (0.003)	1.040 (0.035)	0.494 (3E-04)	0.361 (0.350)	**0.406 (0.388)**
SPEA-2	0.013 (0.019)	1.086 (0.119)	0.278 (0.103)	**0.541 (0.325)**	0.205 (0.260)

TABLE IX. SIMULATION RESULTS FOR DTLZ-7 WITH 300.000 FITNESS FUNCTION EVALUATIONS.

Alg.	S	MS	HV	C(DFR,*)	C(*,DFR)
MOPSO DFR	0.018 (0.012)	0.637 (0.045)	**0.290 (0.054)**	- -	- -
MOPSO CDR	0.001 (2E-04)	0.745 (5E-04)	0.335 (8E-04)	**0.458 (0.103)**	0.0 (0.0)
SMPSO	**0.001 (1E-04)**	0.745 (4E-04)	0.334 (4E-04)	**0.665 (0.104)**	0.0 (0.0)
NSGAII	0.005 (5E-04)	0.745 (0.001)	0.334 (0.002)	**0.657 (0.106)**	0.0 (0.0)
SPEA2	0.004 (4E-04)	**0.745 (4E-04)**	0.334 (7E-04)	**0.698 (0.091)**	0.0 (0.0)

Simulation results showed that MOPSO-DFR converged to Pareto Fronts that dominate Pareto Fronts obtained by well known multi-objective optimization algorithms in most of the functions of a widely used set of benchmark functions. The results shows that our proposal can reach more extreme solutions while converging for the optimum Pareto simultaneously.

We believe that we achieved better results because the DF is a global density estimator that helps to properly select the leaders and to prune the EA.

We did not achieve the best result for the DTLZ-6 function. Our hypothesis for this case is that MOPSO-DFR converges too fast to a local optimum and gets trapped at this local minimum. We intend to analyze in details what happened for the DTLZ-6 function in order to propose improvements for the MOPSO-DFR. We believe it can help to improve the spacing (S) of the non-dominated solutions.

ACKNOWLEDGMENT

The authors would like to thank the Polytechnic School of the University of Pernambuco, FACEPE, CNPq and FINEP.

REFERENCES

[1] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE Internacional Conference on Neural Networks*. IEEE, 1995, pp. 1945–1948.

[2] C. A. C. Coello and M. S. Lechuga, "MOPSO: A proposal for multiple objective particle swarm optimization," in *Proceedings of the Congress on Evolutionary Computation (CEC 2002)*, vol. 2. Honolulu, HI, EUA: IEEE, 2002, pp. 1051–1056.

[3] R. A. Santana, M. R. Pontes, and C. J. A. Bastos-Filho, "A multiple objective particle swarm optimization approach using crowding distance and roulette wheel," in *ISDA '09: Proceedings of the 2009 Ninth International Conference on Intelligent Systems Design and Applications*. IEEE, 2009, pp. 1–5.

[4] X. Hu, R. Eberhart, and Y. Shi, "Particle swarm with extended memory for multiobjective optimization," in *Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE*, 2003, pp. 193–197.

[5] S.-Y. Chiu, T.-Y. Sun, S.-T. Hsieh, and C.-W. Lin, "Cross-searching strategy for multi-objective particle swarm optimization." in *IEEE Congress on Evolutionary Computation*. IEEE, 2007, pp. 3135–3141.

[6] A. J. Nebro, J. J. Durillo, J. García-Nieto, C. A. Coello Coello, F. Luna, and E. Alba, "SMPSO: A New PSO-based Metaheuristic for Multi-objective Optimization," in *IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MCDM 2009)*, 2009, pp. 66–73.

[7] Z.-H. Zhan, J. Zhang, Y. Li, and H. S.-H. Chung, "Adaptive particle swarm optimization," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 39, no. 6, pp. 1362–1381, Dec. 2009. [Online]. Available: http://dx.doi.org/10.1109/TSMCB.2009.2015956

[8] E. Zitzler, "Evolutionary algorithms for multiobjective optimization: Methods and applications," PHD Thesis, ETH Zurich, Switzerland, 1999.

[9] J. L. Risco-Martín, S. Mittal, D. Atienza, J. I. Hidalgo, and J. Lanchares, "Optimization of dynamic data types in embedded systems using devs/soa-based modeling and simulation," in *InfoScale '08: Proceedings of the 3rd international conference on Scalable information systems*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 1–11.

[10] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE transactions on Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[11] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002. [Online]. Available: http://dx.doi.org/10.1109/4235.985692

[12] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[13] E. Zitzler, M. Laumanns, and L. Thiele, “SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization,” in *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*. International Center for Numerical Methods in Engineering, 2001, pp. 95–100.

[14] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, “Scalable test problems for evolutionary multiobjective optimization,” in *Evolutionary Multiobjective Optimization*, ser. Advanced Information and Knowledge Processing, A. Abraham, L. Jain, and R. Goldberg, Eds. Springer London, 2005, pp. 105–145.